

No. 587

June 2018

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

**A. Sokolov, O. Davydov, D. Kuzmin
A. Westermann, S. Turek**

ISSN: 2190-1767

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

Andriy Sokolov* Oleg Davydov† Dmitri Kuzmin*
Alexander Westermann* Stefan Turek*

May 28, 2018

Abstract

In this article we introduce a FCT stabilized Radial Basis Function (RBF)-Finite Difference (FD) method for the numerical solution of convection dominated problems. The proposed algorithm is designed to maintain mass conservation and to guarantee positivity of the solution for an almost random placement of scattered data nodes. The method can be applicable both for problems defined in a domain or if equipped with level set techniques, on a stationary manifold. We demonstrate the numerical behavior of the method by performing numerical tests for the solid-body rotation benchmark in a unit square and for a transport problem along a curve implicitly prescribed by a level set function. Extension of the proposed method to higher dimensions is straightforward and easily realizable.

Key words: Flux correction, Radial basis functions, generalized finite differences, level set, convection dominated problems

1 Introduction

Diffusion-convection equations, which are written with partial differentials, are used to describe many important problems in mechanics, fluid dynamics, medicine, biology and other branches of science and technology. Numerical calculation of these equations are nontrivial, especially when the convection strongly dominates the diffusion. As a result, conventional space discretization methods (e.g. the method of finite differences, the finite volume method, the finite element method) are not able to deliver a sufficiently smooth, positively preserved numerical solution with the mass-conservation. Without additional stabilization technique the resulting numerical solution will fail to predict the physical result. Namely, the presence of numerical oscillations increases as marched in time and spoils the numerical solution in the entire domain. The construction of a stabilization method that would eliminate this effect is one of the most important problems in the numerical mathematics. In the

*Institute for Applied Mathematics, TU Dortmund, Email addresses:
kuzmin@math.uni-dortmund.de, alexander.westermann@tu-dortmund.de,
ture@featflow.de.

†Department of Mathematics, University of Giessen, Email address:
oleg.davydov@math.uni-giessen.de.

last few decades, many profound and effective stabilization methods have been developed, see e.g. [16]. Most of them were developed in the context of finite elements, which require construction of a mesh and the corresponding mesh-related procedures. Creating such a mesh and its maintenance/modification during the simulation process can be very expensive and time-consuming. The kernel methods which are based on radial basis functions are becoming increasingly popular for the numerical simulation of partial differential equations due to their flexibility of working with scattered data nodes, high spectral accuracy, good convergence and significantly easier implementation. These methods demonstrated promising results for various problems of PDEs in two- and three-dimensional domains, see, e.g., [1, 3, 21, 12].

Recently, various attempts have been made to create stabilization methods for convection dominated problems within the meshless framework. Among them are approaches which are based on the hyperviscosity [2]; use upwind-like techniques [15] or act by adding extra nodes in the regions where the numerical solution has a steep gradient. Substantial drawbacks of these methods are: they do not automatically guarantee positivity-preservation and mass-conservation of the numerical solution; they might lack the high order of accuracy near steep gradients; and they might require a very scrupulous and often heuristic hand-tuning of problem-dependent parameters.

In this article we introduce a flux-corrected transport (FCT) algorithm for the method of Radial Basis Functions with generalized Finite-Differences (RBF-FD). The method guarantees mass conservation, positivity preservation and high-order accuracy of the numerical solution. Besides these properties the considered methodology has the following additional benefits:

- the method can be used for almost any set of scattered data nodes;
- the method admits the straightforward extension to higher spatial dimensions;
- the method does not require any artificially tuned problem-dependent parameters.

The FCT techniques are known since 1970s, see e.g. [32]. Recently they have been significantly improved in the context of the finite element method [17, 18, 19]. This paper is the first attempt to apply the state-of-the-art FCT methodology to meshless methods.

We apply the FCT method to the numerical stabilization of convection dominated problems not only on domains, but also on manifolds, which is enabled by combining the RBF-FD method with the level set technique, see [11, 26].

This paper is organized as follows. After this opening section we formulate the problem, which consists of a partial differential equation and which can be defined either in a domain or on some sufficiently smooth and closed manifold of arbitrary curvature. Here, we briefly introduce the level set method to be employed for the numerical treatment of surface-defined PDEs. In section 3 we discuss the method of radial basis functions with generalized finite differences and the corresponding numerical approximation of the (general-purpose) diffusion and convection operators. After that, in section 4 we describe the flux corrected transport paradigm to be applied in combination with the RBF-FD method for the numerical stabilization of the dominant convection. Therein we present numerical results that illustrate the properties of the proposed scheme. After analysis of these results, the last section is dedicated to some further remarks and discussions.

2 Problem formulation

In this article we consider the general transport problem of the following type:

$$u_t - \nabla_{\mathcal{P}} \cdot (D\nabla_{\mathcal{P}}u) + \mathbf{v} \cdot \nabla_{\mathcal{P}}u = 0, \quad (1)$$

where $\nabla_{\mathcal{P}} = \mathcal{P}\nabla$. If $\mathcal{P} = \mathcal{I}$, the general problem (1) transforms into the diffusion-convection equation

$$u_t - \nabla \cdot (D\nabla u) + \mathbf{v} \cdot \nabla u = 0, \quad \text{in } \Omega, \quad (2)$$

where $\Omega \subset \mathbb{R}^d$, $d = 2, 3$ is a computational domain and \mathbf{v} is some velocity vector-field.

In the case when $\mathcal{P} = \mathcal{P}_{\Gamma}$ is a projection onto the tangent space $\mathcal{T}_{\mathbf{x}}\Gamma$ of a manifold Γ , we obtain the diffusion-convection equation

$$u_t - \nabla_{\Gamma} \cdot (D\nabla_{\Gamma}u) + \mathbf{v} \cdot \nabla_{\Gamma}u = 0, \quad \text{on } \Gamma. \quad (3)$$

Here, $\nabla_{\Gamma} \cdot (D\nabla_{\Gamma}\cdot)$ is the generalized Laplace-Beltrami operator. We assume that the solution u of (3) can be (naturally) extended from Γ to an ε -band Ω_{ε} , see Figure 1. The domain of interest Ω is decomposed into $\Omega = \Omega_{in} \cup \Omega_{out} \cup \Gamma$. For the sake of simplicity, we also assume

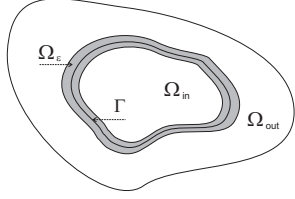


Figure 1: Geometric illustration.

that the boundary $\partial\Omega$ is aligned with some level set and therefore no boundary conditions for u on $\partial\Omega$ are necessary.

2.1 Level set method

For the implicit prescription of a compact, smoothly connected, oriented, time-independent hypersurface $\Gamma \subset \Omega$ we introduce a smooth level set function

$$\phi(t, \mathbf{x}) = \begin{cases} < 0, & \text{if } \mathbf{x} \text{ is inside } \Gamma, \\ = 0, & \text{if } \mathbf{x} \in \Gamma, \\ > 0, & \text{if } \mathbf{x} \text{ is outside } \Gamma, \end{cases} \quad (4)$$

such that $|\nabla\phi| \neq 0$ in Ω_{ε} . Then, an outward normal to Γ at the point \mathbf{x} is

$$\mathbf{n}(\mathbf{x}) = (n^1, n^2, \dots, n^d)^T = \nabla\phi(\mathbf{x})/|\nabla\phi(\mathbf{x})| \quad (5)$$

and the matrix

$$\mathcal{P}_{\Gamma} = I - \mathbf{n}\mathbf{n}^T = (\delta_{ij} - n^i n^j)_{i,j=1}^d \quad (6)$$

is the projection onto the tangent space $\mathcal{T}_{\mathbf{x}}\Gamma$. For a scalar function η and a tangential vector field $\boldsymbol{\eta} = (\eta^1, \eta^2, \dots, \eta^d)^T$ on Γ extended into Ω we can define

$$\nabla_{\Gamma}\eta := (\mathcal{P}_{\Gamma}\nabla)\eta = \left\{ \frac{\partial\eta}{\partial x^i} - \sum_{j=1}^d n^i n^j \frac{\partial\eta}{\partial x^j} \right\}_{i=1}^d, \quad (7)$$

$$\nabla_{\Gamma} \cdot \boldsymbol{\eta} = \sum_{i=1}^d \left(\frac{\partial\eta^i}{\partial x^i} - \sum_{j=1}^d n^i n^j \frac{\partial\eta^i}{\partial x^j} \right), \quad (8)$$

the surface gradient ∇_{Γ} and the surface divergence $\nabla_{\Gamma} \cdot$ operators, respectively. Using this notation, the Laplace-Beltrami operator can be written as

$$\Delta_{\Gamma}\eta = \nabla_{\Gamma} \cdot \nabla_{\Gamma}\eta = \mathcal{P}_{\Gamma}\nabla \cdot \mathcal{P}_{\Gamma}\nabla\eta. \quad (9)$$

3 Numerical discretization in space and in time

3.1 Kernel interpolation and operator approximation

Given a set of scattered nodes $X = \{\mathbf{x}_j\}_{j=1}^N \subset \Omega$ we are looking for a continuous function $u : \Omega \rightarrow \mathbb{R}$ as a kernel interpolant, whose general form is

$$I_\phi u(\mathbf{x}) = \sum_{j=1}^N c_j \Phi(\mathbf{x}, \mathbf{x}_j), \quad \mathbf{x} \in \Omega, \quad (10)$$

such that u is a solution of the corresponding partial differential equation. Here, Φ is a positive definite kernel called a *radial basis function* (RBF) with the property $\Phi(\mathbf{x}, \mathbf{y}) = \varphi(\|\mathbf{x} - \mathbf{y}\|)$. Denoting $\mathbf{r}_j(\mathbf{x}_i) = \|\mathbf{x}_i - \mathbf{x}_j\|$, the interpolation coefficients $\{c_j\}_{j=1}^N$ are determined by enforcing $I_\phi u|_X = u|_X$ as the following linear system:

$$A_X c_X = u_X, \quad (11)$$

where

$$A_X = \begin{bmatrix} \varphi(\mathbf{r}_1(\mathbf{x}_1)) & \varphi(\mathbf{r}_2(\mathbf{x}_1)) & \dots & \varphi(\mathbf{r}_N(\mathbf{x}_1)) \\ \varphi(\mathbf{r}_1(\mathbf{x}_2)) & \varphi(\mathbf{r}_2(\mathbf{x}_2)) & \dots & \varphi(\mathbf{r}_N(\mathbf{x}_2)) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi(\mathbf{r}_1(\mathbf{x}_N)) & \varphi(\mathbf{r}_2(\mathbf{x}_N)) & \dots & \varphi(\mathbf{r}_N(\mathbf{x}_N)) \end{bmatrix}, \quad c_X = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{bmatrix}, \quad u_X = \begin{bmatrix} u(\mathbf{x}_1) \\ u(\mathbf{x}_2) \\ \vdots \\ u(\mathbf{x}_N) \end{bmatrix}.$$

For a positive definite φ , this system is positive definite and hence solvable.

In the following we use the radial basis function finite difference (RBF-FD) method for approximation of all linear differential operators, which arise through our derivations. Let \mathcal{L} be one of these linear operators. Then the approximation of $\mathcal{L}u$ at the point ζ is sought as a weighted sum of function values $u(\xi_j)$ at the points $\Xi = \Xi_\zeta = \{\xi_1, \xi_2, \dots, \xi_K\}$ neighboring to ζ :

$$\mathcal{L}u(\zeta) \approx \sum_{j=1}^K \omega_j u(\xi_j), \quad \xi_j \in \Xi, \quad (12)$$

where the approximation weights $\omega = (\omega_1, \omega_2, \dots, \omega_K)^T$ can be computed by solving the linear system

$$A_\Xi \omega = [\mathcal{L}\varphi(\mathbf{r}_j(\zeta))]_{j=1}^K \quad \text{with} \quad A_\Xi := [\varphi(\mathbf{r}_j(\xi_i))]_{i,j=1}^K. \quad (13)$$

In general, a good choice of sets of influence ξ_i for the accurate approximation of $\mathcal{L}u(\zeta)$ is a nontrivial task which requires additional analysis [4, 21, 6]. In this article, the set Ξ_ζ consists of the $K = 9$ points nearest to ζ in the Euclidean distance, including ζ itself. The more detailed study of the choice of the sets of influence and its consequences for the accuracy of the resulting discretization is of great importance, but further discussion lies outside of the scope of this article. Either Gaussian $\varphi(r) = \exp(-\epsilon^2 r^2)$ with $\epsilon > 0$ close to zero, or the polyharmonic radial basis function $\varphi(r) = r^\gamma$ with $\gamma = 5$ are used in all presented numerical simulations. In the case of Gaussian we use a QR preconditioning technique that allows stable computation of the weights for any value of the shape parameter ϵ [14, 5, 20]. Polyharmonic RBF is only conditionally positive definite and therefore the interpolant (10) is extended in this case by a polynomial term of degree $\lfloor \gamma/2 \rfloor$, see [12, 13] for details.

In the case of a vector-valued operator \mathcal{L} the weights ω_j are vectors, and ω is a matrix. In particular, (12) is replaced by

$$\nabla u(\zeta) \approx \omega_{\nabla}(\zeta, \Xi)^T u_{\Xi} \quad (14)$$

for the gradient operator ∇ , where each column of the matrix $\omega_{\nabla}(\zeta, \Xi) \in \mathbb{R}^{K \times d}$ is obtained by solving (13) for the corresponding partial derivative operator. Clearly, a gradient-type operator $\mathcal{L}_{\text{grad}}^A u = A \nabla u$ with components $\sum_{j=1}^d a_{ij} \frac{\partial u}{\partial x_j}$, $i = 1, \dots, d$, where $A : \Omega \rightarrow \mathbb{R}^{d \times d}$, can be discretized as

$$\mathcal{L}_{\text{grad}}^A u(\zeta) \approx \left[\sum_{i=1}^K \omega_{ij} u(\xi_i) \right]_{j=1}^d = A(\zeta) \omega_{\nabla}^T(\zeta, \Xi) u_{\Xi}, \quad (15)$$

where $\omega := \omega_{\nabla}(\zeta, \Xi) A^T(\zeta)$. A simple calculation shows that the same weight matrix $\omega = \omega_{\nabla}(\zeta, \Xi) A^T(\zeta)$ gives a discretization

$$\mathcal{L}_{\text{div}}^A u(\zeta) \approx \sum_{i=1}^K \sum_{j=1}^d \omega_{ij} u_j(\xi_i) = \text{trace} (A(\zeta) \omega_{\nabla}^T(\zeta, \Xi) u_{\Xi}) \quad (16)$$

of the divergence-type operator $\mathcal{L}_{\text{div}}^A u = A \nabla \cdot u := \sum_{i,j=1}^d a_{ij} \frac{\partial u_i}{\partial x_j}$, where $u = (u_1, \dots, u_d)^T$ is a vector-function, and $u_{\Xi} = [u_j(\xi_i)]_{i,j=1}^{K,d}$.

Formulas (15) and (16) can be combined to obtain an approximation of the anisotropic diffusion operator

$$\Delta^{A,B} u := A \nabla \cdot B \nabla u = \mathcal{L}_{\text{div}}^A \mathcal{L}_{\text{grad}}^B u, \quad A, B : \Omega \rightarrow \mathbb{R}^{d \times d}.$$

To this end, an auxiliary set of points $\Lambda = \{\gamma_1, \dots, \gamma_L\}$ is chosen in the neighborhood of ζ , an approximation of the vector

$$u(\gamma_s) := \mathcal{L}_{\text{grad}}^B u(\gamma_s) \approx \left[\sum_{i=1}^K \omega_{ij}(\gamma_s) u(\xi_i) \right]_{j=1}^d, \quad \omega(\gamma_s) := \omega_{\nabla}(\gamma_s, \Xi) B^T(\gamma_s),$$

is obtained by (15) for each $s = 1, \dots, L$, and inserted into (16), where Λ is used instead of Ξ . Setting $\tilde{\omega} := \omega_{\nabla}(\zeta, \Lambda) A^T(\zeta)$, we arrive at

$$\Delta^{A,B} u(\zeta) \approx \sum_{i=1}^K \omega_i u(\xi_i), \quad \omega_i = \sum_{s=1}^L \sum_{j=1}^d \tilde{\omega}_{sj} \omega_{ij}(\gamma_s), \quad (17)$$

that is

$$\omega_i = \text{trace} (\tilde{\omega} [\omega_{ij}(\gamma_s)]_{j,s=1}^{d,L}), \quad i = 1, \dots, K.$$

In the case when $A = B$ and $\zeta \in \Lambda = \Xi$ the formulas for ω_i in (17) can be simplified since $\tilde{\omega}$ coincides with one of the matrices $\omega(\gamma_s)$, see [11, 26]. We however prefer to choose Λ closer to ζ , in order to obtain more reliable numerical differentiation formulas for $\mathcal{L}_{\text{grad}}^B u(\gamma_s)$. In this paper we use

$$\gamma_j = (\zeta + \xi_j)/2, \quad j = 1, \dots, K, \quad (18)$$

see Figure 2, where $\xi_1 = \zeta$.

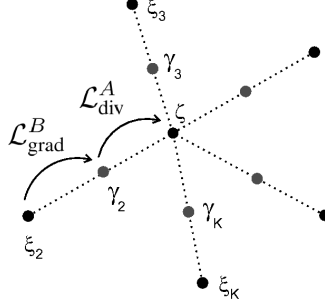


Figure 2: Discretization of the anisotropic diffusion operator.

3.2 RBF-FD discretization in space

Let us denote discrete counterparts of the involved continuous operators as follows:

$$L(t, \mathcal{I})\mathbf{u} \approx -\nabla \cdot (D(\mathbf{x})\nabla u)|_X, \quad (19)$$

$$L(t, \mathcal{P})\mathbf{u} \approx -\nabla_{\mathcal{P}} \cdot (D(\mathbf{x})\nabla_{\mathcal{P}} u)|_X, \quad (20)$$

$$K(t, \mathbf{v}, \mathcal{I})\mathbf{u} \approx -\mathbf{v} \cdot \nabla u|_X, \quad (21)$$

$$K(t, \mathbf{v}, \mathcal{P})\mathbf{u} \approx -\mathbf{v} \cdot \nabla_{\mathcal{P}} u|_X, \quad (22)$$

where $\mathbf{u} = (u_1, u_2, \dots, u_N)^T \approx u|_X = (u(\mathbf{x}_1), u(\mathbf{x}_2), \dots, u(\mathbf{x}_N))^T$, with $X = \{\mathbf{x}_j\}_{j=1}^N \subset \Omega$. We now describe the RBF-FD construction of the discrete operators (19)–(22).

After choosing a set of nodes $X = \{\mathbf{x}_j\}_{j=1}^N \subset \Omega$, we select for each $\zeta \in X$ a set of neighbors $\Xi_{\zeta} \subset X$. Thanks to (9), the value of the operator $-\nabla_{\mathcal{P}} \cdot (D(\mathbf{x})\nabla_{\mathcal{P}} u)$ can be approximated according to (17) with $A = -\mathcal{P}$ and $B = D(\mathbf{x})\mathcal{P}$, and the weights ω_i of this formula become the nonzero entries of the ζ -row of the matrix $L(t, \mathcal{P})$ in (20). In particular, for $\mathcal{P} = \mathcal{P}_{\Gamma}$ we obtain a discretization of the generalized Laplace-Beltrami operator $-\Delta_{\Gamma} u(\zeta)$.

For the RBF-FD approximation of convection operators $\mathbf{v} \cdot \nabla u$ and $\mathbf{w} \cdot \nabla_{\mathcal{P}} u$ we use

$$(\mathbf{w} \cdot \nabla_{\mathcal{P}} u)(\zeta) \approx \mathbf{w}^T(\zeta)\mathcal{P}(\zeta)\omega_{\nabla}^T(\zeta, \Xi_{\zeta})u_{\Xi_{\zeta}}$$

as in (15), leading to the weights for the ζ -row of $K(t, \mathbf{v}, \mathcal{P})$ in (22). In particular, for $\mathbf{w} \cdot \nabla_{\Gamma} u$ we make an assumption that both vector fields \mathbf{v} and \mathbf{w} can be extended outside of Γ to the whole domain Ω . Then by (7), $\nabla_{\Gamma} u = \mathcal{P}_{\Gamma}\nabla u$, and the above formula is used with $\mathcal{P} = \mathcal{P}_{\Gamma}$.

Construction of corresponding discretizations for the discrete operators (19) and (21) is done in a similar way by setting $\mathcal{P} = \mathcal{I}$.

3.3 Discretization in time

For simplicity of notation we will not distinguish between operators $K(t, \mathbf{v}, \mathcal{I})$ and $K(t, \mathbf{v}, \mathcal{P})$ and will denote them as $K(t, \mathbf{v})$. Analogously, we will do for diffusion operators $L(t, \mathcal{I})$ and $L(t, \mathcal{P})$ by denoting them as $L(t)$. After the RBF-FD discretization of problem (2),

resp. (3), one obtains a semi-discrete problem of the form:

$$\mathcal{M} \frac{d\mathbf{u}}{dt} = \mathcal{K}\mathbf{u}, \quad (23)$$

where $\mathcal{K} = -\mathcal{M}(L(t) + K(t, \mathbf{v}))$, and $\mathcal{M} = \text{diag}(m_1, \dots, m_N)$ is a diagonal mass matrix, where m_i determines the area of a subdomain belonging to the i th node. The most straightforward, though not optimal approach to constructing a matrix \mathcal{M} is to use the areas of the cells of the Voronoi tessellation of Ω generated by X . This method is, however, quite laborious, so it is advisable to use other methods, for example the construction of primal-dual grids fitted to meshless methods [30]. For a uniform grid the unit matrix $M = \mathcal{I}$ can be used because the areas associated with all nodes are equal.

For the discretization in time of problem (23) we use the θ -scheme method: given \mathbf{u}^n and the time step $\Delta t = t_{n+1} - t_n$, solve for \mathbf{u}^{n+1}

$$(\mathcal{M} - \theta \Delta t \mathcal{K}) \mathbf{u}^{n+1} = (\mathcal{M} + (1 - \theta) \Delta t \mathcal{K}) \mathbf{u}^n. \quad (24)$$

Here, the choice $\theta = 1, \frac{1}{2}, 0$ corresponds to the Implicit-Euler, Crank-Nicolson and Explicit-Euler schemes respectively.

4 Numerical stabilization

As shown by Kuzmin *et al.* [17, 18, 19], positivity constraints can be readily enforced at the algebraic level using a conservative manipulation of the matrices $\mathcal{M} = \{m_i\}$ and $\mathcal{K} = \{k_{ij}\}$ in equation (23). In the RBF-FD context the mass-matrix \mathcal{M} is already diagonal and therefore requires no additional changes of its entries. To enforce monotonicity, all negative off-diagonal elements of the matrix \mathcal{K} are eliminated by adding an artificial diffusion operator $\mathcal{D} = \{d_{ij}\}$:

$$\mathcal{K}^L = \mathcal{K} + \mathcal{D}. \quad (25)$$

Physical meaning of (25) is addition of artificial diffusion; so that, the numerical solution u_L of

$$\mathcal{M} \frac{du}{dt} = \mathcal{K}^L u, \quad (26)$$

satisfies positivity constraints but is of low order. For conservation reasons, the matrix \mathcal{D} must be symmetric with zero row and column sums. For any pair of neighboring nodes i and j , the entry d_{ij} is defined as [17, 18]

$$d_{ij} = \max\{-k_{ij}, 0, -k_{ji}\}, \quad j \neq i. \quad (27)$$

Note that $d_{ji} = d_{ij}$, so that \mathcal{D} is a symmetric matrix. The diagonal coefficients d_{ii} are defined so that the row and column sums of \mathcal{D} are equal to zero

$$d_{ii} = - \sum_{j \neq i} d_{ij}. \quad (28)$$

It should be noted if the non-diagonal elements of \mathcal{K} are positive in the absence of \mathcal{D} , no addition of artificial diffusion is necessary and $\mathcal{K} = \mathcal{K}^L$. In this case the physical diffusion is enough to guarantee positivity preservation of the numerical solution from the beginning.

The procedure of addition of artificial diffusion is also applicable in the case $\theta \in [0, 1]$ in (24). By this we obtain the following discrete equation:

$$(\mathcal{M} - \theta \Delta t \mathcal{K}^L) \mathbf{u}^{n+1} = (\mathcal{M} + (1 - \theta) \Delta t \mathcal{K}^L) \mathbf{u}^n. \quad (29)$$

Now, we wish to achieve high resolution while keeping the scheme positivity-preserving. For this purpose, we rewrite the equation (29) as follows:

$$(\mathcal{M} - \theta \Delta t \mathcal{K}^L) \mathbf{u}^{n+1} = \mathcal{M} \mathbf{u}^{n+\theta} + \Delta t f(\mathbf{u}^n, \mathbf{u}^{n+1}), \quad (30)$$

where

$$\mathcal{M} \mathbf{u}^{n+\theta} = (\mathcal{M} + (1 - \theta) \Delta t \mathcal{K}^L) \mathbf{u}^n \quad (31)$$

and

$$f(\mathbf{u}^n, \mathbf{u}^{n+1}) = -(1 - \theta) \mathcal{D} \mathbf{u}^n - \theta \mathcal{D} \mathbf{u}^{n+1}. \quad (32)$$

Here, by f we denote the difference between the residuals of the low-order scheme and that of the original high-order scheme. By virtue of the above decomposition, we have

$$f_i = \sum_{j \neq i} f_{ij}, \quad f_{ji} = -f_{ij}, \quad \forall j \neq i. \quad (33)$$

To achieve high resolution while keeping the scheme positivity-preserving, each flux f_{ij} is multiplied by a solution-dependent correction factor $\alpha_{ij} \in [0, 1]$ and inserted into the right-hand side of the nonoscillatory low-order scheme (30):

$$(\mathcal{M} - \theta \Delta t \mathcal{K}^L) \mathbf{u}^{n+1} = \mathcal{M} \mathbf{u}^* := \mathcal{M} \mathbf{u}^{n+\theta} + \Delta t \bar{f}, \quad \bar{f}_i = \sum_{j \neq i} \alpha_{ij} f_{ij}. \quad (34)$$

The original discretization (24) corresponds to the setting $\alpha_{ij} := 1$. It may be used in regions where the numerical solution is smooth and well-resolved. The setting $\alpha_{ij} := 0$ is appropriate in the neighborhood of steep fronts.

The limiting process begins with cancelling all fluxes that are diffusive in nature and tend to flatten the solution profiles. The required modification is:

$$f_{ij} := 0 \quad \text{if} \quad f_{ij} \cdot (u_j^{n+\theta} - u_i^{n+\theta}) > 0,$$

where $\mathbf{u}^{n+\theta}$ is the positivity-preserving solution of low order defined by (31).

The computation of correction factors α_{ij} is accomplished by using Zalezak's algorithm [32] and involves the following algorithmic steps:

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

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

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

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

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

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

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

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

By defining nodal correction factors α_{ij} in such a way, one immediately obtains the boundedness of the right-hand-side of (34) =: $m_i u_i^*$:

$$u_{i,\min}^{n+\theta} := u_i^{n+\theta} + Q_i^- \leq u_i^* \leq u_i^{n+\theta} + Q_i^+ =: u_{i,\max}^{n+\theta}.$$

Furthermore, the limited anti-diffusion does not amplify local extrema, since

$$Q_i^\pm = 0 \Rightarrow R_i^\pm = 0 \Rightarrow \alpha_{ij} = 0.$$

Finally, the FCT scheme can be summarized by the following algorithm:

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

$$(\mathcal{M} - \theta \Delta t \mathcal{K}) \bar{\mathbf{u}}^{n+1} = (\mathcal{M} + (1 - \theta) \Delta t \mathcal{K}) \mathbf{u}^n. \quad (35)$$

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

$$\mathcal{M} \mathbf{u}^{n+\theta} = (\mathcal{M} + (1 - \theta) \Delta t \mathcal{K}^L) \mathbf{u}^n. \quad (36)$$

3. Compute antidiffusive fluxes $f_{ij}(\mathbf{u}^n, \mathbf{u}^{n+\theta})$ from (32) and correction factors $\alpha_{ij}^{n+\theta}$ to find the intermediate solution \mathbf{u}^* :

$$\mathcal{M} \mathbf{u}^* = \mathcal{M} \mathbf{u}^{n+\theta} + \Delta t \sum_{j \neq i} \alpha_{ij}^{n+\theta} f_{ij}(\mathbf{u}^n, \mathbf{u}^{n+\theta}). \quad (37)$$

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

$$(\mathcal{M} - \theta \Delta t \mathcal{K}^L) \mathbf{u}^{n+1} = \mathcal{M} \mathbf{u}^*. \quad (38)$$

Using the M-matrix property of $\mathcal{M} - \theta \Delta t \mathcal{K}^L$, a discrete maximum principle can be shown for \mathbf{u}^{n+1} [17, 18].

5 Numerical results

In this section, the developed FCT RBF-FD algorithm is applied to several domain- and surface-defined convection dominated problems that require the use of positivity-preserving discretization techniques. To ease the understanding of numerical results, we use the following notation in our examples:

u_{analyt} – analytical solution;

u_{pure} – numerical solution without any stabilization;

u_{stab} – FCT-stabilized numerical solution;

u_{diff} – overdiffusive numerical solution with no flux-limiting technique applied;

In our numerical experiments we use the polyharmonic radial basis function $\varphi(r) = r^\gamma$ with $\gamma = 5$. As uniformly distributed nodes seem most appropriate for the demonstration of the benefits of FCT in the benchmark problems we consider, we choose gridded nodes, and use $\mathcal{M} = \mathcal{I}$ in (23).

5.1 Example 1

In the first example we apply the proposed algorithm to the benchmark problem of the solid body rotation in 2D [17, 18, 19]. We solve the following pure transport equation:

$$\partial_t u + \mathbf{v} \cdot \nabla u = 0, \quad \text{in } \Omega = (0, 1)^2, \quad (39)$$

where $\mathbf{v} = (0.5 - y, x - 0.5)$ is the incompressible velocity field which corresponds to a counterclockwise rotation about the center $P = (0.5, 0.5)^T$ of the computational domain. The initial condition is shown in Figure 3.

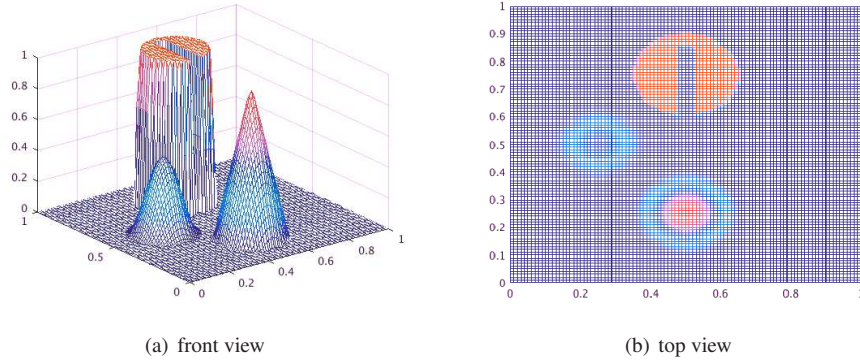
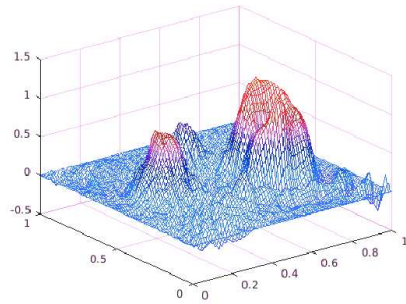


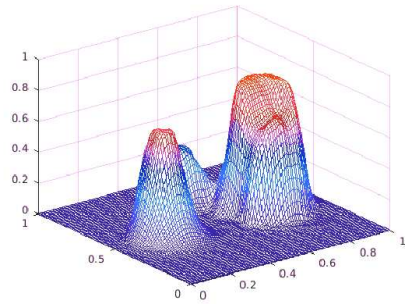
Figure 3: Initial condition.

In Figures 4(a)–4(f) we compare the pure RBF-FD discretization method with the FCT stabilized approach for various levels of spatial discretizations of Ω : $N = 50 \times 50$, $N = 100 \times 100$ and $N = 200 \times 200$. Here, the angle of rotation $\alpha = \frac{2\pi}{3}$ and the time-step $\Delta t = 0.002$ are chosen. One can clearly observe artificial oscillations and negative values by u_{pure} near regions of steep gradients, see Figures 4(a), 4(c) and 4(e). These nonphysical negative values grow rapidly as time evolves, which leads to the divergence of the algorithm and

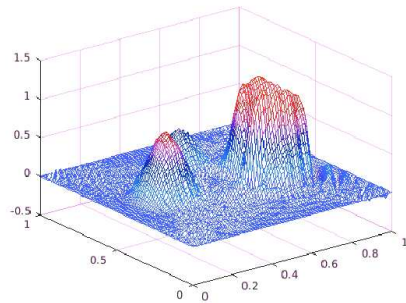
the corresponding termination of the simulation run. The corresponding FCT methodology helps to stabilize this type of problems and delivers a sufficiently accurate solution, see u_{stab} in Figures 4(b), 4(d) and 4(f).



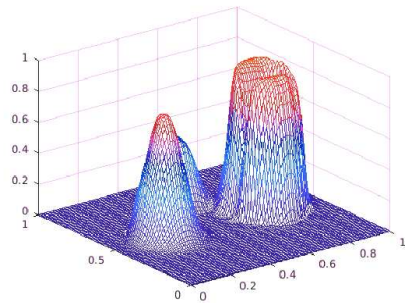
(a) non-stabilized, $N = 50 \times 50$



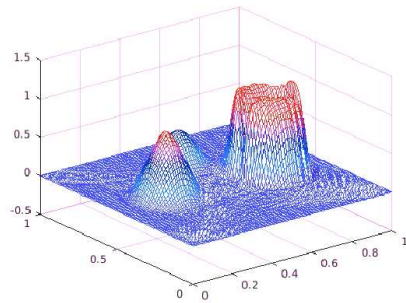
(b) FCT-stabilized, $N = 50 \times 50$



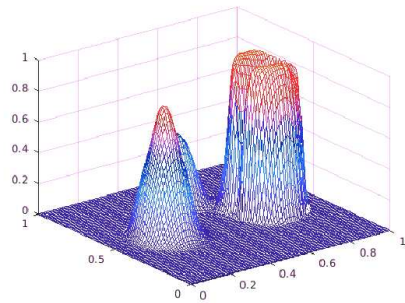
(c) non-stabilized, $N = 100 \times 100$



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



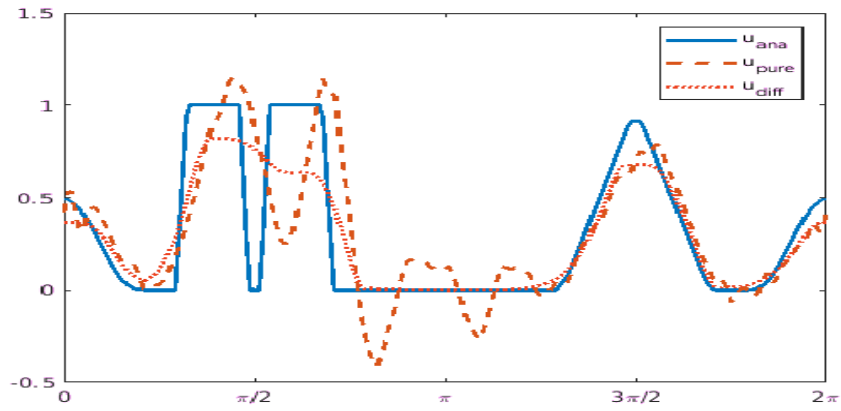
(e) non-stabilized, $N = 200 \times 200$



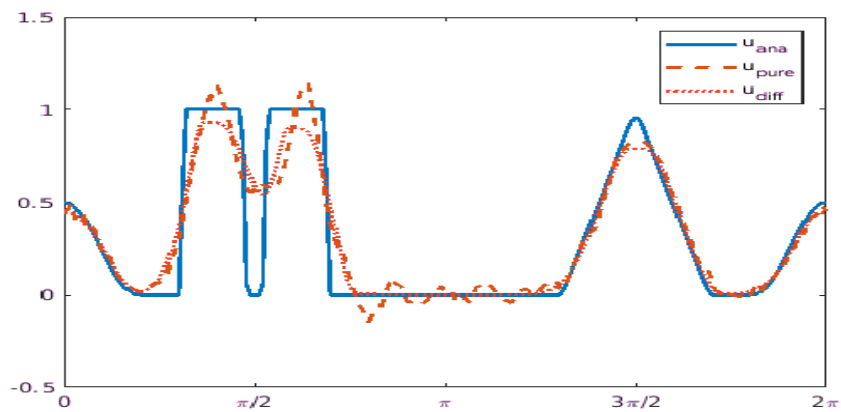
(f) FCT-stabilized, $N = 200 \times 200$

Figure 4: Numerical solution after rotation by the angle $\alpha = \frac{2\pi}{3}$.

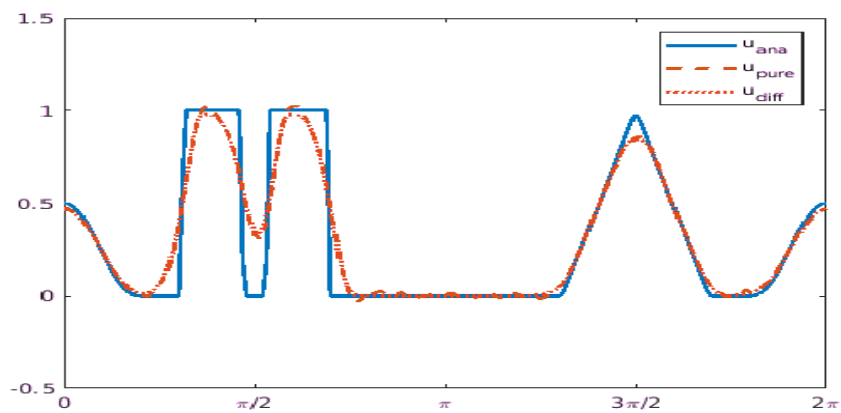
To demonstrate the comparative accuracy of solutions u_{pure} and u_{stab} , we compare them with the known u_{analyt} by plotting corresponding cutlines along a circle of a radius 0.25 which is centered at $(0.5, 0.5)^T$, see figures 5(a)–5(c). We can observe convergence of stabilized u_{stab} and non-stabilized u_{pure} numerical solutions to the analytical one u_{analyt} as a number of scattered data nodes increases. Also, as before we notice that the non-stabilized solution does not preserve the positivity constraint; at the same time the stabilized solution



(a) *Cutline*, $N = 50 \times 50$



(b) *Cutline*, $N = 100 \times 100$



(c) *Cutline*, $N = 200 \times 200$

Figure 5: *Cutlines* after rotation by the angle $\alpha = \frac{2\pi}{3}$.

preserves positivity even for a minimal number of unknowns $N = 50 \times 50$.

To conclude this example, in Figures 6(a)–6(d) we demonstrate the stabilized numerical solution u_{stab} at various time instances during the full counterclockwise rotation.

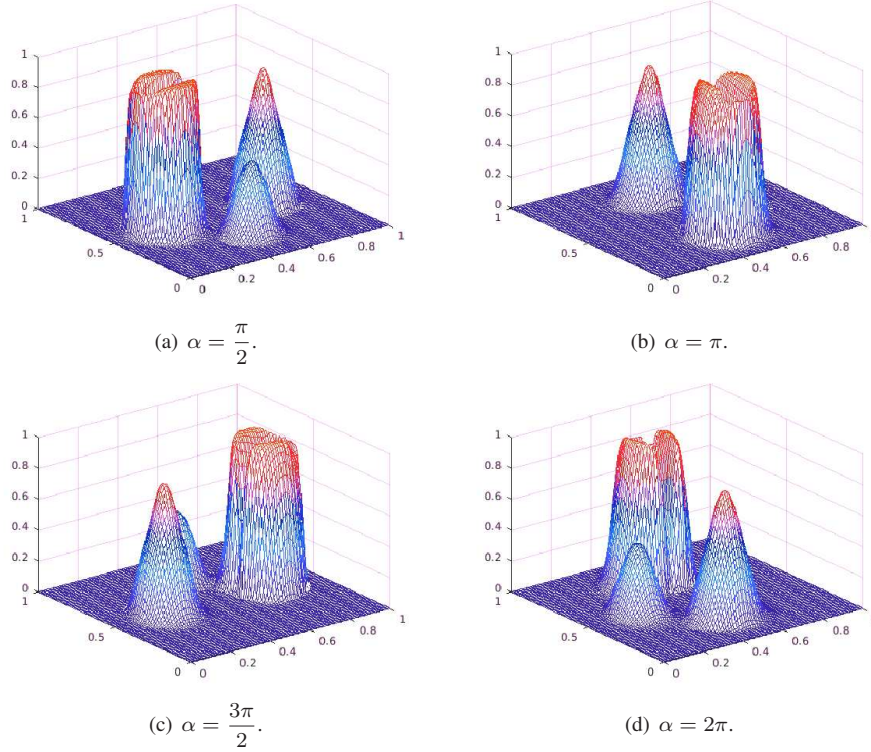


Figure 6: An FCT-stabilized solution at various time instances, $N = 200 \times 200$.

5.2 Example 2

As the next example, we consider the pure transport equation on a curve:

$$\partial_t \rho + \mathbf{v} \cdot \nabla_{\Gamma} \rho = 0 \quad \text{on } \Gamma. \quad (40)$$

Here, $\Gamma = \{\mathbf{x} : |\mathbf{x}| = 1\}$ is a unit circle, which is implicitly prescribed by the zero level-set of the corresponding indicator function

$$\phi(\mathbf{x}) = |\mathbf{x}| - 1.0.$$

For the sake of simplicity our computational domain $\Omega = \{\mathbf{x} : 0.5 \leq |\mathbf{x}| \leq 1.5\}$ is chosen to be a union of all level sets $\Gamma_c = \{\mathbf{x} : \phi(\mathbf{x}) = c\}$. The following initial condition

$$u(\mathbf{x}) = \begin{cases} 1 & \text{if } |\mathbf{x} - (0, 1)^T| \leq 0.1, \\ 0 & \text{otherwise,} \end{cases} \quad (41)$$

which is visualized in figure 7(b), and the advective velocity vector-field

$$\mathbf{v} = (-1, 0)^T$$

are employed.

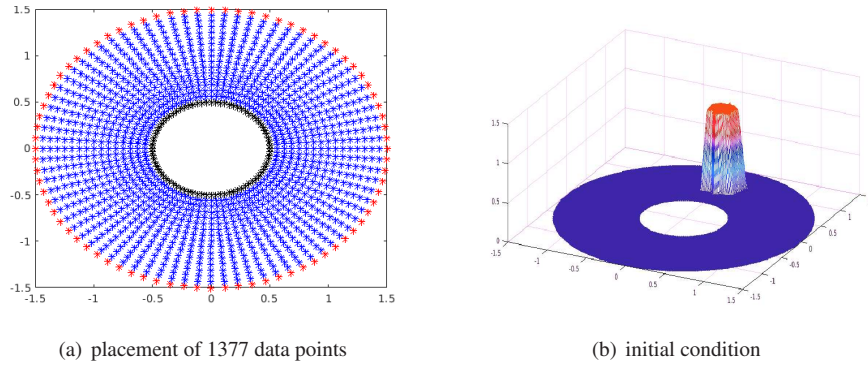


Figure 7: Initial setting.

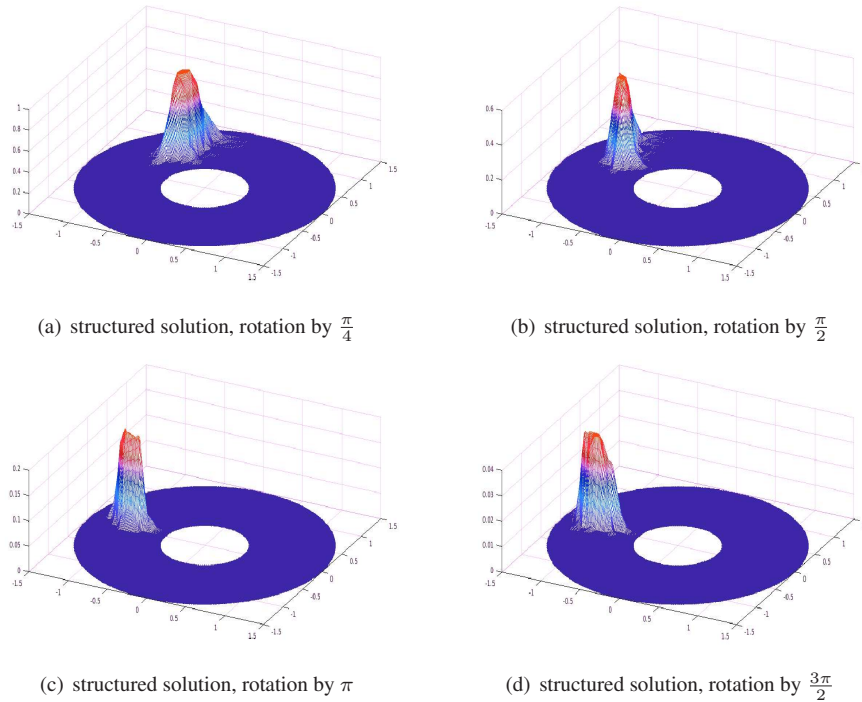


Figure 8: An FCT-stabilized numerical solution.

The obtained FCT-stabilized numerical solution, which is shown in Figures 8(a)–8(d), remains smooth and preserves positivity during the whole time of simulation. For the better resolution of u on Γ , one requires additional data nodes.

In the RBF-FD approach there is more freedom for choosing the sets of influence than, e.g., in the finite element method [21]. Since for the approximation of surface operators the tangential direction has a priority over the normal one, placement of nodes should to be much

denser and more carefully chosen along the level set (tangential directions) than along its gradient (the normal direction). For the construction of convection or, resp., advection operators, it is more practically efficient to precalculate the convection or, resp., advection field v and then choose nodes for a stencil taking into account that field. These manipulations with nodes can significantly increase accuracy of a resulting numerical solution. This is an interesting and relatively unexplored field of research. More detailed examination of this issue is beyond the scope of this article and will be considered in our further work.

5.3 Example 3

In a final test we would like to demonstrate the universal applicability of the presented method. To illustrate the applicability of our FCT stabilized RBF-FD method to general initial data, a portrait of the famous mathematician Carl Friedrich Gauss* is rotated around the point $P = (0.5, 0.5)^T$ in a counterclockwise direction. The initial condition viewed from the front (left) and from the top (right) are shown in Figure 9 for $N = 200 \times 200$ spatial data nodes.

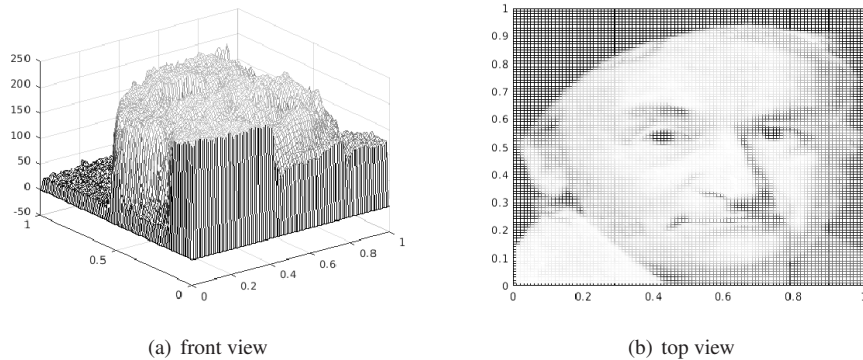


Figure 9: Initial condition.

Now the portrait of Gauss is rotated by $\alpha = 2\pi$ and the intermediate results for the pure u_{pure} , diffusive u_{diff} , and FCT-stabilized u_{stab} numerical solutions are displayed in Figure 10-12.

*extract from a portayal from Gottlieb Biermann, 1887; Link: https://commons.wikimedia.org/wiki/File:Carl_Friedrich_Gauss.jpg

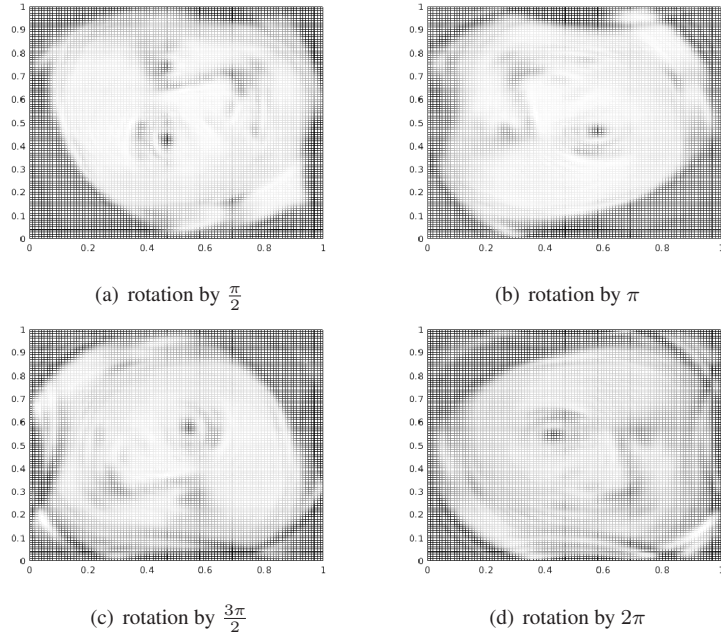


Figure 10: Pure u_{pure} numerical solution and various time instances.

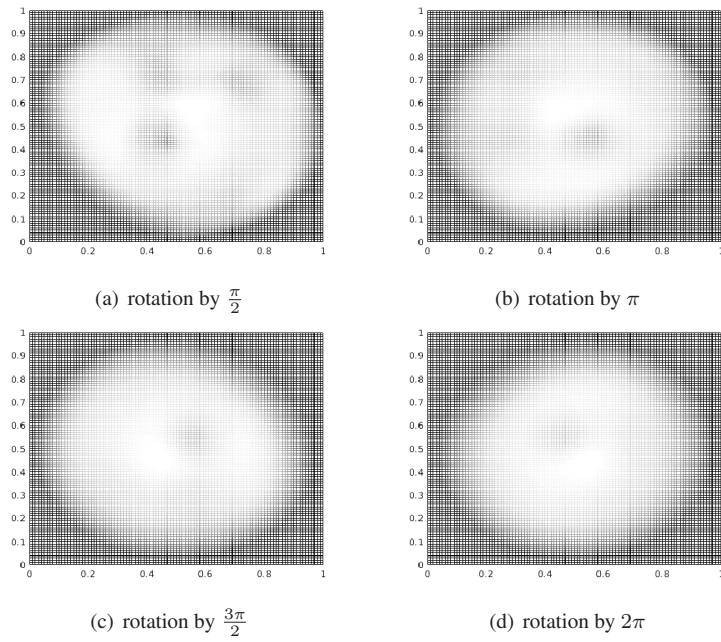


Figure 11: Diffusive u_{diff} numerical solution and various time instances.

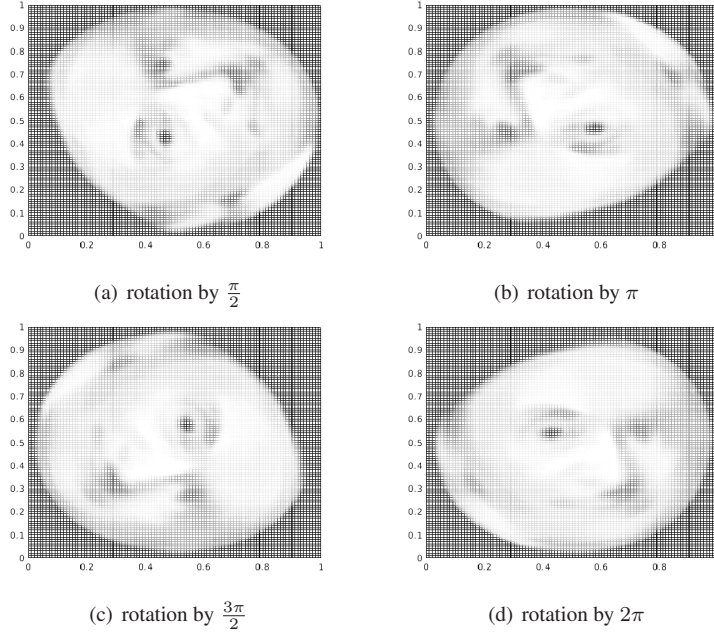


Figure 12: FCT-stabilized u_{stab} numerical solution and various time instances.

After a full rotation by using pure-, diffusive- and FCT-stabilized approaches, we compare the resulting numerical solutions. Oscillations of u_{pure} become noticeable in near-to-face areas. The diffusive property of u_{diff} is so strong, that the complete face is smeared out and becomes unrecognizable. However, u_{stab} shows no oscillatory behavior and the level of smearing remains low: significant areas as, e.g. eyes or the nose, are recognizable. This shows that the proposed stabilized method is able to approximate the problem for an initial condition with many fine-scale features which are difficult to capture without generating numerical artifacts. We think that the quality of a numerical solution will improve as more adaptive data nodes are employed to achieve higher resolution.

6 Conclusion

In the present article we introduced an FCT stabilized Radial Basis Function (RBF)-Finite Difference (FD) scheme for the numerical solution of partial differential equations (PDEs) of reaction-diffusion type with dominating convection terms. The proposed methodology is fully multidimensional and applicable to arbitrary placements of scattered data nodes. The scheme resolves steep gradients of the numerical solution without excessive smearing and satisfies the discrete maximum principle.

The proposed method has a general-purpose nature: it can be applied not only to convective terms of the form $\mathbf{v} \cdot \nabla u$, but also to more general convection-, resp. advection-like, operators, see, e.g. numerical simulation of chemotaxis equations, tumor-growth models, pattern formation in biology, etc. [28, 29]. We also showed that by using the level set methodology, one can extend the proposed method to surface-PDEs. Thus, offering a promising alternative to the currently existing stabilization methods, such as upwind-like

methods, methods which use more nodes in regions of steep gradients and methods where some amount of hyperviscosity is added into the system.

Our numerical results confirm the reliability of the proposed computational framework in terms of numerical convergence and capturing of typical/expected solution profiles. We have thus developed an FCT stabilized RBF-FD approach that can be employed for practical applications that involve PDEs in domains and on stationary surfaces.

The framework has a straightforward extension to three dimensional models which is mandatory when considering real-life applications, though some computational and code optimization are required, since the computational and analytical complexity significantly increases in the three dimensional case.

References

- [1] G. A. Barnett, N. Flyer and L. J. Wicker, *An RBF-FD polynomial method based on polyharmonic splines for the Navier-Stokes equations: Comparisons on different node layouts*, Journal of Computational Physics, arXiv:1509.02615, (2015).
- [2] N. Flyer, G. A. Barnett and Louis J. Wicker, *Enhancing finite differences with radial basis functions: Experiments on the Navier-Stokes equations*, Journal of Computational Physics **316**, (2016), pp. 39–62.
- [3] T. Cecil, J. Qian and S. Osher, *Numerical methods for high dimensional Hamilton-Jacobi equations using radial basis functions*, Journal of Computational Physics **196**, (2004), pp. 327-347.
- [4] O. Davydov and D. T. Oanh, *Adaptive meshless centres and RBF stencils for Poisson equation*, Journal of Computational Physics **230**, (2011), pp. 287–304.
- [5] O. Davydov and D. T. Oanh, *On optimal shape parameter for Gaussian RBF-FD approximation of Poisson equation*, Comput. Math. Appl. **62**, (2011), pp. 2143–2161.
- [6] O. Davydov and R. Schaback, *Error Bounds for Kernel-Based Numerical Differentiation*, Numer. Math. **132**, (2016), pp. 243–269.
- [7] G. Dziuk and C.M. Elliott, *An Eulerian approach to transport and diffusion on evolving implicit surfaces*, Comput Visual Sci. **13**, (2010), pp. 17–28.
- [8] G. Dziuk, C.M. Elliott, *A fully discrete evolving surface finite element method*, SIAM J. Numer. Anal. **50**, no. 5 (2012), pp. 2677-2694.
- [9] G. Dziuk and C.M. Elliott, *Finite element method for surface PDEs*, Acta Numerica (2013), pp. 289-396, doi:10.1017/S0962492913000056.
- [10] C. M. Elliott, B. Stinner and C. Venkataraman, *Modelling cell motility and chemotaxis with evolving surface finite elements*, J. R. Soc. Interface, PMID: 22675164, pp. (2012).
- [11] Edward J. Fuselier, Grady B. Wright, *A High-Order Kernel Method for Diffusion and Reaction-Diffusion Equations on Surfaces*, Journal of Scientific Computing **56**, Issue 3 (2013), pp. 535-565.
- [12] G. E. Fasshauer, *Meshfree Approximation Methods with Matlab*, Interdisciplinary Mathematical Sciences - Vol. 6. World Scientific Publishers, Singapore (2007).

- [13] N. Flyer and B. Fornberg and V. Bayona and G. A. Barnett, *On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy*, Journal of Computational Physics **321**, (2016), pp. 21–38.
- [14] B. Fornberg and E. Larsson and N. Flyer, *Stable computations with Gaussian radial basis functions*, SIAM J. Sci. Comput. **33**, no. 2, (2011), pp. 869–892.
- [15] Y. T. Gu and G. R. Liu, *Meshless techniques for convection dominated problems*, Comput. Mech. **38**, (2006), pp. 171–182.
- [16] Volker John, Petr Knobloch and Julia Novo, *Finite elements for scalar convection-dominated equations and incompressible flow problems: a never ending story?*, Computing and Visualization in Science, DOI: 10.1007/s00791-018-0290-5, (2017).
- [17] D. Kuzmin and S. Turek, *Flux correction tools for finite elements*, J. Comput. Phys. **175**, (2002), pp. 525–558.
- [18] D. Kuzmin and M. Möller, *Algebraic flux correction I. Scalar conservation laws*, in: D. KUZMIN, R. LÖHNER, S. TUREK (Eds.), *Flux-Corrected Transport: Principles, Algorithms, and Applications*, Springer, Berlin, (2005), pp. 155–206.
- [19] D. Kuzmin, *Explicit and implicit FEM-FCT algorithms with flux linearization*, J. Comput. Phys. **228**, (2009), pp. 2517–2534.
- [20] E. Larsson and E. Lehto and A. Heryudono and B. Fornberg, *Stable computation of differentiation matrices and scattered node stencils based on Gaussian radial basis functions*, SIAM Journal on Scientific Computing **35**, no. 4, (2013), pp. A2096–A2119.
- [21] D. T. Oanh and O. Davydov and H. X. Phu, *Adaptive RBF-FD Method for Elliptic Problems with Point Singularities in 2D*, Applied Mathematics and Computation **313**, (2017), pp. 474–497.
- [22] M. A. Olshanskii, A. Reusken and J. Grande, *A Finite Element method for elliptic equations on surfaces*, SIAM J. Numer. Anal. **47**, no. 5 (2009), pp. 3339–3358.
- [23] M. A. Olshanskii, A. Reusken and X. Xu, *An Eulerian space-time Finite Element method for diffusion problems on evolving surfaces*, SIAM J. Numer. Anal. **52**, no. 3 (2014), pp. 13541377.
- [24] M.A. Olshanskii, A. Reusken, *Error analysis of a spacetime finite element method for solving PDEs on evolving surfaces*, SIAM J. Numer. Anal. **52**, no. 4 (2014), pp. 20922120.
- [25] A. Rätz and A. Voigt, *PDE’s on surfaces — a diffuse interface approach*, Comm. Math. Sci. **4**, no. 3 (2006), pp. 575–590.
- [26] Varun Shankar, Grady B. Wright, Robert M. Kirby and Aaron L. Fogelson, *A Radial Basis Function (RBF)-Finite Difference (FD) Method for Diffusion and Reaction-Diffusion Equations on Surfaces*, Journal of Scientific Computing **63**, Issue 3 (2015), pp. 745–768.
- [27] A. Sokolov, R. Ali and Stefan Turek, *An AFC-stabilized implicit finite element method for partial differential equations on evolving-in-time surfaces*, Journal of Computational and Applied Mathematics **289**, (2015), pp. 101–115.

- [28] R. Strehl, A. Sokolov, D. Kuzmin and S. Turek, *A flux-corrected finite element method for chemotaxis problems*, Computational Methods in Applied Mathematics **10**, no. 2 (2010), pp. 118–131.
- [29] R. Strehl, A. Sokolov, D. Kuzmin, D. Horstmann and S. Turek, *A positivity-preserving finite element method for chemotaxis problems in 3D*, Journal of Computational and Applied Mathematics **239**, (2013), pp. 290–303.
- [30] N. Trask, M. Perego, P. Bochev, *A high-order staggered meshless method for elliptic problems*, SIAM Journal on Scientific Computing **39**, no. 2, (2016).
- [31] G. Wright, *Radial Basis Function: Numerical and Analytical Developments*, Phd thesis, University of Colorado (2003).
- [32] S. T. ZALESK, *Fully multidimensional flux-corrected transport algorithms for fluids*, J. Comput. Phys. **31**, (1979), pp. 335–362.