Journal of Computational Physics 498 (2024) 112667



Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp





Very high-order finite difference method on arbitrary geometries with Cartesian grids for non-linear convection diffusion reaction equations

S. Clain [a,*], D. Lopes [b], Rui M.S. Pereira [c,d], Paulo A. Pereira [e,d]

[a] Center of Mathematics of the FCTUC, Largo D. Dinis, 3000-143 Coimbra, Portugal
[b] 2Ai – Applied Artificial Intelligence Laboratory, Campus do IPCA, Vila Frescainha S. Martinho, 4750-810 Barcelos, Portugal
[c] Centre of Physics UM - UP, Campus de Gualtar, 4710 - 057 Braga, Portugal
[d] Department of Mathematics, University of Minho, Campus de Azurém, 4080-058 Guimarães, Portugal
[e] Centre of Mathematics, Campus de Gualtar, 4710 - 057 Braga, Portugal

ARTICLE INFO

Keywords:
Very high-order
Finite difference
Arbitrary geometries
ROD polynomial
Immersed boundary

ABSTRACT

An arbitrary order finite difference method for solving non-linear convection Diffusion Reaction equations in curved boundary domains with Cartesian grid is proposed. Ghost points' values are determined with the Reconstruction Off-Site Data based on a polynomial interpolation using the least square method with constraints to enforce the boundary conditions. We propose a second-, fourth-, and sixth-order schemes for linear non-constant coefficients problem in both the conservative and non-conservative scalar equations. Extensions to non-linear scalar problems and systems are then implemented while preserving the optimal orders. Numerical simulations are carried out to provide evidence about the convergence order and the stability of the method.

1. Introduction

High order finite difference methods on Cartesian grids are a key player in High-Performance Computing due to the simplicity, low memory storage and efficiency [1] of the traditional five-point (in 2D) or seven-point (in 3D) schemes together with high scalability when using dimensional splitting techniques such as the Alternative Direction Implicit (ADI) method [2–5]. Nevertheless, a critical issue, when dealing with very high order approximations, is to handle curved boundary domains that do not match with the grid and the design of numerical schemes that preserve the method order [6]. Many numerical technologies have been proposed to treat boundary conditions on non-polygonal domains, and are still ongoing challenges. The Immersed Boundary Method (IMB) was one of the first approaches [7] and gave rise to several families of immersed techniques: Immersed Interface Method (IIM) [8], matched interface and boundary (MIB) [9]. Alternatively, the introduction of curvilinear coordinates was proposed to remapping the curved domain onto a simple quadrilateral using local transformations [10,11].

Ghost-cells or ghost-points close to the boundary provide the necessary information for the internal equation's discretisation. We evaluate their values with additional algebraic equations deriving from the boundary conditions [12–14] with interpolation/extrapolation procedures leading to small linear systems (one for each ghost point). Accuracy and stability are the major issues. Higher accuracy requires larger stencils of computation (fluid) cells, hence larger condition number, and the local linear systems may in-

* Corresponding author.
E-mail address: clain@mat.uc.pt (S. Clain).

https://doi.org/10.1016/j.jcp.2023.112667
Received 14 June 2022; Received in revised form 8 October 2023; Accepted 20 November 2023

Available online 28 November 2023
0021-9991/© 2023 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

volve negative coefficients that would jeopardise the stability. The Ghost-point method, or Direct Boundary Condition Imposition [15], dates back to 2000s and by introducing a linear, bilinear or quadratic reconstruction that incorporates the boundary condition at a point on the physical boundary [16]. The most popular technique consists in selecting a ghost point G of the grid and computing the boundary intercept point I together with the Mirror point M (Image point) in the numerical domain. The Intercept point satisfies the property that GI is collinear to the boundary normal vector n at point I while the mirror point M is simply given by $GM = 2IM$. A reconstructed value for M is deduced from interpolation with the neighbouring points and a linear extrapolation using the points M and I provides a second-order approximation for the ghost point [17–20].

For very high-order schemes *i.e.* strictly higher than the second-order one, on curved domains, it is a more challenging task and very few publications have tackled such an issue. We can distinguish two different approaches. On the one hand, an extension of the extrapolation/interpolation following the normal direction or the axis direction [21–23] to provide fourth- or fifth-order of accuracy. On the other hand, some authors [24–27] use a polynomial interpolation over a stencil constituted with interior points around the ghost point. The boundary condition is determined at the intercept point and added to a cost function. A constraint-free least squares minimisation is then carried out to provide the polynomial function we use to set the value at the ghost point.

More recently, the least square interpolation with restriction has been proposed in the context of the finite volume method to achieve very high-order of accuracy on curved boundary domains [28]. The Reconstruction Off-site Data (ROD) method consists in determining a polynomial approximation over a stencil of cells, together with restrictions given by boundary conditions located on the physical boundary, which are strongly fulfilled. The method has been applied for general boundary conditions [29], interior curved interfaces between two subdomains [30], three-dimensional geometries [31], and anisotropic meshes [32]. The method has been revisited to be adapted to the finite difference context in [33] for the steady-state convection diffusion case, and a substantial variation has been proposed in [34] using the ghost points methodology.

In Luo et al. [25], the method requires the determination of the intercept point I to build the polynomial reconstruction centred on I. Secondly, the value ϕ_{GC} of the ghost cell is introduced in the functional to provide a linear relation between ϕ_{GC} and the rest of the stencil. Then, the boundary condition, evaluated at the intercept point, provides the last relation to determine ϕ_{GC}. Our approach is quite different since: (1) we use a collar of *a priori* given points of the boundary and no intercept point is used; (2) our polynomial, centred at the ghost point, does not include the ϕ_{GC} in the functional; (3) we add the boundary conditions (one or more) via additional restrictions on the least squares minimisation procedure that we solve with Lagrange multipliers. We highlight that our method does not require the knowledge of the whole interface since we do not use intercept points but the information of a collar of points all along the physical boundary.

Problems involving linear scalar convection diffusion with constant coefficients on curved domain have been tackled in [33]. The non-linear case for scalar problems or systems is now investigated. Non-linearity is treated with a fix-point method that requires the resolution of successive linear problems with non-constant coefficients. Conservative or non-conservative formulation gives rise to new numerical schemes near the boundary together with the ROD reconstructions to achieve the optimal order. Consequently, discretisation of the coefficients is needed and has to be carefully treated to preserve the order. In particular, when no analytical expression of the coefficients is available, typically the case of nonlinear coefficient, a specific strategy based on local reconstruction both for the solution and the coefficients is required.

The paper is organised as follows. In section 2, we introduce the notations, while section 3 is dedicated to the ROD procedure. The generic linear convection-diffusion problem with space dependent coefficients is tackled in section 4 where we consider the conservative and non-conservative forms. A particular interest is the global accuracy preservation up to the sixth-order with curved boundary domain for one of the proposed methods. Section 5 is dedicated to the non-linear problems, where we present different benchmarks for the accuracy and stability by considering a large class of models (heat transfer, electromagnetism) and boundary conditions (Dirichlet, Neumann, non-linear Robin conditions). In all benchmarks, we consider a uniform grid. We draw some conclusions and perspectives in section 6.

2. Notations and discretisation

Let $\Omega \subset \mathbb{R}^2$ be an open-bounded set with Lipschitz, smooth piecewise, boundary $\partial\Omega$. We denote by $\Lambda = [0, L_x] \times [0, L_y] \subset \mathbb{R}^2$ a rectangle that contains the subdomain $\Omega \subset \Lambda$. For I, J, two given integer numbers, we set $\Delta x = L_x/I$, $\Delta y = L_y/J$ the mesh sizes while $x_{i-1/2} = i\Delta x$ and $y_{j-1/2} = j\Delta y$ are the nodes of the grid. Moreover, we denote the cell $C_{i,j} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$ of centre $X_{i,j} = (x_i, y_j)$ and $N^1_{i,j}$, $\cdots$, $N^4_{i,j}$ stand for the four nodes of the cell $C_{i,j}$.

2.1. Numerical domain

The mesh $\mathcal{M}_\Delta = \mathcal{M}_\Delta(\Lambda)$ gathers the active cells of the domain Λ, *i.e.* the quadrangle which all the nodes belong to Ω. We then define the active grid associated to the domain Ω by,

$$\mathcal{M}_\Delta(\Omega) = \left\{ C \in \mathcal{M}_\Delta, \text{ such that } N^1, \cdots, N^4 \in \Omega \right\}, \qquad \Omega_\Delta = \bigcup_{C \in \mathcal{M}_\Delta(\Omega)} C,$$

where Ω_Δ stands for the numerical domain. Cells $C \in \mathcal{M}_\Delta(\Omega)$ are tagged active or computational cells since they correspond to the computational domain (see Fig. 1, left panel).

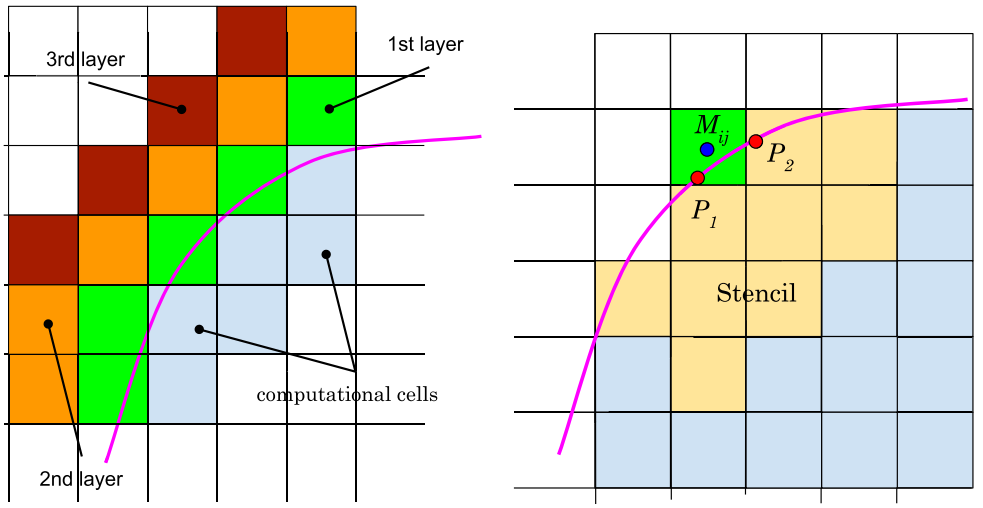


Fig. 1. (left panel) Physical and numerical domain. The three layers are plotted in green (1st layer), orange (2nd layer) and brown (3rd layer). (right panel) Stencil with the two collar points to compute the ROD polynomial associated to the ghost cell depicted in green. (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

2.2. Ghost cells

To define the ghost cells, we introduce the rook distance between two cells with

$$d_r(C_{i,j}, C_{i',j'}) = \begin{cases} |i-i'| \text{ if } j = j', \\ |j-j'| \text{ if } i = i', \\ +\infty \text{ otherwise,} \end{cases}$$

and the distance between a cell and domain Ω_Δ with

$$d_r(C_{i,j}, \Omega_\Delta) = \min_{C_{i',j'} \in \mathcal{M}_\Delta(\Omega)} d_r(C_{i,j}, C_{i',j'}).$$

The first layer of ghost cells is then characterised by cells with a distance to Ω_Δ equal to 1. We shall denote $\mathcal{L}_1(\Omega_\Delta) = \{C_{i,j} \in \mathcal{M}_\Delta,\ d_r(C_{i,j}, \Omega_\Delta) = 1\}$ with a similar definition for the second layer $\mathcal{L}_2(\Omega_\Delta)$ and third layer $\mathcal{L}_3(\Omega_\Delta)$ of ghost cells. Finally, for a given cell $C_{i,j}$ and $\ell \in \mathbb{N}$, we define the ℓ-stencil $\mathcal{V}_\ell(C_{i,j}) \subset \mathcal{M}_\Delta(\Omega)$, as the list of the ℓ-closest active cells of $\mathcal{M}_\Delta(\Omega)$ to $C_{i,j}$ (see Fig. 1, right panel).

2.3. Collar

The boundary is discretised with just a set of K points $P_k \in \partial\Omega$, $k = 1, \cdots, K$ we name the collar $\mathcal{C}_\Delta$ (see Fig. 1). We only assume that $|P_k P_{k+1}| = O(\Delta s)$, where $P_{K+1} = P_1$ by convention and, Δs is the characteristic length of the collar. We shall take $\Delta s = \sqrt{\Delta x \Delta y}$ to guarantee the original accuracy of the numerical schemes. Additionally, the collar also provides the outward normal vector n_k of the physical boundary at points P_k.

Remark. Notice that our method does not require a full description of the boundary (parametrised curve, level set function) but just a set of points, situated on the physical boundary. Indeed, up to our knowledge, all the strategies are based on the projections of ghost points on the physical boundary (coordinate direction [35] or normal direction [22,23]) and require an analytical representation to calculate such projections.

2.4. Approximations

Let $\phi = \phi(x, y)$ be a function defined on the domain Λ. We denote by $\mathbf{\Phi}[i, j]$ the approximations of $\phi(X_{i,j}) = \phi(x_i, y_j)$ we gather in matrix $\mathbf{\Phi} \in \mathbb{R}^{I \times J}$. Adopting a new single-index numbering $[i, j] \to k$, the matrix $\mathbf{\Phi}$ gives rise to two vectors: $\mathbf{\Phi}_{\text{IN}}$ corresponds to the values on the active cells that belong to $\mathcal{M}_\Delta(\Omega)$, while $\mathbf{\Phi}_{\text{GC}}$ gathers the values in the ghost cells of the different layers. The equation we consider throughout the paper is the convection-diffusion-reaction that can be read in the linear conservative form as,

$$\mathfrak{F}_c(\phi) = f - \nabla \cdot (u\phi - \kappa \nabla \phi) - r\phi = 0, \quad \text{in } \Omega,$$

where κ stands for the diffusion variable coefficient, and r is the reaction coefficient, $u=(u^x,u^y)$ is the velocity and f is the source term.

Since the coefficients are variable, the function $\kappa(X)$ is discretised with $K=(\kappa_{i,j})_{i,j}$ the matrix that gathers all the values of $K[i,j]=\kappa(X_{i,j})$ at the centre of the cells and, for the velocity, we denote by U^x, U^y the two matrices that gather the entries $U^x[i,j]=u^x(X_{i,j})$ and $U^y[i,j]=u^y(X_{i,j})$ respectively.

3. ROD operator

The Robin boundary condition encompasses general linear boundary conditions such as Dirichlet or Neumann ones.

$$\mathfrak{B}(\phi)=g-\alpha\phi-\beta\nabla\phi\cdot n=0,\quad \text{on } \partial\Omega, \tag{1}$$

with g a given function on the physical boundary $\partial\Omega$, α and β are real valued functions (possibly depending on x) and n is the outward normal vector on $\partial\Omega$. The Reconstruction Off-site Data (ROD) is a linear operator that, given the vector $\mathbf{\Phi}_{\text{IN}}$ associated to the computational cells and the boundary conditions prescribed on the collar, transfers the boundary information into the ghost cells, namely

$$\mathbf{\Phi}_{\text{IN}}\to\mathbf{\Phi}_{\text{GC}}=\text{R}(\mathbf{\Phi}_{\text{IN}};\mathfrak{B}). \tag{2}$$

3.1. A least square with constraints problem

We improve the technique detailed in [33] (section 3) by introducing a weighted loss functional to handle more difficult situations (corner, anisotropic stencil). Adopting the multi-index notation $\gamma=(\gamma_x,\gamma_y)$ with $|\gamma|=\gamma_x+\gamma_y$, and given a ghost cell $C_{i,j}\in\mathcal{L}_1(\Omega_\Delta)$, we define the polynomials $\pi\in\mathbb{P}_d(\mathbb{R}^2)$ of degree d, centred at $X_{i,j}$ evaluated at point $X\in\mathbb{R}^2$ as follows,

$$\pi(X)=\pi(X;X_{i,j})=\sum_{|\gamma|\le d}a_\gamma\left(\frac{x-x_i}{\Delta x}\right)^{\gamma_x}\left(\frac{y-y_j}{\Delta y}\right)^{\gamma_y}, \tag{3}$$

where a_γ is the vector of coefficients to determine.

Let $\mathcal{V}=\mathcal{V}_\ell(C_{i,j})$ be the stencil of size ℓ associated to cell $C_{i,j}$, and P_1 and P_2 two points of the collar chosen following the cone property (see [33], section 3.1). We seek the polynomial of degree d that locally approximates the function around $C_{i,j}$ and exactly satisfies the constraints at the two points on the boundary. Note that the number ℓ of cells has to be large enough to guarantee the computation of the polynomial. To take the proximity of the interpolated cells with the ghost cell into account, we define the weighted functional

$$\mathfrak{E}_{i,j}(\pi;\mathcal{V},\mathbf{\Phi})=\frac{1}{2}\sum_{X_{i',j'}\in\mathcal{V}}\omega_{ij;i'j'}\Big(\pi(X_{i',j'})-\mathbf{\Phi}[i',j']\Big)^2$$

that represents the weighted quadratic error between the polynomial representation centred at $X_{i,j}$ and the discrete values over the stencil $\mathcal{V}=\mathcal{V}(C_{i,j})$ constituted of active cells. When implementing, we consider the size of the stencil is twice the number of unknowns to determine, to make sure enough local information is available. The weights are computed using the kernel function $\rho_\tau(s)=e^{-(s/\tau)^2}$, where $\tau\approx O(\Delta x)$,

$$\omega_{ij;i'j'}=\rho_\tau(|X_{i,j}X_{i',j'}|)$$

and τ a scaling parameter. On the other hand, we prescribe that the polynomial reconstruction exactly satisfies the boundary condition at the two collar points and reads

$$\mathfrak{B}^r_{i,j}(\pi)=\alpha(P_r)\pi(P_r)+\beta(P_r)\nabla_X\pi(P_r)\cdot n_r-g(P_r)=0,\quad r=1,2.$$

Following [33] (section 3), we sketch the solution of the least square problem with the constraints. We drop indices i,j for simplicity and introduce the Lagrangian functional

$$\mathcal{L}(a,\lambda;\mathcal{V},\mathbf{\Phi})=\mathfrak{E}(\pi;\mathcal{V},\mathbf{\Phi})+\sum_{r=1,2}\lambda_r\mathfrak{B}^r(\pi),$$

where vector a gathers the polynomial coefficients of $\pi(X)=\pi(X;a)$. Minimum with restriction is achieved with $\overline{a},\overline{\lambda}$ such that we have both $\nabla_a\mathcal{L}(\overline{a},\overline{\lambda})=0$ and $\nabla_\lambda\mathcal{L}(\overline{a},\overline{\lambda})=0$. Introducing a local unidimensional index $\mu=0,\cdots,\ell$ for the cells of the stencil around the target cell C_0, and $\nu=1,\cdots,m=(d+1)(d+2)/2$ that maps with the multi-index $\gamma\to\nu=\nu(\gamma)$, the minimisation reads

$$\begin{bmatrix} MWM^{\text{t}} & B^{\text{t}}\\ B & 0\end{bmatrix}\begin{bmatrix}\overline{a}\\ \overline{\lambda}\end{bmatrix}=\begin{bmatrix} MW\Phi\\ g\end{bmatrix}, \tag{4}$$

with $M\in\mathbb{R}^{m\times(\ell+1)}$, $W\in\mathbb{R}^{(\ell+1)\times(\ell+1)}$ given by

$$M = \underbrace{\begin{bmatrix} \cdots & \cdots & \cdots \\ \cdots & \dfrac{(X_\mu - X_0)^\gamma}{(\Delta x)^{\gamma_x}(\Delta y)^{\gamma_y}} & \cdots \\ \cdots & \cdots & \cdots \end{bmatrix}}_{\ell+1}, \qquad W = \underbrace{\begin{bmatrix} \ddots & 0 & 0 \\ 0 & \omega_\mu & 0 \\ 0 & 0 & \ddots \end{bmatrix}}_{\ell+1},$$

and $B \in \mathbb{R}^{2\times m}$ given by the entries for $|\gamma| \le d$, $r = 1, 2$

$$B[\nu(\gamma), r] = \alpha \frac{(P_r - X_0)^{(\gamma_x,\gamma_y)}}{(\Delta x)^{\gamma_x}(\Delta y)^{\gamma_y}} + \beta\gamma_x \frac{(P_r - X_0)^{(\gamma_x-1,\gamma_y)}}{(\Delta x)^{\gamma_x}(\Delta y)^{\gamma_y}} \cdot n_x(P_r) + \beta\gamma_y \frac{(P_r - X_0)^{(\gamma_x,\gamma_y-1)}}{(\Delta x)^{\gamma_x}(\Delta y)^{\gamma_y}} \cdot n_y(P_r).$$

3.2. ROD benchmarking

3.2.1. Introduction to the benchmarks

We propose a series of numerical experiments to assess the quality of the ROD method by computing the norm-max error between the approximation and the exact solution. Before going into details, we shall explain the organisation of the tests and the issues we tackle. Indeed, the reconstruction involves several parameters (kernel choice, kernel size, order method, smooth and rough interface) and we pretend to provide a careful study of all the situations.

- **The method order.** We have implemented three reconstructions: ROD2 with a quadratic reconstruction, ROD4 with polynomial of degree 4, and ROD6 for the polynomial of degree 6. The point is to check the convergence order for the L^1-error.
- **Conditioning.** The ROD method requires the solution of a dense linear system, which the conditioning depends on the relative position of the cells with the ghost cell and the collar points. The condition number deeply affects the accuracy of the method. That can be seen in the number of decimal places that are correct, as the size of the grid increases.
- **The stencil.** The construction of the stencil is a function of the degree and the boundary regularity. When dealing with a corner, the number of available active cells in the neighbourhood is limited and one has to pick up cells far away, leading to a dramatic increase in the conditioning.
- **The kernel.** Weighted functional is a powerful tool to reduce the condition number. It is a common technique in the SPH community, and we shall experiment with different popular kernels such as the Gaussian, cubic and quintic kernel. Moreover, the scaling parameter has to be wisely chosen to optimise the reconstruction, *i.e.* to minimise the condition number. Notice that the best choice of τ depends on the polynomial degree of the reconstruction.

To assess the approximation accuracy for the L^1-norm and ability to handle complex boundaries, numerical simulations are carried out on a circular boundary domain (Fig. 2 left panel) and a nozzle-like domain with corners (Fig. 2 right panel).

For smooth boundaries, we easily pick up active cells close to the ghost cell to build the stencil with a uniform spatial distribution, whereas the stencil at the corner is restricted to a cone with a smaller aperture, leading to a degradation of the scheme because some of the stencil cells will be further away from the ghost cell.

To compute the errors in L^1-norm, we use the regular function $\phi(x, y) = \exp(x + 2y)$ and set the exact values in all the active cells. On the other hand, we prescribe the Dirichlet condition all around the physical boundary. In all the configurations, the collar is constituted of a uniform, distributed set of points with $\Delta s \approx \Delta x$.

The benchmarks aim at quantifying the quality of the reconstruction regarding the choice of the kernel parameter τ by computing the condition number associated to the local linear reconstruction matrix MWM^t. On the one hand, the uniform kernel corresponds to unit weight for all nodes (*i.e.*, $\tau = +\infty$). On the other hand, we take three representative values in function of the space parameter $\tau = \Delta x,\ 2\Delta x,\ 3\Delta x$ providing a small, medium and large kernel support respectively. Since MWM^t depends on the ghost cell where the reconstruction is achieved, we only report the average of the condition number. Accuracy and stability are also assessed in all the situations to determine the best option in function of the scheme order.

3.2.2. ROD in a circular domain

Fig. 3 shows the errors (left panel), and the condition number average (right panel) of the reconstruction matrices for the smooth boundary domain given by a circle centred at the origin with radius 0.8 as depicted in the left panel of Fig. 2. We have carried out the simulation for the four situations: the uniform kernel and the Gaussian kernel with parameter $\tau = \Delta x$, $\tau = 2\Delta x$ and $\tau = 3\Delta x$. We recover the optimal second-order method in all cases but $\tau = \Delta x$ provide the smallest errors. Condition numbers are almost constant regarding the mesh size but strongly different with the kernel size parameter τ. Particularly, the uniform kernel provides the worst conditioning and, by that way, highlights the interest in using a kernel.

The weighted minimisation provides better approximation and managed to reduce the condition number of the matrix. We observe that the smallest kernel support with $\tau = \Delta x$ is very well adapted for second-order reconstructions. The condition number and the error are halved when compared to the uniform kernel. Notice that the kernel usage is quite efficient when dealing with coarse grids (30, 60 and 120) while the benefits are less obvious when dealing with very fine grid. The boundary appears as a smooth interface when the local curvature radius is ten times larger than the space parameter Δx. Indeed, the interface is locally a circle with radius larger than $10\Delta x$ while the distance between two successive collar points is of order Δx. We then discretise the piece of

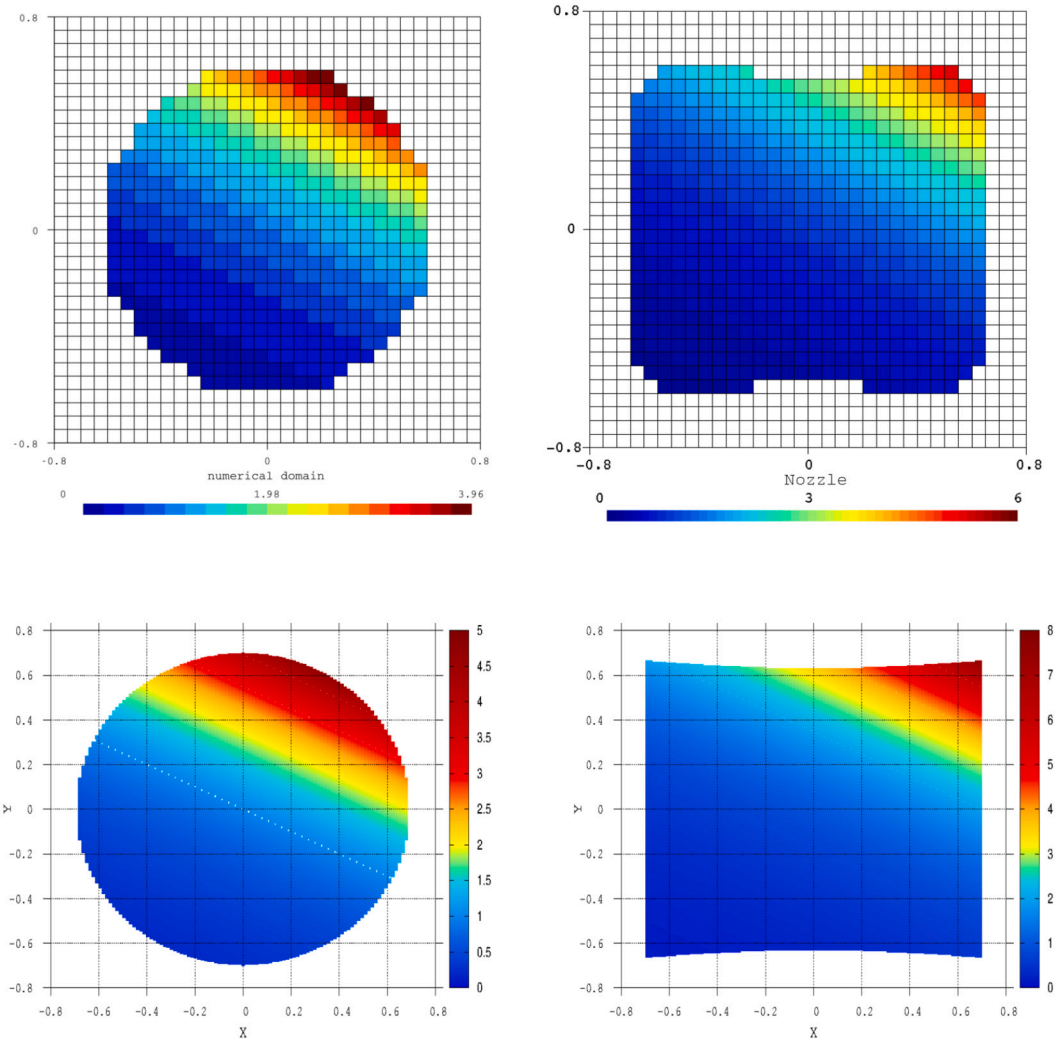


Fig. 2. Numerical domains for the order assessments. The smooth domain is the disk of radius 0.8 (left column) and the nozzle-like configuration presents four corners (right column) the first row depicts the domain with a 30×30 grid, while the second row plots the same domain with the highest resolution of 120×120.

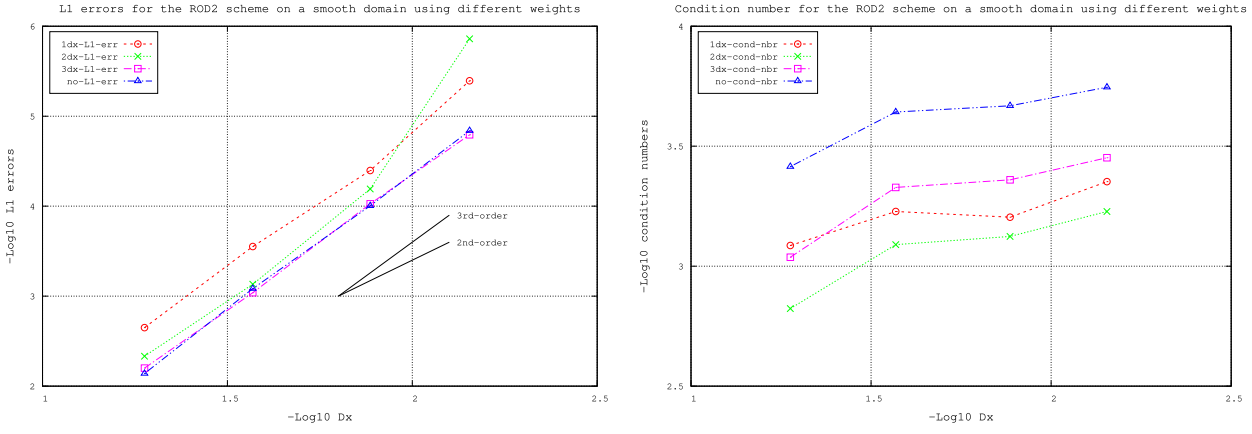


Fig. 3. L^1-errors (left panel) and conditioning numbers (right panel) for the ROD2 schemes on a smooth domain with different scales (uniform, $\tau = \Delta x$, $\tau = 2\Delta x$, $\tau = 3\Delta x$).

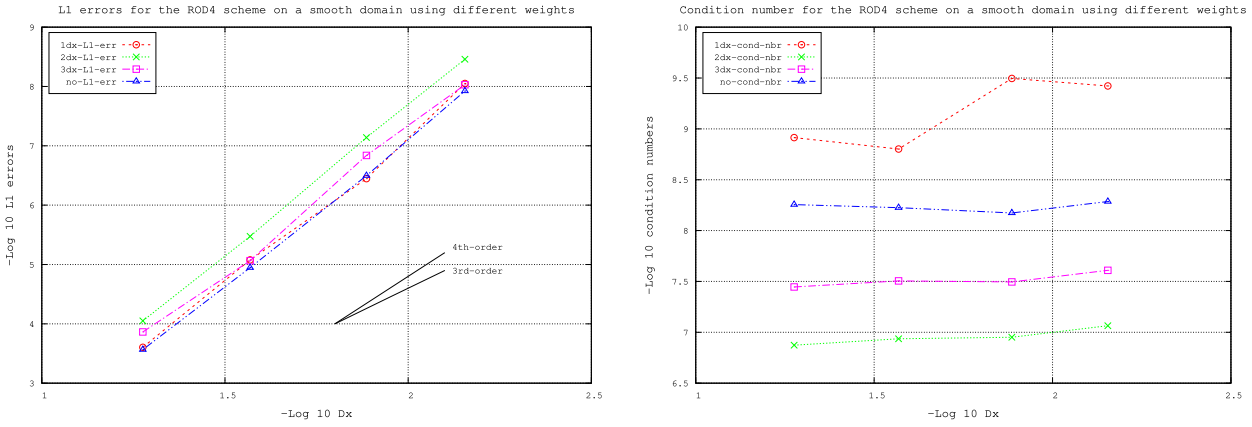


Fig. 4. L^1-errors (left panel) and conditioning numbers (right panel) for the ROD4 schemes on a smooth domain with different scales (uniform, $\tau = \Delta x$, $\tau = 2\Delta x$, $\tau = 3\Delta x$).

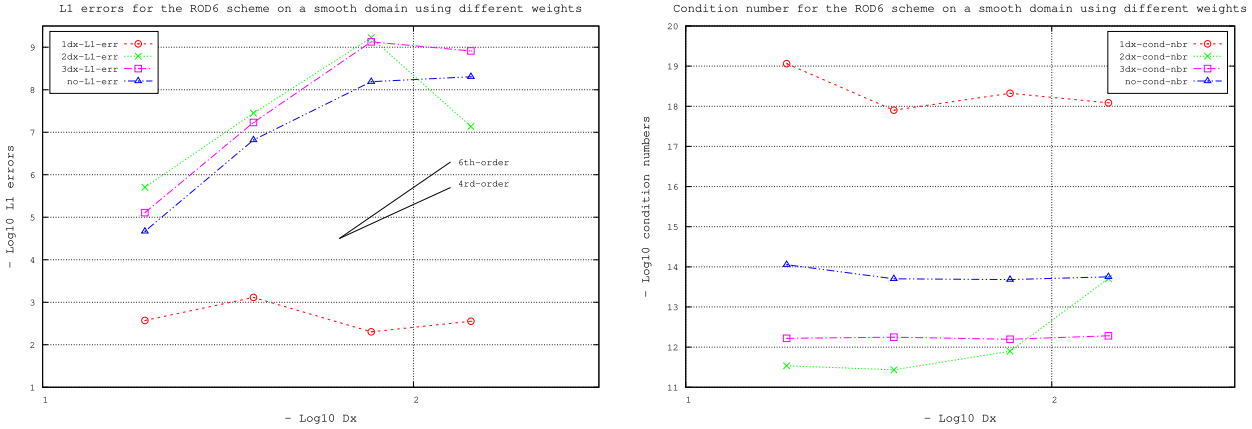


Fig. 5. L^1-errors (left panel) and conditioning numbers (right panel) for the ROD6 schemes on a smooth domain with different scales (uniform, $\tau = \Delta x$, $\tau = 2\Delta x$, $\tau = 3\Delta x$).

boundary with enough number of collar points to provide a fine approximation. On the contrary, if the local curvature is lower than some few Δx, (let's say $2\Delta x$), we do not manage to catch the whole geometry since the structure has a smaller scale than the grid.

We now turn to the ROD4 case and display in Fig. 4 the errors, convergence rates (left panel) together with the condition numbers deriving from the different kernels (right panel). We first notice that, the conditioning does not depend on the grid size. We underline the dramatic growth of the condition number (around 4 to 5 magnitudes), regarding the kernel parameter, that is responsible for a loss of precision due to the truncation of the `double` format. Kernel parameter $\tau = 2\Delta x$ provides the best results for the fourth-order. Notice that by using this kernel, the condition number is reduced by an order of magnitude and the error is halved. Smaller radius such as $\tau = \Delta x$ provokes an increase in the condition number and shall be kept away.

At last, we plot in Fig. 5 the errors (left panel) and the condition number (right panel) for the ROD6 method for different kernel parameter values. The huge growth of the conditioning derives from the stencil expansion, where one faces a trade-off between the method order and the convergence limitation of the scheme inherent to the `double` precision. For the finest grids, $I = 120,240$ the accuracy stalls due to the very high condition number and represents a serious issue. For small kernel parameter $\tau = \Delta x$, the errors and convergence rates are considerable because the support of the kernel is too small and provides insignificant weights for the second and third layers of cells that belong to the stencil. The intermediate radius and largest radius considerably reduce the errors, with a slight advantage for $\tau = 2\Delta x$ where the condition number is reduced by two orders of magnitude and the errors in the first ring are reduced by an order of magnitude. From this benchmark, we were able to see that the use of a weighted kernel for the cells in the stencil in the ROD process leads us to smaller condition numbers and smaller errors. The greater the degree of the polynomial reconstruction, the greater that τ should be.

3.2.3. Kernels' comparison

The Compact Kernel function to compute the weights is widely used in Smooth Particles Hydrodynamic method [36] or Moving Least Square approximation [37]. The three most popular Kernels are the third order cubic, fifth-order quintic and Gaussian representation of the kernel $s \to \rho_\tau(s)$ with τ the scaling factor. To assess the accuracy and efficiency of the different kernels, we have

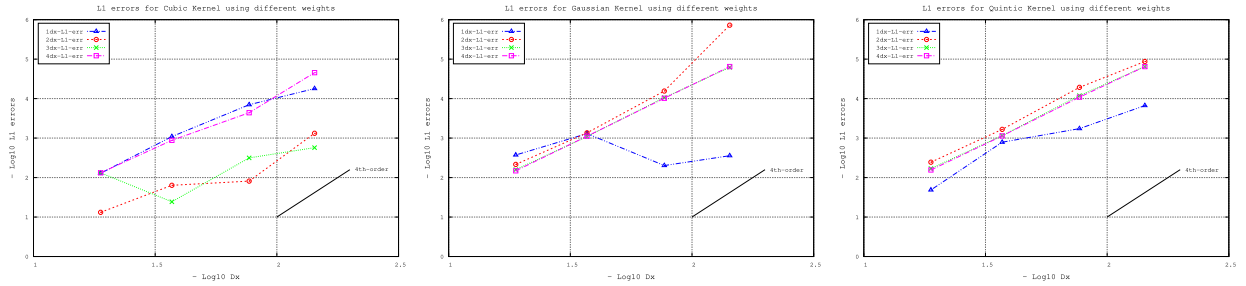


Fig. 6. L^1-errors and convergence order for the first ring with a 2nd order scheme using the cubic (left), quintic (middle) and Gaussian (right) Kernel with different values of the kernel parameter τ.

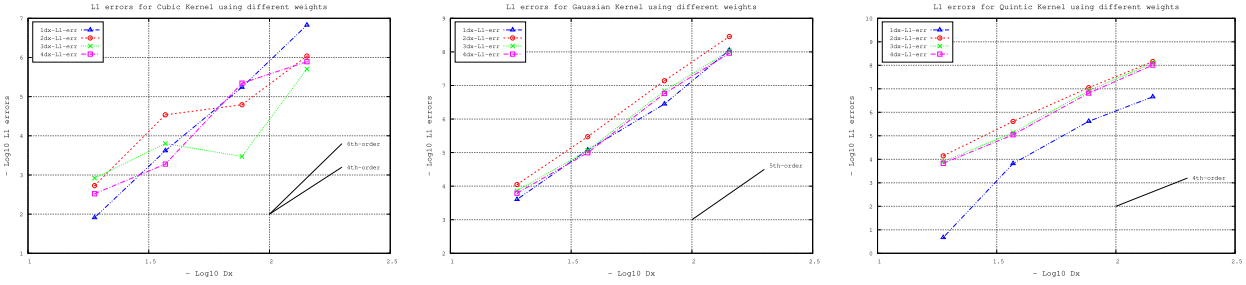


Fig. 7. L^1-errors and convergence order for the first ring with a 4th order scheme using the cubic (left), quintic (middle), Gaussian (right) Kernel with different values of the scaling parameter τ.

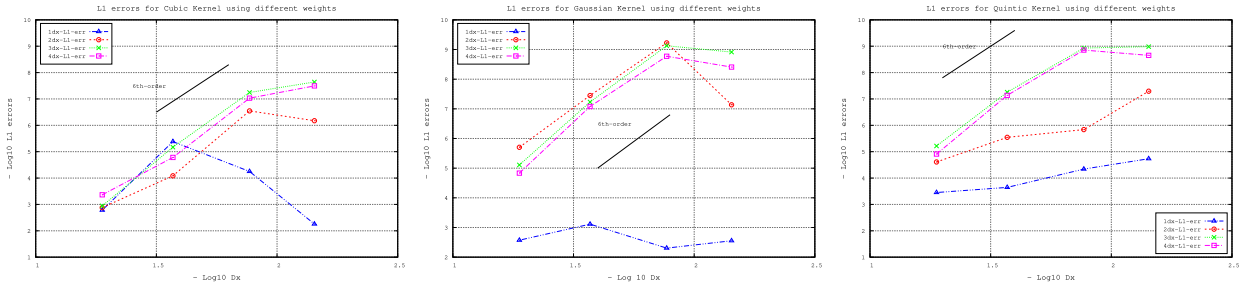


Fig. 8. L^1-errors and convergence order for the ROD6 method using the cubic (left), quintic (middle), Gaussian (right) Kernel with different values of the kernel parameter τ.

carried out the different ROD of degree 2, 4 and 6 over the ghost cells all along the circle boundary, the exact solution being given in the active cells.

We display the errors of the ROD2 in Fig. 6 using the cubic (left), cubic (middle) and Gaussian (right) kernel for different kernel parameter values of τ to compute the weights. The cubic kernel provides the worst results, and is less competitive regarding the quintic and the Gaussian kernel. The Quintic kernel produces similar error than the Gaussian one. The relative advantages change in function of the order and Δx.

Errors and convergence orders of the ROD4 are displayed in Fig. 7 for the different kernels. Once again, the cubic kernel provides the worst results, whereas quintic and Gaussian kernels present the same behaviour. No particular advantages, in terms of accuracy, seem to indicate a more efficient kernel. We just noticed that the Gaussian kernel may sustain a good convergence order even with smaller scaling parameters (obvious for $\tau = \Delta x$).

At last, we consider the ROD6 version and plot in Fig. 8 the errors and convergence order for the ROD6. The cubic kernel provides the worst results, while the Gaussian kernel presents a slight advantage regarding the quintic kernel. At that stage, the choice depends on the trade-off between accuracy and computational efficiency. The quintic kernel is very fast due to its polynomial expression regarded to the evaluation of the exponential function. On the other hand, the Gaussian kernel provides a slightly better accuracy.

3.2.4. Convergence analysis for the three layers

All the convergence tests proposed above just consider the L^1 error for the first layer/ring of ghost points but we have two other rings r2 and r3 as displayed in Fig. 1 (left panel). Adopting the quintic kernel, we report in Fig. 9 the L^1-errors and convergence for

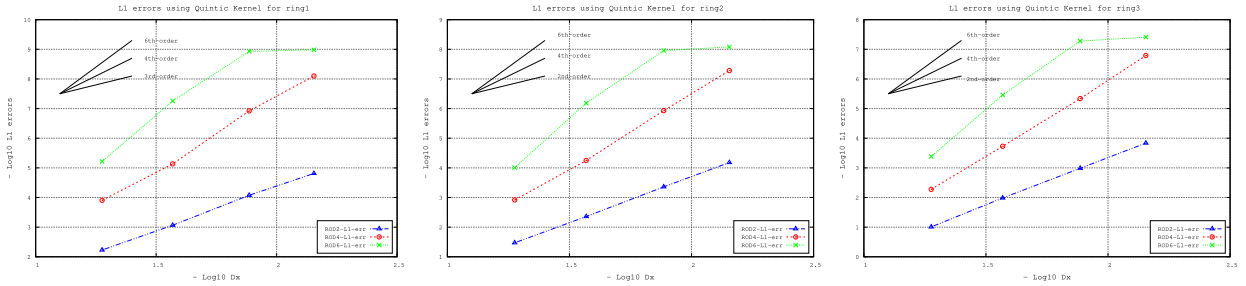


Fig. 9. L^1-error and convergence order for ring 1 (left), ring 2 (middle) and ring 3 (right).

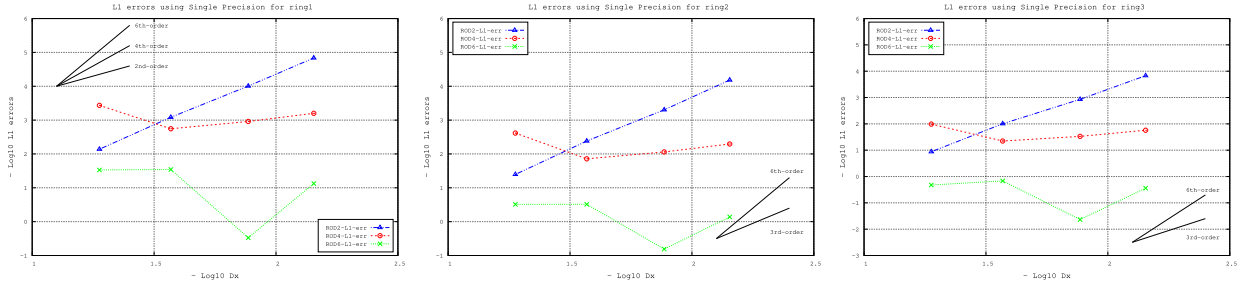


Fig. 10. L^1-errors and convergence with `single` precision number: ring 1 (left), ring 2 (middle) and ring 3 (right).

the first (left), second (middle) and third ring (right). We recall that the fourth-order methods use rings 1 and 2 while the sixth-order method requires the additional ring 3. Kernel parameter is adjusted to the optimal regarding the order method. Convergence orders are the expected ones except for ROD6 with the finest mesh due to a large condition number of the matrix.

3.2.5. ROD with single precision

To assess the floating-point precision impact on the conditioning, we have carried out the simulation using the `single` format (4 bytes floating-point number with 11 significant digits) in place of the usual `double` format (8 bytes floating-point number with 17 significant digits). We plot in Fig. 10 the $L1$-error for the three kinds of construction. The ROD2 method provides the expected convergence rate and errors. Nevertheless, ROD4 and ROD6 fail to evaluate an accurate solution correctly due to the large condition number. A wider stencil together with unbalanced Vandermonde matrices leads to an awful approximation. A final note regarding ROD6 is the possible accuracy improvement if one used `quad`-precision format, which in turn would come with additional computational costs.

3.2.6. ROD in a nozzle domain

The physical domain Ω is delimited by the functions $y = \pm\sqrt{1 + 2\log(\cosh(x))}$, for $x \in [-1; 1]$ providing a nozzle-like shape as shown in Fig. 2. The domain presents sharp corners that locally reduce the number of close neighbour cells, and requires a wider stencil. As in the previous geometry, we systematically test the convergence rate and the conditioning of the reconstruction for different order and kernel parameter values.

We draw in Fig. 11 the errors and convergence rates for the ROD2 method with different kernel parameters (Gaussian). The stencils only require a few cells (at least 6), and hence we manage to pick-up close enough active cells to provide good local reconstructions, even at the corner. The condition number is rather low, and we observe an optimal convergence order whatever the size of the kernel support is. Note that the conditioning presents important variation regarding the τ with two orders of magnitude ($\tau = 2\Delta x$ being the best and the errors tend to be smaller, while the conditioning decreases by almost an order of magnitude).

We print in Fig. 12 the accuracy and convergence order curves for the ROD4 method. The stencils are larger than the ones used in the second-order case, thus the stencil for a ghost cell situated around a corner is quite larger than the smooth interface case. Indeed, the number of activated cells in the first and second layer is not enough to provide a matrix M with maximum rank. One has to pick up more active cells to fulfil the maximum rank property. The stencil radius is larger than the one we elaborate for smooth interfaces, leading to a larger conditioning and then a dramatic reduction of the accuracy. The second relevant point concerns the kernel parameter, where $\tau = \Delta x$ clearly fails to catch enough active cells to provide a relevant reconstruction. At last, the parameter $\tau = 3\Delta x$ is better than the uniform kernel and highlights the improvement in using weights in the reconstruction. Notice the condition number is almost an order of magnitude smaller, and the errors in the first ring decrease by almost half of an order of magnitude.

The ROD6 turns to be the most sensitive situation since the condition numbers are huge. Fig. 13 shows the strong limitation of the accuracy for finer meshes due to the conditioning. Nevertheless, for coarse meshes we recover the optimal order with $\tau = 3\Delta x$ (better than the uniform kernel, condition number decreases by two orders of magnitude, together with the errors) whereas $\tau = \Delta x, 2\Delta x$

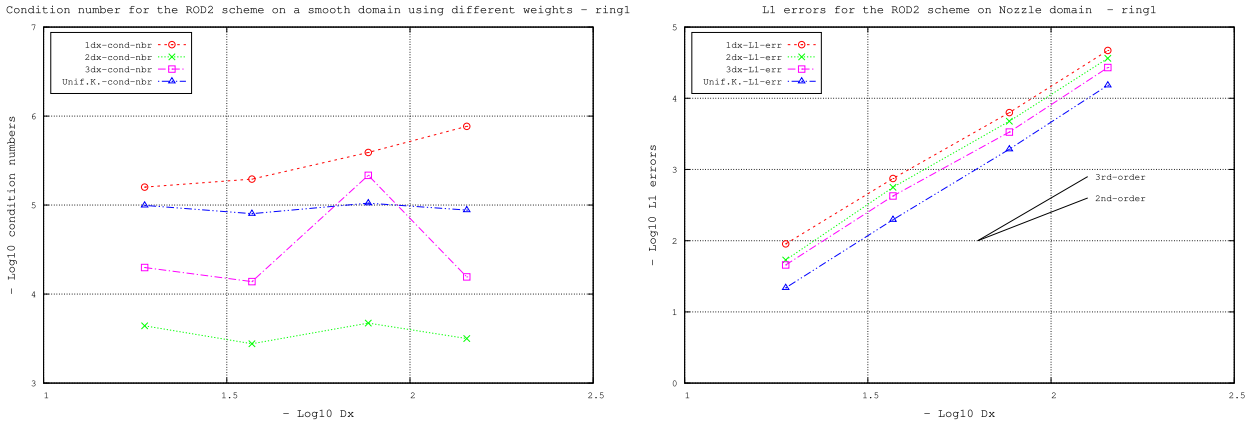


Fig. 11. Condition number (left) and L^1-errors (right) for the ROD2 method on the nozzle domain using the Gaussian Kernel with different weights.

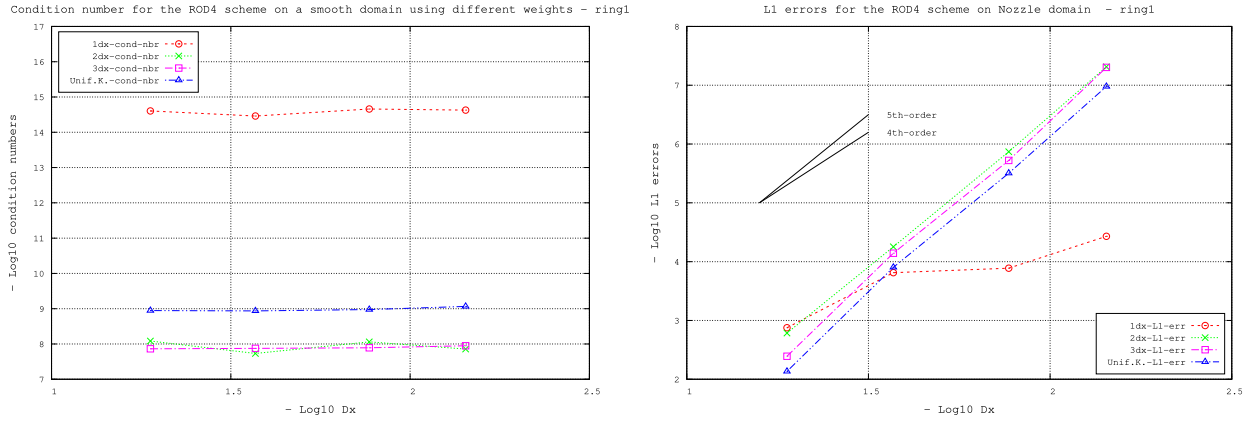


Fig. 12. Condition number (left) and L^1-errors (right) for the ROD4 method on the nozzle domain using the Gaussian Kernel with different weights.

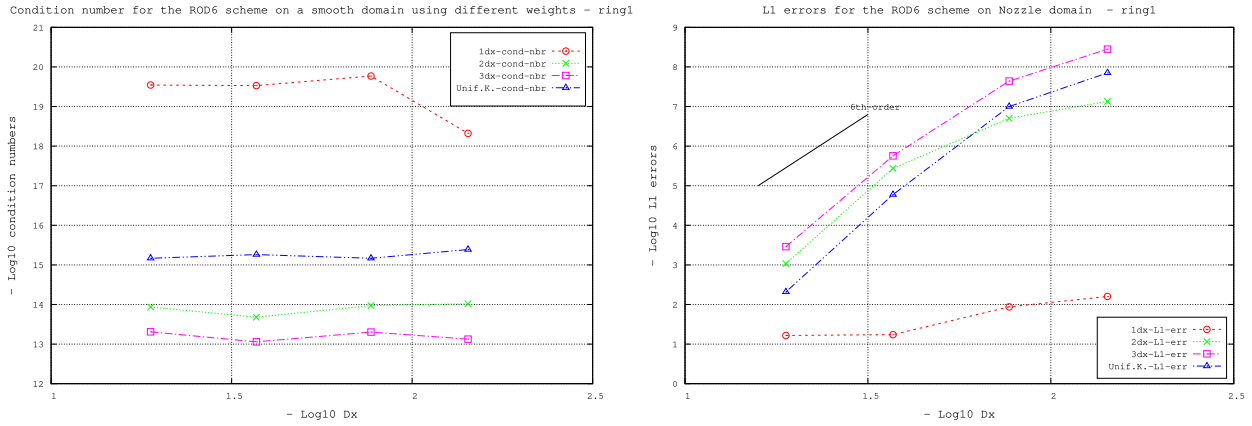


Fig. 13. Condition number (left) and L^1-errors (right) for the ROD6 method on the nozzle domain using the Gaussian Kernel with different weights.

gives non-sustainable conditioning. Compared with the smooth case, the condition number increases by one magnitude, which we attribute to the corners. Indeed, to provide a stencil enough cells to guarantee an eligible reconstruction, one has to select active cells very far from the ghost cell, resulting in a dramatic loose of accuracy.

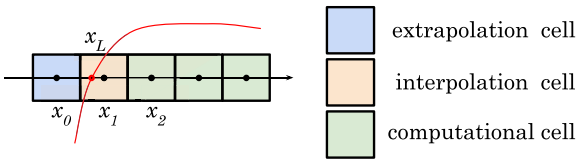


Fig. 14. Interpolation and extrapolation method to compute the ghost cell value. We use a node x_1 as the ghost point in the interpolation case, while we take a node x_0 for the extrapolation.

4. Stability

Several pathologic numerical issues are covered under the same umbrella "stability" such as positivity preservation, maximum principle, TVD preservation, no oscillation creation, negative eigenvalues that may amplify small perturbations. Stability of Immersed Boundary method strongly relies on the interpolation/extrapolation procedure to fill the ghost cell value. In [38], the author analyses the interpolation reconstruction, where the ghost cell is situated inside the physical domain, while the mirror method [39,11,15,25] assumes that the ghost cell is situated outside the physical domain (see Fig. 14).

The choice of the ghost cell's location (inside or outside the physical domain) deeply affects the scheme properties, such as the maximum principle or the positivity conservation. We hereafter provide a more in-depth analysis for the interpolation case (section 4.1) and extrapolation case (section 4.2).

4.1. Ghost cell outside the physical domain

We first tackle the situation where the ghost cell is situated outside the domain. We substitute the Dirichlet condition prescribed at x_L by another Dirichlet condition at x_0, following the ROD method principle, that is, to shift the boundary condition from x_L to a boundary condition at node x_0. We shall see that positivity preservation is not necessarily achieved, even if one prescribes the exact solution at the point x_0.

To highlight such an issue, let $\varepsilon > 0$ and consider the function

$$\phi_\varepsilon(x) = (1+2\varepsilon)^3(x+\varepsilon) - (x+\varepsilon)^4.$$

One has $\phi_\varepsilon(x) \geq 0$ on $\Omega_\varepsilon =]-\varepsilon, 1+\varepsilon[$ and $f = -\phi_\varepsilon''(x) \geq 0$. Therefore, ϕ_ε is the non-negative solution of the Poisson equation $-\phi'' = f$ with homogeneous Dirichlet condition on the domain Ω_ε. Consider a regular grid $x_i = i\Delta x$ with $\Delta x = 1/I$ and take $\Omega_\Delta =]-\Delta x, 1+\Delta x[$ for the numerical domain. Setting $\varepsilon = \Delta x/10$, one has $\Omega_\varepsilon \subset \Omega_\Delta$. Since $-\varepsilon \in]-\Delta x, 0[$, we substitute the exact boundary condition $\phi(-\varepsilon) = 0$ with the Dirichlet condition $\phi_{-1} = \phi_\varepsilon(-\Delta x) < 0$. Consequently, the exact value at node $-\Delta x$ provides a negative value, whereas no reconstruction procedure is involved in the definition of the ghost point value. Shifted the boundary condition from x_L to x_0 provides a negative Dirichlet condition at x_0. Worst, the numerical approximation now provides a negative value at node $x = 0$ turns. To check that last affirmation, we take $I = 10$ and using the centred second-order finite difference method (providing an M-matrix system that preserves the positivity), the approximation at node x_0 is `-9.3988e-05` while the exact value is `1.0060e-03`. Consequently, the positivity property is not achieved.

In conclusion, using a cell outside the physical domain may provide a negative value at the ghost point, and, on the other hand, a stable and standard numerical method may deliver negative values close to the physical boundary. Such a situation is independent of the reconstruction, since we have used the exact values at the point $x_{-1} = -\Delta x$. In particular, the ROD method will provide an excellent approximation at x_{-1}, hence the positivity property will be violated since the problem already exists with the exact value at node x_{-1}.

4.2. Ghost cell inside the computational domain

Following [38], the choice of ghost cells inside the physical domain guarantees the maximal principle and the positivity preservation for linear reconstruction. The positivity is no longer achieved if one considers higher order reconstruction. To this end, let us consider the domain $\Omega_\varepsilon =]-\varepsilon, 1+\varepsilon[$, point x_0 is now discarded while the ghost point is located at node $x_1 = 0$ and the unknowns are situated in $x_2 = \Delta x$, $x_3 = 2\Delta x$.

On the one hand, linear reconstruction between using $x_L = -\varepsilon$ and $x_2 = \Delta x$ leads to a convex combination between the values at x_L and x_2, hence the reconstruction preserves the positivity together with the maximum principle. On the other hand, if one considers a quadratic reconstruction, it is well-known that the interpolated value at node x_1 may not satisfy the positivity preserving property (Gibbs phenomenon). For example, take $\phi_\varepsilon(x) = (x-\varepsilon)(x+\varepsilon)$ with $\varepsilon < \Delta x$. One can check that $\phi_\varepsilon(-\varepsilon) = 0$, $\phi_\varepsilon(x_2) > 0$, $\phi_\varepsilon(x_3) > 0$, while $\phi_\varepsilon(x_1) < 0$. In conclusion, any polynomial reconstruction may not preserve the positivity unless introducing some non-linear procedure to control that issue. In particular, the ROD reconstruction also suffers with this drawback when considering non-regular solution.

4.3. ROD method stability

The iterative algorithm computes successively approximations inside the domain with the classical 7-points or 13-point finite difference method, while an approximation of the boundary condition in the ghost points is achieved with the ROD reconstruction. We only focus on the stability of the ROD reconstruction, since the stability of classical FD on Cartesian mesh is well-documented.

The polynomial reconstruction given by relation (3) is centred at the ghost point, hence the value at the point (x_i, y_j) is exactly a_0, that is the zero-degree monomial. On the other hand, the linear relation (4) is independent of Δx and Δy by construction of the polynomial π, matrix M together with the matrix W. Finally, assuming that we only prescribe the Dirichlet condition on the physical boundary, the matrix B is independent of Δx. Consequently, the eigenvalues of the symmetric matrix

$$R_{ij} = \begin{bmatrix} M_{ij}W_{ij}(M_{ij})^{\mathrm{t}} & (B_{ij})^{\mathrm{t}} \\ B_{ij} & 0 \end{bmatrix}$$

only depend on the relative position of the points of the stencil and the two points of the collar regarding the ghost point.

We assume that the configuration of the stencil and the two collar points provide a non-singular matrix R_{ij}. One has

$$\begin{bmatrix} \overline{a} \\ \overline{\lambda} \end{bmatrix} = (R_{ij})^{-1} \begin{bmatrix} M_{ij}W_{ij}\Phi \\ g \end{bmatrix},$$

where $\overline{a}$ is the vector of the polynomial coefficients (3) while $\phi_{ij} = \overline{a}_0$ is the reconstructed value.

Noting by $|a|$ the Euclidean norm of the vector a and its inductive norm $|A|$ for the matrix, we then have

$$|R_{ij}^{-1}| = \tau(R_{ij})$$

where $\tau(R_{ij}) > 0$ is the smallest eigenvalue of the matrix R_{ij}. We deduce the inequality,

$$|\phi_{ij}| = |\bar{a}_0| \le |\overline{a}| \le \sqrt{|\bar{a}|^2 + |\lambda|^2} \le \frac{1}{\tau(R_{ij})} \left| \begin{bmatrix} M_{ij}W_{ij}\Phi \\ g \end{bmatrix} \right| \le \frac{\sqrt{|M_{ij}W_{ij}\Phi|^2 + |g|^2}}{\tau(R_{ij})}.$$

On the other hand, we have

$$|M_{ij}W_{ij}\Phi| \le \rho(M_{ij}W_{ij})|\Phi|$$

with $\rho(M_{ij}W_{ij})$ the largest singular value of the matrix $M_{ij}W_{ij}$. We finally established the stability relation

$$|\phi_{ij}| \le K_{ij}\sqrt{|\Phi|^2 + |g|^2} \tag{5}$$

with $K_{ij} = \dfrac{\rho(M_{ij}W_{ij}) + 1}{\tau(R_{ij})}$.

Notice that quantities $\tau(R_{ij})$ and $\rho(M_{ij}W_{ij})$ are independent of the grid size, but strongly rely on the relative position of the points. As an example, we consider the simple case of the disk and plot in Fig. 15 and compute the quantities $\tau(R_{ij})$ (first row), $\rho(M_{ij}W_{ij})$ (second row) and K_{ij} (third row) for all the ghost cells with three levels of grids 40×40, 80×80, 120×120 (columns). Such figures reveal the impact of the two quantities on the stability of the reconstruction, and the last row represents the coefficient that controls the solution ϕ_{ij} regarding the data.

We observe that K_{ij} ranges between 50 and 300 with very little change regarding the grid size. The main issue is the control of $\tau(R_{ij})$ that could be tiny if the stencil configuration is not well-balanced in all the directions. Notice that the stability coefficient K_{ij} is a nice indicator of the solution quality, and some optimisation of the stencil design to provide the smallest value would be considered to reduce the error and decrease the sensitivity of the scheme to grid.

4.4. Consistency

Since the reconstruction is linear, a variation $\Delta\Phi$ of the values on the computational domain together with a variation Δg of the Dirichlet conditions provide a variation $\Delta\phi_{ij}$ at the ghost node we simply control with

$$|\Delta\phi_{ij}| \le K_{ij}\sqrt{|\Delta\Phi|^2 + |\Delta g|^2}.$$

Noting by $K = \max K_{ij}$ over all the ghost point provides, we deduce a global estimate of the error

$$|\Delta\phi_{GC}| \le K_{ij}\sqrt{|\Delta\Phi|^2 + |\Delta g|^2},$$

where $\Delta\phi_{GC}$ are the value's variation at the ghost cells resulting from the variation of the data $\Delta\Phi$ and Δg.

If the solution ϕ is a polynomial function of degree d, the constraints are satisfied and the reconstruction is exact, leading to a numerical solution that exactly matches with the numerical one, that is $\phi_{i,j} = \phi(x_i, y_i)$.

Let now consider a smooth function $\phi(x, y)$ and set the exact values at all the computation nodes, together with the boundary condition at the collar points. A Taylor expansion of ϕ around the ghost point (x_i, y_j) reads

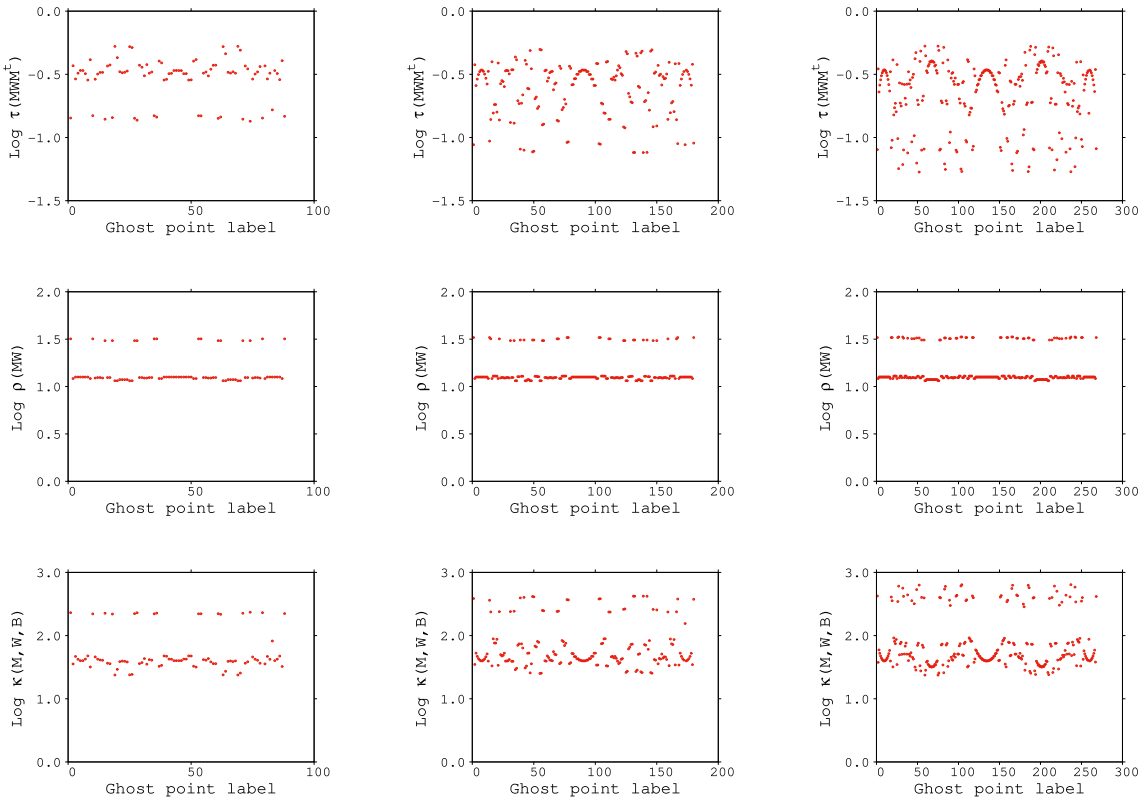


Fig. 15. Stability of the ROD reconstruction (logarithm scale). Computation of the $\tau(R_{ij})$ (first row), $\rho(M_{ij}W_{ij})$ (second row) and $\kappa(M,W,B)$ (third row), with a grid of 40×40 (first column), 80×80 (second column) and 120×120 (third column).

$$\phi(x,y)=\pi_{ij}(x,y)+r(x,y),\quad |r(x,y)|=O(\Delta x)^{d+1}.$$

On the one hand, we define $\Phi=(\phi(x_k,y_l))$ and on the other hand, we set $\widetilde{\Phi}=(\pi_{ij}(x_k,y_l))$ that contains the values for the node (x_k,y_l) of the stencil. Similarly, we construct the two entries vector g with the function ϕ for the points of the collar P_1 and P_2 while $\widetilde{g}$ comes from the polynomial π_{ij}.

The key point is that the ROD provides the exact Taylor Development polynomial π of ϕ when using the values $\widetilde{\Phi}$. Consequently, from the stability relation (5), we deduce

$$|\phi_{ij}-\pi(x_i,y_j)|\le K_{ij}\sqrt{|\Phi-\widetilde{\Phi}|^2+|g-\widetilde{g}|^2}.$$

Since $\pi_{ij}(x_i,y_j)=\phi(x_i,y_j)$ and $|\Phi-\widetilde{\Phi}|=O(\Delta x)^{d+1}$, $|g-\widetilde{g}|=O(\Delta x)^{d+1}$, we deduce the consistency estimate

$$|\phi_{ij}-\phi(x_i,y_j)|=O(\Delta x)^{d+1}.$$

5. Linear C-D-R equation with variable coefficients

Numerical resolution of the non-linear problem is achieved with a fixed point method using a local linearisation of the operators. Hence, the discretisation of variable coefficients C-D-R linear equation is the masterpiece of the method. The linear scalar convection diffusion reaction writes under two forms:
the conservative form

$$\mathfrak{F}_c(\phi)=f-\nabla\cdot(u\phi-\kappa\nabla\phi)-r\phi=0,\quad \text{in }\Omega, \tag{6}$$

with $\kappa(X)\ge 0$, $u(X)=(u^x,u^y)$, $r(X)\in\mathbb{R}$, equipped with the boundary condition and the non-conservative form

$$\mathfrak{F}_{nc}(\phi)=f+\kappa\Delta\phi-v\cdot\nabla\phi-q\phi=0,\quad \text{in }\Omega, \tag{7}$$

with $v(X)=(v^x,v^y)$ and $q(X)\in\mathbb{R}$. For regular solutions, the two forms are equivalent by setting $v=u-\nabla\kappa$, and $q=r+\nabla\cdot u$, but the numerical treatment is slightly different due to the usage of ghost cell.

In [33], the sixth-order method with constant coefficients is presented, and we detail its extension to variable coefficients combined with the ROD strategies to handle complex geometries. Such an extension is not straightforward when considering the C-D-R problem under the conservative form due to the ghost cells.

Indeed, translating a conservative equation into the non-conservative one involves the gradient of κ and the divergence of u. On the one hand, if one has an analytic expression of the coefficients κ and u, evaluation of the coefficients v and q is straightforward. On the other hand, if no analytic expression is available, we mean only the values at the nodes are known, as it happens in a linearisation, an approximation of v and q at the active node together with the collar points that preserves the order is more challenging.

5.1. High-order finite difference schemes

Given the values in the ghost cells represented with a vector $\mathbf{\Phi}_{\text{GC}}$, we seek the vector $\mathbf{\Phi}_{\text{IN}}$ of the approximation in the active cells, deriving from traditional centred finite difference schemes with Dirichlet condition at the ghost cells following [33] (section 5). We then obtain a linear operator

$$\mathbf{\Phi}_{\text{GC}} \to \mathbf{\Phi}_{\text{IN}} = \text{S}(\mathbf{\Phi}_{\text{GC}}; f) \tag{8}$$

that provides the approximation on the active cells, given the values on the ghost cells and the right-hand side function.

Numerical derivatives are performed with a discrete operator over the vector $\mathbf{\Phi}$. The discrete operator $(D_x^m\mathbf{\Phi})[i,j]$ represents the m-order finite differential approximation evaluated at the point $M_{i,j}$ of the partial derivative $\partial_x\phi(x_i,y_j)$ and we define the other discrete operators for $\partial_y\phi$, $\partial_{xx}\phi$ and $\partial_{yy}\phi$. For instance, the fourth-order approximation of operator $\partial_{xx}\phi$ reads

$$\partial_{xx}\phi(x_i,y_j) \approx (D_{xx}^4\mathbf{\Phi})[i,j] = \frac{1}{12\Delta x^2}\big(-\phi_{i-2,j} + 16\phi_{i-1,j} - 30\phi_{i,j} + 16\phi_{i+1,j} - \phi_{i+2,j}\big).$$

We shall denote (FD2), (FD4) and (FD6) the finite difference methods of order 2, 4 and 6 respectively.

5.2. Discretisation of the non-conservative form

Assume that κ, v and q are given functions. We set $K[i,j] = \kappa(x_i,y_j)$, $V^x[i,j] = v^x(x_i,y_j)$ and the same for $V^y[i,j]$ and $Q[i,j]$. The m-order discretisation of equation (7) with variable coefficients then reads,

$$-\big(K.(D_{xx}^m\mathbf{\Phi} + D_{yy}^m\mathbf{\Phi})\big) + (V^x.D_x^m\mathbf{\Phi} + V^y.D_y^m\mathbf{\Phi}) + (Q.\mathbf{\Phi}) = f. \tag{9}$$

Notice that the parameters κ, v, q only involve the active cells. Assuming that the m layers of ghost cells contain an approximation $\mathbf{\Phi}_{\text{GC}}$ of ϕ, the linear system reads

$$A^m\mathbf{\Phi}_{\text{IN}} = F_{\text{IN}} + G_{\text{GC}}^m$$

where A^m stands for the matrix associated to the m-order finite difference, F_{IN} corresponds to the right-hand side term (f_{ij}) on the active cells while $G_{\text{GC}}^m = G_{\text{GC}}^m(\mathbf{\Phi}_{\text{GC}})$ gathers the contributions of the ghost cells. Solving the linear problem provides the operator

$$\mathbf{\Phi}_{\text{GC}} \to \mathbf{\Phi}_{\text{IN}} = \text{S}(\mathbf{\Phi}_{\text{GC}}; f).$$

5.3. Discretisation of the conservative form

The conservative form (6) requires more attention, since the discrete operators are not applied in a straightforward way. A classical approach consists in rewriting the equation in its non-conservative form with $v = u - \nabla\kappa$ and, $q = r + \nabla\cdot u$ which requires an m-order approximation of $\nabla\kappa$ and $\nabla\cdot u$ at the centroid of the computation cells. If the coefficients are given with analytical functions, calculations of the gradient and the divergence easily provide the new parameters we immediately use in the non-conservative discretisation (9). In some cases, for instance for non-linear problems, we do not have analytical expressions for derivatives of the parameters and numerical approximations have to be carried out. To this end, we assume that approximations of both κ and u are known at the centroid of the active cells and **the ghost cells**, stored in matrices K, U^x, U^y, and R. Consequently, discrete operators D_x^m, D_y^m are applied to the coefficients, and we define

$$\nabla\kappa(M_{i,j}) \approx (D^mK)[i,j] = \big[(D_x^mK)[i,j],(D_y^mK)[i,j]\big]^T, \tag{10}$$

$$\nabla\cdot u(M_{i,j}) \approx (D^m\cdot U)[i,j] = (D_x^mU^x + D_y^mU^y)[i,j]. \tag{11}$$

We then compute for any active cell the parameters with

$$V = U - D^mK, \quad Q = R + D^m\cdot U$$

we plug into the non-conservative discretisation (9).

Remark. An alternative technique could be based on the successive partial derivatives such as $D_x^m(K.D_x^m\mathbf{\Phi})$ to approach $\partial_x(\kappa\partial_x(\phi))$ in order to preserve the conservative aspect of the equation. Nevertheless, we face a problem regarding the stencil size. Indeed,

the sixth-order scheme ($m = 6$) requires a 7-points stencil for the inner first-derivative and another 7-points stencil for the outer first-derivative, leading to an 11-points stencil in each direction. Due to this specific constraint, we do not have considered this issue. □

5.4. ADI algorithm

To take advantage of the multi-threading of the modern computer architecture with many cores, we use an Alternative Direction Implicit (ADI) method that splits the two-dimensional problem into I mono-dimensional independent problems on direction y (sweep y) and J mono-dimensional independent problems on direction x (sweep x). Initially developed for time-dependent problems [40], the method is reshaped as an algebraic solver in the steady-state context [41] and extended to the coupling of ROD with ADI in [33]. We introduce the linear operators

$$L^x\mathbf{\Phi} = -K.D^m_{xx}\mathbf{\Phi} + V^x.D^m_x\mathbf{\Phi} + \frac{1}{2}Q.\mathbf{\Phi}, \quad L^y\mathbf{\Phi} = -K.D^m_{yy}\mathbf{\Phi} + V^y.D^m_y\mathbf{\Phi} + \frac{1}{2}Q.\mathbf{\Phi}.$$

Let $\theta > 0$ be a small parameter. We reformulate the problem as

$$\mathbf{\Phi} + \theta(L^x + L^y)\mathbf{\Phi} = \mathbf{\Phi} + \theta F.$$

We introduce the dimensional splitting by solving successively

$$(\boldsymbol{I} - \theta L^x)\mathbf{\Phi}^{n+1/2} = (\boldsymbol{I} + \theta L^y)\mathbf{\Phi}^n + \theta F,$$
$$(\boldsymbol{I} - \theta L^y)\mathbf{\Phi}^{n+1} = (\boldsymbol{I} + \theta L^x)\mathbf{\Phi}^{n+1/2} + \theta F.$$

Notice that the operator $(\boldsymbol{I} - \theta L^x)$ is now applying on the $x-$components independently in j, thus we get J one-dimensional problems in variable x. Similarly, operator $(\boldsymbol{I} - \theta L^y)$ corresponds to I one-dimensional independent problems in variable y. We then derive a highly parallelised scheme where each sub-problem can be solved with an independent thread as in [33].

5.5. Improvement analysis of the kernel contribution

We assess what extends the ROD method equipped with the kernel to perform the ADI algorithm, namely the accuracy, the conditioning and the number of iterations. We provide the benchmarks for the smooth domain and the nozzle-like domain with corners. We build a sequence of approximations $(\mathbf{\Phi}^n_{\text{IN}})_n$ that converge to the fix-point solution by iteratively solving equations (2)-(8):

$$\mathbf{\Phi}^n_{\text{IN}} \to \mathbf{\Phi}^n_{\text{GC}} = \text{R}(\mathbf{\Phi}^n_{\text{IN}}; g) \to \mathbf{\Phi}^{n+1}_{\text{IN}} = \text{S}(\mathbf{\Phi}^n_{\text{GC}}; f)$$

where $\mathbf{\Phi}^0_{\text{IN}}$ is an initial approximation we compute using a simple second order method with rough boundary conditions (directly set the ghost cell values with the boundary condition of the closest collar point).

5.5.1. Smooth domain

We performed a first set of simulations on the disk domain using the Dirichlet condition. The manufactured solution is $\phi(x, y) = \exp(x + 2y)$ and we take $\kappa = 1$, $u = (1, 1)$, $r = 0$. The right-hand side source term is then given by $f(x, y) = -2\exp(x + 2y)$.

Fig. 16 displays the errors and convergence rate for the ROD2 coupling with the second-order finite difference method. Simulation were carried out with the uniform kernel and the Gaussian kernel with $\tau = \Delta x$, $2\Delta x$ and $3\Delta x$. We observe a slight improvement of the accuracy when using $\tau = \Delta x$ while optimal second-order of convergence is achieved.

Similarly, we provide in Figs. 17 and 18 the L^1-errors (left) and L^∞-errors (right) for the fourth-order FD4-ROD4 and sixth-order FD6-ROD6 schemes with different kernel parameter values. Small values of τ do not manage to provide satisfactory solutions, and the choice of $\tau = 3\Delta x$ is required to achieve a solution with very high accuracy. The use of kernel gives some accuracy improvement, but does not reduce the condition number enough to overcome the error limitation close to `1-e10`.

5.5.2. Nozzle domain

The Nozzle-like domain (see Fig. 2) presents four corners that strongly impact the ROD method. To select a stencil capable of providing a minimum amount of information to guarantee an eligible reconstruction, we must pick cells situated further away from the reference cell. Wider stencil yields to a substantial deterioration of the condition number, and we aim at alleviating their consequences for using a wise kernel parametrisation.

We display in Fig. 19 the errors and convergence rates for the second-order scheme using different kernel parameter values. The change of Kernel parameter does not affect the errors' curve, except for the $\tau = 2\Delta x$ case which present some erratic behaviour.

Figs. 20 provides the errors and convergence rates for the fourth-order schemes with different kernel parameter values. Similarly to the smooth case, small support of the kernel may provide erroneous results and one has to choose $\tau = 2\Delta x$ and $\tau = 3\Delta x$ to obtain the optimal 4th-order of accuracy. Substantial improvements are reported and highlight the efficiency of the kernel to mitigate the corner situation. Notice that the 4th-order method does not present any limitation due to the `double` precision issue, since the condition number is small enough and does not perturb the fixed point convergence.

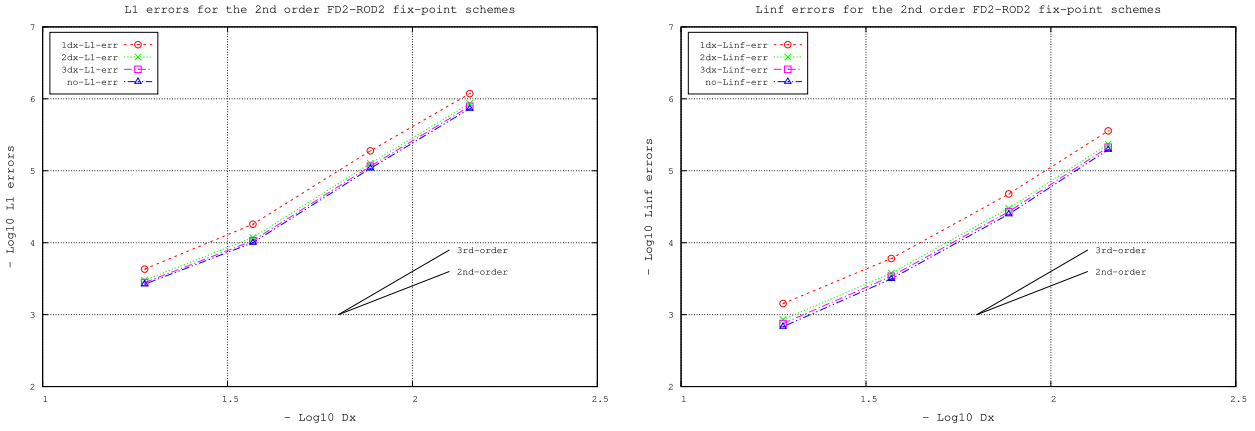


Fig. 16. $L1$ errors (left) and $L\infty$ errors (right) for the 2nd order finite difference with ROD2 on a smooth domain with for different kernel parameters.

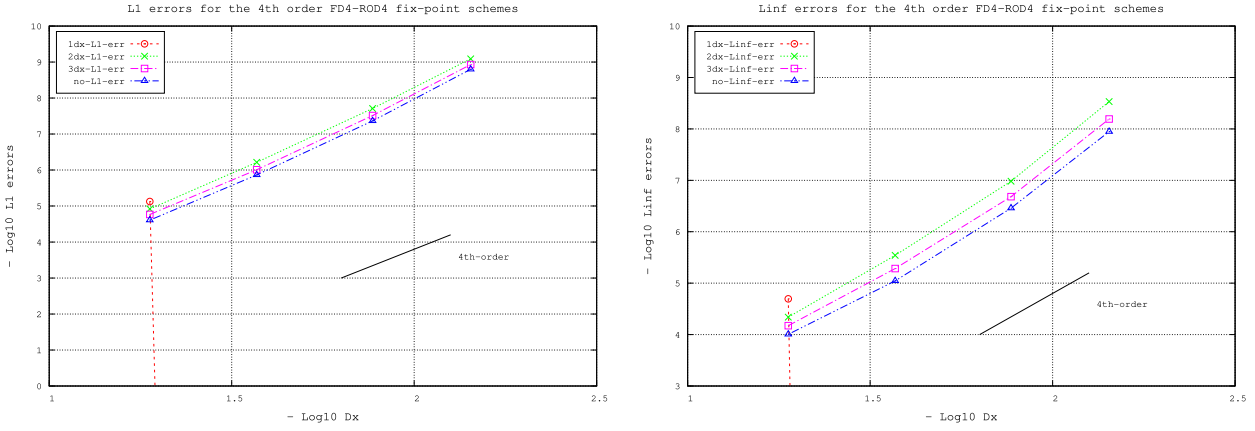


Fig. 17. $L1$ errors (left) and $L\infty$ errors (right) for the 4th-order finite difference method coupling with the ROD4 on a smooth domain with kernel.

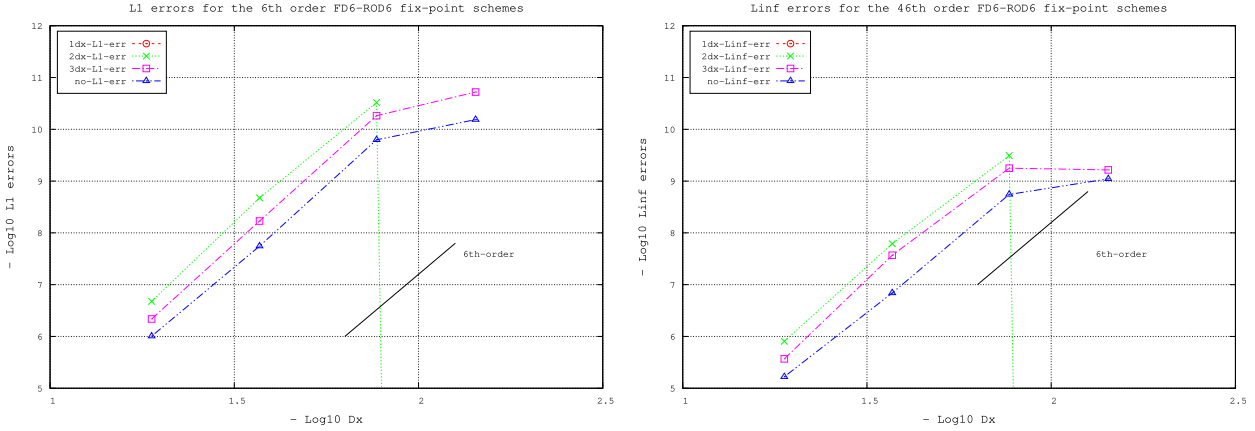


Fig. 18. $L1$-errors (left) and $L\infty$-errors (right) for the 6th-order finite difference coupling with the ROD6 on a smooth domain with different kernel parameters.

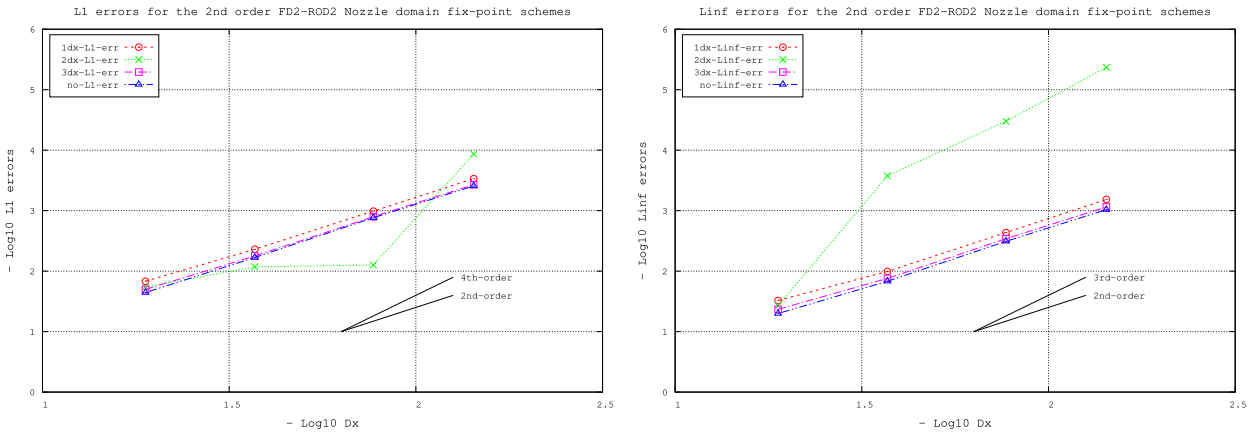


Fig. 19. $L1$ errors (left) and $L\infty$ errors (right) for the 2nd order fixed-point FD2-ROD2 method on a nozzle domain with different kernels.

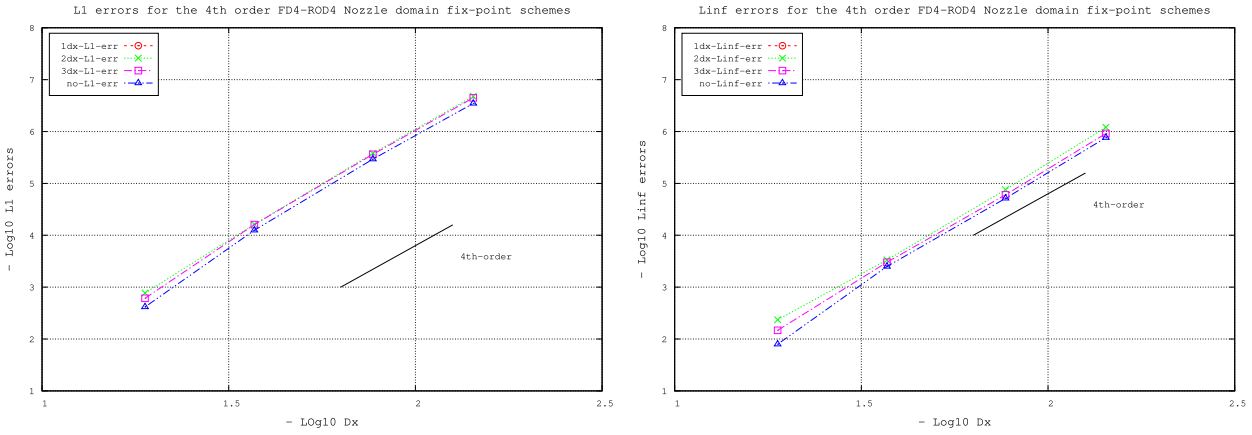


Fig. 20. $L1$ errors (left) and $L\infty$ errors (right) for the 4th order FD4-ROD4 fix-point methods on a nozzle domain with different kernels.

We give in Fig. 21 the errors and convergence rates for the sixth-order schemes with the kernel parameter $\tau = 3\Delta x$ and the uniform kernel. Substantial improvements are reported and highlight the efficiency of the kernel to mitigate the corner situation. For instance, we improve one order of magnitude of accuracy between the uniform kernel and the Gaussian kernel for the 6th-order case. Nevertheless, the accuracy limitation around `1e-10` is visible and derives from the polynomial coefficients' approximation. Stencils are too large and keep the accuracy from reaching the optimal errors.

5.6. Conservative form with analytical coefficients

We now deal with a non-constant coefficients convection diffusion problem initially given under its conservative form as in (6) with

$$u = \begin{pmatrix} \dfrac{x^2}{\sqrt{x^2+y^2+1}} \\ \dfrac{y^2}{\sqrt{x^2+y^2+1}} \end{pmatrix}, \qquad \kappa = 1 + \frac{x^2+y^2}{2}.$$

We rewrite the problem under the non-conservative form (7) taking advantage of the analytical expression of the coefficients, that is

$$v = u - \nabla\kappa = u - \begin{pmatrix} x \\ y \end{pmatrix},$$

$$q = \nabla \cdot u = \frac{x(x^2+2y^2+2) + y(y^2+2x^2+2)}{(x^2+y^2+1)^{\frac{3}{2}}}.$$

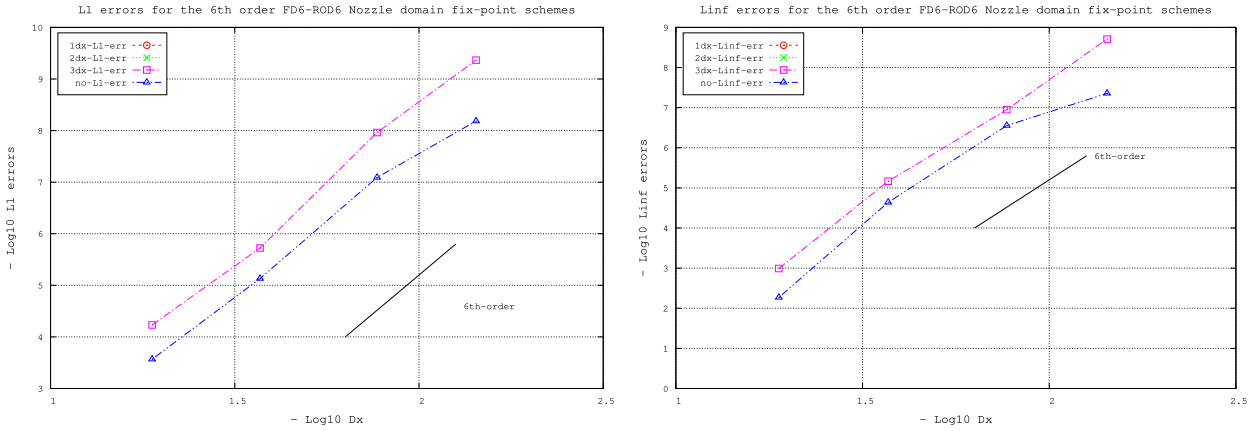


Fig. 21. $L1$ errors (left) and $L\infty$ errors (right) for the 6th-order FD6-ROD6 fix-point schemes on a nozzle domain with different kernels.

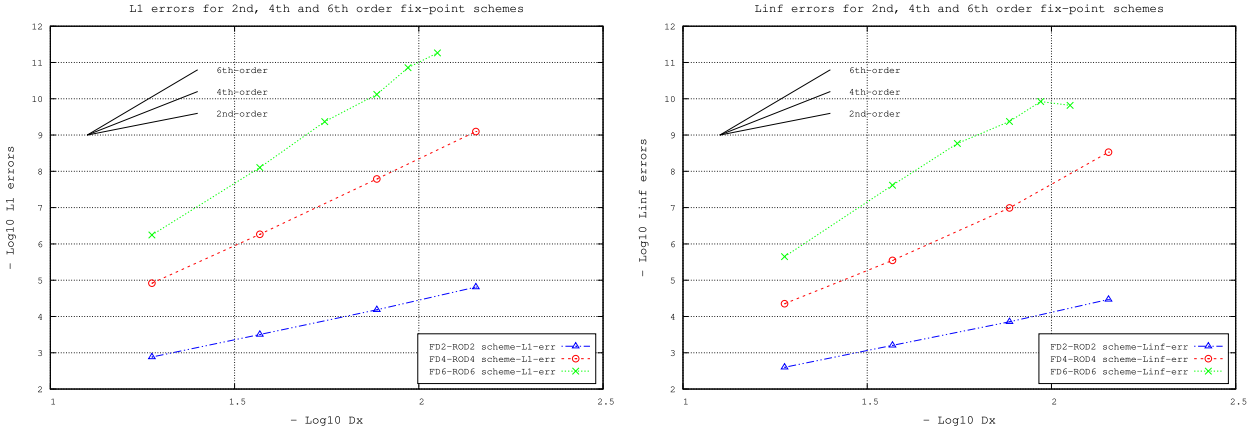


Fig. 22. $L1$ errors (left) and $L\infty$ errors (right) for the 2nd, 4th and 6th order fix-point schemes on a circular domain solving the conservative convection-diffusion problem.

We take $\phi = \exp(x + 2y)$ as a manufactured solution, and the right-hand side term f is computed using the velocity and conductivity analytical expression. The domain is the disk of radius $R = 0.8$ depicted in Fig. 2-left panel, where Dirichlet's condition is prescribed all along the boundary using the collar points.

5.6.1. *Conservative form with the analytical coefficients*

Given the analytic expression of κ and u used in the conservative form, we deduce the equivalent expression for v and q we use in the non-conservative form of the equation. In Fig. 22, we plot the errors and convergence rate for the second-, fourth- and sixth-order methods using the Gaussian kernel with $\tau = \Delta x$, $2\Delta x$ and $3\Delta x$ respectively. We reach the optimal order for the three methods, with the usual limitation for the 6th-order scheme due to the high condition number for finest grids. However, we emphasise that with just 90 points using a 6^{th} order scheme, errors are smaller than using 240 points with an 4^{th} order scheme.

5.6.2. *Conservative form with approximated coefficients*

An issue arises where no analytical expression of the coefficients is available and velocity, diffusion are only given point-wise. Deriving the non-conservative form requires numerical approximation for the derivative of the coefficient using the discrete expressions (10) and (11). We revisited the convection diffusion problem by substituting the exact expression for v and q with its numerical approximations. Fig. 23 shows the errors and convergence rates using the approximated coefficients for the second-, fourth-, sixth-order method. Comparison with the analytical version given in Fig. 22 shows a perfect matching of the two methods. Orders are optimal until reaching the `double` truncation limitation.

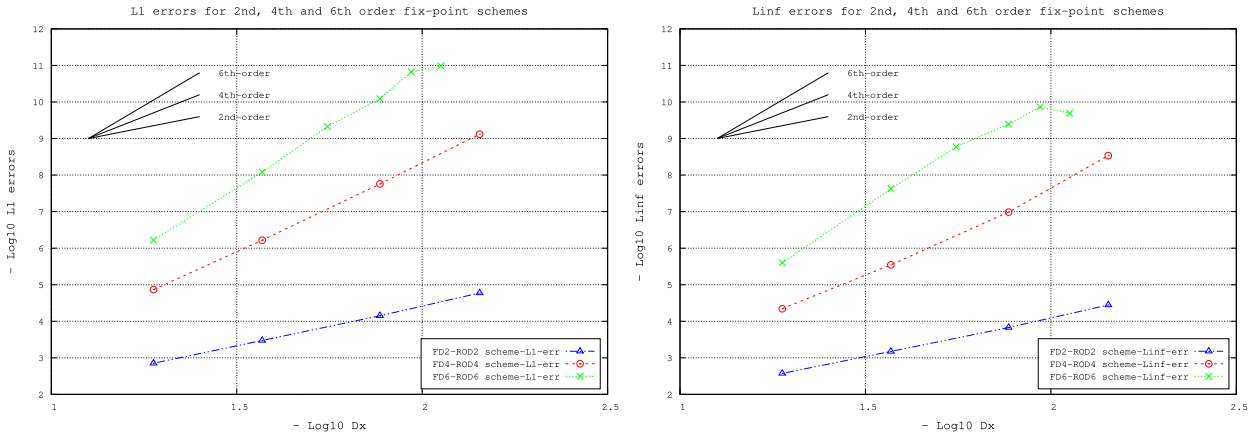


Fig. 23. $L1$ errors (left) and $L\infty$ errors (right) for the 2nd, 4th and 6th order fix-point schemes on a circular domain solving a conservative convection-diffusion problem with approximated values for $\nabla \cdot u$, and $\nabla\kappa$.

6. Non-linear problem

The fix-point (also called Picard) procedure is a popular technique to solve non-linear problems [42] based on a local linearisation by using the solution of the previous step for the coefficients' evaluation. As an example, the non-linear convection diffusion reaction problem

$$-\kappa(\phi)\Delta\phi + u(\phi)\nabla\phi + q(\phi)\phi = f$$

will be solved by determining a sequence $(\phi^n)_{n\in\mathbb{N}}$ given by the solution

$$-\kappa(\phi^n)\Delta\phi^{n+1} + u(\phi^n)\nabla\phi^{n+1} + q(\phi^n)\phi^{n+1} = f$$

with ϕ^0 the initial guess. At the discrete level, one has to consider the two operators

$$(\mathbf{\Phi}^n_{\text{GC}}, \mathbf{\Phi}^n_{\text{IN}}) \to \mathbf{\Phi}^{n+1}_{\text{IN}} = \text{S}(\mathbf{\Phi}^n_{\text{GC}}, \mathbf{\Phi}^n_{\text{IN}}; f),$$
$$(\mathbf{\Phi}^{n+1}_{\text{IN}}, \mathbf{\Phi}^n_{\text{GC}}) \to \mathbf{\Phi}^{n+1}_{\text{GC}} = \text{R}(\mathbf{\Phi}^{n+1}_{\text{IN}}, \mathbf{\Phi}^n_{\text{GC}}; g),$$

that consider the linearisation with the solution at iteration n. Fig. A.36 shows the details of the fix-point method. The values of the convection-diffusion coefficients are updated in the external n-loop, while the linear problem is solved within the inner ADI cycle.

The numerical methods are implemented with C++ using OpenMP for multi-threading and using the Eigen library for matrix operations (ROD). The tests were carried out on an 8-cores AMD Ryzen 7 3700U with Radeon Vega 10 Graphics and 12 GB DDR4-2400 SDRAM.

6.1. A non-linear thermal problem

We elaborate the first benchmark by considering a steady-state nonlinear thermal problem where the heat conduction depends on the temperature $\kappa = \sqrt{\phi}$ while velocity, given by $v = (1-(2y)^2, 0)^T$, is independent of the temperature. Domain Ω is the nozzle-shaped geometry bounded by the two curves $y = \pm\sqrt{0.2 + 0.1\log(\cosh(x))}$, with, $x \in [-1; 1]$ as depicted in Fig. 24. We construct a manufactured solution by setting $\phi = \eta\phi_b + (1-\eta)\phi_c$, where we take

$$\eta = \frac{\cosh(y) - 1.0}{\cosh(0.5) - 1.0}$$
$$\phi_c = \phi_0 + \frac{1-x^2}{2}(\phi_{max} - \phi_0)$$
$$\phi_b = \phi_{min} + \frac{1-x^2}{2}(\phi_{max} - \phi_{min}),$$

with ϕ_{min}, ϕ_{max} and ϕ_0 given values. We depict in Fig. 24 the exact solution for the coarse mesh and the fine mesh respectively.

We prescribe the Dirichlet conditions all along the boundary and carry out the simulations for the 2nd, 4th and 6th orders schemes using optimised values for the kernel. The Effective convergence order is reported in Fig. 25. The accuracy limitation for the sixth-order is still the consequence of `double` precision together with a large condition number.

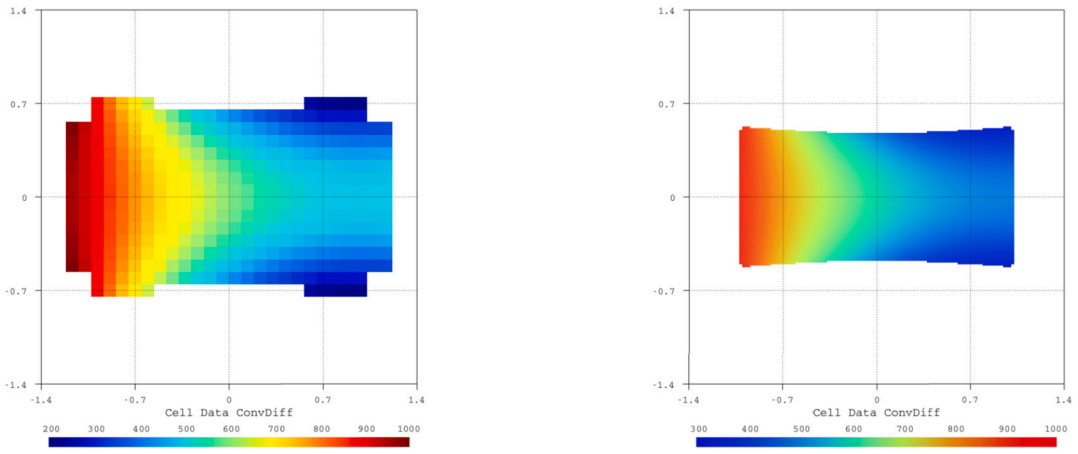


Fig. 24. Solution of the non-linear convection diffusion problem in the nozzle with 30×30 cells (left) and 240×240 cells (right). Notice that the ghost cells are included in the pictures.

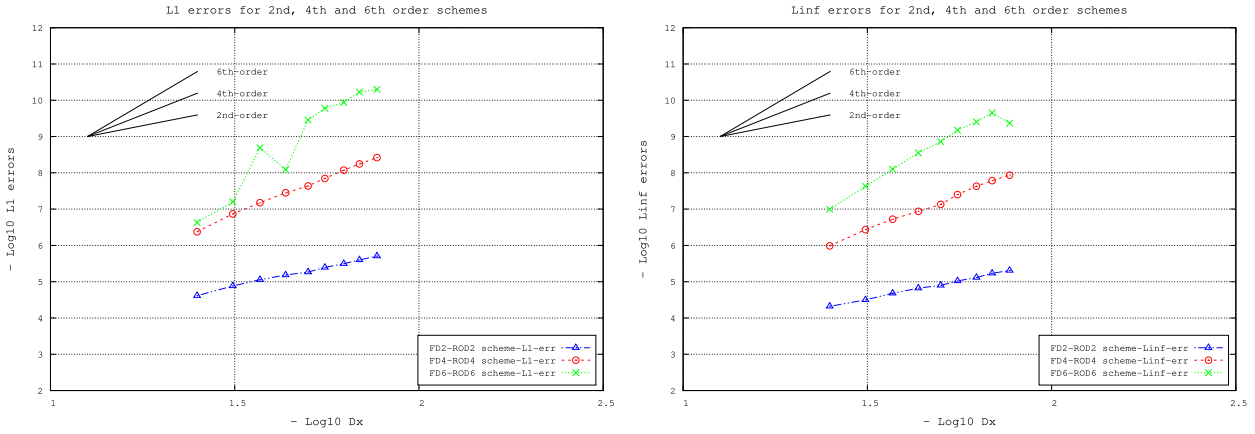


Fig. 25. L^1-errors (left) and L^∞ errors (right) for the 2nd, 4th and 6th order fix-point schemes on a nozzle domain solving a non-conservative nonlinear convection-diffusion problem.

6.1.1. Convergence of the fix point

To assess the convergence of the fix-point method, we compute the L^∞ norm of the difference between two successive solutions and plots the residual history in Fig. 26 for the 2nd, 4th and 6th order schemes, namely FD2-ROD2, FD4-ROD4 and FD6-ROD6. We use two different grids 50x50 (plain line) and 80x80 (dashed line) for comparison. The stop criterion is controlled by the tolerance parameter $\epsilon =$`1E-13`.

For the 50x50 grid, the second-order scheme is the faster since the condition number of the whole problem is smaller, whereas the sixth-order scheme needs around 50% more iterations. A finer mesh increases the condition number, and the second-order method needs 30% more iterations. The 4th-order case is a bit different, since the additional computation effort is rather low with the fine grid. Convergence of the sixth-order method with the 50x50 grids is almost stalled due to the large condition number and may not reach the user tolerance.

6.1.2. Grid convergence

We assess the grid convergence using a sequence of refined grids, while the reference solution is computed with the finest grid 120×120. Numerical approximations are evaluated at 15 points inside the domain $(-1 + i/2, -1/2 + j/2)$ with $i = 0, \cdots, 4$, $j = 0, 1, 2$, using a high-order interpolation procedure. We evaluate a pseudo L^1-error by averaging the errors over the grid. Fig. 27 displays the errors and convergence rates for the previous non-linear convection diffusion problem. Second-order is well achieved for all the grids, together with the fourth-order scheme. The sixth-order case is almost preserved for coarse grids until the 90×90 resolution. Above that size, we definitely loose the convergence order due to the high condition number of the problem. Nevertheless, the convergence in grid matches with the convergence with the exact solution and provides a reliable tool to evaluate the error when no analytic solution is available.

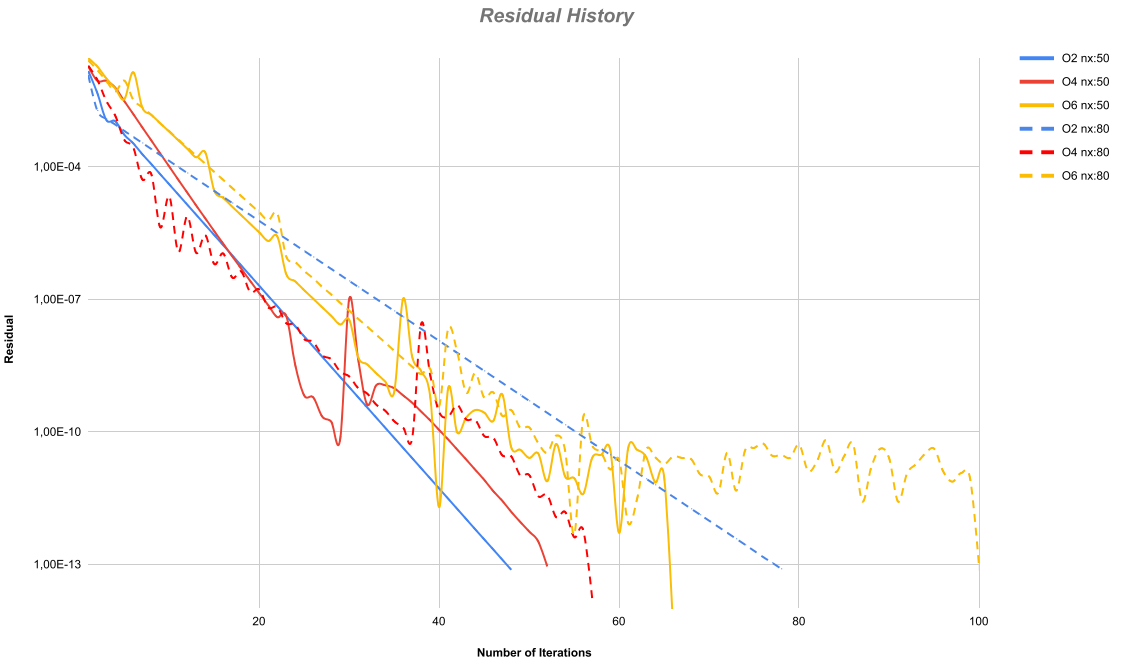


Fig. 26. Residual history for two different grid sizes using the 2nd, 4th and 6th order schemes.

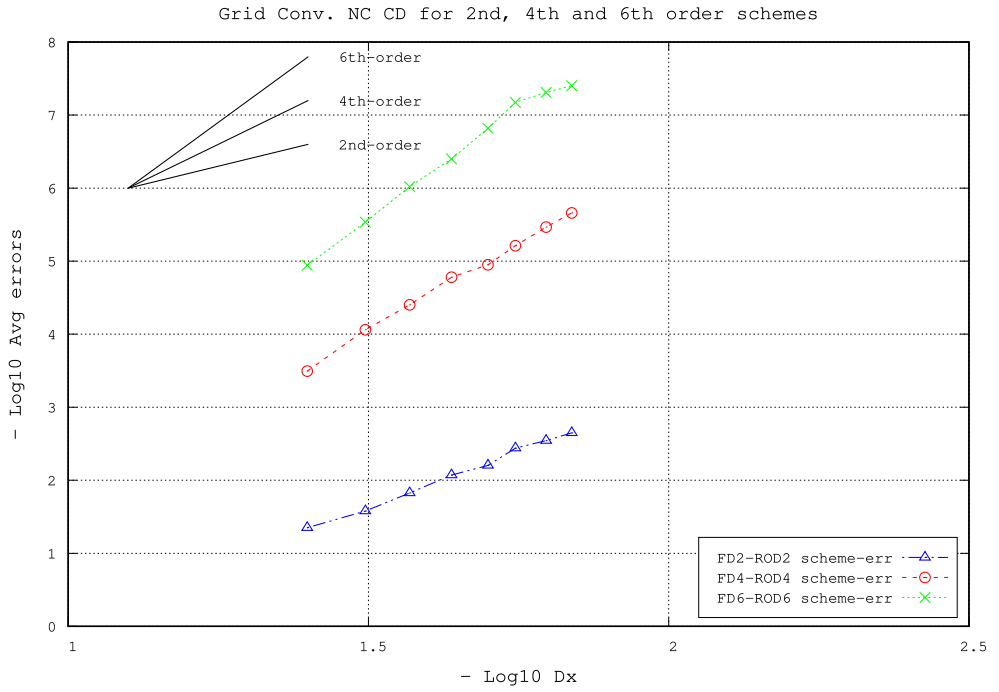


Fig. 27. Grid convergence orders for the 2nd, 4th and 6th order fix-point schemes on a nozzle domain solving a non-conservative nonlinear convection-diffusion problem.

6.1.3. Time computation versus accuracy

We now turn to the analysis of the computational time in function of the accuracy for the three scenarios FD2-ROD2, FD4-ROD4 and FD6-ROD6. We report in Table 1 the running time, the number of iterations and the error for the previous non-conservative nonlinear problem.

We first note that, given a grid, the time per iteration is of the same order, and in the worst cases we have a ratio 1:2 between FD2-ROD2 and FD6-ROD6. In particular, for larger grid, we have barely the same computational cost per iteration. The number of

Table 1
Time, iterations and accuracy for the 2nd, 4th and 6th order fix-point schemes on a nozzle domain solving a non-conservative nonlinear convection-diffusion problem.

I	FD2-ROD2				FD4-ROD4				FD6-ROD6			
	time (s)	its	time/it (s)	err	time (s)	its	time/it (s)	err	time (s)	its	time/it (s)	err
40	0,36	31	1,16e-02	4,79e-05	0,78	36	2,17e-02	1,03e-06	0,82	37	2,22e-02	1,01e-07
50	1,2	51	2,35e-02	3,18e-05	1,93	55	3,51e-02	3,66e-07	2,19	69	3,17e-02	2,35e-08
60	1,15	40	2,88e-02	2,09e-05	2,26	46	4,91e-02	1,89e-07	1,85	37	5,00e-02	7,92e-09
70	2,42	59	4,10e-02	1,51e-05	4,29	68	6,31e-02	1,15e-07	3,07	56	5,48e-02	2,82e-09
80	3,63	81	4,48e-02	1,26e-05	4,85	60	8,08e-02	7,41e-08	5,47	103	5,31e-02	1,38e-09
90	3,81	64	5,95e-02	9,55e-06	8,64	91	9,49e-02	3,98e-08	10,5	160	6,56e-02	6,67e-10
100	4,29	52	8,25e-02	7,67e-06	6,06	46	1,32e-01	2,34e-08	9.64	118	8,17e-02	3,95e-10
110	6,24	71	8,79e-02	5,85e-06	8,53	50	1,71e-01	1,64e-08	13,63	155	8,79e-02	2,22e-10
120	5,92	56	1,06e-01	4,91e-06	9,27	57	1,63e-01	1,16e-08	16,05	104	1,54e-01	4,26e-10

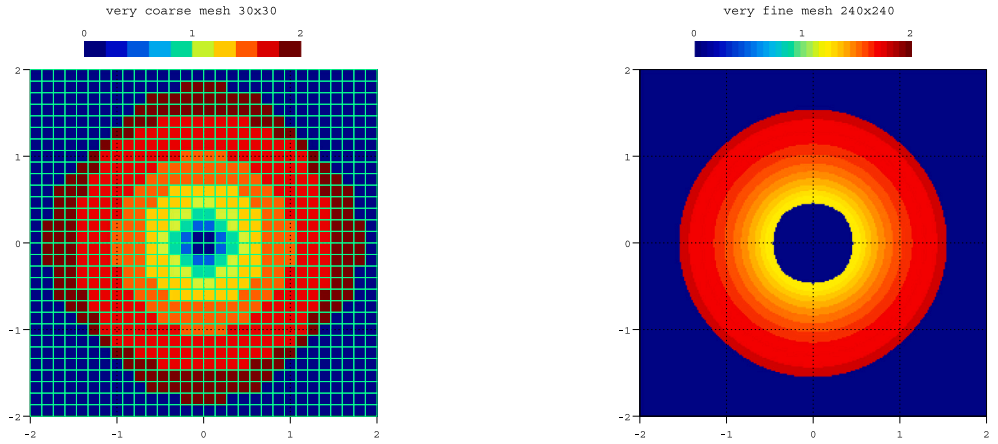


Fig. 28. Exact solution on the annulus domain for two grid sizes.

iterations clearly increases with the method order, but also remains in a scale of 1:2 between 2nd-order and 6th-order methods. To sum up, the computational cost is, at most, three times higher for the FD6-ROD6.

On the other hand, the accuracy gain is tremendous. The 50x50 grid with the 6th-order method provides an error of `2.35E-08` in 2.2 seconds. We achieve the same error with the 100x100 grid with FD4-ROD4 in 6 seconds while, by extrapolation, we expect a 1800x1800 grid to reach the same accuracy with a running time of 1400 seconds.

6.2. The annulus problem

We proceed with a non-linear convection-diffusion problem involving both Neumann and Dirichlet conditions on the boundary of an annulus with inner radius $r_{int} = 0.5$, and outer radius $r_{out} = 1.5$. The exact solution is given by $\phi(r) = a(\ln(r) + b)^{\frac{2}{3}}$ with $r = \sqrt{x^2 + y^2}$, $a, b \in \mathbb{R}$ and velocity reads, $U = (y; -x)$ while the diffusive term is given by $\kappa = \sqrt{\phi(r)}$ (see Fig. 28). We set $a = 1$ and $b = 2$ in our application and the fix point method is achieved by carrying out the linearisation both for the active cells and the ghost cells. Indeed, the Neumann condition involves the non-linear coefficient κ that has to be updated.

We print in Fig. 29 the errors and convergence rates for the different grids. Second-, fourth- and sixth-order schemes are in line with the expected orders. Even the sixth-order method presents the optimal convergence due to the very smooth curved boundary that does not create excessive condition number. The ROD reconstruction is quite effective for all the grids, both for the Dirichlet and Neumann cases. Similar behaviours with the convergence in grid are reported in Fig. 30 where the second-, fourth- and sixth order of convergence are achieved for all the grids.

6.3. Heat induction

The last benchmark concerns a really more complex problem combining electromagnetic and thermal equations over a curved domain. A piece of metal $\Omega \subset \mathbb{R}^2$ is heated by electromagnetic induction. The magnetic field is assumed to be periodic in time, *i.e.* $H(t, x) = e^{\mathbf{i}\omega t} H(x)$, with $\mathbf{i}$ the imaginary number and $H(x)$ a complex function (see [43]). The steady-state solution (H, T) is then given by the non-linear system

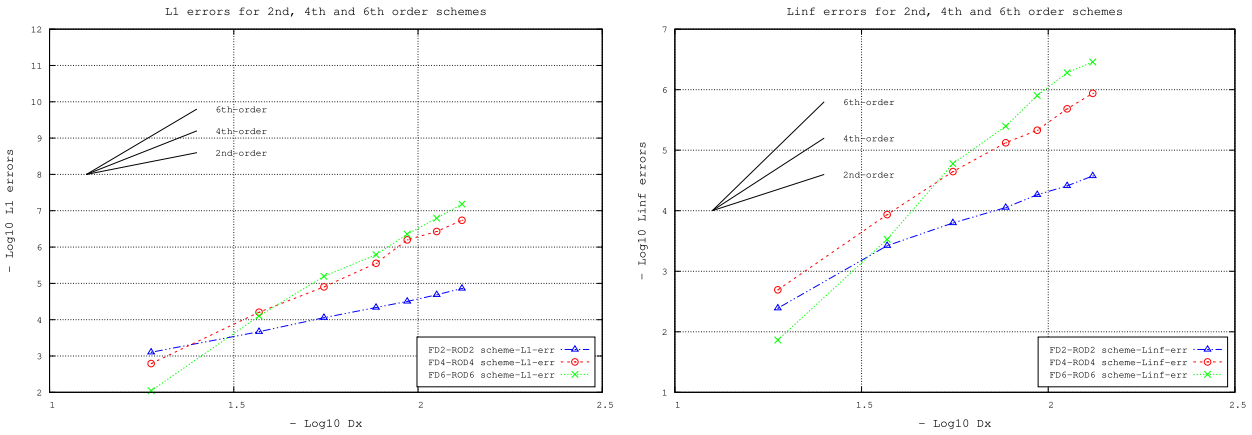


Fig. 29. L^1-errors (left) and L^∞ errors (right) with their respective convergence orders for the 2nd, 4th and 6th order fix-point schemes on an annulus domain solving a conservative nonlinear convection-diffusion problem.

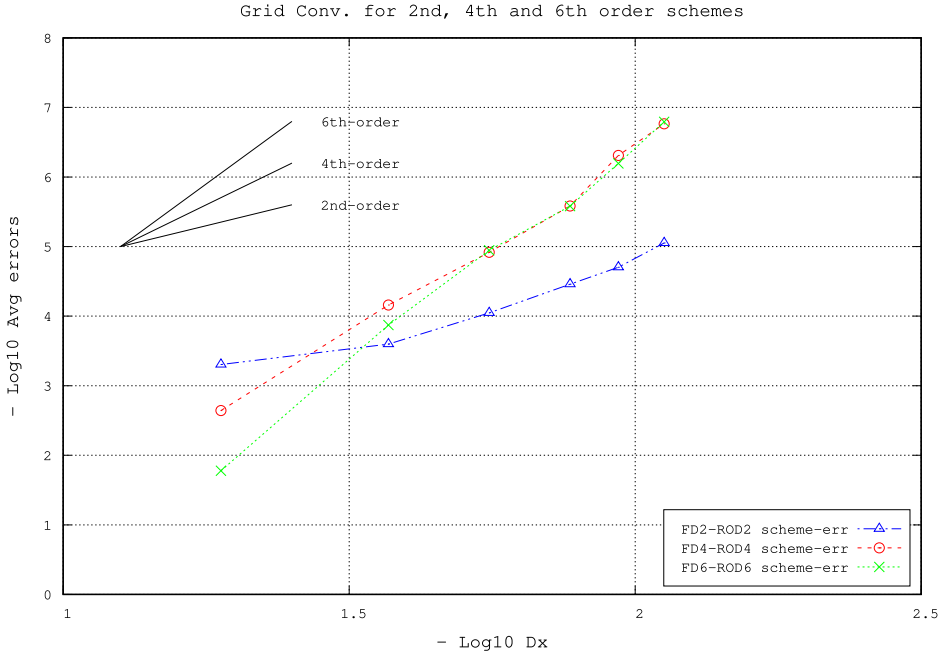


Fig. 30. L^1-errors with their respective convergence orders for the 2nd, 4th and 6th order fix-point schemes on an annulus domain solving a conservative nonlinear convection-diffusion problem.

$$\mathbf{i}\omega\mu H - \nabla\cdot(\rho(T)\nabla H) = 0,$$
$$-\nabla\cdot(\kappa(T)\nabla T) = \rho(T)|\nabla H|^2,$$

where T is the temperature, ρ the electric conductivity, and κ the thermal conductivity. We fix the magnetic field on the boundary $\partial\Omega$ with, $H = H_0 \in \mathbb{C}$ while we assume a radiation-like condition for the temperature given by $\kappa(T)\nabla T\cdot\boldsymbol{n} = \sigma T^\gamma$.

We reformulate the magnetic field problem with real functions (H_r, H_i) and get

$$-\nabla\cdot(\rho(T)\nabla H_r) = \omega\mu H_i,$$
$$-\nabla\cdot(\rho(T)\nabla H_i) = -\omega\mu H_r,$$
$$-\nabla\cdot(\kappa(T)\nabla T) = \rho(T)(|\nabla H_r|^2 + |\nabla H_i|^2),$$

with the boundary condition $H_r = H_0$ and $H_i = 0$ for the real and imaginary parts respectively.

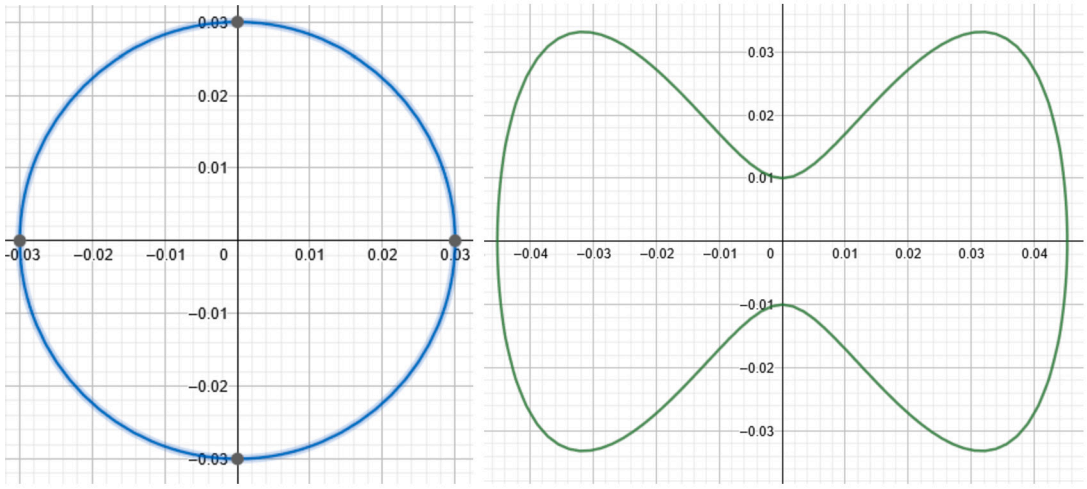


Fig. 31. Test domains.

To deal with the nonlinearity, a simple fix point algorithm is proposed where we build a sequence (H_r^n, H_i^n, T^n), $n = 1, \cdots$ by solving successively

$$
\begin{aligned}
&-\nabla \cdot (\rho(T^n)\nabla H_r^{n+1}) = \omega\mu H_i^n,\\
&-\nabla \cdot (\rho(T^n)\nabla H_i^{n+1}) = -\omega\mu H_r^n,\\
&-\nabla \cdot (\kappa(T^n)\nabla T^{n+1}) = \rho(T^n)(|\nabla H_r^{n+1}|^2 + |\nabla H_i^{n+1}|^2)
\end{aligned}
$$

while the radiation condition is linearised with the Robin condition

$$\kappa(T^n)\nabla T^{n+1} \cdot \boldsymbol{n} - \sigma.(T^n)^{\gamma-1} T^{n+1} = 0$$

The loop is carried out until that the difference between two successive solutions is lower than a given tolerance.

To proceed, we consider two intermediate problems with increasing difficulties. The Magnetic field problem is solved with a constant coefficient. Then we solve the thermal problem using the energy term dissipation. Finally, we couple the two problems assuming that the coefficients depend on the temperature. Furthermore, two different curved domains will be considered for testing the capacity to handle complex boundary: the disk domain and the butterfly shaped domain (Fig. 31).

6.3.1. Magnetic fields system

We start by solving the magnetic system

$$
\begin{aligned}
&-\nabla \cdot (\rho\nabla H_r^{n+1}) = \omega\mu H_i^n,\\
&-\nabla \cdot (\rho\nabla H_i^{n+1}) = -\omega\mu H_r^n,
\end{aligned}
$$

using different domain shapes. We set the conductivity $\rho = 1E-7$ Ohms, the vacuum permeability $\mu = 1.25E-6$ H/m, and the frequency is given by $\omega = 2\pi 1E+2$. We prescribe Dirichlet boundary condition, adopting $H_0 = 2E+7$ on the whole boundary for real magnetic field intensity, while we impose a null condition for the imaginary part H_i.

We assess the convergence in grid and report in Fig. 32 the errors and orders for H_r and H_i respectively. The circle domain is very smooth and convergence is close to the expected one. Convergence in grid is not as regular as the convergence regarding an exact solution, but we observe a strong reduction of the absolute error with the finest grid by comparing the 2nd-, 4th- and 6th-order method. Compared to the boundary condition $H_0 = 2E+7$, the number of digits correctly evaluated is five, nine and ten respectively. We highlight that the fourth-order method is a good trade-off between accuracy and computational cost.

We carry out the same simulations for the butterfly shape equipped with a sharp concave curvature. We report in Fig. 33 the errors in the grid and the orders for successive refinements. We observe large errors compared to the annulus case, but the order is not quite affected with ratio in line with the expected ones. Such an example highlights the conditioning issue for the ROD method, typical when one handles strong curvatures. Note that the problem arises both for the real and the imaginary part of the magnetic field.

6.3.2. Full heat induction problem but with constant thermal conductivity

We tackle the complete heat induction problem by using the energy density derived from the magnetic field simulation. We still consider a constant conductivity $\rho = 1E-7$ Ohms, the vacuum's permeability $\mu = 1.25E-6$ H/m and the frequency $\omega = 2\pi 1E+2$

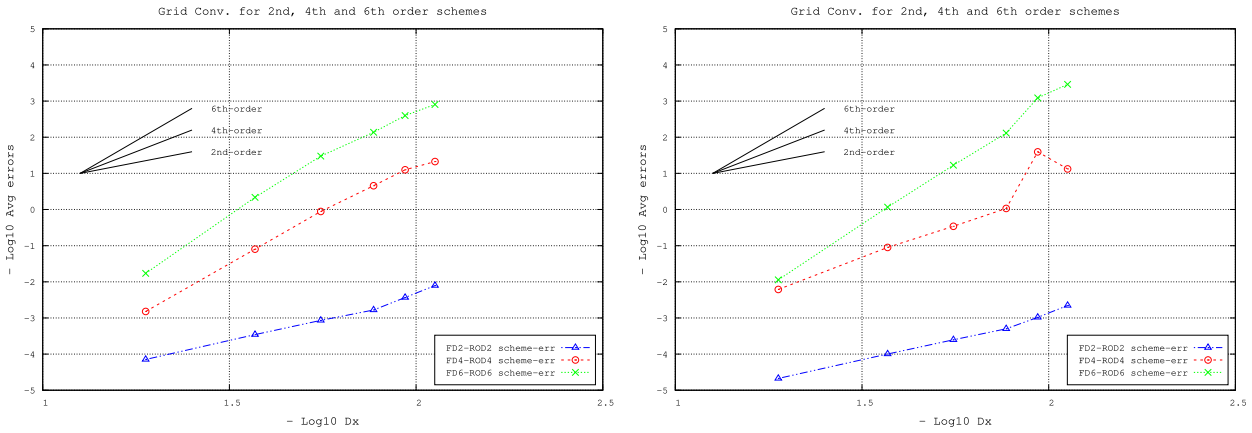


Fig. 32. Grid convergence orders of Hr (left) and Hi (right) for the 2nd, 4th and 6th order fix-point schemes on a circle domain solving the magnetic system of an electromagnetic problem.

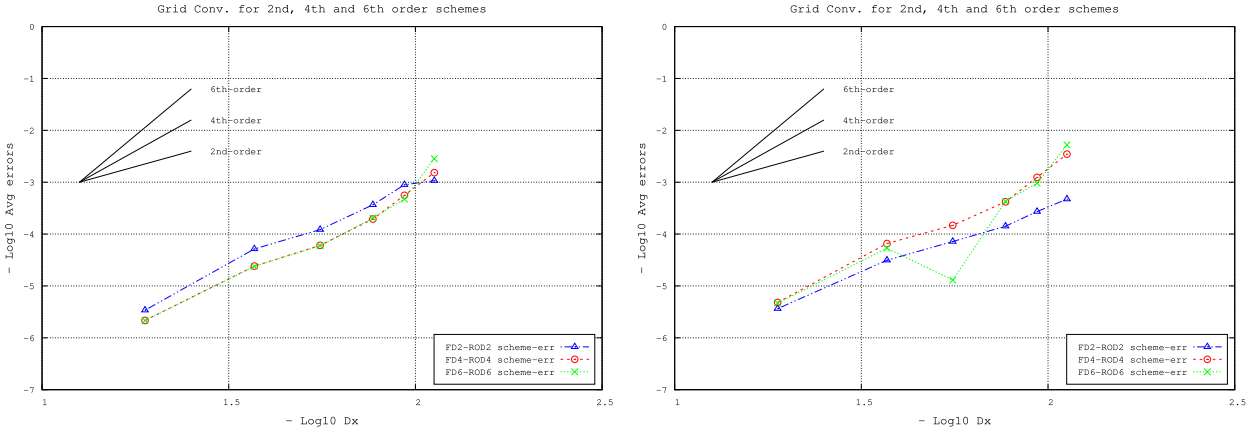


Fig. 33. Grid convergence orders of Hr (left) and Hi (right) using the 2nd, 4th and 6th order fix-point schemes on the butterfly domain solving the magnetic system of an electromagnetic problem.

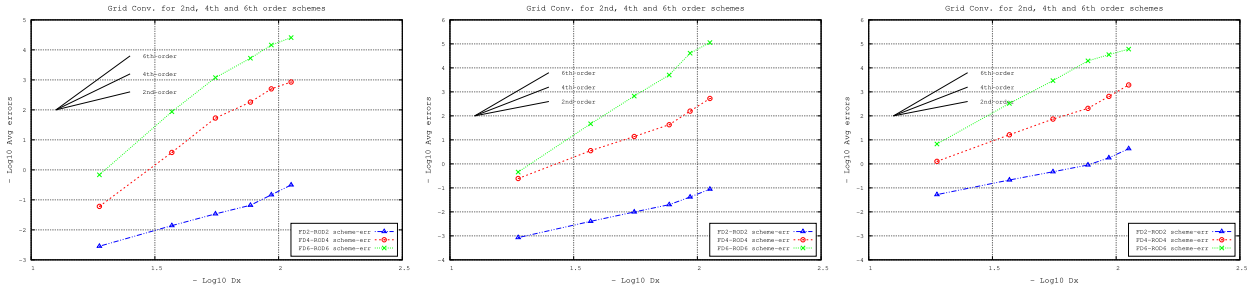


Fig. 34. L^1-errors grid convergence for Hr (left), Hi (middle) and temperature (right) using the 2nd, 4th and 6th order fix-point schemes on a circle domain solving the whole magnetic and thermal system, but with a constant conductivity σ and thermal diffusion κ.

for the magnetic problem while the thermal conductivity $\kappa = 1e+2$ is fixed, and we take $\sigma = 5.67E-8$ for the Stefan–Boltzmann constant. We prescribe $H_0 = 5E+5$ for magnetic field intensity on the boundary and take $\gamma = 4$. Notice that the problem is only one-way coupled, since the magnetic field equation does not depend on the temperature.

We print in Fig. 34 the L^1-errors and convergence rates in grid for the magnetic field H_r (left), H_i (middle) and temperature (right). We obtain effective convergence orders very close to the optimal ones. Notice that the equilibrium temperature derived from the balance between the source term involving the gradient of H and the highly nonlinear sensitive radiation flux. We did not face any convergence problem to achieve the fix-point, and the temperature approximation does not suffer from accuracy reduction that may have resulted from the quadratic term $\rho|\nabla H|^2$.

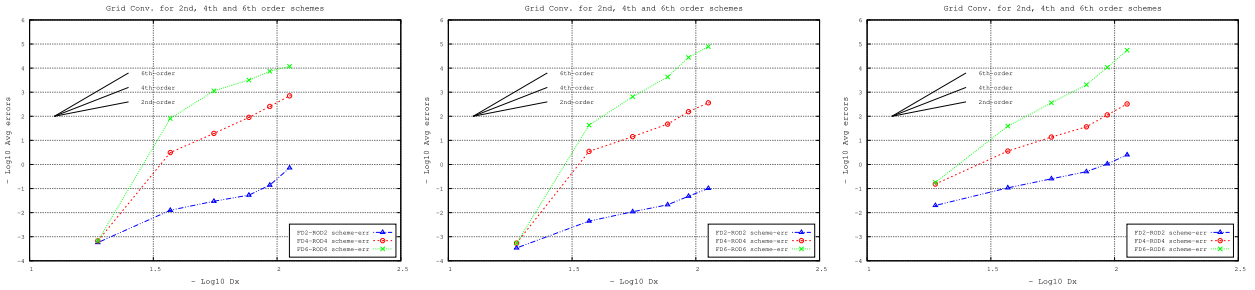


Fig. 35. L^1-error grid convergence orders of Hr (left), Hi (middle) and temperature (right) for the 2nd, 4th and 6th order fix-point schemes on a circle domain solving the fully coupled magnetic and thermal system of heat induction.

6.3.3. Full heat induction problem with variable conductivity

We finally solved the fully coupled model where the electrical conductivity depends on the temperature. For instance, we use the iron material and $\rho(T) = R_{\text{ref}}(1 + \alpha(T - T_{\text{ref}}))$ with $R_{\text{ref}} = 9.71e-8$, $\alpha = 3.86e-3$ and $T_{\text{ref}} = 293.15$. The other parameters are the same as the previous subsection, and simulation are carried for the disk.

Fig. 35 provides the error and convergence in grid for the real and imaginary magnetic field together with the temperature. We first report that the global fix-point procedure converges and the errors are quite similar to the ones obtained in the previous sub-problem. It confirms that we manage to handle curved boundary up to the sixth-order and correctly solve the problem with a very accurate approximation that considers the shape of the interfaces.

7. Conclusion

A new numerical treatment for curved boundary in the context of the finite difference method has been proposed to achieve very high-order approximation of immersed domains that do not fit with the Cartesian grid. We develop a strategy to handle non-constant coefficients for both conservative and non-conservative linear problems. An extension to the non-linear problems (both for the PDE and the boundary condition) has also been provided, and several benchmarks involving quite different systems demonstrate the efficiency of the method. The main drawback we report is the large conditioning number of the local linear systems for the ROD reconstructions, leading to a numerical limitation of the accuracy of the sixth-order method. Future research is planned to address the issue to avoid the stall of the convergence for fine grids.

CRediT authorship contribution statement

S. Clain: Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Writing – original draft. **D. Lopes:** Investigation, Software, Validation, Writing – original draft, Writing – review & editing. **Rui M.S. Pereira:** Investigation, Writing – original draft, Writing – review & editing. **Paulo A. Pereira:** Investigation, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

R.M.S. Pereira acknowledges the financial support by FEDER – Fundo Europeu de Desenvolvimento Regional, through COMPETE 2020 – Programa Operational Fatores de Competitividade, and the National Funds through FCT – Fundação para a Ciência e a Tecnologia, project no. UID/FIS/04650/2019.

P. A. Pereira acknowledges the financial support by Portuguese Funds through the Foundation for Science and Technology (FCT) in the framework of the Projects UIDB/00013/2020 and UIDP/00013/2020 of CMAT-UM.

S. Clain acknowledges the financial support by FEDER – Fundo Europeu de Desenvolvimento Regional, through COMPETE 2020 – Programa Operational Fatores de Competitividade, and the National Funds through FCT – Fundação para a Ciência e a Tecnologia, project Nº UIDB/00324/2020.

Appendix A. Non-linear solver workflow

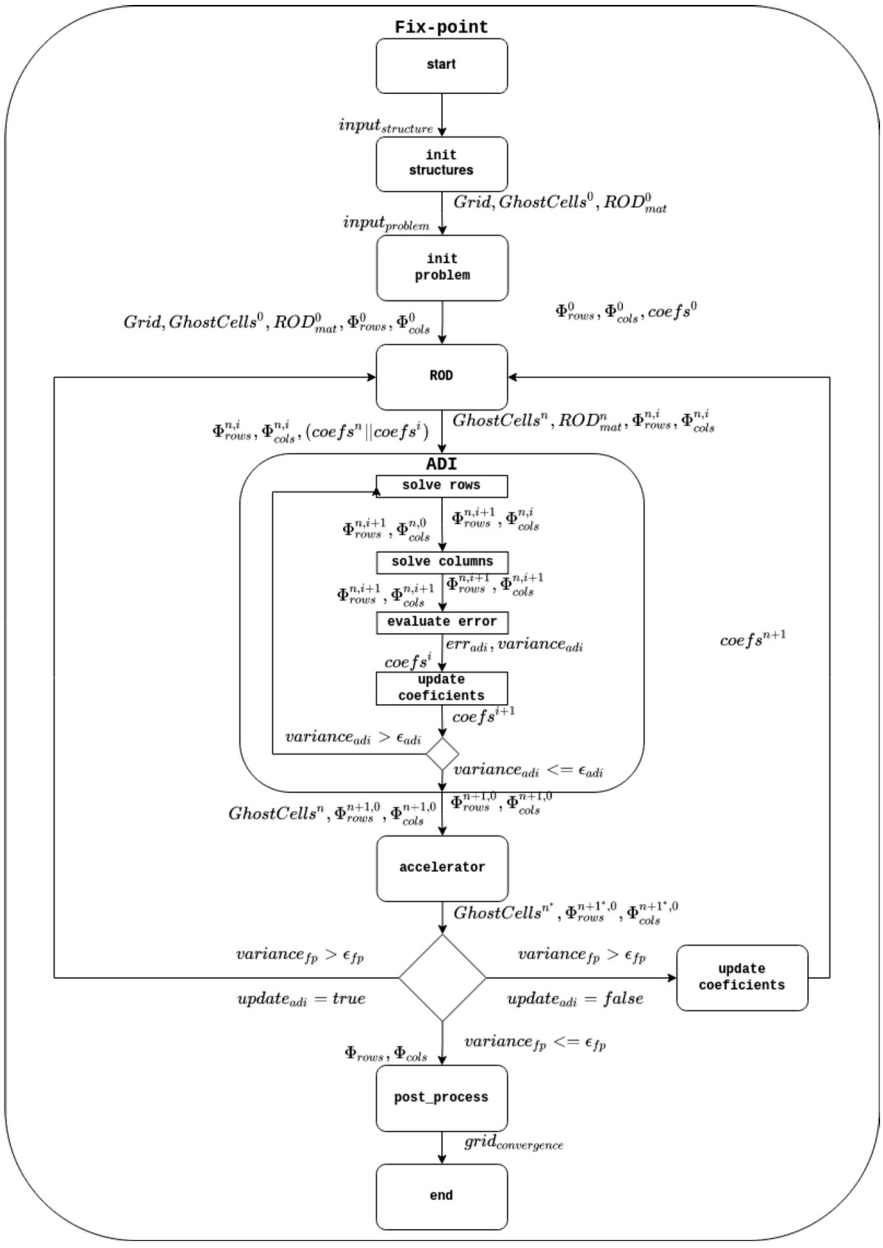


Fig. A.36. Fix-point method overview [33].

References

[1] S.P. Jammy, C.T. Jacobs, N.D. Sandham, Performance evaluation of explicit finite difference algorithms with varying amounts of computational and memory intensity, J. Comput. Phys. 36 (2019) 100565, https://doi.org/10.1016/j.jocs.2016.10.015.

[2] C. Li, G. Long, Y. Li, S. Zhao, Accelerating cfd simulation with high order finite difference method on curvilinear coordinates for modern gpu clusters, Computation 9 (79) (2021), https://doi.org/10.3390/computation9070079.

[3] A. Wakatani, A parallel and scalable algorithm for adi method with pre-propagation and message vectorization, Parallel Comput. 30 (12) (2004) 1345–1359.

[4] Z. Wei, B. Jang, Y. Zhang, Y. Jia, Parallelizing alternating direction implicit solver on gpus, Proc. Comput. Sci. 18 (2013) 389–398.

[5] D. Lopes, R. Pereira, S. Clain, High performance computation with adi on cartesian grid to solve the steady state 2d convection diffusion equation, 2019.

[6] C.-C. Ye, P.-J.-Y. Zhang, Z.-H. Wan, R. Yan, D.-J. Sun, Accelerating cfd simulation with high order finite difference method on curvilinear coordinates for modern gpu clusters, Adv. Aerodyn. 4 (7) (2022), https://doi.org/10.1186/s42774-021-00098-3.

[7] C. Peskin, Numerical analysis of blood flow in heart, J. Comput. Phys. 25 (1977) 220–252.

[8] R. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (1994) 1019–1044.

[9] Y. Zhou, S. Zhao, M. Feig, G. Wei, High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources, J. Comput. Phys. 213 (2006) 1–30.
[10] J. Nordström, M.H. Carpenter, High-order finite difference methods, multidimensional linear problems, and curvilinear coordinates, J. Comput. Phys. 173 (2001) 149–174, https://doi.org/10.1006/jcph.2001.6864.
[11] M.R. Visbal, D.V. Gaitonde, On the use of higher-order finite-difference schemes on curvilinear and deforming meshes, J. Comput. Phys. 181 (2002) 155–185, https://doi.org/10.1006/jcph.2002.7117.
[12] J. Mohd-Yusof, Combined immersed-boundary/b-spline methods for simulations of flow in complex geometries, in: Annual Research Briefs, Center for Turbulence Research, Stanford University, 1997, pp. 317–327.
[13] E. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (1) (2000) 35–60.
[14] E. Balaras, Modeling complex boundaries using an external force field on fixed cartesian grids in large-eddy simulations, Comput. Fluids 33 (3) (2004) 375–404.
[15] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.
[16] S. Majumbar, G. Iaccarino, P. Durbin, Rans solver with adaptative structured boundary non-conforming grids, in: Annu. Res. Briefs, 2001, pp. 353–364.
[17] Y.-H. Tseng, J.H. Fersiger, A ghost-cell immersed boundary method for flow in complex geometry, J. Comput. Phys. 192 (2003) 539–623.
[18] A. Gilmanov, F. Sotiropoulos, E. Balaras, A general reconstruction algorithm for simulating flows with complex 3d immersed boundaries on cartesian grids, J. Comput. Phys. 191 (2003) 660–669.
[19] A. Gilmanov, F. Sotiropoulos, A hybrid cartesian immersed boundary method for simulation flows with 3d geometrically complex moving bodies, J. Comput. Phys. 207 (2005) 457–492.
[20] A. Coco, G. Russo, Finite difference ghost point multigrid methods on cartesian grids for elliptic problems in arbitrary domains, J. Comput. Phys. 241 (2013) 464–501.
[21] F. Gibou, R. Fedkiw, A fourth order accurate discretization for the Laplace and heat equations on arbitrary domain with applications to the Stefan problem, J. Comput. Phys. 202 (2005) 577–601.
[22] D. Appelo, N.A. Peterson, A fourth-order accurate embedded boundary method for the wave equation, SIAM J. Sci. Comput. 50 (6) (2012) 2982–3008, https://doi.org/10.1137/09077223X.
[23] A. Baeza, P. Mulet, D. Zorío, High order boundary extrapolation technique for finite difference methods on complex domains with cartesian meshes, J. Sci. Comput. 66 (2016) 761–791, https://doi.org/10.1007/s10915-015-0043-2.
[24] M.E. Khalili, M. Larsson, B. Müller, High order ghost-point immersed boundary method for viscous compressible flows based on summation-by-parts operators, Int. J. Numer. Methods Fluids 89 (2019) 256–292.
[25] H. Luo, R. Mittal, X. Zheng, S. Bielamowicz, R. Walsh, J.K. Hahn, An immersed-boundary method for flow-structure interaction in biological systems with application to phonation, J. Comput. Phys. 227 (2008) 9303–9332.
[26] J.H. Seo, R. Mittal, A high-order immersed boundary method for acoustic wave scattering and low-Mach number flow-induced sound in complex geometries, J. Comput. Phys. 230 (2011) 1000–1019.
[27] J. Xia, K. Luo, J. Fan, A ghost-cell based high-order immersed boundary method for inter-phase heat trans simulation, Int. J. Heat Mass Transf. 75 (2014) 302–312.
[28] R. Costa, S. Clain, R. Loubère, G. Machado, Very high-order accurate finite volume scheme on curved boundaries for the two-dimensional steady-state convection-diffusion equation with Dirichlet condition, Appl. Math. Model. 54 (2018) 752–767.
[29] R. Costa, J.M. Nóbrega, S. Clain, G.J. Machado, R. Loubère, Very high-order accurate finite volume scheme for the convection-diffusion equation with general boundary conditions on arbitrary curved boundaries, Int. J. Numer. Methods Eng. 117 (2019) 188–220.
[30] R. Costa, J.M. Nóbrega, S. Clain, G.J. Machado, Very high-order accurate polygonal mesh finite volume scheme for conjugate heat transfer problems with curved interfaces and imperfect contacts, Comput. Methods Appl. Mech. Eng. 357 (2019) 112560.
[31] R. Costa, J.M. Nóbrega, S. Clain, G.J. Machado, Efficient very high-order accurate polyhedral mesh finite volume scheme for 3d conjugate heat transfer problems in curved domains, J. Comput. Phys. 445 (2021) 110604.
[32] R. Costa, J.M. Nóbrega, S. Clain, G.J. Machado, High-order accurate conjugate heat transfer solutions with a finite volume method in anisotropic meshes with application in polymer processing, Int. J. Numer. Methods Eng. 123 (2022) 1146–1185.
[33] S. Clain, D. Lopes, R.M.S. Pereira, Very high-order cartesian-grid finite difference method on arbitrary geometries, J. Comput. Phys. 434 (1) (2021) 110217, https://doi.org/10.1016/j.jcp.2021.110217.
[34] J. Fernández-Fidalgo, S. Clain, L. Ramírez, I. Colominas, X. Nogueira, Very high-order method on immersed curved domains for finite difference schemes with regular cartesian grids, Comput. Methods Appl. Mech. Eng. 360 (2020) 112782, https://doi.org/10.1016/j.cma.2019.112782.
[35] S. Udaykumar, R. Mittal, P. Rampunggoon, A. Khanna, A sharp interface cartesian grid method for simulating flows with complex moving boundaries, J. Comput. Phys. 174 (1) (2001) 345–380.
[36] M.B. Liu, G.R. Liu, K.Y. Lam, Constructing smoothing functions in smoothed particle hydrodynamics with applications, J. Comput. Appl. Math. 155 (2) (2003) 263–284.
[37] W.Y. Tey, N.A.C. Sidik, Y. Asako, M.W. Muhieldeen, O. Afshar, Moving least squares method and its improvement: a concise review, J. Appl. Comput. Mech. 7 (2) (2021) 883–889.
[38] N. Peller, A. Le-Duc, F. Tremblay, M. Manhart, High-order stable interpolations for immersed boundary methods, Int. J. Numer. Methods Fluids 52 (2006) 1175–1193.
[39] M. Yousefzadeh, I. Battiato, High order ghost-cell immersed boundary method for generalized boundary conditions, Int. J. Heat Mass Transf. 137 (2019) 585–598.
[40] J. Douglas, D. Peaceman, Numerical solution of two-dimensional heat-flow problems, AIChE J. 1 (1955) 505–512.
[41] J. Douglas, Alternating direction methods for three space variables, Numer. Math. 4 (1962) 41–63.
[42] R. Agarwal, D. O'Regan, Picard's Method of Successive Approximations, Springer-Verlag, New York, 2008.
[43] S. Clain, J. Rappaz, M. Swierkosz, R. Touzani, Numerical modeling of induction heating for two-dimensional geometries, Math. Models Methods Appl. Sci. 3 (6) (1993) 805–822.