PRÉ-PUBLICAÇÕES DMUC

Optimized Stencil Strategy for the Generalized Finite Difference Method: Application to Steady-State Non-Linear Problems

Stéphane Clain[a,*], Jorge Figueiredo[b]

[a] *Mathematics Center of the Coimbra University, Largo D. Dinis, Coimbra, 3000-143, Portugal*

[b] *Center of Mathematics, Campus de Gualtar, 4710-057 Braga, Portugal*

DMUC Preprint 26-34, 08 Jul 2026

Abstract

We propose an optimized stencil strategy for the Generalized Finite Difference Method (GFDM) applied to non-linear problems. We take advantage of the flexibility of GFDM to engineer specific stencils by agglomerating nodes and balancing size with numerical accuracy. Previous work, focusing on the stencil construction for the linear convection-diffusion problem, showed that optimizing stencils and scaling parameters improves the scheme. The present study aims to investigate similar benefits for non-linear problems such as the Burgers' equations and the weakly compressible Navier-Stokes system.

Keywords: General Finite Difference Method, Stencil optimization, High-order method, weakly compressible Navier-Stokes

1. Introduction

Meshless methods, *i.e.* numerical methods based on a set of points with no specific connectivity, have received great attention over the last two decades due to their ability to handle complex geometries, ability to adapt the points density in function of the local regularity and to support both Lagrangian and Eulerian formulations. Such an approach has been broken down into several methods such as Smooth Particles Hydrodynamics (SPH) [1, 2], Finite Pointset method (FPsM) [3], Finite Point method (FPtM) [4, 5, 6], or Generalized Finite Difference Method (GFDM) [7, 8, 9].

Finite Pointset and Finite Point methods use the entire neighborhood of points belonging to a compact ball, controlled by a threshold parameter included in the weighted function. Moreover, both rely on a moving least-squares method where the reconstruction is achieved at any point (not necessarily a point of the cloud). On the contrary, the Generalized Finite Difference Method depends on particular stencils (not all the points of the neighborhood are necessarily used), and a classical least-squares technique centered on the cloud points is used to derive relations between derivatives and the function values over the stencil. Consequently, the GFDM provides an additional advantage over FPsM and FPtM, the ability to play with the stencil choice and, by that way, obtain the best trade-off between stencil size and accuracy.

*corresponding author

Email addresses: clain@mat.uc.pt (Stéphane Clain), jmfiguei@math.uminho.pt (Jorge Figueiredo)

[v1] Wed, 08 Jul 2026

In [10], the authors take advantage of several properties inherent to the local representation in the GFDM. The choice of the stencil together with the parameterization of the weight function and the polynomial's scaling parameter enable to optimize the quality of the representation of the derivatives while preserving small-sized stencils. The study was developed for the linear convection diffusion operator in two-dimensional space, where computational cost reduction and better accuracy for a given cloud of points are reported. The question then arises as to whether such gains are still present for non-linear problems. To this end, we conduct a new study on two very popular systems, the viscous Burgers' equation [11] and the weakly compressible Navier-Stokes equations (Tait equation of state). We do not integrate any upwind techniques [12, 13] since we carry out the simulations with low Reynolds numbers.

2. Design of the numerical method

Let Ω be a bounded open set of $\mathbb{R}^2$ with boundary $\partial\Omega$ and denote by $\boldsymbol{x} = (x_1, x_2)$ a generic point of $\overline{\Omega}$. We denote by ϕ any smooth function defined on $\overline{\Omega}$ and sufficiently regular such that the first- and second-order derivatives are continuously bounded on $\overline{\Omega}$. We restrict the presentation to the two-dimensional geometries for the sake of simplicity, but extension to higher dimensions is straightforward.

2.1. Cloud of points, approximations, and stencil

A cloud consists of a finite set of points $\boldsymbol{x}_i \in \overline{\Omega}$ characterized by the indices set $\mathcal{G} = \{1, \ldots, I\}$ which we split into $\mathring{\mathcal{G}}$, the indices of the points that belong to the open set Ω and $\partial\mathcal{G}$ the indices of points on the boundary $\partial\Omega$. For any node $i \in \mathcal{G}$, we denote by $\mathcal{S}_i$ the associated stencil constituted by a sublist of indices from $\mathcal{G}$ such that $i \in \mathcal{S}_i$, while $|\mathcal{S}_i|$ stands for the stencil size.

We introduce the multi-index notation $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$, with $\alpha_1, \alpha_2 \in \mathbb{N}_0$ with $|\boldsymbol{\alpha}| = \alpha_1 + \alpha_2$ and $\boldsymbol{\alpha}! = (\alpha_1!)(\alpha_2!)$, while $\boldsymbol{x}^{\boldsymbol{\alpha}} = (x_1)^{\alpha_1}(x_2)^{\alpha_2}$. For $i \in \mathcal{G}$ we also define the monomial function of degree $\boldsymbol{\alpha}$ centred at point $\boldsymbol{x}_i$ with

$$\pi_{\boldsymbol{\alpha}}(\boldsymbol{x}; \boldsymbol{x}_i, h_i) = \left(\frac{\boldsymbol{x} - \boldsymbol{x}_i}{h_i}\right)^{\boldsymbol{\alpha}}$$

where $h_i > 0$ is a scaling parameter.

Given $\boldsymbol{\beta} = (\beta_1, \beta_2)$, with $\beta_1, \beta_2 \in \mathbb{N}_0$, we associate the partial derivative $\partial^{\boldsymbol{\beta}}\phi = \partial_{x_1}^{\beta_1}\partial_{x_2}^{\beta_2}\phi(x_1, x_2)$. Inequality $\boldsymbol{\alpha} \geq \boldsymbol{\beta}$ means that $\alpha_1 \geq \beta_1$ and $\alpha_2 \geq \beta_2$. In particular, if $\boldsymbol{\alpha} \geq \boldsymbol{\beta}$, one has

$$\partial^{\boldsymbol{\beta}}\pi_{\boldsymbol{\alpha}}(\boldsymbol{x}; \boldsymbol{x}_i, h_i) = \frac{\boldsymbol{\alpha}!}{(\boldsymbol{\alpha} - \boldsymbol{\beta})!} \frac{1}{(h_i)^{|\boldsymbol{\beta}|}} \pi_{\boldsymbol{\alpha}-\boldsymbol{\beta}}(\boldsymbol{x}; \boldsymbol{x}_i, h_i),$$

while $\partial^{\boldsymbol{\beta}}\pi_{\boldsymbol{\alpha}}(\boldsymbol{x}; \boldsymbol{x}_i, h_i) = 0$, otherwise. Note that taking $\boldsymbol{x} = \boldsymbol{x}_i$ provides the relations

$$\partial^{\boldsymbol{\alpha}}\pi_{\boldsymbol{\alpha}}(\boldsymbol{x}_i; \boldsymbol{x}_i, h_i) = \frac{\boldsymbol{\alpha}!}{(h_i)^{|\boldsymbol{\alpha}|}} \quad \text{and} \quad \partial^{\boldsymbol{\beta}}\pi_{\alpha}(\boldsymbol{x}_i; \boldsymbol{x}_i, h_i) = 0, \text{ otherwise.} \tag{1}$$

2.2. Polynomial degree set and derivative order set

The polynomial degree set $\mathcal{A}$. The space $\mathbb{P}_i^{\mathcal{A}}$ is constituted by the span of the monomial family

$$\mathbb{P}_i^{\mathcal{A}} = \text{span}\big\{\pi_{\boldsymbol{\alpha}}(\boldsymbol{x}; x_i, h_i),\ \boldsymbol{\alpha} \in \mathcal{A}\big\}$$

we shall call the $\mathcal{A}$-polynomial space centred at node i.

Remark 1. For the particular case where $\mathcal{A} = \{\boldsymbol{\alpha},\ |\boldsymbol{\alpha}| \leq p\}$, the set simply reads $\mathbb{P}^p$ since the reference point has no longer importance. Indeed, the centred basis is equivalent to the canonical one. For example, for the polynomial basis of degree $p = 2$, we have a basis attached to node i

$$\mathbb{P}^2 = \text{span}\Big\{\pi_\alpha(\boldsymbol{x}; \boldsymbol{x}_i, h_i),\ \boldsymbol{\alpha} \in \mathcal{A} = \big\{(0,0),(1,0),(0,1),(2,0),(1,1),(0,2)\big\}\Big\}.$$

In that case, the reference node has no importance since we can rewrite the polynomial space with the canonical monomial. ■

Consequently, any polynomial function of $\mathbb{P}_i^{\mathcal{A}}$ is written under the form

$$\pi(\boldsymbol{x}, \boldsymbol{c}_i; \boldsymbol{x}_i) = \pi(\boldsymbol{x}, \boldsymbol{c}_i; \boldsymbol{x}_i, h_i, \mathcal{A}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} c_{i,\boldsymbol{\alpha}} \pi_{\boldsymbol{\alpha}}(\boldsymbol{x}; \boldsymbol{x}_i, h_i),$$

where the vector $\boldsymbol{c}_i$ collects the entries $\big(c_{i,\boldsymbol{\alpha}}\big)_{\boldsymbol{\alpha} \in \mathcal{A}}$.

The set $\mathcal{B}$ contains the list of derivatives $\boldsymbol{\beta}$ we aim at discretizing via differential operators. For example, $\big\{\partial^0_{x_1}\partial^0_{x_2},\ \partial^1_{x_1}\partial^0_{x_2},\ \partial^0_{x_1}\partial^1_{x_2},\ \partial^1_{x_1}\partial^1_{x_2}\big\} \iff \mathcal{B} = \big\{(0,0),(1,0),(0,1),(1,1)\big\}$.

Finally, for any regular function ϕ, we denote $\phi_i \approx \phi(\boldsymbol{x}_i)$ an approximation of ϕ at point $\boldsymbol{x}_i$ and for any partial derivative $\partial^{\boldsymbol{\beta}}$, we use the notation $\phi_i^{\boldsymbol{\beta}} \approx \partial^{\boldsymbol{\beta}}\phi(\boldsymbol{x}_i)$. We then organize the data following two vectors:

- $\Phi_{\mathcal{S}_i} = (\phi_j)_{j \in \mathcal{S}_i}$ the approximation vector associated with the stencil $\mathcal{S}_i$;
- $\Phi = \Phi_{\mathcal{G}} = (\phi_i)_{i \in \mathcal{G}}$ the vector that gathers all the approximations associated to the cloud $\mathcal{G}$.

2.3. Interpolation and differential operators' discretization

Given a stencil $\mathcal{S}_i$ associated to a node $i \in \mathcal{G}$, $\Phi_{\mathcal{S}_i}$ the vector of approximations over the stencil, and $\mathbb{P}_i^{\mathcal{A}}$ the polynomial subspace, we define the following functional for any $\pi \in \mathbb{P}_i^{\mathcal{A}}$,

$$E(\boldsymbol{c}_i; \mathcal{S}_i, \Phi_{\mathcal{S}_i}, \boldsymbol{w}_i, \mathcal{A}, h_i) = \sum_{j \in \mathcal{S}_i} w_{ij} \Big[\pi(\boldsymbol{x}_j, \boldsymbol{c}_i; \boldsymbol{x}_i, h_i, \mathcal{A}) - \phi_j\Big]^2,$$

where $\boldsymbol{w}_i = (w_{ij})_{j \in \mathcal{S}_i}$ contains the weights associated to each point of the stencil $\mathcal{S}_i$. A usual choice is $w_{ij} = \rho(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|; \tau_i)$, where $\rho(s; \tau)$ is a decreasing function of $s \in [0, +\infty[$ controlled by a scaling parameter τ. For instance, the Gaussian weights read

$$w_{ij} = \rho(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|; \tau_i) = \exp\left(-\frac{\|\boldsymbol{x}_j - \boldsymbol{x}_i\|^2}{\tau_i^2}\right).$$

We define the $|\mathcal{A}| \times |\mathcal{S}_i|$ matrix $M_i = \left(m_{i;\boldsymbol{\alpha},j}\right)_{j\in\mathcal{S}_i}^{\boldsymbol{\alpha}\in\mathcal{A}}$ with

$$m_{i;\boldsymbol{\alpha},j} = \pi_{\boldsymbol{\alpha}}(\boldsymbol{x}_j;\boldsymbol{x}_i,h_i) = \left(\frac{\boldsymbol{x}_j-\boldsymbol{x}_i}{h_i}\right)^{\boldsymbol{\alpha}}, \quad \boldsymbol{\alpha}\in\mathcal{A},\ j\in\mathcal{S}_i.$$

Using the Euclidean norm, the functional rewrites under the form

$$E_i(\boldsymbol{c}_i;\mathcal{S}_i,\Phi_{\mathcal{S}_i},\boldsymbol{w}_i,\mathcal{A},h_i) = \|W_i(M_i\boldsymbol{c}_i-\Phi_{\mathcal{S}_i})\|^2,$$

with $W_i = \mathrm{diag}(\boldsymbol{w}_i)$ the $|\mathcal{S}_i| \times |\mathcal{S}_i|$ diagonal matrix of the weights.

Minimization of the functional with respect to vector $\boldsymbol{c}_i$ leads to the linear system

$$(M_i)^{\mathrm{t}}W_iM_i\,\boldsymbol{c}_i = (M_i)^{\mathrm{t}}W_i\Phi_{\mathcal{S}_i}.$$

Assuming that the choice of $\mathcal{A}$ and $\mathcal{S}_i$ provides a matrix M_i that satisfies the maximal rank property, then $M_i^{\dagger} = (M_i)^{\mathrm{t}}W_iM_i$ is a non-singular $|\mathcal{A}| \times |\mathcal{A}|$ matrix and the polynomial coefficients are given by $\boldsymbol{c}_i = M_i^{\ddagger}\Phi_{\mathcal{S}_i}$ with

$$M_i^{\ddagger} = \left(M_i^{\dagger}\right)^{-1}(M_i)^{\mathrm{t}}W_i.$$

Consequently, we deduce the coefficient of a $\mathcal{A}$-polynomial approximation given by

$$\widetilde{\phi}_i(\boldsymbol{x}) = \phi_i(\boldsymbol{x},\boldsymbol{c}_i;\boldsymbol{x}_i,\mathcal{S}_i,h_i,\mathcal{A}) = \sum_{\alpha\in\mathcal{A}} c_{i,\boldsymbol{\alpha}}\pi_{\boldsymbol{\alpha}}(\boldsymbol{x};\boldsymbol{x}_i,h_i).$$

Under the condition $\mathcal{B}\subset\mathcal{A}$, we get

$$\partial^{\boldsymbol{\beta}}\pi_{\alpha}(\boldsymbol{x}_i) = \delta_{\boldsymbol{\alpha},\boldsymbol{\beta}}\frac{\boldsymbol{\alpha}!}{(h_i)^{\boldsymbol{\alpha}}}.$$

and deduce

$$\partial^{\beta}\widetilde{\phi}_i(\boldsymbol{x}_i) = \frac{\boldsymbol{\beta}!}{(h_i)^{\boldsymbol{\beta}}}c_{i,\boldsymbol{\beta}} = \frac{\boldsymbol{\beta}!}{(h_i)^{|\boldsymbol{\beta}|}}\sum_{j\in\mathcal{S}_i} m_{i;\boldsymbol{\beta},j}^{\ddagger}\phi_j = \sum_{j\in\mathcal{S}_i} a_{i;\boldsymbol{\beta},j}\phi_j,\ \text{with } a_{i;\boldsymbol{\beta},j} = \frac{\boldsymbol{\beta}!}{(h_i)^{|\boldsymbol{\beta}|}}\,m_{i;\boldsymbol{\beta},j}^{\ddagger}. \quad (2)$$

Relation (2) provides the discretization of the partial derivatives, in terms of the stencil-node values, which we use in the physical equations.

Remark 2 (Pointset method versus generalized finite difference method). There exists a subtle difference between the two methods when dealing with the reconstruction of $\phi_i(x_i)$. Indeed, we have two ways. We take $\phi_i(x_i) = \phi_i$ (Generalized Finite Difference Method) or we take $\phi_i(x_i) = \sum_{j\in\mathcal{S}_i} a_{i;(0,0),j}\phi_j$ (Pointset Diffuse approximation). The rest of the derivatives' approximation for $\boldsymbol{\beta} \neq (0,0)$ are all the same.

Example 1. Consider the equation $-\Delta\phi = f$. Then the discretization at node i simply reads

$$-\sum_{j\in\mathcal{S}_i}(a_{i;(2,0),j}+a_{i;(2,0),j})\phi_j = f_i.$$

Similarly, the boundary condition $\nabla\phi \cdot \boldsymbol{n} = g$ at node i reads

$$\sum_{j\in\mathcal{S}_i} (a_{i;(1,0),j} n_{i,1} + a_{i;(0,1),j} n_{i,2})\phi_j = g_i.$$

2.4. Discretization optimization

Given a point $\boldsymbol{x}_i \in \mathcal{G}$, matrix $M_i^{\ddagger} = M_i^{\ddagger}(h_i, \tau_i, \mathcal{S}_i)$ depends on the polynomial scaling factor h_i, the weights scaling factor τ_i, and the stencil $\mathcal{S}_i$. In [10] we develop an optimization procedure based on the condition number of matrix $M_i^{\dagger}$ since it corresponds to the non-singular matrix we have to inverse to determine the derivatives. We then introduce a quantification of the polynomial reconstruction efficiency by computing the condition number $\chi_2(M_i^{\dagger})$ and we aim at determining the parameters that minimize it. In other words, we seek

$$(h_i^{\star}, \tau_i^{\star}, \mathcal{S}_i^{\star}) = \arg\min\, \chi_2(M_i^{\dagger}(h_i, \tau_i, \mathcal{S}_i))$$

under the constraint $|\mathcal{S}_i| \geq |\boldsymbol{\alpha}|$, since we need more points (*i.e.*, equations) than unknown polynomial coefficients.

Optimization over a discrete set $\mathcal{S}$ and continuous sets $]0, +\infty[\times]0, +\infty[$ is a difficult task, so the authors combine a heuristic approach for the stencil choice along with a gradient method for the continuous parameters. In this way, we obtain a rather small stencil ($|\mathcal{A}| \leq |\mathcal{S}_i| \leq 1.5|\mathcal{A}|$) with a dramatic cut in the condition number in comparison with an unsophisticated stencil consisting of the closest points to the reference point $\boldsymbol{x}_i$ (about twice the size of $\mathcal{A}$ to produce a non-singular matrix $M_i^{\dagger}$).

Remark 3. In our previous work the stencil associated with each reference point used to start the heuristic approach is constructed by selecting the $2|\mathcal{A}|$ closest neighbour points. For linear PDEs the strategy enables us to achieve optimal convergence rates, but for certain non-linear problems it occasionally results in suboptimal convergence rates. To improve robustness, we select the $3|\mathcal{A}|$ closest points and discard $|\mathcal{A}|$ of them to avoid poor angular distribution (e.g., near collinearity with respect to the reference point). The resulting set of $2|\mathcal{A}|$ points has a more uniform angular spread, and this enriched initial configuration is then used as the input for the subsequent heuristic stencil-optimization procedure.

Since we report excellent orders of accuracy for linear PDE problems on complex domains (curved boundary domains) in [10], we aim to extend the same framework to non-linear PDE problems and systems by benchmarking several important non-linear problems and assessing its efficiency in terms of both accuracy and stability.

3. The two-dimensional Burgers' equations

The main goal is to assess the accuracy improvement achieved by the optimized stencil, as well as the robustness of the method. The linear case has been studied in [10] where we showed the advantages and gains of the optimization, but the question arises when we are dealing with non-linear problems.

3.1. Modeling and linearization

We consider the two-dimensional stationary forced Burgers' equations with viscous term,

$$\begin{aligned} u_1\,\partial_{x_1}u_1 + u_2\,\partial_{x_2}u_1 - \varepsilon\big(\partial^2_{x_1}u_1 + \partial^2_{x_2}u_1\big) &= f_1, \quad \text{in } \Omega,\\ u_1\,\partial_{x_1}u_2 + u_2\,\partial_{x_2}u_2 - \varepsilon\big(\partial^2_{x_1}u_2 + \partial^2_{x_2}u_2\big) &= f_2, \quad \text{in } \Omega, \end{aligned}$$

where $\boldsymbol{u} = (u_1, u_2)$ is the unknown velocity, $\boldsymbol{f} = (f_1, f_2)$ the source term, and $\varepsilon > 0$ the viscosity parameter. The general Robin boundary conditions are given by

$$\mu_1(\boldsymbol{x}_i)\,u_1 + \nu_1(\boldsymbol{x}_i)\nabla u_1 \cdot \boldsymbol{n} = g_1(\boldsymbol{x}_i), \quad \mu_2(\boldsymbol{x}_i)\,u_2 + \nu_2(\boldsymbol{x}_i)\nabla u_2 \cdot \boldsymbol{n} = g_2(\boldsymbol{x}_i), \quad \text{on } \partial\Omega,$$

where $\boldsymbol{\mu} = (\mu_1, \mu_2)$, $\boldsymbol{\nu} = (\nu_1, \nu_2)$, and $\boldsymbol{g} = (g_1, g_2)$ are given functions and $\boldsymbol{n} = \boldsymbol{n}(\boldsymbol{x})$ is the outward unit normal vector. Notice that the velocity components are not coupled on the boundary. In this context, we define for any node $i \in \mathcal{G}$ the label $\ell(i)$ that indicates the physical equation associated with the node. For example, $\ell(i) = 1$ means that one has to satisfy the inner equations, while $\ell(i) = 2$ corresponds to the case where the boundary equations hold.

To solve the non-linear system, we use a simple fixed-point method based on a local linearization where we build a sequence $\boldsymbol{u}^{[k]} = (u_1^{[k]}, u_2^{[k]})$, $k = 0, 1, \ldots$, of approximations of the solution $\boldsymbol{u}$ on $\overline{\Omega}$. To this end, assuming $(u_1^{[k]}, u_2^{[k]})$ is known, we seek a solution $(u_1^{[k+1]}, u_2^{[k+1]})$ of the linearized problem

$$\begin{aligned} u_1^{[k]}\,\partial_{x_1}u_1^{[k+1]} + u_2^{[k]}\,\partial_{x_2}u_1^{[k+1]} - \varepsilon\Big(\partial^2_{x_1}u_1^{[k+1]} + \partial^2_{x_2}u_1^{[k+1]}\Big) &= f_1,\\ u_1^{[k]}\,\partial_{x_1}u_2^{[k+1]} + u_2^{[k]}\,\partial_{x_2}u_2^{[k+1]} - \varepsilon\Big(\partial^2_{x_1}u_2^{[k+1]} + \partial^2_{x_2}u_2^{[k+1]}\Big) &= f_2. \end{aligned}$$

Note that the problem is decoupled and one solves for the velocities u_1 and u_2 independently. The iterative procedure stops when both $\|u_1^{[k+1]} - u_1^{[k]}\|$ and $\|u_2^{[k+1]} - u_2^{[k]}\|$ are lower than a given tolerance. Convergence of the fixed-point is not guaranteed, in particular for large Péclet numbers, since the stencil choice does not take into account the upwind direction, leading to a centered scheme for the convective term. Therefore, benchmarks have been carried out with a relatively low Péclet number to focus on the accuracy and convergence order of the numerical scheme.

3.2. Discretization and assembly

We denote by U_1 and U_2 the vectors of $\mathbb{R}^I$ with the numerical solution of the non-linear system, that is,

$$U_1 = (u_{1,i})_{i\in\mathcal{G}}, \quad U_2 = (u_{2,i})_{i\in\mathcal{G}}.$$

Moreover, for any point $\boldsymbol{x}_i$, $i \in \mathcal{G}$, and $\phi = u_1, u_2$, we have $\phi(\boldsymbol{x}_i) \approx \phi_i^{(0,0)} = \phi_i$, and we use relations (2) for the discrete derivatives, that is

$$\partial_{x_1}\phi(\boldsymbol{x}_i) \approx \phi_i^{(1,0)} = \sum_{j\in\mathcal{S}_i} a_{i;(1,0),j}\phi_j, \quad \partial_{x_2}\phi(\boldsymbol{x}_i) \approx \phi_i^{(0,1)} = \sum_{j\in\mathcal{S}_i} a_{i;(0,1),j}\phi_j, \tag{3}$$

$$\partial^2_{x_1}\phi(\boldsymbol{x}_i) \approx \phi_i^{(2,0)} = \sum_{j\in\mathcal{S}_i} a_{i;(2,0),j}\phi_j, \quad \partial^2_{x_2}\phi(\boldsymbol{x}_i) \approx \phi_i^{(0,2)} = \sum_{j\in\mathcal{S}_i} a_{i;(0,2),j}\phi_j. \tag{4}$$

Since the simulation is carried out using a simple fixed-point method, we construct a sequence of approximations using the following iterative procedure: given $(U_1^{[k]}, U_2^{[k]})$, we compute $(U_1^{[k+1]}, U_2^{[k+1]})$, the approximation of the associated continuous linearized problem at iteration $k+1$:

$$u_{1,i}^{[k],(0,0)}\, u_{1,i}^{[k+1],(1,0)} + u_{2,i}^{[k],(0,0)}\, u_{1,i}^{[k+1],(0,1)} - \varepsilon\left(u_{1,i}^{[k+1],(2,0)} + u_{1,i}^{[k+1],(0,2)}\right) = f_1(\boldsymbol{x}_i),$$
$$u_{1,i}^{[k],(0,0)}\, u_{2,i}^{[k+1],(1,0)} + u_{2,i}^{[k],(0,0)}\, u_{2,i}^{[k+1],(0,1)} - \varepsilon\left(u_{2,i}^{[k+1],(2,0)} + u_{2,i}^{[k+1],(0,2)}\right) = f_2(\boldsymbol{x}_i).$$

Replacing the derivatives by the local representation given by (3) and (4), we then assemble two independent linear systems $A_1^{[k]}U_1 = b_1$ and $A_2^{[k]}U_2 = b_2$. For example, the assembled matrix corresponding to vector U_1 is obtained as follows:

- if $\ell(i) = 1$, then for $j \in \mathcal{S}_i$ we set

$$A_1^{[k]}[i,j] = u_{1,j}^{[k]}\, a_{i;(1,0),j} + u_{2,j}^{[k]}\, a_{i;(0,1),j} - \varepsilon\Big(a_{i;(2,0),j} + a_{i;(0,2),j}\Big)$$

  and $A_1^{[k]}[i,j] = 0$ otherwise. The right-hand side is simply $b_1[i] = f_1(\boldsymbol{x}_i)$.

- if $\ell(i) = 2$, then for $j \in \mathcal{S}_i$ the discrete Robin condition provides the following entries

$$A_1^{[k]}[i,j] = \delta_{i,j}\mu_1(\boldsymbol{x}_i) + \nu_1(\boldsymbol{x}_i)(a_{i;(1,0),j}n_{i;1} + a_{i;(0,1),j}n_{i;2}),$$

  and $A_1^{[k]}[i,j] = 0$ otherwise. The right-hand side is given by $b_1[i] = g_1(\boldsymbol{x}_i)$. Note that whenever the Dirichlet conditions hold, one has $A_1^{[k]}[i,j] = \delta_{i,j}$ and $b_1[i] = u_{D_1}(\boldsymbol{x}_i)$.

We update the numerical approximation by computing $U_1^{[k+1]}$ and $U_2^{[k+1]}$, solutions of $A_1^{[k]}\, U_1^{[k+1]} = b_1$ and $A_2^{[k]}\, U_2^{[k+1]} = b_2$, respectively.

3.3. Test case 3.1

In this first test case devoted to the two-dimensional Burgers' equations, we prescribe the Dirichlet boundary conditions for both horizontal and vertical velocities, that is $\boldsymbol{\mu}(\boldsymbol{x}_i) = \mathbf{1}$, $\boldsymbol{\nu}(\boldsymbol{x}_i) = \mathbf{0}$, for all $i \in \partial\mathcal{G}$. We further take $\varepsilon = 10^{-2}$ and consider the following irregular domain,

$$\Omega = \left\{\boldsymbol{x} \in \,]-1,1[\,\times\,]-1,1[\,: x_1^2 + x_2^2 > 0.3^2 \wedge |x_1 x_2| < 0.3\right\}.$$

In order to perform a convergence analysis we set $\boldsymbol{f} = (f_1, f_2)$ and $\boldsymbol{u}_D = (u_{D_1}, u_{D_2})$ such that

$$u_1(\boldsymbol{x}) = -\frac{\varepsilon x_2}{x_1^2 + x_2^2}, \qquad u_2(\boldsymbol{x}) = \frac{\varepsilon x_1}{x_1^2 + x_2^2}$$

is the unique solution of the boundary value problem (BVP). We note that this solution is invariant to rotation with respect to the origin of the coordinate system. The resulting expression for (f_1, f_2) is

$$f_1(\boldsymbol{x}) = -\frac{\varepsilon^2 x_1}{\left(x_1^2 + x_2^2\right)^2}, \qquad f_2(\boldsymbol{x}) = \frac{\varepsilon^2 x_2}{\left(x_1^2 + x_2^2\right)^2}.$$

The initial velocity for the fixed-point method, $(U_1^{[0]}, U_2^{[0]})$, is taken equal to the average value over the nodes located on $\partial\Omega$. The geometrical setting and the exact solution of the BVP are illustrated in Figure 1.

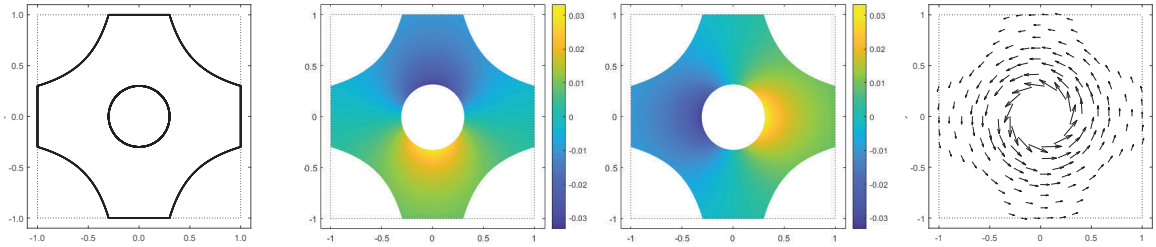

Figure 1: *Test case 3.1. Geometrical setting (left panel) and exact solution (from left to right: u_1, u_2, $\boldsymbol{u}$).*

Numerical simulations are carried out with 622, 1228, 2458, and 4886 nodes, obtained with a Delaunay triangulation procedure using the Gmsh© package. We note that the number of nodes on the inner and outer boundaries is almost the same. Figure 2 illustrates the different clouds used in the present test case.

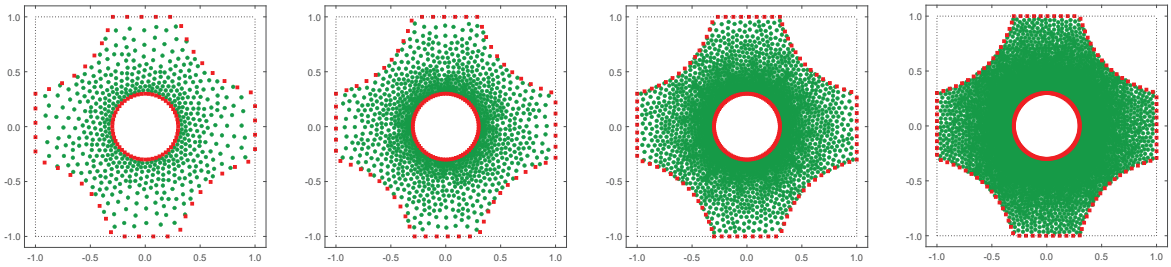

Figure 2: *Test case 3.1. For $\overline{\Omega}$ the panels illustrate, from left to right, four clouds obtained with a Delaunay triangulation procedure with* 622, 1288, 2458, *and* 4886 *nodes. The green filled circles and the red squares correspond, respectively, to interior nodes and boundary nodes where Dirichlet conditions are imposed.*

We present the numerical solution error (using L^∞- and L^1-norms) and the corresponding average convergence order (ACO) for $p = 2, \ldots, 6$ in Figure 3.

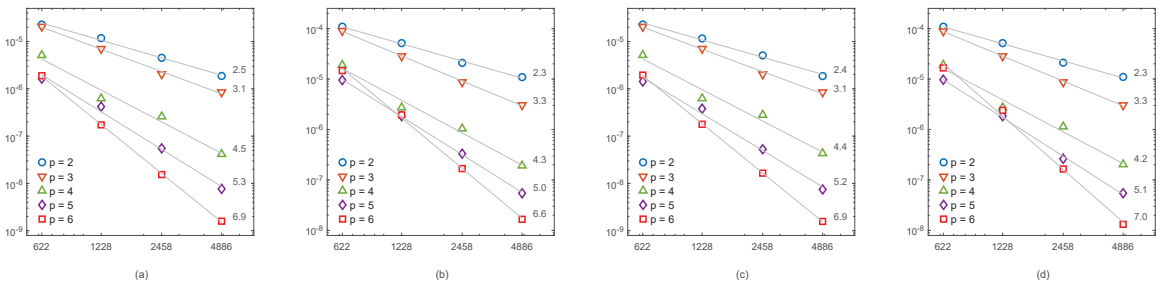

Figure 3: *Results for the test case 3.1 for $p = 2, \ldots, 6$. The L^1- and L^∞- numerical solution errors for u_1 (panels (a) and (b), respectively) and u_2 (panels (c) and (d), respectively) are presented for a set of four grids with* 622, 1288, 2458, *and* 4886 *nodes. The corresponding average convergence order is also presented for each p.*

Looking at ACO, we conclude that the optimal convergence order is obtained (or even exceeded) for both the L^∞- and the L^1-errors, independently of the approximation degree. We also report

that the ratio L^∞-error / L^1-error obtained for $p = 2, \ldots, 6$ (4.9 ± 0.5, 4.1 ± 0.2, 4.2 ± 0.4, 5.9 ± 0.9, and 9.8 ± 2.0, respectively) is well bounded, indicating the good quality of the numerical solution across the computational domain for all values of p.

3.4. Test case 3.2

We consider once again the two-dimensional Burgers' equations supplemented by Dirichlet boundary conditions over a non-polygonal domain expressed in polar coordinates with respect to the origin of the coordinate system (see Figure 4):

$$\Omega = \left\{ (r, \theta) \in \mathbb{R}^+ \times [0, 2\pi[\, : 0.3 \leq r \leq 0.95 + 0.05 \sin(12\theta + \pi/2) \right\}.$$

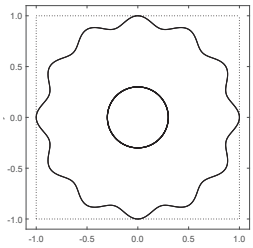

Figure 4: *Test case 3.2. Geometrical setting.*

In order to perform a convergence analysis, we set $\boldsymbol{f}$ and $\boldsymbol{u}_D$ such that the unique solution of the BVP is given by

$$u_1(r, \theta) = -q(r, \varepsilon) \sin\theta, \qquad u_2(r, \theta) = q(r, \varepsilon) \cos\theta,$$

with

$$q(r, \varepsilon) = \frac{r_4}{\varepsilon} \exp\left[\frac{r_3 \varepsilon}{(r - r_1)(r - r_2)} \right],$$

where we adopt $\varepsilon = 10^{-2}$, while r_1, r_2, r_3, and r_4 are free parameters we shall set in the sequel. The solution of the BVP is invariant to rotation with respect to the origin of the coordinate system and we obtain

$$f_1(r, \theta) = -\frac{q^2}{r} \cos\theta + \varepsilon \left(\frac{\partial^2 q}{\partial r^2} + \frac{1}{r} \frac{\partial q}{\partial r} - \frac{q}{r^2} \right) \sin\theta,$$
$$f_2(r, \theta) = -\frac{q^2}{r} \sin\theta - \varepsilon \left(\frac{\partial^2 q}{\partial r^2} + \frac{1}{r} \frac{\partial q}{\partial r} - \frac{q}{r^2} \right) \cos\theta.$$

Test case 3.2a. We start the numerical simulations by setting $r_1 = 3$, $r_2 = 0.225$, $r_3 = 100$, and $r_4 = 2 \times 10^{-4}$. The corresponding shape for the function q (leading to a shallow slope curve) and the exact solution of the BVP are depicted in Figure 5.

For the numerical simulations, we use four grids with 674, 1316, 2533, and 5118 nodes, obtained with a Delaunay triangulation procedure. The number of nodes in the inner boundary doubles that of its outer counterpart. Figure 6 illustrates the different grids used in the test case 3.2a numerical simulations. The nonlinearity is addressed using a fixed-point approach, as described previously.

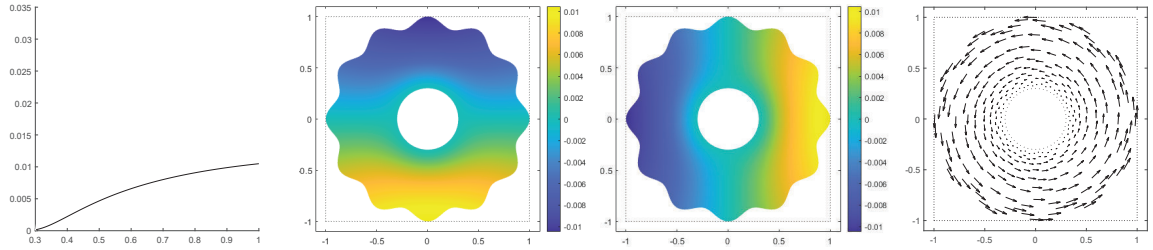

Figure 5: *Test case 3.2a. Shape of function q (left panel) and exact solution (from left to right: u_1, u_2, $\boldsymbol{u}$).*

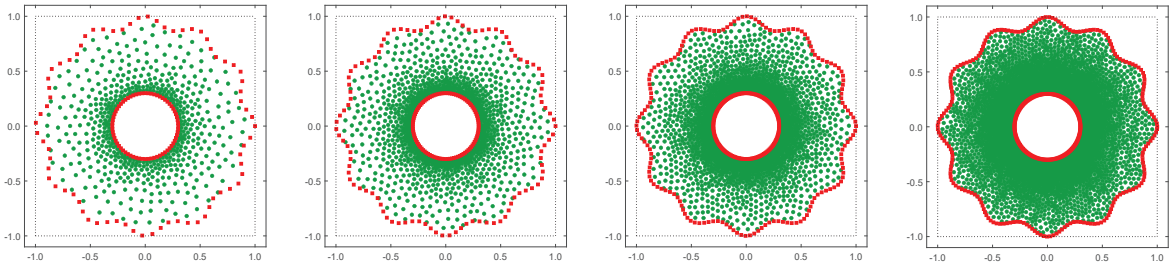

Figure 6: *Test case 3.2a. For $\overline{\Omega}$ the panels illustrate, from left to right, four grids obtained with a Delaunay triangulation procedure with* 674, 1316, 2533, *and* 5118 *nodes. The green filled circles and the red squares correspond, respectively, to interior nodes and boundary nodes where Dirichlet conditions are imposed.*

In Figure 7, we present the numerical solution error (using L^∞- and L^1-norms) and the corresponding ACO for $p = 2, \ldots, 6$.

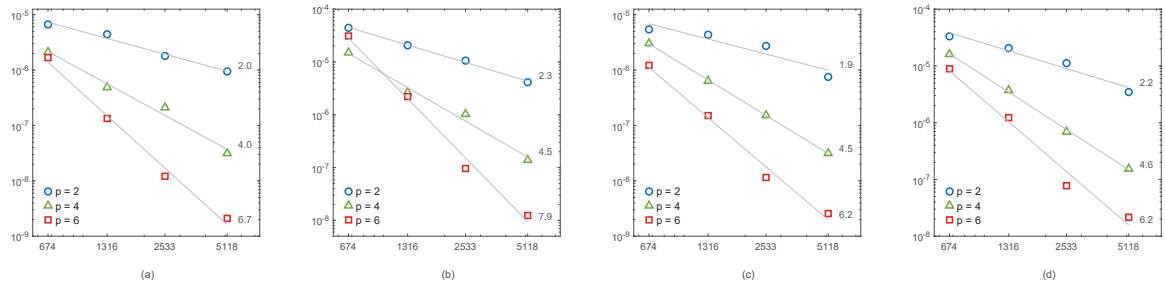

Figure 7: *Results for the test case 3.2a for $p = 2, \ldots, 6$ (the results obtained for $p = 3$ and $p = 5$ are not presented for the sake of clarity since they are very similar to those obtained for $p = 2$ and $p = 4$, respectively). The L^1- and L^∞- numerical solution errors for u_1 (panels (a) and (b), respectively) and u_2 (panels (c) and (d), respectively) are presented for a set of four grids with* 674, 1316, 2533, *and* 5118 *nodes. The corresponding average convergence order is also presented for each p.*

From the analysis of the results obtained for the ACO, we conclude that the optimal convergence order is obtained (or even exceeded) for both the L^∞- and the L^1-errors, as long as the approximation degree is even. When p is odd, the ACO obtained equals the one obtained for the $p-1$ case, a feature already reported in our previous work.

We also report that the ratio L^∞-error / L^1-error obtained for $p = 2, \ldots, 6$ (5.1 ± 0.8, 3.2 ± 0.2, 5.3 ± 0.7, 4.0 ± 0.7, and 9.9 ± 4.0, respectively) remains moderate and very similar to the values obtained in case 3.1, indicating the good quality of the numerical solution for the whole computational domain for all values of p.

Test case 3.2b. We now consider a solution that has an increased gradient near the inner boundary by setting $r_1 = 4$, $r_2 = 0.2775$, $r_3 = 50$, and $r_4 = 2.8 \times 10^{-4}$. The corresponding shape of function q and the exact solution of the BVP are illustrated in Figure 8.

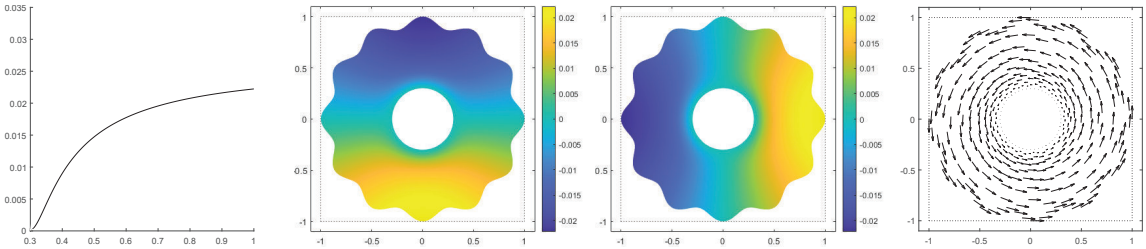

Figure 8: *Test case 3.2b. Shape of function q (left panel) and exact solution (from left to right: u_1, u_2, $\boldsymbol{u}$).*

For the numerical simulations, we use four grids with 667, 1317, 2521, and 5178 nodes, obtained with a Delaunay triangulation procedure. In order to account for the increased gradient near the inner boundary, the number of nodes in the inner boundary is now four times the number of nodes in its outer counterpart. We note that the total number of nodes in the four grids is very similar to the one in test case 3.2a. Figure 9 illustrates the different grids used in the test case 3.2b simulations.

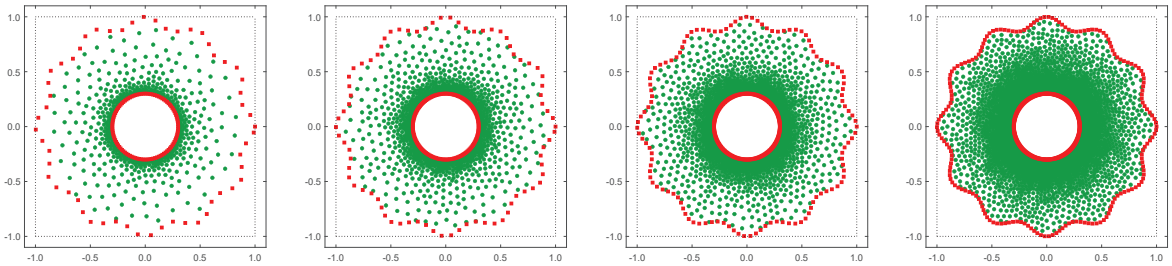

Figure 9: *Test case 3.2b. For $\overline{\Omega}$ the panels illustrate, from left to right, four grids obtained with a Delaunay triangulation procedure with* 667*,* 1317*,* 2521*, and* 5178 *nodes. The green filled circles and the red squares correspond, respectively, to interior nodes and boundary nodes where Dirichlet conditions are imposed.*

The results obtained for the numerical solution error and convergence order for $p = 2, \ldots, 6$ are presented in Figure 10.

We conclude that even though the errors are higher than those obtained in test case 3.2a, increasing the gradient of function q near the inner boundary does not impact the convergence order, since it follows the same trends already described for test case 3.2a. Likewise, the results observed for the ratio L^∞-error / L^1-error for $p = 2, \ldots, 6$ (5.1 ± 0.8, 4.3 ± 0.5, 4.7 ± 1.5, 4.9 ± 1.1, 7.0 ± 1.2, respectively) are also in line with the good values obtained for the previous test cases.

Test case 3.2c. Finally, setting $r_1 = 2$, $r_2 = 0.295$, $r_3 = 5$, and $r_4 = 3.45 \times 10^{-4}$, we consider a solution that is even steeper near the inner boundary, mimicking a boundary layer scenario. The corresponding shape for function q and the exact solution of the BVP are shown in Figure 11.

In order to account for the rather steep gradient near the inner boundary, the number of nodes in the inner boundary is set to ten times the number of nodes in its outer counterpart. The density of

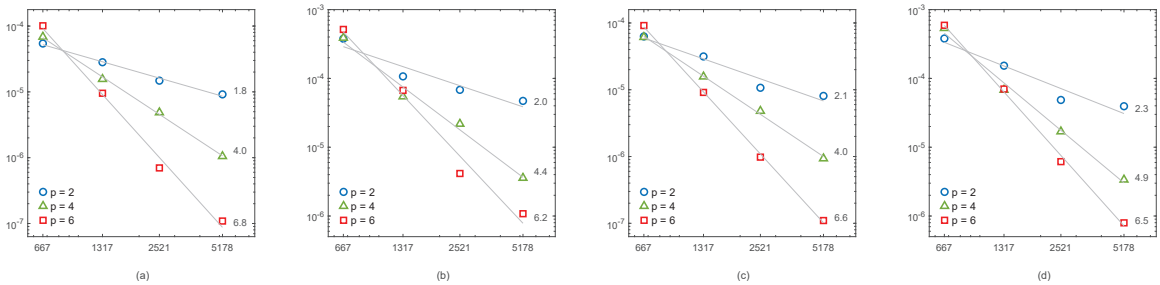

Figure 10: *Results for the test case 3.2b for* $p = 2, \ldots, 6$ *(the results obtained for* $p = 3$ *and* $p = 5$ *are not presented for the sake of clarity since they are very similar to those obtained for* $p = 2$ *and* $p = 4$*, respectively). The* L^1*- and* L^∞*- numerical solution errors for* u_1 *(panels (a) and (b), respectively) and* u_2 *(panels (c) and (d), respectively) are presented for a set of four grids with* 667*,* 1317*,* 2521*, and* 5178 *nodes. The corresponding average convergence order is also presented for each* p*.*

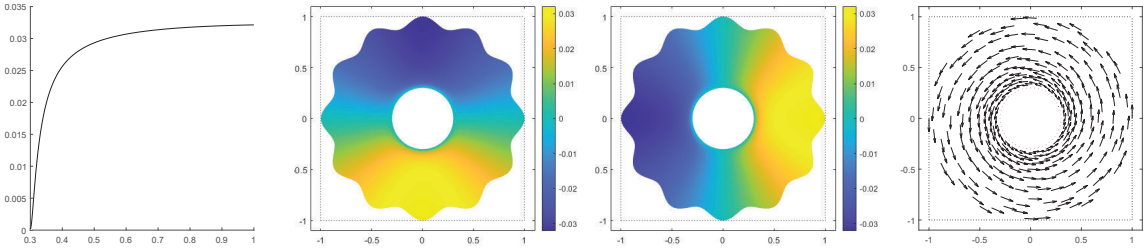

Figure 11: *Test case 3.2c. Shape of function* q *(left panel) and exact solution (from left to right:* u_1*,* u_2*,* $\boldsymbol{u}$*).*

nodes in the outer region has to be kept similar to the one used in test case 3.2b in order to ensure the convergence of the fixed point method for all values of p. Therefore, the number of nodes has to be increased with respect to the test cases 3.2a and 3.2b. Hence, for the numerical simulations, we use four grids with 2711, 3723, 5159, and 7169 nodes, obtained with a Delaunay triangulation procedure. Figure 12 illustrates the four grids used in the test case 3.2c simulations.

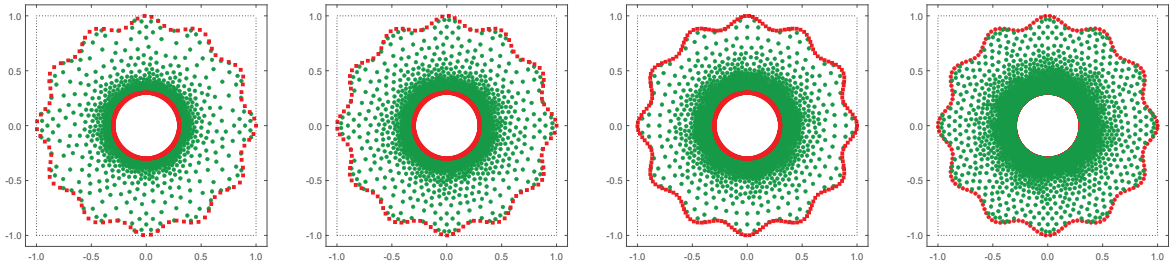

Figure 12: *Test case 3.2c. For* $\overline{\Omega}$ *the panels illustrate, from left to right, four grids obtained with a Delaunay triangulation procedure with* 2711*,* 3723*,* 5159 *and* 7169 *nodes. The green filled circles and the red squares correspond, respectively, to interior nodes and boundary nodes where Dirichlet conditions are imposed.*

The results obtained for the numerical solution error and the ACO for $p = 2, \ldots, 6$ are presented in Figure 13.

The results show that the use of a more demanding function q inevitably leads to higher errors, but the convergence order remains essentially the same as in test cases 3.2a and 3.2b, reflecting the good ability of the numerical scheme to handle a boundary layer-like scenario. On the other hand,

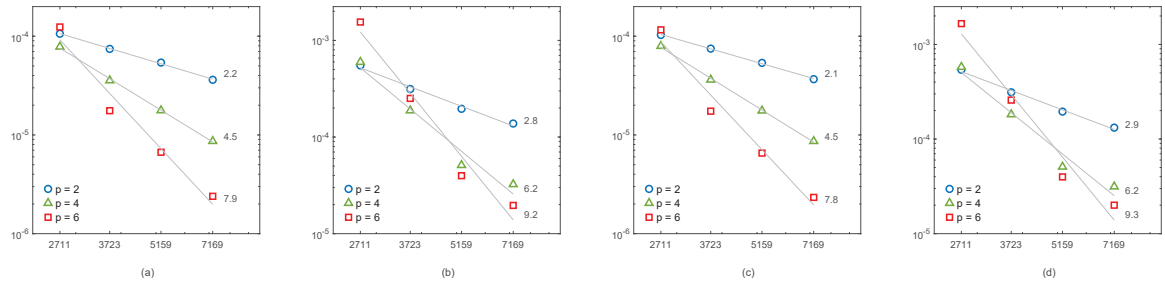

Figure 13: *Results for the test case 3.2c for* $p = 2, \ldots, 6$ *(the results obtained for* $p = 3$ *and* $p = 5$ *are not presented for the sake of clarity since they are very similar to those obtained for* $p = 2$ *and* $p = 4$*, respectively). The* L^1*- and* L^∞*- numerical solution errors for* u_1 *(panels (a) and (b), respectively) and* u_2 *(panels (c) and (d), respectively) are presented for a set of four grids with* 2711*,* 3723*,* 5159*, and* 7169 *nodes. The corresponding average convergence order is also presented for each* p*.*

the results observed for the ratio L^∞-error / L^1-error for $p = 2, \ldots, 6$ (4.2 ± 0.6, 4.1 ± 0.4, 4.8 ± 1.6, 3.2 ± 0.5, 10.6 ± 3.2, respectively) support the good quality of the numerical solution across the computational domain for all values of p.

3.5. Test case 3.3

For this last test case concerning the two-dimensional Burgers' equations, we aim to test the ability of the numerical scheme to address Neumann boundary conditions in the presence of complex geometries. As in the previous test case, we take $\varepsilon = 10^{-2}$. We now consider a 2D nozzle-like domain given by

$$\Omega = \left\{ \boldsymbol{x} \in \mathbb{R}^2 : 0.5 < x_1 < 1,\ -h(x_1) < x_2 < h(x_1) \right\},$$

where

$$\begin{aligned}
\Gamma_t &= \left\{ \boldsymbol{x} \in \mathbb{R}^2 : 0.5 < x_1 < 1,\ x_2 = h(x_1) \right\}, \\
\Gamma_b &= \left\{ \boldsymbol{x} \in \mathbb{R}^2 : 0.5 < x_1 < 1,\ x_2 = -h(x_1) \right\}, \\
\Gamma_l &= \left\{ \boldsymbol{x} \in \mathbb{R}^2 : x_1 = 0.5,\ -h(x_1) \leq x_2 \leq h(x_1) \right\}, \\
\Gamma_r &= \left\{ \boldsymbol{x} \in \mathbb{R}^2 : x_1 = 1,\ -h(x_1) \leq x_2 \leq h(x_1) \right\},
\end{aligned}$$

with $h(x_1) = 0.025 \arccos(0.1 \cos 40 x_1)$. The solution of the BVP satisfies homogeneous Neumann conditions on the upper and lower boundaries, Γ_t and Γ_b, and Dirichlet conditions on the left and right boundaries, Γ_l and Γ_r, that is,

$$\begin{aligned}
\boldsymbol{u}(\boldsymbol{x}) &= \boldsymbol{u}_D(\boldsymbol{x}), \quad \text{on } \Gamma_l \cup \Gamma_r, \\
\partial_{\mathbf{n}} \boldsymbol{u}(\boldsymbol{x}) &= \mathbf{0}, \quad \text{on } \Gamma_t \cup \Gamma_b,
\end{aligned}$$

where $\mathbf{n}$ is the outer unit normal vector. Therefore, we set $\boldsymbol{\mu}(\boldsymbol{x}_i) = \mathbf{1}$ and $\boldsymbol{\nu}(\boldsymbol{x}_i) = \mathbf{0}$, for all $\boldsymbol{x}_i \in \Gamma_l \cup \Gamma_r$, and $\boldsymbol{\mu}(\boldsymbol{x}_i) = \mathbf{0}$ and $\boldsymbol{\nu}(\boldsymbol{x}_i) = \mathbf{1}$, for all $\boldsymbol{x}_i \in \Gamma_t \cup \Gamma_b$. In order to perform a convergence

analysis we set $\boldsymbol{f}$ and $\boldsymbol{u}_D$ such that

$$u_1(\boldsymbol{x}) = \frac{1}{5}(1 + \cos 40x_1 \cos 40x_2), \quad u_2(\boldsymbol{x}) = \frac{1}{5}(1 - \cos 40x_1 \cos 40x_2),$$

is the unique solution of the BVP. The choice of function h ensures homogeneous Neumann conditions hold on Γ_t and Γ_b for this particular choice of the solution of the BVP (*cf.* [14]).

The geometrical setting and the exact solution of the BVP are illustrated in Figure 14.

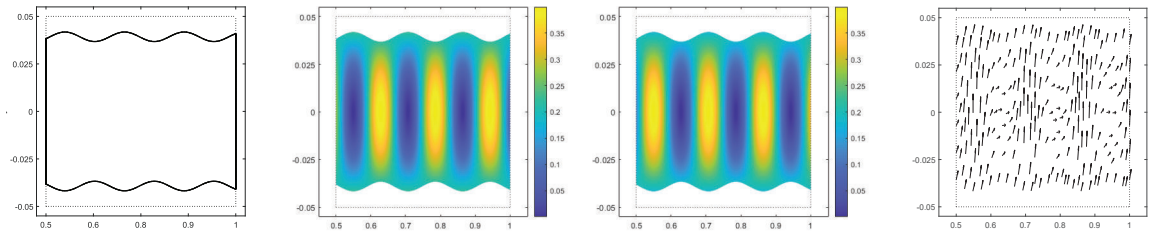

Figure 14: *Test case 3.3. Geometrical setting (left panel) and exact solution (from left to right: u_1, u_2, $\boldsymbol{u}$).*

For the numerical simulations, we use four grids with 625, 1264, 2505, and 5008 nodes, obtained using a Delaunay triangulation procedure. Figure 15 illustrates the different grids used in test case 3.3 numerical simulations.

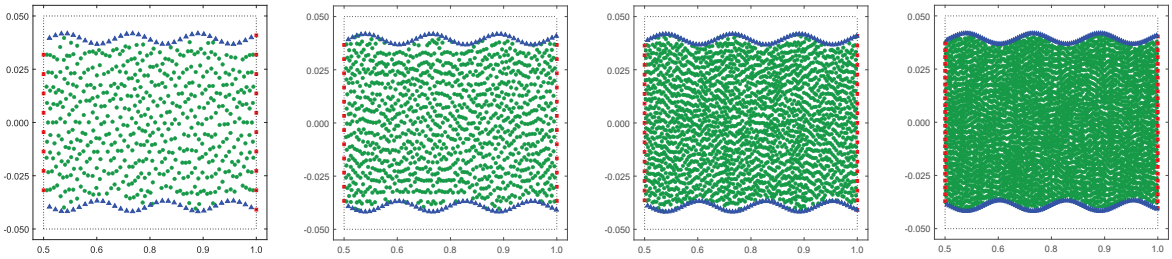

Figure 15: *Test case 3.3. For $\overline{\Omega}$ the panels illustrate, from left to right, four grids obtained with a Delaunay triangulation procedure with* 625*,* 1264*,* 2505*, and* 5008 *nodes. The green filled circles, the red squares and the blue triangles correspond, respectively, to interior nodes, boundary nodes where Dirichlet conditions are imposed and boundary nodes where Neumann conditions hold.*

The results obtained for the numerical solution error and convergence order for $p = 2, \ldots, 6$ are presented in Figure 16.

From the analysis of the results, we conclude that optimal convergence orders are obtained for the two norms used to evaluate the numerical solution error when the polynomial approximation degree is even. As in test case 3.2, the results obtained for $p = 3$ and $p = 5$, in particular the ACO, are rather similar to those found for $p = 2$ and $p = 4$, respectively. The good quality of the numerical solution across the computational domain is also confirmed by the low values obtained for the ratio L^∞-error / L^1-error (3.9 ± 0.8, 2.6 ± 0.3, 3.4 ± 0.9, 3.1 ± 1.1, 3.7 ± 0.5, for $p = 2, \ldots, 6$, respectively), which remains below 4.

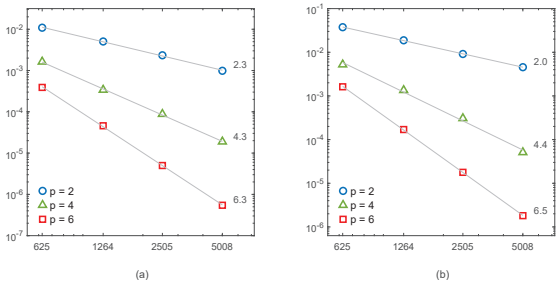

Figure 16: *Results for the test case 3.3 for* $p = 2, \ldots, 6$ *(the results obtained for* $p = 3$ *and* $p = 5$ *are not presented for the sake of clarity since they are very similar to those obtained for* $p = 2$ *and* $p = 4$*, respectively). The* L^1*- and* L^∞*- numerical solution errors for* u_1 *(panels (a) and (b), respectively) are presented for a set of four grids with* 625*,* 1264*,* 2505*, and* 5008 *nodes. The corresponding average convergence order is also presented for each* p*. We note that the results obtained for* u_2 *are omitted since they are identical to those presented for* u_1*.*

4. The weakly compressible Navier-Stokes system

4.1. Modeling and linearization

We recall the two-dimensional stationary forced Euler isentropic equations with viscosity

$$\begin{aligned} \partial_{x_1}(\rho u_1) + \partial_{x_2}(\rho u_2) &= 0, \quad \text{in } \Omega, \\ -\mu\left(\partial^2_{x_1} u_1 + \partial^2_{x_2} u_1\right) + \partial_{x_1}(\rho u_1^2 + P) + \partial_{x_2}(\rho u_1 u_2) &= f_{u_1}, \quad \text{in } \Omega, \\ -\mu\left(\partial^2_{x_1} u_2 + \partial^2_{x_2} u_2\right) + \partial_{x_1}(\rho u_1 u_2) + \partial_{x_2}(\rho u_2^2 + P) &= f_{u_2}, \quad \text{in } \Omega, \end{aligned}$$

where ρ is the density, $\boldsymbol{u} = (u_1, u_2)$ the fluid velocity vector, and $\mu > 0$ the dynamic viscosity coefficient, while the pressure P is related to the density by the Tait equation of state, that is, $P(\rho) = C[(\rho/\rho_0)^\gamma - 1]/\gamma + P_0$, with $P_0 \geq 0$, $\gamma > 1$ and $C, \rho_0 > 0$. In the following, we shall take $P_0 = 0$, $\gamma = 1.4$, $C = 1$, $\rho_0 = 1$, and $\mu \in [0.001, 0.01]$. This combination of values of γ and μ holds for some light oils, hydrocarbons, alcohols, liquid carbon dioxide, and low-viscosity silicone oils.

The steady-state situation is considered as the limit case of a non-stationary problem in which we prescribe constant velocity in time on the boundary, with the additional condition $\boldsymbol{u} \cdot \boldsymbol{n} = 0$ on $\partial\Omega$. In that regard, mass conservation yields that the total mass $M_\rho = \displaystyle\int_\Omega \rho(x)dx$ is defined by the initial condition, and such a constraint has to be added to close the steady-state problem. An alternative way to fix the degree of freedom consists of explicitly setting the density value at some point of the domain. We addopt such an approach by using a velocity $\boldsymbol{u} = \boldsymbol{u}_D$ such that $\boldsymbol{u}_D \cdot \boldsymbol{n} = 0$ and we prescribe the density at the origin with the condition $\rho(0, 0) = \rho_0$.

Similarly to the Burgers' equations case, the Picard fixed-point iteration is carried out by building

a sequence $(\rho^{[k]}, u_1^{[k]}, u_2^{[k]})_k$ of the linearized problem

$$u_1^{[k]}\partial_{x_1}\rho^{[k+1]} + \rho^{[k]}\partial_{x_1}u_1^{[k+1]} + u_2^{[k]}\partial_{x_2}\rho^{[k+1]} + \rho^{[k]}\partial_{x_2}u_2^{[k+1]} = 0,$$

$$-\mu\left(\partial_{x_1}^2 u_1^{[k+1]} + \partial_{x_2}^2 u_1^{[k+1]}\right) + \rho^{[k]}u_1^{[k]}\partial_{x_1}u_1^{[k+1]} + \rho^{[k]}u_2^{[k]}\partial_{x_2}u_1^{[k+1]} + C\left(\rho^{[k]}\right)^{\gamma-1}\partial_{x_1}\rho^{[k+1]} = f_{u_1},$$

$$-\mu\left(\partial_{x_1}^2 u_2^{[k+1]} + \partial_{x_2}^2 u_2^{[k+1]}\right) + \rho^{[k]}u_1^{[k]}\partial_{x_1}u_2^{[k+1]} + \rho^{[k]}u_2^{[k]}\partial_{x_2}u_2^{[k+1]} + C\left(\rho^{[k]}\right)^{\gamma-1}\partial_{x_2}\rho^{[k+1]} = f_{u_2}.$$

We stop the procedure when the norm of the difference between two successive iterations is below a given threshold.

4.2. Discretization and assembling

Denoting by D, U_1, and U_2 the vectors of $\mathbb{R}^I$ containing the numerical solution of the non-linear system, that is,

$$D = (\rho_i)_{i\in\mathcal{G}}, \quad U_1 = (u_{1,i})_{i\in\mathcal{G}}, \quad U_2 = (u_{2,i})_{i\in\mathcal{G}},$$

the system of equations reads

$$u_1^{[k],(0,0)}\rho^{[k+1],(1,0)} + \rho^{[k],(0,0)}u_1^{[k+1],(1,0)} + u_2^{[k],(0,0)}\rho^{[k+1],(0,1)} + \rho^{[k],(0,0)}u_2^{[k+1],(0,1)} = 0,$$

$$-\mu\left(u_1^{[k+1],(2,0)} + u_1^{[k+1],(0,2)}\right) + \rho^{[k],(0,0)}u_1^{[k],(0,0)}u_1^{[k+1],(1,0)} + \rho^{[k],(0,0)}u_2^{[k],(0,0)}u_1^{[k+1],(0,1)} + C\left(\rho^{[k],(0,0)}\right)^{\gamma-1}\rho^{[k+1],(1,0)} = f_{u_1}(\boldsymbol{x}_i),$$

$$-\mu\left(u_2^{[k+1],(2,0)} + u_2^{[k+1],(0,2)}\right) + \rho^{[k],(0,0)}u_1^{[k],(0,0)}u_2^{[k+1],(1,0)} + \rho^{[k],(0,0)}u_2^{[k],(0,0)}u_2^{[k+1],(0,1)} + C\left(\rho^{[k],(0,0)}\right)^{\gamma-1}\rho^{[k+1],(0,1)} = f_{u_2}(\boldsymbol{x}_i).$$

Substituting the derivatives with the local representation given by (3) and (4), we obtain the system

$$\begin{pmatrix} A_{\rho\rho}^{[k]} & A_{\rho u_1}^{[k]} & A_{\rho u_2}^{[k]} \\ A_{u_1\rho}^{[k]} & A_{u_1u_1}^{[k]} & A_{u_1u_2}^{[k]} \\ A_{u_2\rho}^{[k]} & A_{u_2u_1}^{[k]} & A_{u_2u_2}^{[k]} \end{pmatrix} \begin{pmatrix} D^{[k+1]} \\ U_1^{[k+1]} \\ U_2^{[k+1]} \end{pmatrix} = \begin{pmatrix} b_\rho \\ b_{u_1} \\ b_{u_2} \end{pmatrix}$$

with, for $i \in \mathcal{G}$ and $j \in \mathcal{S}_i$,

$$
\begin{aligned}
A^{[k]}_{\rho\rho}[i,j] &= u_1^{[k],(0,0)} a_{i;(1,0),j} + u_2^{[k],(0,0)} a_{i;(0,1),j},\\
A^{[k]}_{u_1u_1}[i,j] &= -\mu\left(a_{i;(2,0),j} + a_{i;(0,2),j}\right) + \rho^{[k],(0,0)} u_1^{[k],(0,0)} a_{i;(1,0),j},\\
A^{[k]}_{u_2u_2}[i,j] &= -\mu\left(a_{i;(2,0),j} + a_{i;(0,2),j}\right) + \rho^{[k],(0,0)} u_2^{[k],(0,0)} a_{i;(0,1),j},\\
A^{[k]}_{\rho u_1}[i,j] &= \rho^{[k],(0,0)} a_{i;(1,0),j}, \qquad & A^{[k]}_{u_1\rho}[i,j] &= C\left(\rho^{[k],(0,0)}\right)^{\gamma-1} a_{i;(1,0),j},\\
A^{[k]}_{\rho u_2}[i,j] &= \rho^{[k],(0,0)} a_{i;(0,1),j}, \qquad & A^{[k]}_{u_2\rho}[i,j] &= C\left(\rho^{[k],(0,0)}\right)^{\gamma-1} a_{i;(0,1),j},\\
A^{[k]}_{u_1u_2}[i,j] &= \rho^{[k],(0,0)} u_2^{[k],(0,0)} a_{i;(0,1),j}, \qquad & A^{[k]}_{u_2u_1}[i,j] &= \rho^{[k],(0,0)} u_1^{[k],(0,0)} a_{i;(1,0),j},
\end{aligned}
$$

and zero otherwise. The right-hand side is given by $b_\rho[i] = 0$, $b_{u_1}[i] = f_{u_1}(\boldsymbol{x}_i)$, and $b_{u_2}[i] = f_{u_2}(\boldsymbol{x}_i)$.

4.3. Benchmarking

4.3.1. Test case 4.1

The geometrical setting for the BVP is given by the square domain $\Omega =]-1,1[\times]-1,1[$, and we consider Dirichlet boundary conditions of the form

$$\rho(\boldsymbol{x}) = \frac{4}{5}, \quad \boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{u}_D(\boldsymbol{x}), \quad \boldsymbol{x} \in \partial\Omega.$$

In order to perform a convergence analysis we take $\mu = 0.01$ and set $\boldsymbol{f}$ and $\boldsymbol{u}_D$ such that

$$\rho(\boldsymbol{x}) = \frac{4}{5} + \frac{2}{5}\cos\frac{\pi x_1}{2}\cos\frac{\pi x_2}{2}, \quad u_1(\boldsymbol{x}) = \frac{1}{2}\cos\frac{\pi x_1}{2}\sin\frac{\pi x_2}{2}, \quad u_2(\boldsymbol{x}) = -\frac{1}{2}\sin\frac{\pi x_1}{2}\cos\frac{\pi x_2}{2},$$

is the unique solution of the BVP. We note that the velocity field $\boldsymbol{u}$ satisfies no-penetration boundary conditions since

$$\boldsymbol{u}(\boldsymbol{x}) \cdot \mathbf{n}(\boldsymbol{x}) = 0, \quad \boldsymbol{x} \in \partial\Omega.$$

The geometrical setting and the exact solution of the BVP are illustrated in Figure 17.

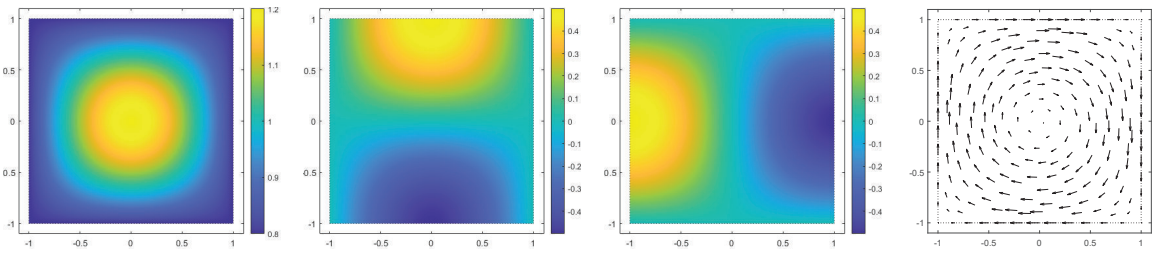

Figure 17: *Test case 4.1. Exact solution (from left to right: ρ, u_1, u_2, $\boldsymbol{u}$).*

For the numerical simulations, we use four grids with 833, 1222, 1792, and 2523 nodes, obtained with a Delaunay triangulation procedure. These grids are depicted in Figure 18.

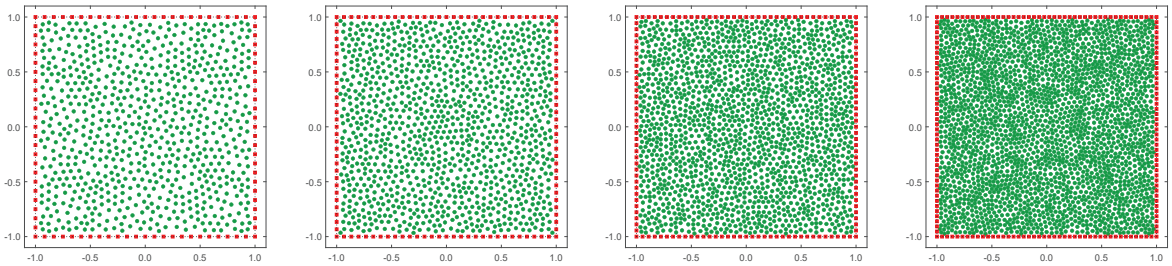

Figure 18: *Test case 4.1. For $\overline{\Omega}$ the panels illustrate, from left to right, four grids obtained with a Delaunay triangulation procedure with* 833, 1222, 1792, *and* 2523 *nodes. The green filled circles correspond to interior nodes and the red squares to boundary nodes where Dirichlet conditions are imposed.*

The results obtained for the numerical solution error and convergence order for $p = 2, \ldots, 6$ are presented in Figure 19.

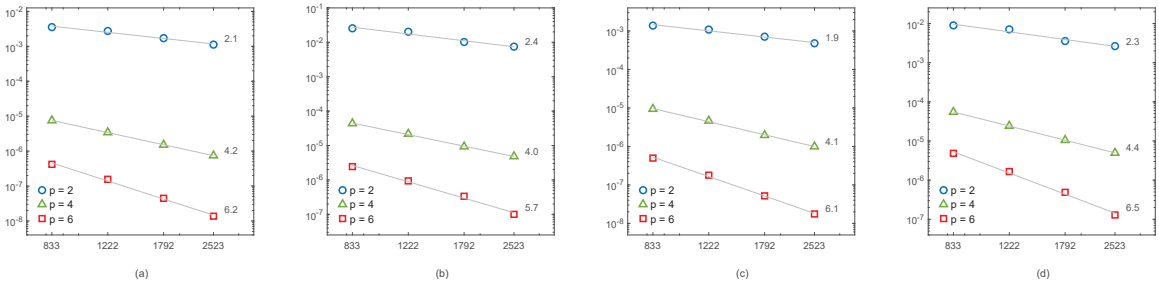

Figure 19: *Results for the test case 4.1 for $p = 2, \ldots, 6$ (the results obtained for $p = 3$ and $p = 5$ are not presented for the sake of clarity since they are very similar to those obtained for $p = 2$ and $p = 4$, respectively). The L^1- and L^∞- numerical solution errors for ρ (panels (a) and (b), respectively) and u_1 (panels (c) and (d), respectively) are presented for a set of four grids with* 833, 1222, 1792, *and* 2523 *nodes. The corresponding average convergence order is also presented for each p. We note that the results obtained for u_2 are omitted since they are very similar to those obtained for u_1.*

The results obtained show that optimal convergence rates prevail for all values of p, independently of the unknown considered. On the other hand, the values obtained for the ratio L^∞-error / L^1-error (6.4 ± 0.7, 4.3 ± 0.3, 5.6 ± 0.3, 4.6 ± 1.2, 8.1 ± 1.0, for $p = 2, \ldots, 6$, respectively) allow us to conclude that the numerical solution is of good quality for the entire computational domain for all values of p.

It is worth noting that the value adopted in this test case for the viscosity coefficient, $\mu = 0.01$, is the minimum value for which we obtained convergence of the fixed-point procedure for all the grids and values of p considered in the numerical simulations. This corresponds to a maximum value for the Reynolds number, $Re = \rho\,||\boldsymbol{u}||L/\mu$ (taking $L = 2$), of 220 obtained for the coarsest grid when $p = 6$. One expects that different grids and values of p will lead to different minimum values of μ for which the fixed-point method converges and, therefore, maximum values of Re. Table 1 depicts the results obtained for these two parameters when different Delaunay-type grids and values of p are used. From the analysis of the results presented in Table 1 we report that finer grids lead to smaller values of μ_{min} and, consequently, to higher values of Re_{max}, for all the values of p. In particular, for the 7179 points grid, when $p = 4$ and $p = 5$, we obtain the smallest value of the viscosity coefficient, 5×10^{-4}, and therefore the highest value of the Reynolds number ($\simeq 1760$).

We also observe that for a given grid the two coefficients do not correlate with the polynomial degree used for the approximation, with the exception of the 833 points grid for which an increase of the approximation degree leads to a systematic decrease of the (maximum) Reynolds number.

Table 1: *Results for the test case 4.1. For different Delaunay type grids and degrees of the polynomial approximation, p, we present the corresponding minimum value of the viscosity coefficient, μ_{min}, for which the fixed-point method converges, as well as the associated maximum Reynolds number, Re_{max}.*

DOF		833	1222	1792	2523	3627	5190	7179
$p=2$	μ_{min}	4e-03	7e-03	6e-03	4e-03	2e-03	2e-03	1e-03
	Re_{max}	220	126	148	221	440	440	884
$p=3$	μ_{min}	5e-03	5e-03	5e-03	3e-03	2e-03	2e-03	1e-03
	Re_{max}	176	176	176	294	440	440	884
$p=4$	μ_{min}	5e-03	2e-03	2e-03	4e-03	2e-03	1e-03	5e-04
	Re_{max}	176	220	440	220	440	884	1760
$p=5$	μ_{min}	8e-03	8e-03	8e-03	6e-03	2e-03	1e-03	5e-04
	Re_{max}	110	110	110	147	440	884	1761
$p=6$	μ_{min}	1e-02	5e-03	5e-03	3e-03	2e-03	1e-03	8e-04
	Re_{max}	88	110	176	294	445	884	1100

4.3.2. Test case 4.2

In this test case, we handle the stationary 2D lid-driven cavity problem, which has the particularity of involving a curved lid instead of the straight-lid case addressed above (see further below). The set of partial differential equations governing the flow are the homogeneous Euler isentropic equations (*i.e.*, $\boldsymbol{f} = \mathbf{0}$) with viscosity (we take $\mu = 0.001$). The equation of state remains unchanged with respect to the previous test case.

The velocity boundary conditions for the left (Γ_l), bottom (Γ_b), and right (Γ_r) walls correspond to no-slip conditions, that is, $\boldsymbol{u} = \mathbf{0}$. For the lid ($\Gamma_t$), which is the arc of a circle with center at point $(0.5, 0)$ and passing through points $(0, 1)$ and $(1, 1)$ (see Figure 20), we impose no-penetration conditions with velocity $\boldsymbol{u}_D$ such that $\boldsymbol{u}_D \cdot \mathbf{n} = 0$ and $||\boldsymbol{u}_D|| = U_D$, where U_D is a constant that will be set later. Since the lid moves toward the right, $u_{D_1} > 0$, while $u_{D_2} > 0$ for $0 < x_1 < 0.5$, $u_{D_2} = 0$ for $x_1 = 0.5$, and $u_{D_2} < 0$ for $0.5 < x_1 < 1$.

As for the boundary conditions involving the density, we set $\rho = \rho_0 = 1$ at $(0, 0)$ to ensure that the BVP has a unique solution and impose that the mass conservation equation holds at the other boundary points.

In summary, the boundary conditions for the BVP can be written as follows:

$$\begin{cases} \partial_{x_1}(\rho u_1) + \partial_{x_2}(\rho u_2) = 0, \quad \boldsymbol{u} = \mathbf{0}, & \text{on } \Gamma_l \cup \Gamma_b \cup \Gamma_r \setminus \{(0,0)\}, \\ \rho = 1, \quad \boldsymbol{u} = \mathbf{0}, & \text{at } (0,0), \\ \partial_{x_1}(\rho u_1) + \partial_{x_2}(\rho u_2) = 0, \quad \boldsymbol{u} = \boldsymbol{u}_D, & \text{on } \Gamma_t. \end{cases}$$

For the numerical simulations illustrating this test case, we consider a grid with 10173 nodes obtained with a Delaunay triangulation procedure. We take $p = 6$ for the numerical approximation

of the solution and adopt $U_D = 0.4$, leading to a Reynolds number of 400. In Figure 21, we depict the velocity profile along the centerlines: for u_1 at $x_2 = x_{2max}/2$ and for u_2 at $x_1 = 0.5$. For the sake of qualitative comparison, Figure 21 also presents the results obtained by [15] for an incompressible flow for a straight lid using the Navier-Stokes equation and a multigrid method on rectangular grids. We report that although the underlying physics and geometries are different, our results follow the same trend as those presented in [15].

Finally, we address the streamlines of the compressible flow. For that purpose, we consider the density-weighted stream function, ψ, satisfying

$$\partial_{x_1}\psi = -\rho u_2, \quad \partial_{x_2}\psi = \rho u_1.$$

We compute an approximation of ψ at each node of the grid using $p = 6$ and solving the following BVP numerically:

$$\begin{cases} \partial^2_{x_1}\psi + \partial^2_{x_2}\psi = -\partial_{x_1}(\rho u_2) + \partial_{x_2}(\rho u_1), & \text{in } \Omega, \\ \psi = 0, & \text{on } \partial\Omega. \end{cases}$$

where ρ, u_1, and u_2 are known.

In Figure 22, we present the streamlines for the compressible flow together with the velocity amplitude. We see that, as in the incompressible flow straight-lid case, the primary vortex is accompanied by several smaller secondary vortices. Unlike in [15], only one additional secondary vortex appears in the lower left corner of the domain. In Table 2 we present the position of the center of each vortex, as well as the corresponding value of the stream function.

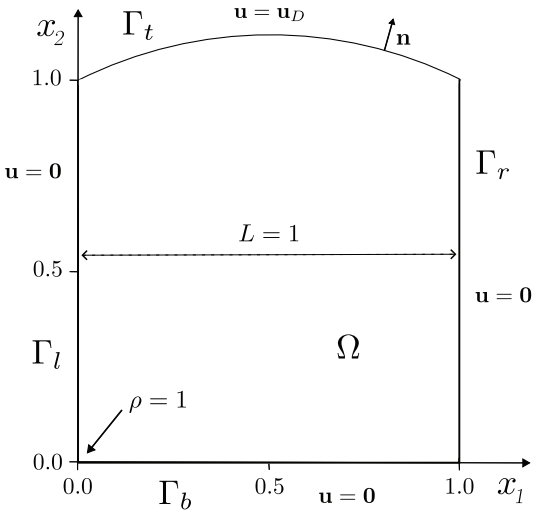

Figure 20: *Test case 4.2. Illustration of the lid-cavity geometry and boundary conditions for the velocity and density. The moving lid, denoted* Γ_t*, corresponds to the arc of circumference having center at point* $(0.5, 0)$ *and passing through points* $(0, 1)$ *and* $(1, 1)$*. The lid displacement is from the left to the right, the norm of the lid velocity* $\mathbf{u}_D$ *being constant and equal to* U_D*. The value of the density is set to 1 at point* $(0, 0)$ *to provide uniqueness of solution. As usual,* $\mathbf{n}$ *stands for the outer unit normal (not to scale).*

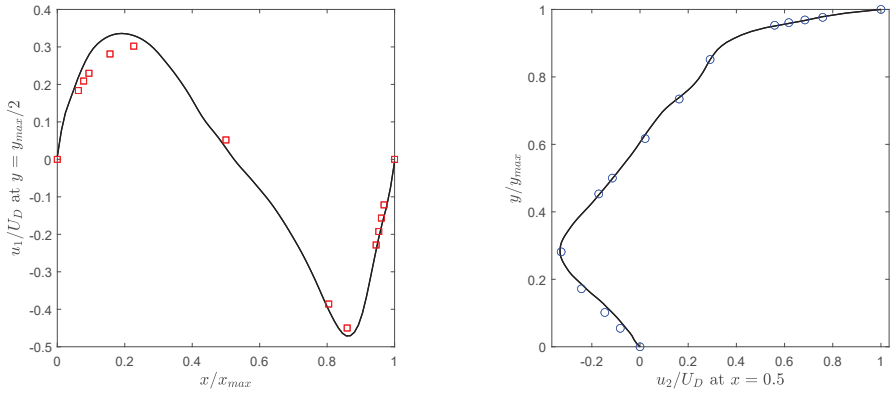

Figure 21: *Test case 4.2. Velocity profiles at the centerlines:* u_1 *at* $x_2 = x_{2max}/2$ *(left panel) and for* u_2 *at* $x_1 = 0.5$ *(right panel). The squares and circles correspond to the results obtained by [15] for an incompressible flow for a straight lid.*

Table 2: *Results for the test case 4.2. Position of the center of primary and secondary vortices and the corresponding local extreme values of the stream function* ψ*.*

		this work	Ghia et al. (1982)
Primary vortex	Location (x_1, x_2)	(0.5304,0.6755)	(0.5547,0.6065)
	ψ_{min}	-0.047699	-0.113909
Secondary vortex #1	Location (x_1, x_2)	(0.0745,0.0382)	(0.0508,0.0469)
(lower left corner)	ψ_{max}	0.99782e-05	1.41951e-05
Secondary vortex #2	Location (x_1, x_2)	(0.8548,0.1172)	(0.8906,0.1250)
(lower right corner)	ψ_{max}	4.7983e-04	6.4235e-04
Secondary vortex #3	Location (x_1, x_2)	(0.9761,0.0097)	(0.9922,0.0078)
(lower right corner)	ψ_{min}	-1.8496e-06	-1.8660e-08

5. Conclusion

The general finite difference method, by its nature, enables one to wisely choose the nodes of the stencil attached to a generic node i while other popular meshless methods, such as FPsM or FPtM, always consider the nodes belonging to the compact ball given by a user-dependent radius. Such a degree of freedom provides both accuracy and stability while maintaining a small stencil size. As a continuation of our work on the scalar linear PDE, we prove that the strategy works very well for the non-linear steady-state problem using a fixed-point technique based on a local linearization. Only low local Péclet/Reynolds-number discrete problems were considered to assess the optimization algorithm initially based on an isotropic stencil. The next stage will be to introduce some upwind bias, both for the stencil and the weights, to tackle convection-dominated problems and even hyperbolic systems with smooth solutions.

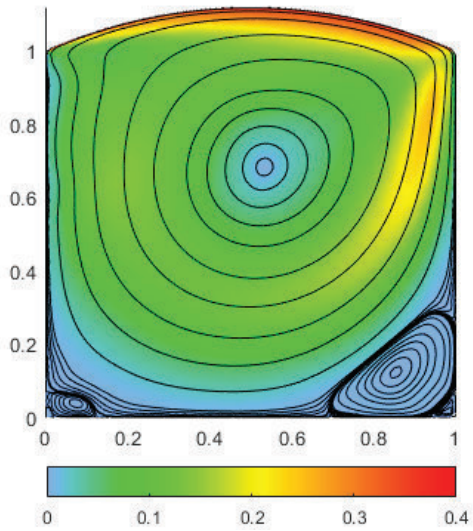

Figure 22: *Test case 5.2. Streamlines and velocity amplitude for the (curved) lid-driven cavity flow.*

Acknowledgments

The authors Stéphane Clain and Jorge Figueiredo were funded by national funds from FCT - Fundação para a Ciência e a Tecnologia, I.P., within the context of the project HiPMos4C setting with reference 2023.16854.ICDT2023.16854.ICDT, https://doi.org/10.54499/2023.16854.ICDT.
Stéphane Clain would like to acknowledge the support of Centro de Matemática of the Coimbra University (CMUC), funded by the Portuguese Foundation for Science and Technology, under the references UIDB/00324/2020 (https://doi.org/10.54499/UIDB/00324/2020) and UIDP/00324/2020 (https://doi.org/10.54499/UIDP/00324/2020).
Jorge Figueiredo was supported by national funds through FCT - Fundação para a Ciência e a Tecnologia, under the Program Contract of the Centre of Mathematics of the University of Minho (CMAT/UM), UID/00013/2025 (https://doi.org/10.54499/UID/00013/2025).

Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

[1] R. A. Gingold, J. J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Monthly Notices of the Royal Astronomical Society 181 (1977) 375–389.

[2] L. B. Lucy, A numerical approach to the testing of the fission hypothesis, The Astronomical Journal 82 (1977) 1013–1024.

[3] S. Tiwari, J. Kuhnert, A meshfree method for compressible navier–stokes equations with moving boundaries, Nonlinear Theory and Its Applications 3 (2002) 221–230.

[4] E. Oñate, S. Idelsohn, O. C. Zienkiewicz, R. L. Taylor, A finite point method in computational mechanics: Applications to convective transport and fluid flow., International Journal for Numerical Methods in Engineering 39 (1996) 3839–3866.

[5] E. Oñate, S. Idelsohn, A mesh-free finite point method for compressible and incompressible flow problems, International Journal for Numerical Methods in Engineering 43 (1998) 743–767.

[6] E. Ortega, E. Oñate, S. R. Idelsohn, R. Flores, A meshless finite point method for three-dimensional analysis of compressible flow problems involving moving boundaries and adaptivity, International Journal for Numerical Methods in Fluids 73 (2013) 323–343.

[7] J. J. Benito, F. Ureña, L. Gavete, Influence of several factors in the generalized finite difference method, Applied Mathematical Modelling 27 (2001) 1039–1053.

[8] Z. Zheng, X. Li, Theoretical analysis of the generalized finite difference method, Computers and Mathematics with Applications 120 (2022) 1–14. doi:10.1016/j.camwa.2022.06.017.

[9] K. Wang, Y. Li, A. Wu, X. Yin, F. Liu, X. Feng, Numerical analysis of a linear generalized finite difference scheme for burgers' equation, Computers and Mathematics with Applications 193 (2025) 117–131. doi:10.1016/j.camwa.2025.05.027.

[10] S. Clain, J. Figueiredo, Stencil and kernel optimisation for mesh-free very high-order generalised finite difference method, Mathematics and Computers in Simulation 218 (2024) 49–78.

[11] C.-M. Fan, P.-W. Li, Generalized finite difference method for solving two-dimensional burgers' equations, Procedia Engineering 79 (2014) 55–60. doi:10.1016/j.proeng.2014.06.310.

[12] P.-W. Li, C.-M. Fan, Generalized finite difference method for two-dimensional shallow water equations, Engineering Analysis with Boundary Elements 80 (2017) 58–71. doi:10.1016/j.enganabound.2017.03.012.

[13] X. Rao, An upwind generalized finite difference method (gfdm) for meshless analysis of heat and mass transfer in porous media, Computational Particle Mechanics 10 (2023) 533–554. doi:10.1007/s40571-022-00501-w.

[14] S. Clain, D. Lopes, R. M. S. Pereira, Very high-order cartesian-grid finite difference method on arbitrary geometries, J. Comput. Phys. 434 (2021) 110217.

[15] U. Ghia, K. N. Ghia, C. T. Shin, High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, J. Comput. Phys. 48 (1982) 387–411.