A sixth-order finite volume method for diffusion problem with curved boundaries

Abderrhramane Boularas, Stéphane Clain, Fulbert Baudoin

A sixth-order finite volume method for diffusion problem with curved boundaries

A. Boularas, [a] S. Clain, [b,c] F. Baudoin, [a,d]

[a] *University of Toulouse, UPS, INPT, LAPLACE, 118 route de Narbonne, F-31062 Toulouse cedex 9, France.*

[b] *Centre of Mathematics, University of Minho Campus de Azurém, 4080-058 Guimarães, Portugal*

[c] *Institut de Mathématiques de Toulouse,Université Paul Sabatier, 31062 Toulouse, France*

[d] *CNRS ; LAPLACE ; F-31062 Toulouse, France.*

Abstract

A sixth-order finite volume method is proposed to solve the Poisson equation for two- and three-dimensional geometries involving curved boundaries. A specific polynomial reconstruction is used to provide accurate fluxes for diffusive operators even with discontinuous coefficients while we introduce a new technique to preserve the sixth-order approximation for non-polygonal domains. Numerical tests covering a large panel of situations are addressed to assess the performances of the method.

Key words: Finite Volume, high-order, convection-diffusion, polynomial reconstruction, Poisson equation, curved boundary.

Email addresses: boularas@laplace.univ-tlse.fr (A. Boularas,), clain@math.uminho.pt (S. Clain,), fulbert.baudoin@laplace.univ-tlse.fr (F. Baudoin,).

1 Introduction

Very high-order finite volume method (higher than the second-order) for elliptic and parabolic operators is a recent trend and has received considerable attention during the last decade. The coupling of the Euler system with a viscous term [1–6] or the shallow-water system with turbulence [7] are, among others, strong motivations to design efficient and accurate schemes for elliptic operators in the finite volume context. There exists a large literature on second-order approximations and convergence analysis [8–15] but few studies have been done for very high-order [16,17]. In a recent paper [18], a new high-order finite volume method for convection-diffusion problem has been developed for two-dimensional geometries providing up to sixth-order approximations. The technique is based on specific polynomial reconstructions to evaluate the fluxes across the cell interfaces. Nevertheless, the original technique does no take into account curved boundaries (non-polygonal domains) and the method accuracy is lowered to a second-order one due to the approximation of arcs by segments. The goal of the present study is, on the one hand, to achieve an extension for the three-dimensional case and, on the other hand, to develop a new class of polynomial representation for the boundary to preserve an effective sixth-order approximation even with curved boundaries.

When dealing with second-order finite volume schemes for elliptic operator, the curved boundary is substituted by a polygon or a polyhedra and the Dirichlet condition is evaluated at the vertices or the midpoint of the boundary edges. Very few papers in the finite volume community tackles the question of curved boundary since the problem is relevant for at least third-order schemes. In [3], the authors propose a polynomial reconstruction which exactly matches the Dirichlet condition at the Gauss points on the curved boundary. The procedure then involves two new linear constraints added to the matrix system we use to compute the polynomial coefficients. Such an algorithm is extended to the Neumann condition on curved boundary. Another approach consists to apply a local mapping $(x, y) \rightarrow (\xi, \eta)$ to transform the cell into a new computational domain which matches the boundary [19]. Integrals therefore involves the Jacobian matrix of the transformation and polynomial reconstruction procedure is performed in the local basis leading to a rather complex algorithm. The method we propose in the present study is related to the works of [3] but the Dirichlet condition is enforced in a different way by using the mean value on the edge as a free parameter to fit with the boundary condition at the Gauss points. On the other hand, very high-order method leads to an important computational effort due to the reconstruction process and a large part of the time consuming derives from the polynomial coefficients evaluation. We proposed a new procedure to dramatically reduce the computational cost by a pre-evaluation of local vector for each Gauss point one can identify as a partial assembly procedure. It results that the polynomial evaluation at the Gauss points is reduced to a simple inner product between the local vector and the data saving a lot of memory and time.

In the present document, we do not deal with the convective part on purpose since the main difficulty concerns the diffusive contribution so we only focus on the Poisson problem. The second section is devoted to the generic very high-order finite volume scheme while the third section focus on the polynomial reconstruction and detail all the improvements we propose.

Numerical experiments are presented in the four section to assess the scheme accuracy and robustness as well as the efficiency of the iterative solver coupling with a new preconditioning strategy.

2 Generic high-order Finite Volume Scheme

We consider an open bounded domain Ω of $\mathbb{R}^2$ or $\mathbb{R}^3$ with a piecewise regular curved boundary. Γ_D and Γ_N define a partition of the boundary $\partial\Omega$ where we shall prescribe the Dirichlet and the Neumann conditions respectively. We intend to compute accurate approximations of function V solution of the Poisson equation

$$-\nabla.(\varepsilon\nabla V) = g, \quad \text{in } \Omega, \quad \varepsilon\nabla V.n = 0, \text{ on } \Gamma_N, \ V = V_D \text{ on } \Gamma_D, \tag{1}$$

with ε a positive function which may present some discontinuities, V_D is a given function defined on Γ_D and g is the source term. We assume that boundary Γ_N is composed of lines (2D case) or planes (3D case) while Γ_D is curved. In some applications, we shall split the domain into two sub-domains Ω_1 and Ω_2 shared by an interface Γ and we denote by $\varepsilon_\ell = \varepsilon_{|\Omega_\ell}$, $\ell = 1, 2$ the restrictions on each sub-domain. Functions ε_1 and ε_2 are regular but ε presents a discontinuity at the interface Γ. Such an assumption is required when dealing with two different materials (for instance the air and a dielectric material).

We introduce the notations to derive the finite volume scheme (see Fig. 1). For the two-dimensional case, the domain is divided into non-overlapping convex polygonal cells c_i, $i = 1, \ldots, I$ and we denote by $e_{ij} = c_i \cap c_j$ the common interface shared by two adjacent cells while n_{ij} stands for the unit normal vector from c_i towards c_j. To handle the edges on the boundary, we introduce the notation e_{iD} which corresponds to an edge of cell c_i which belongs to Γ_D while n_{iD} is the unit outward normal vector. In the same way we define e_{iN} and n_{iN} for the edge associated to Γ_N. At last ν_i is the index subset of the adjacent cells including symbols D, N when dealing with a boundary edge. To perform accurate numerical integration, we denote by $q_{ij,r}$, $r = 1, \ldots, R_2$ the Gauss points on edge e_{ij} while ξ_r, $r = 1, \ldots, R_2$ represent the respective weights. We adopt a similar notation setting $q_{iD,r}$ (resp. $q_{iN,r}$) for the Gauss points on edge e_{iD} (resp, e_{iN}).

The three-dimensional situation is more complicated due to the existence of possible non-coplanar interfaces. For the sake of simplicity, we shall only consider tetrahedral cells c_i, $i = 1, \ldots, I$ with triangular faces $f_{ij} = c_i \cap c_j$ with n_{ij} the unit normal vector from c_i pointing to c_j. As in the two-dimensional case, we use symbol D (resp. N) to represent a boundary face f_{iD} (resp. f_{iN}) and its associated outward normal vector n_{iD} (resp. n_{iN}). We also introduce the set ν_i to represent the index of the neighbour cells which share a common face, including D, N for the boundaries. Since the face are triangular, we denote by $q_{ij,r}$, $r = 1, \ldots, R_3$ the Gauss points with their associated weights ξ_r.

Remark 2.1 *We use the same notation $q_{ij,r}$ and ξ_r for the two- and three-dimensional case for the sake of simplicity since it will not give rise to any confusion.*

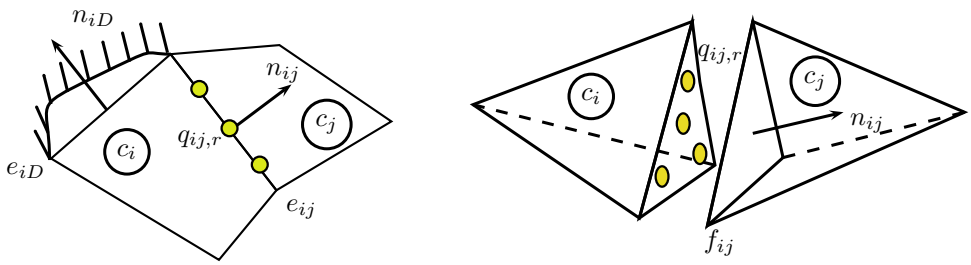

Fig. 1. Notation for the two-dimensional (left) and three-dimensional mesh

Integration by part on cell $c_i \subset \mathbb{R}^2$ (resp. $c_i \subset \mathbb{R}^3$) and substituting the exact integration on edge (resp. faces) with their respective quadrature formula provide the relations

$$G_i = \sum_{j\in\nu_i} \frac{|e_{ij}|}{|c_i|} \sum_{r=1}^{R_2} \xi_r F_{ij,r} - \bar{g}_i, \qquad \text{(resp.) } G_i = \sum_{j\in\nu_i} \frac{|f_{ij}|}{|c_i|} \sum_{r=1}^{R_3} \xi_r F_{ij,r} - \bar{g}_i,$$

where $|e_{ij}|$, $|f_{ij}|$, $|c_i|$ represent the respective measures of the geometrical entities while $\bar{g}_i = \dfrac{1}{|c_i|}\displaystyle\int_{c_i} g(x)\,dx$ stands for the mean value of function g on cell c_i and $F_{ij,r} = \varepsilon(q_{ij,r})\nabla V(q_{ij,r}).n_{ij}$. We substitute the exact (but unknown) flux by a numerical approximation $\mathcal{F}_{ij,r}$ at each Gauss points we shall detail in the next section and introduce the residual form

$$\mathcal{G}_i = \sum_{j\in\nu_i} \frac{|e_{ij}|}{|c_i|} \sum_{r=1}^{R_2} \xi_r \mathcal{F}_{ij,r} - g_i, \qquad \text{(resp.) } \mathcal{G}_i = \sum_{j\in\nu_i} \frac{|f_{ij}|}{|c_i|} \sum_{r=1}^{R_3} \xi_r \mathcal{F}_{ij,r} - g_i,$$

where g_i is a sixth-order approximation of the source term mean value using the Gauss points on cells c_i.

3 Polynomial reconstructions

Polynomial reconstructions in the finite volume context has been introduced by [20] but we use the technique initially proposed for hyperbolic system [21–23] and adapted for the convection-diffusion situation in [18]. Let us denote by V_i an approximation of $\bar{V}_i = \dfrac{1}{|c_i|}\displaystyle\int_{c_i} V(x)\,dx$ and $V = (V_i) \in \mathbb{R}^I$ be the vector which gathers all the components (reps. $\bar{V}$ with components $\bar{V}_i$). The section aims to define the numerical fluxes across edges or faces in function of vector V. Basically the idea is to build local polynomial representations based on the mean value approximations V_i and the boundary conditions. We have to distinguish several situations to adapt the polynomial reconstruction whether the edge (or the face) is on the boundary, on the interface Γ or is an inner edge. We rephrase the method presented in [18] for the sake of consistency and to introduce the notations but we highlight that several important new improvements will also be detailed: curved boundaries are now considered, a faster algorithm to compute values and derivatives at the Gauss points is designed and a new preconditioning matrix is proposed.

3.1 Inner face situation

We first consider the situation of cells and faces f_{ij} (or e_{ij}) which do not belong to the boundary. The case for e_{iD} and f_{iD} as well as e_{iN} and f_{iN} will be treated in the next subsections. We present the reconstruction for faces (3D case) but a very similar reconstruction is available for edges as presented in [18].

3.1.1 Conservative reconstruction for Cell

We first start with the conservative reconstruction associated to a cell. Such a reconstruction is usually dedicated to the convection contribution but, in the present study, we need it for cells in contact with the interface Γ. Let c_i be a cell, d the degree of the polynomial reconstruction we intend to build, and $S(c_i, d)$ the associated stencil constituted of cells **belonging to the same sub-domain** of cell c_i. We assume that the approximation V_i on c_i is given and the polynomial function of degree d writes

$$\widehat{V}_i(x;d) = V_i + \sum_{1\leqslant|\alpha|\leqslant d} \mathfrak{R}_i^{d,\alpha}\Big\{(x-m_i)^\alpha - M_i^\alpha\Big\}, \tag{2}$$

with $\alpha = (\alpha_1, \alpha_2, \alpha_3)$, $|\alpha| = \alpha_1 + \alpha_2 + \alpha_3$, $x = (x_1, x_2, x_3)$, m_i the cell centroid, and we adopt the convention $x^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} x_3^{\alpha_3}$. We set $M_i^\alpha = \dfrac{1}{|c_i|}\displaystyle\int_{c_i} (x-b_i)^\alpha \, dx$ such that the conservation property

$$\frac{1}{|c_i|}\int_{c_i} \widehat{V}_i(x;d)\, dx = V_i \tag{3}$$

holds. To fix the coefficients $\mathfrak{R}_i^{d,\alpha}$ of polynomial function (2) we assume that values V_ℓ on cells c_ℓ, $\ell \in S(c_i, d)$ are known, and we introduce the functional

$$E_i(\mathfrak{R}_i^d; d) = \sum_{\ell\in S(c_i,d)} \Big[\frac{1}{|c_\ell|}\int_{c_\ell} \widehat{V}_i(x;d)\, dx - V_\ell\Big]^2, \tag{4}$$

where $\mathfrak{R}_i^d$ is the vector gathering all the coefficients $\mathfrak{R}_i^{d,\alpha}$. Existence and uniqueness of the functional minimizer $\widehat{\mathfrak{R}}_i^d$ is guaranteed and provides the best approximation. We refer to [18] for a detailed description of the method and the pre-conditioning strategy to obtain vector $\widehat{\mathfrak{R}}_i^d$.

3.1.2 Conservative reconstruction for a face on interface Γ

We assume that $f_{ij} \subset \Gamma$ such that $f_{ij} = c_i \cap c_j$ with $c_i \subset \Omega_1$ and $c_j \subset \Omega_2$. To compute an accurate diffusion flux across interface Γ with a continuous transition of function V, we consider a polynomial function under its conservative form

$$\breve{V}_j(x;d) = V_j + \sum_{1\leqslant|\alpha|\leqslant d} \mathfrak{R}_j^{d,\alpha}\Big\{(x-m_j)^\alpha - M_j^\alpha\Big\}.$$

where we assume that V_j is a relevant information on cell c_j. We slightly modify the former functional to minimize, namely

$$E_j(\mathfrak{R}_j^d; d) = \sum_{\ell \in S(c_j,d)} \Big[\frac{1}{|c_\ell|} \int_{c_\ell} \breve{V}_j(x; d)\ dx - V_\ell \Big]^2 + \omega_{ij} \Big[\frac{1}{|f_{ij}|} \int_{f_{ij}} \breve{V}_j(x; d)\ ds - V_{ij} \Big]^2 \tag{5}$$

introducing a positive weight ω_{ij} while V_{ij} is a relevant information on face f_{ij}. The functional takes the continuity condition into account across Γ, controlled by parameter ω_{ij} to be fixed in the numerical applications. The minimizing vector $\breve{\mathfrak{R}}_j^d$ is solution of an over-determined system, in the least-square sense, pre-conditioned as in [18].

3.1.3 Non-conservative reconstruction for an inner face

An inner face of Ω_1 or Ω_2 is a face of the form $f_{ij} = c_i \cap c_j$ such that cells c_i, c_j both belong to the same sub-domain. For a given degree d, we associate a stencil $S(f_{ij}, d)$ **constituted of cells of the same sub-domain** than f_{ij}. We consider the polynomial reconstruction

$$\widetilde{V}_{ij}(x; d) = \sum_{0 \leqslant |\alpha| \leqslant d} \mathfrak{R}_{ij}^{d,\alpha} (x - m_{ij})^\alpha, \tag{6}$$

where m_{ij} is the centroid of face f_{ij}. Assuming that we know V_ℓ on cells c_ℓ, $\ell \in S(f_{ij}, d)$, we introduce the functional

$$E_{ij}(\mathfrak{R}_{ij}^d; d) = \sum_{\ell \in S(f_{ij},d)} \omega_{ij,\ell} \Big[\frac{1}{|c_\ell|} \int_{c_\ell} \widetilde{V}_{ij}(x; d)\ dx - V_\ell \Big]^2, \tag{7}$$

where $\omega_{ij,\ell}$ are positive weights and $\mathfrak{R}_{ij}^d$ is the vector gathering all the coefficients $\mathfrak{R}_{ij}^{d,\alpha}$. As mentioned in [18], weights are introduced to provide the positivity preserving property (see the numerical test 4.1) and are defined in the following way: if $\ell = i, j$, $\omega_{ij,\ell} = \omega \geqslant 1$ else $\omega_{ij,\ell} = 1$. Vector $\widetilde{\mathfrak{R}}_{ij}^d$ stands for the unique vector minimizing the functional and provides the best approximation.

3.2 Polynomial reconstruction for the Dirichlet condition on curved boundary

When dealing with curved boundaries, the substitution of domain Ω with the polygonal (or polyhedral) domain $\Omega_h = \bigcup_{i=1,\dots,I} c_i$ dramatically reduces the method order due to a rough approximation of the boundary with line segments. A specific treatment of the polynomial reconstruction associated to edge e_{iD} or face f_{iD} is required. For the Neumann case, we have assumed that Γ_N is polygonal (or polyhedral) in the present study for the sake of simplicity hence no particular treatment is required.

3.2.1 Second-order approach

As a first approach, we recall the simple second-order method where one uses the mean value of the Dirichlet condition on the boundary edge. We present the two-dimensional situation since the extension for the three-dimensional case is straightforward. Let e_{iD} be an edge situated on the boundary Γ_D and set the mean value $\bar{V}_{iD} = \dfrac{1}{|e_{iD}|}\displaystyle\int_{e_{iD}} \phi_D(s)\ ds$. In [18] the following conservative polynomial function of degree d has been considered

$$\widehat{V}_{iD}(x;d) = V_{iD} + \sum_{1\leqslant|\alpha|\leqslant d} \mathfrak{R}_{iD}^{d,\alpha}\Big\{(x-m_{iD})^{\alpha} - M_{iD}^{\alpha}\Big\}, \tag{8}$$

taking $V_{iD} = \bar{V}_{iD}$, m_{iD} the midpoint of edge e_{iD} and $M_{iD}^{\alpha} = \dfrac{1}{|e_{iD}|}\displaystyle\int_{e_{iD}} (x-m_{iD})^{\alpha}\ ds$ such that the conservative property $\dfrac{1}{|e_{iD}|}\displaystyle\int_{e_{iD}} \widehat{V}_{iD}(x;d)\ ds = V_{iD}$ holds. To fix the coefficients, we introduce the functional

$$E_{iD}(\mathfrak{R}_{iD}^{d};d) = \sum_{\ell\in S(f_{iD},d)} \omega_{iD,\ell}\Big[\frac{1}{|c_{\ell}|}\int_{c_{\ell}} \widehat{V}_{iD}(x;d)\ dx - \phi_{\ell}\Big]^2, \tag{9}$$

where $\omega_{iD,\ell}$ are positive weights and vector $\widehat{\mathfrak{R}}_{iD}^{d}$ stands for the unique vector minimizing the functional and providing the best approximation. The three-dimensional situation is identical (just substitute e_{iD} with f_{iD}).

If the boundary is a line (a plane), we get a $d+1$th-order method but such an approach achieves, at most, a second-order approximation when we substitute the mean value on a curved arc by the mean value on edge e_{ij}. The key-point is to evaluate the polynomial reconstruction with a better V_{iD} choice in formulae (8), different to the candidate $\bar{V}_{iD}$ proposed above, in order to provide better approximations of $\widehat{V}_{iD}(x;d)$ in the sense we shall specify in the sequel.

3.2.2 The two-dimensional case

3.2.2.1 Geometrical ingredients To provide a better approximation of the boundary condition, a local parametrization is introduced and a new quadrature formula is used to perform accurate numerical integrations on the arc. Let e be a generic boundary edge. We denote by v_1, v_2 the vertices and by $e = v_1v_2$ the segment with length $|v_1v_2|$ while $\widehat{v_1v_2}$ represents the boundary arc between v_1 and v_2 with length $|\widehat{v_1v_2}|$ (see Fig. 2 left panel).
We introduce the edge parametrization $q(t) = (1-t)v_1 + tv_2$, $t\in[0,1]$ which satisfies $|q'(t)| = |v_1v_2|$ while $p(t)$ is a parametrization of the arc such that

$$p(0) = v_1,\ p(1) = v_2,\ |p'(t)| = |\widehat{v_1v_2}| \text{ is constant.}$$

Let us denote by q_1, ..., q_{R_2} the Gauss points on edge e associated to parameters t_1, ..., t_{R_2}, then $p_1 = p(t_1)$, ..., $p_{R_2} = p(t_{R_2})$ are the corresponding Gauss points on the boundary arc.

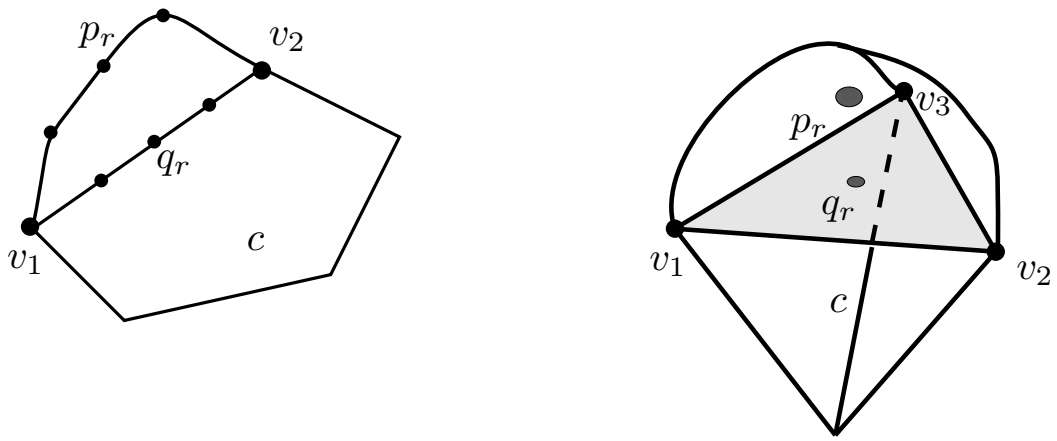

Fig. 2. Geometrical ingredients for the sixth-order approximation of the Dirichlet condition with curved boundary: the two-dimensional (left) and three-dimensional case (right). Points q_r are the Gauss points on the edge (face) while points p_r are the Gauss points on the curved boundary.

Indeed, using the quadrature rule for the numerical integration over the arc, one has

$$\int_{\widehat{v_1v_2}} V_D(p)dp = \int_0^1 V_D(p(t))|p'(t)|dt \approx \sum_{r=1}^{R_2} \xi_r V_D(p_r)|p'(t_r)| = |\widehat{v_1v_2}| \sum_{r=1}^{R_2} \xi_r V_D(p_r).$$

Notice that the following property then holds for $r = 1, \cdots, R_2$

$$\frac{|\widehat{v_1p_r}|}{|v_1q_r|} = \frac{|\widehat{v_2p_r}|}{|v_2q_r|} = \frac{|\widehat{v_1v_2}|}{|v_1v_2|}$$

which, in practice, enable to determine the location of points p_r on the arc.

3.2.2.2 The polynomial reconstruction on boundary Let us now consider $\widehat{e}_{iD}$ the boundary arc associated to the edge e_{iD} and assume that mean value approximations V_i, $i = 1, \ldots, I$ are given. We consider the following linear operator deriving from equation (8)

$$V_{iD} \to \widehat{V}_{iD}(x; d, V_{iD}) \in \mathbb{P}_d$$

where V_{iD} represents an approximation of the mean value on e_{iD}. We explicitly mention the variable V_{iD} in the polynomial function arguments since the dependence is of crucial importance in the procedure we shall implement. The main difficulty we have to face is that the Dirichlet condition is defined on $\widehat{e}_{iD}$ **and not** on e_{iD}. To overcome the problem, we introduce the functional

$$H(V_{iD}) = \sum_{r=1}^{R_2} \left(\widehat{V}_{iD}(p_{ij,r}; d, V_{iD}) - V_D(p_{ij,r}) \right)^2$$

which corresponds to the error between the polynomial approximation and the real Dirichlet condition at the Gauss points on the boundary arc. Since we are dealing with a quadratic functional, existence and uniqueness of the minimum $V_{iD}^\star$ is guaranteed and $\widehat{V}_{iD}(x; d) = \widehat{V}_{iD}(x; d, V_{iD}^\star)$ will be the polynomial reconstruction we use to compute the diffusive flux on edge e_{iD}.

To determine $V_{iD}^\star$ in practice, we propose the following simple algorithm. We consider the sequence V_{iD}^k initialized with $V_{iD}^0 = \bar{V}_{iD}$ and given by

(1) with V_{iD}^k in hand, compute the associated polynomial function $\widehat{V}_{iD}(x; d, V_{iD}^k)$,
(2) evaluate the errors $\delta_r^k = V_D(p_{ij,r}) - V_{iD}(p_{ij,r}; d, V_{iD}^k)$ at the Gauss points of the arc,
(3) update the mean value on e_{iD} with $V_{iD}^{k+1} = V_{iD}^k + \displaystyle\sum_{r=1}^{R_2} \xi_r \delta_r^k$,
(4) stop if $|V_{iD}^{k+1} - V_{iD}^k| < \epsilon_D \bar{V}_{iD}$ where the tolerance ϵ_D has been prescribed and set $V_{iD}^\star = V_{iD}^{k+1}$, else goto step (1).

Numerical experiments shows that we quickly converge with two or three steps using $\epsilon_D = 10^{-12}$. Furthermore, when $|\bar{V}_{iD}| < \epsilon_D$, we use the absolute error criterion $|V_{iD}^{k+1} - V_{iD}^k| < \epsilon_D$ in place of the relative error criterion.

Remark 3.1 *Notice that the method only requires the arc length $|\widehat{e}_{iD}|$, the Gauss points $q_{iD,r}$ on the edge and the associated Gauss points $p_{iD,r}$ on the boundary arc. No geometrical transformation is performed which provides a very simple method, easy to implement. The main difficulty is the determination of $p_{iD,r}$ for a general arc of curve.*

3.2.3 The three-dimensional case

Let f_{iD} be a triangular face situated on the boundary Γ_D and set the mean value $\bar{V}_{iD} = \dfrac{1}{|f_{iD}|} \displaystyle\int_{f_{iD}} V_D(s)\, ds$. As in the two-dimensional case, the polynomial reconstruction $\widehat{V}_{iD}$ with $V_{iD} = \bar{V}_{iD}$ only provides at most a second-order approximation with curved boundary. We propose an extention of the method to achieve an effective sixth-order accuracy.

3.2.3.1 Geometrical ingredients Let f be a boundary triangle face characterized by vertices v_1, v_2, v_3. We set $f = v_1v_2v_3$ with area $|v_1v_2v_3|$ while $\widehat{f} = \widehat{v_1v_2v_3}$ represents the boundary surface between v_1, v_2, v_3 with area $|\widehat{v_1v_2v_3}|$ (see Fig. 2 right panel). Let $\triangle = \{(s,t) \in [0,1]^2$ such that $s + t \leqslant 1\}$. We introduce the parametrization $q(s,t) = (1-t-s)v_1 + sv_2 + tv_3$, $(s,t) \in \triangle$ of face f which provides $|\partial_s q \wedge \partial_t q| = |v_1v_2 \wedge v_1v_3| = 2|v_1v_2v_3|$. On the other hand, we assume that $p(s,t)$, $(s,t) \in \triangle$ is a parametrization of the surface $\widehat{f}$ such that

$$p(0,0) = v_1,\ p(1,0) = v_2,\ p(0,1) = v_3,\ |\partial_s p \wedge \partial_t p| = 2|\widehat{v_1v_2v_3}| \text{ is constant.}$$

Let denote by $q_1, \ldots, q_{R_3}$ the Gauss points on f associated to parameters $(s_1,t_1), \ldots, (s_{R_3},t_{R_3})$, then $p_1 = p(s_1,t_1), \ldots, p_{R_3} = p(s_{R_3},t_{R_3})$ are the corresponding Gauss points on the boundary surface and the quadrature rule for the numerical integration over the surface writes

$$\int_{\widehat{v_1v_2v_3}} V_D(p)dp = \int_{\triangle} V_D(p(s,t))|\partial_s p \wedge \partial_t p|dsdt \quad (\text{see that} |\Delta| = \frac{1}{2})$$
$$\approx \frac{1}{2}\sum_{r=1}^{R_2} \xi_r V_D(p_r)|\partial_s p(s_r,t_r) \wedge \partial_t p(s_r,t_r)| = |\widehat{v_1v_2v_3}| \sum_{r=1}^{R_2} \xi_r V_D(p_r).$$

Notice that the following property holds for $r = 1, \cdots, R_3$

$$\frac{|\widehat{v_1v_2p_r}|}{|v_1v_2q_r|} = \frac{|\widehat{v_2v_3p_r}|}{|v_2v_3q_r|} = \frac{|\widehat{v_3v_1p_r}|}{|v_3v_1q_r|} = \frac{|\widehat{v_1v_2v_3}|}{|v_1v_2v_3|} \tag{10}$$

which enable, in practice, to determine points p_r. Indeed, the main difficulty is the determination of the Gauss points p_r associated to q_r for any surface $\widehat{v_1v_2v_3}$. To this end, assume that there exists a function such that for each point $p \in \widehat{v_1v_2v_3}$ provides the three areas $|\widehat{v_1v_2p}|$, $|\widehat{v_2v_3p}|$, $|\widehat{v_3v_1p}|$. Based on that function, we determine point p_r associated to q_r using relation (10) where we introduce the functional

$$p \to \left(\frac{|\widehat{v_1v_2p}|}{|v_1v_2q_r|} - \chi\right)^2 + \left(\frac{|\widehat{v_2v_3p}|}{|v_2v_3q_r|} - \chi\right)^2 + \left(\frac{|\widehat{v_3v_1p}|}{|v_3v_1q_r|} - \chi\right)^2$$

with $\chi = \dfrac{|\widehat{v_1v_2v_3}|}{|v_1v_2v_3|}$. The Gauss point p_r corresponds to the minimum of the quadratic functional.

3.2.3.2 The polynomial reconstruction on boundary Let us denote by $\widehat{f}_{iD}$ the boundary surface associated to the triangular face f_{iD} and assume that mean value approximations V_i, $i = 1, \ldots, I$ are given. We consider one more time the following linear operator

$$V_{iD} \to \widehat{V}_{iD}(x; d, V_{iD}) \in \mathbb{P}_d$$

where V_{iD} represents an approximation of the mean value on f_{iD} while the Dirichlet condition are defined on $\widehat{f}_{iD}$. As in the two-dimensional case, we introduce the functional

$$H(V_{iD}) = \sum_{r=1}^{R_3} \left(\widehat{V}_{iD}(p_{ij,r}; d, V_{iD}) - V_D(p_{ij,r})\right)^2$$

which corresponds to the error between the polynomial approximation and the real Dirichlet condition at the Gauss points of the boundary surface. Existence and uniqueness of the minimum $V_{iD}^\star$ is obtained and $\widehat{V}_{iD}(x; d) = \widehat{V}_{iD}(x; d, V_{iD}^\star)$ will be the polynomial reconstruction. The algorithm to provide $V_{iD}^\star$ is the same as the one proposed in the two-dimensional case.

3.3 Numerical fluxes

With the the polynomial reconstructions in hand, we explicit the fluxes in function of the face. One has to distinguish four cases.

(1) Assume that f_{ij} is an inner face of Ω_1 or Ω_2, then we define the flux at the quadrature point by

$$\mathcal{F}_{ij,r} = -\varepsilon(q_{ij,r})\nabla\widetilde{V}_{ij}(q_{ij,r};d).n_{ij}.$$

(2) Assume that f_{iD} belongs to Γ_D, then the flux at the quadrature point writes

$$\mathcal{F}_{iD,r} = -\varepsilon(q_{iD,r})\nabla\widehat{V}_{iD}(q_{iD,r};d).n_{iD}.$$

Notice that we compute the flux at the Gauss points of the edge $q_{iD,r}$ and not at the Gauss points $p_{iD,r}$ of the boundary.

(3) Assume that f_{iN} is on Γ_N, then the flux at the quadrature point writes

$$\mathcal{F}_{iN,r} = 0.$$

(4) Assume that $f_{ij} \subset \Gamma$. We first compute $\widehat{V}_i(x;d)$ for the cell in $c_i \subset \Omega_1$ which shares the face f_{ij}. Then, we evaluate the quantity

$$V_{ij} = \sum_{r=1}^{R_3} \xi_r \widehat{V}_i(q_{ij,r};d)$$

which provides the mean value of $\widehat{V}_i$ over f_{ij}. At last, we compute $\breve{V}_j(x;d)$ on the other side of the interface for cell $c_j \subset \Omega_2$. To determine the flux across the face, we then use the formula

$$\mathcal{F}_{ij,r} = -\varepsilon_2(q_{ij,r})\nabla\breve{V}_j(q_{ij,r};d).n_{ji}.$$

For any vector $V \in \mathbb{R}^I$, we define the residual operator on cell c_i by

$$G_i(V) = \sum_{j\in\nu(i)} \frac{|f_{ij}|}{|c_i|} \sum_{r=1}^{R_3} \zeta_r \mathcal{F}_{ij,r}(V) - g_i \tag{11}$$

and introduce the affine operator $G(V) = (G_1(V), \ldots, G_I(V))^T$ from $\mathbb{R}^I$ into $\mathbb{R}^I$. The solution is vector $V^\star$ such that $G(V^\star) = 0$ while the consistency error is given by $G(\overline{V})$ with $\overline{V} = (\overline{V}_i)$ the vector of the mean values of the exact solution. We get a matrix-free problem we solve using the GMRES method (see [24,25] for a detail review of the method).

3.4 Improvements

Several modifications have been introduced to increase the method efficiency and reduce the computational cost in regards to the former version developed in [18].

3.4.1 Faster polynomial evaluation

Polynomial reconstructions are always evaluated at the Gauss points so it is a waste of computational resource to evaluate several times the same monomial functions such as $(x - m_i)^\alpha$ at

the same points. To cut the time consumption, we take advantage of the linearity of the reconstruction with respect to vector V. New notations are required to present the fast evaluation since we need local indexations for the stencil and the polynomial coefficients.

3.4.1.1 Polynomial $\widetilde{V}_{ij}$ We start with polynomial function $\widetilde{V}_{ij}$ associated to stencil $S(f_{ij}, d)$. Let $\#S(f_{ij}, d)$ be the number of elements of the stencil. We define a local indexation setting

$$\lambda \in \{1, \ldots, \#S(f_{ij}, d)\} \to \ell(\lambda) \in S(f_{ij}, d).$$

On the other hand, we introduce the one-parameter indexation for the multi-index α. Let denote

$$\#d = \frac{d(d+1)(d+2)}{6}$$

and $M(d) = \{\alpha \in \mathbb{N}^3; |\alpha| \leqslant d\}$. We define the one-to-one mapping by

$$\kappa \in \{1, \ldots, \#(d+1)\} \to \alpha(\kappa) = (\alpha_1, \alpha_2, \alpha_3) \in M(d)$$

corresponding to the following sequence

κ	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	$\ldots$
α	(0,0,0)	(1,0,0)	(0,1,0)	(0,0,1)	(2,0,0)	(1,1,0)	(0,2,0)	(1,0,1)	(0,1,1)	(0,0,2)	(3,0,0)	(2,1,0)	(1,2,0)	(0,3,0)	(2,0,1)	$\ldots$

such that $\alpha(\kappa)$, $\kappa \in \{1, \ldots, \#(d+1)\}$ ranges all the coefficients $\mathfrak{R}^{k,\alpha}$ of any polynomial function of degree d. The reciprocal operator from $M(d)$ onto $\{1, \ldots, \#(d+1)\}$ is given by

$$\alpha \to \kappa(\alpha) = \#|\alpha| + \alpha_3(\alpha_1 + \alpha_2) + \frac{\alpha_3(\alpha_3+3)}{2} + \alpha_2 + 1.$$

Thanks to the mapping, we introduce the local one-parameter indexation for coefficients setting $\widetilde{\mathfrak{R}}_{ij,\kappa} = \widetilde{\mathfrak{R}}^{\alpha}_{ij}$ with $\kappa = \kappa(\alpha)$ and we denote by $\widetilde{\mathfrak{R}}_{ij} = (\widetilde{\mathfrak{R}}_{ij,\kappa})_{\kappa=1,\ldots,\#(d+1)}$ the vector which gathers all the components. In the same way, for a given stencil $S(f_{ij}, d)$, we denote by $W_{ij,\lambda} = V_\ell$ with $\ell = \ell(\lambda)$ and define $W_{ij} = (W_{ij,\lambda})_{\lambda \in \{1,\ldots,\#S(f_{ij},d)\}}$ the vector which gathers all the components. At last, the minimization of functional (7) writes in the least square sense (see [18]) $\widetilde{A}_{ij}\widetilde{\mathfrak{R}}_{ij} = W_{ij}$ with

$$\widetilde{A}_{ij} = (\widetilde{A}_{ij,\lambda\kappa})_{\substack{\lambda=1,\ldots,\#S(f_{ij},d) \\ \kappa=1,\ldots,\#(d+1)}}$$

where the coefficients are determined in function of the stencil and the geometrical data of the mesh. The problem is equivalent to the matrix-vector product

$$\widetilde{\mathfrak{R}}_{ij} = \widetilde{A}^{\dagger}_{ij} W_{ij}, \quad \widetilde{A}^{\dagger}_{ij} = (\widetilde{A}^{\dagger}_{ij,\kappa\lambda})_{\substack{\kappa=1,\ldots,\#(d+1) \\ \lambda=1,\ldots,\#S(f_{ij},d)}}$$

with $\widetilde{A}^{\dagger}_{ij}$ the Moore-Penrose pseudo-inverse matrix computed during a pre-processing stage. Moreover the corresponding tables $\lambda \to \ell(\lambda)$ and $\kappa \to \alpha(\kappa)$ are also stored in memory at the beginning of the computation.

For each Gauss point $q_{ij,k}$, we define the vector

$$Q_{ij,r} = \begin{pmatrix} (q_{ij,r} - m_{ij})^{\alpha(1)} \\ (q_{ij,r} - m_{ij})^{\alpha(2)} \\ \dots \\ (q_{ij,r} - m_{ij})^{\alpha(\#(d+1))} \end{pmatrix} \in \mathbb{R}^{\#(d+1)}.$$

and one has

$$\widetilde{V}_{ij}(q_{ij,k}, d) = \sum_{\kappa=1}^{\#(d+1)} \widetilde{\mathfrak{R}}_{ij,\kappa}(q_{ij,r} - m_{ij})^{\alpha(\kappa)} = Q_{ij,r}^T \widetilde{\mathfrak{R}}_{ij} = (Q_{ij,r}^T \widetilde{A}_{ij}^{\dagger}) W_{ij}.$$

We also introduce the column vector

$$\widetilde{\chi}_{ij,r} = (\widetilde{\chi}_{ij,r\lambda})_{\lambda=1,\dots,\#S(f_{ij},d)} = Q_{ij,r}^T \widetilde{A}_{ij}^{\dagger} \in \mathbb{R}^{\#S(f_{ij},d)},$$

and the polynomial value at the Gauss point is then evaluated with the product

$$\widetilde{V}_{ij}(q_{ij,r}; d) = \sum_{\lambda=1}^{\#S(f_{ij},d)} \widetilde{\chi}_{ij,r\lambda} V_{\ell(\lambda)} = \widetilde{\chi}_{ij,r}^T W_{ij}. \tag{12}$$

We now deal with the gradient of the polynomial reconstruction. We introduce the $\#(d+1)$-length vector of elements of $\mathbb{R}^3$ (or a $3 \times \#(d+1)$ real matrix)

$$JQ_{ij,r} = \left(\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \dots \begin{pmatrix} \alpha_1 (q_{ij,r} - m_{ij})^{\alpha - (1,0,0)} \\ \alpha_2 (q_{ij,r} - m_{ij})^{\alpha - (0,1,0)} \\ \alpha_3 (q_{ij,r} - m_{ij})^{\alpha - (0,0,1)} \end{pmatrix}, \dots \right)$$

with the rule $\alpha - \beta = (\alpha_1 - \beta_1, \alpha_2 - \beta_2, \alpha_3 - \beta_3)$. In other words, for any $\kappa = 1, \cdots, \#(d+1)$, we have the $\mathbb{R}^3$-value entries

$$JQ_{ij,r\kappa} = \begin{pmatrix} \alpha_1(\kappa)(q_{ij,r} - m_{ij})^{\alpha(\kappa) - (1,0,0)} \\ \alpha_2(\kappa)(q_{ij,r} - m_{ij})^{\alpha(\kappa) - (0,1,0)} \\ \alpha_3(\kappa)(q_{ij,r} - m_{ij})^{\alpha(\kappa) - (0,0,1)} \end{pmatrix}.$$

Applying the same procedure, we deduce that

$$\nabla \widetilde{V}_{ij}(q_{ij,k}, d) = \sum_{\kappa=1}^{\#(d+1)} JQ_{ij,r\kappa} (\widetilde{A}_{ij}^{\dagger} W_{ij})_{\kappa}.$$

Setting matrix $\widetilde{J\chi}_{ij,r}$ with entries

$$\widetilde{J\chi}_{ij,r\lambda} = \sum_{\kappa=1}^{\#(d+1)} JQ_{ij,r\kappa}\widetilde{A}^{\dagger}_{ij,\kappa\lambda} \in \mathbb{R}^3,$$

the polynomial gradient at each Gauss point then corresponds to the product

$$\nabla\widetilde{V}_{ij}(q_{ij,r};d) = \sum_{\lambda=1}^{\#S(f_{ij},d)} \widetilde{J\chi}_{ij,r\lambda}V_{\ell(\lambda)} = \widetilde{J\chi}_{ij,r}W_{ij}. \tag{13}$$

The main advantage is that coefficients $\widetilde{\chi}_{ij,r\lambda} \in \mathbb{R}$ and $\widetilde{J\chi}_{ij,r\lambda} \in \mathbb{R}^3$ are pre-computed just one time at the initial stage. Hence, evaluations of $\widetilde{V}_{ij}(q_{ij,r};d)$ or $\nabla\widetilde{V}_{ij}(q_{ij,r};d)$ turn to be very efficient and the storage is strongly reduced since we just need several small vectors for each Gauss point to use relations (12) or (13). We dramatically reduce the computational effort since neither the polynomial coefficients, nor the monomial expressions have to be recomputed. Moreover the Moore-Penrose matrices used in [18] are now useless. We save a lot of memory since the coefficients storage is really lower that the storage of the Moore-Penrose matrix $\widetilde{A}^{\dagger}_{ij}$ and the polynomial coefficients $\widetilde{\mathfrak{R}}_{ij}$ for each face f_{ij}.

3.4.1.2 Polynomial $\widehat{V}_{iD}$ We now turn to the case of a conservative reconstruction for the faces associated to the Dirichlet condition. The main difference with the former case is the indexation since we the value on face f_{iD} is prescribed. Let $\#S(f_{iD},d)$ be the number of stencil elements. We define a local indexation

$$\lambda \in \{1,\dots,\#S(f_{iD},d)\} \to \ell(\lambda) \in S(f_{iD},d)$$

and introduce vector $W_{iD} \in \mathbb{R}^{\#S(f_{iD},d)+1}$ adding the extra-component $W_0 = V_{iD}$ and $W_{iD,\lambda} = V_{\ell(\lambda)}$ for $\lambda \in \{1,\dots,\#S(f_{iD},d)\}$. Using similar notations, vector

$$\widehat{\mathfrak{R}}_{iD} = (\widehat{\mathfrak{R}}_{iD,\kappa})_{\kappa=1,\dots,\#(d+1)}$$

is evaluated with $\widehat{\mathfrak{R}}_{iD} = \widehat{A}^{\dagger}_{iD}W_{iD}$ where

$$\widehat{A}^{\dagger}_{iD} = (\widehat{A}^{\dagger}_{iD,\kappa\lambda})_{\substack{\kappa=1,\dots,\#(d+1)\\ \lambda=0,\dots,\#S(f_{iD},d)}}.$$

Notice that the conservative property yields $\widehat{\mathfrak{R}}_{iD,1} = V_{iD}$.

For each Gauss point $q_{iD,k} \in f_{iD}$, we define vector

$$Q_{iD,r} = \begin{pmatrix} (q_{iD,r} - m_{iD})^{\alpha(1)} \\ (q_{iD,r} - m_{iD})^{\alpha(2)} \\ \cdots \\ (q_{iD,r} - m_{iD})^{\alpha(\#(d+1))} \end{pmatrix} \in \mathbb{R}^{\#(d+1)}.$$

and one has

$$\widehat{V}_{iD}(q_{iD,k}, d) = \sum_{\kappa=1}^{\#(d+1)} \widehat{\mathfrak{R}}_{iD,\kappa}(q_{iD,r} - m_{iD})^{\alpha(\kappa)} = Q_{iD,r}^T \widehat{\mathfrak{R}}_{iD} = (Q_{iD,r}^T \widehat{A}_{iD}^{\dagger}) W_{iD}.$$

Setting

$$\widehat{\chi}_{iD,r} = (\widehat{\chi}_{iD,r,\lambda})_{\lambda=0,\ldots,\#S(f_{iD},d)} = Q_{iD,r}^T \widehat{A}_{iD}^{\dagger} \in \mathbb{R}^{\#S(f_{iD},d)+1},$$

the polynomial value at each Gauss point is then evaluated with the product

$$\widehat{V}_{iD}(q_{iD,r}; d) = V_{iD} + \sum_{\lambda=1}^{\#S(f_{iD},d)} \widehat{\chi}_{iD,r\lambda} V_{\ell(\lambda)}.$$

For the gradient evaluation, we introduce the vector of elements of $\mathbb{R}^3$

$$JQ_{iD,r} = \left(\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \ldots \begin{pmatrix} \alpha_1 (q_{iD,r} - m_{iD})^{\alpha - (1,0,0)} \\ \alpha_2 (q_{iD,r} - m_{iD})^{\alpha - (0,1,0)} \\ \alpha_3 (q_{iD,r} - m_{iD})^{\alpha - (0,0,1)} \end{pmatrix}, \ldots \right)$$

and we write

$$\nabla \widehat{V}_{iD}(q_{iD,k}, d) = \sum_{\kappa=1}^{\#(d+1)} JQ_{iD,r\kappa} (\widehat{A}_{iD}^{\dagger} W_{iD})_{\kappa}.$$

Setting for $\lambda = 0, \cdots, \#S(f_{iD}, d)$

$$\widehat{J\chi}_{iD,r\lambda} = \sum_{\kappa=1}^{\#(d+1)} JQ_{iD,r\kappa} \widehat{A}_{iD,\kappa\lambda}^{\dagger} \in \mathbb{R}^3,$$

the polynomial gradient at each Gauss point then corresponds to the product

$$\nabla \widehat{V}_{iD}(q_{iD,r}; d) = \widehat{J\chi}_{iD,r0} V_{iD} + \sum_{\lambda=1}^{\#S(f_{iD},d)} \widehat{J\chi}_{iD,r\lambda} V_{\ell(\lambda)} = \sum_{\lambda=1}^{\#S(f_{iD},d)} \widehat{J\chi}_{iD,r\lambda} V_{\ell(\lambda)}.$$

3.4.2 A new preconditioning matrix

Due to the linearity of the problem and the reconstruction process, operator $G(V)$ is an affine mapping $V \to G(V) = AV - b$ with $b \in \mathbb{R}^I$ and A a $I \times I$ real numbers matrix. In [18], a new

preconditioning matrix method has been proposed taking advantage that the problem derived from the finite volume method. With coefficients $\widehat{J\chi}_{ij,r\lambda}$ in hand, we propose an other efficient preconditioning matrix, that is a matrix $P^\dagger \in \mathbb{R}^{I\times I}$ such as one solves the problem $P^\dagger G(V) = 0$ in place of $G(V) = 0$. Clearly the two problems are equivalent ($P^\dagger$ is assumed non-singular) but we expect a better conditioning problem by multiplying with $P^\dagger$.

To this end, we slightly modified some previous notations and introduce new ingredients. For a given cell c_i and $j \in \nu(i)$, the face f_{ij} is associated to a stencil $S(f_{ij}, d)$. Let $\#S(f_{ij}, d)$ be the number of elements of the stencil. We redefine the local indexation setting

$$\lambda \in \{1, \ldots, \#S(f_{ij}, d)\} \to \ell(i, j; \lambda) \in S(f_{ij}, d)$$

to highlight the dependence of function $\ell(i, j; .)$ to face f_{ij}. Since the function is one-to-one, there exists a unique reciprocal function

$$\ell \in S(f_{ij}, d) \to \lambda(i, j; \ell) \in \{1, \ldots, \#S(f_{ij}, d)\}.$$

Index i always belongs to $S(f_{ij}, d)$ by construction, then $\lambda(i, j; i)$ is the local index associated to i. For $j \in \nu(i)$, if f_{ij} is an inner cell, *i.e.* cell c_j exists, then by construction $j \in S(f_{ij}, d)$ and $\lambda(i, j; j)$ is the local index associated to j. If f_{ij} is on the boundary, we skip the index and denote by $\underline{\nu}(i) \subset \nu(i)$ the index set such that f_{ij} only are inner faces. Matrix $P \in \mathbb{R}^{I\times I}$ is then defined as follow: for the diagonal coefficient we set

$$P(i, i) = \sum_{j\in\underline{\nu}_i} \frac{|f_{ij}|}{|c_i|} \sum_{r=1,\ldots,R_3} \xi_r \widehat{J\chi}_{ij,r\lambda(i,j;j)}.n_{ij},$$

and the extra-diagonal coefficients for $j \in \underline{\nu}_i$ writes

$$P(i, j) = \frac{|f_{ij}|}{|c_i|} \sum_{r=1,\ldots,R_3} \xi_r \widehat{J\chi}_{ij,r\lambda(i,j;i)}.n_{ij}.$$

Matrix P is not necessary symmetric but have a symmetric structure *i.e.* the non-null entries are symmetric. In some extent, matrix P is the main part of matrix A where we have performed the assembly procedure only for specific entries, namely the diagonal and the extra-diagonal entries corresponding to the adjacent cells.

To provide the preconditioning matrix, we consider a matrix $P^\dagger$ **with the same null entries** than matrix P (the other entries still unknown) and denote by $C = P^\dagger P$ the product of the two matrices. Since a face f_{ij}, $j \in \underline{\nu}_i$, only shares two cells, the following equalities hold

$$\begin{aligned} C(i, i) &= P^\dagger(i, i)P(i, i) + \sum_{j\in\underline{\nu}_i} P^\dagger(i, j)P(j, i), \\ C(i, j) &= P^\dagger(i, j)P(j, j) + P^\dagger(i, i)P(i, j), \ \forall j \in \underline{\nu}_i. \end{aligned}$$

To fix the $P^\dagger$ matrix coefficients, we state that $C = I$ **only** for the non-null entries and we get

the system

$$P^\dagger(i,i)P(i,i) + \sum_{j\in\underline{\nu}_i} P^\dagger(i,j)P(j,i) = 1, \quad P^\dagger(i,j)P(j,j) + P^\dagger(i,i)P(i,j) = 0, \ \forall j \in \underline{\nu}_i.$$

After some algebraic manipulations, we obtain an explicit expression for $P^\dagger$, namely

$$P^\dagger(i,j) = -P(i,j)\frac{P^\dagger(i,i)}{P(j,j)}, \ j \in \underline{\nu}_i, \quad \text{with} \quad P^\dagger(i,i) = \frac{1}{P(i,i) - \sum_{j\in\underline{\nu}(i)} \frac{P(i,j)P(j,i)}{P(j,j)}}.$$

4 Numerical tests

The present section is dedicated to the quantitative and qualitative assessments of the scheme robustness and accuracy. We first tackle the stability question, then we propose convergence tests for regular and discontinuous diffusion coefficients. Numerical simulations with curved boundary are carried out to show the critical issues that a specific treatment is mandatory for the polynomial reconstruction to preserve the optimal order. At last, we present some numerical experiences deriving from a concrete physical application to highlight the performances of the improvements proposed in section 3.4

To compare the numerical approximation with the exact solution, we introduce two kinds of errors, namely:

$$E^1 = \sum_{i=1}^{I} |c_i||\phi_i - \bar{\phi}_i|, \quad E^\infty = \max_{i=1,\cdots,I} |\phi_i - \bar{\phi}_i|,$$

while the convergence order between two successive meshes of N_1 and N_2 cells is given by

$$\mathcal{O}^1 = d\frac{\ln(E_1^1/E_2^1)}{\ln(N_2/N_1)}, \quad \mathcal{O}^\infty = d\frac{\ln(E_1^\infty/E_2^\infty)}{\ln(N_2/N_1)},$$

with d the space dimension. In this study, we employ triangular meshes for the two-dimensional geometries and tetrahedral meshes for the three-dimensional simulations. Figures are obtained using the `gmsh` software [29].

4.1 Maximum principle

We first assess the scheme ability to preserve the positivity or satisfy the maximum principle. Indeed, in [18], numerical tests show that the choice of the weights in the non-conservative reconstruction process is of crucial importance. We have proceeded in the same way as [18],

experimenting several weight values and check the non-negativity of the inverse matrix of the global linear system. To this end, let consider the unit-length cube. Function $V(x,y,z) = \sin(\pi x)\sin(\pi y)\sin(\pi z)$ is the solution of problem $-\Delta V = g$ with $g = 3\pi^2 \sin(\pi x)\sin(\pi y)\sin(\pi z)$ and homogeneous boundary condition $V_D = 0$ hence the exact solution ranges in the interval $[0,1]$. The polynomial reconstruction process on face f_{ij} depends on weights $\omega_{ij,\ell}$ we introduce in the functional (7). We fix the values in the following way: if $\ell = i,j$ we prescribe $\omega_{ij,\ell} = \omega$ whereas $\omega_{ij,\ell} = 1$ for the other cells. All the numerical tests have been performed with the $\mathbb{P}_1$ reconstruction but the conclusion holds with a $\mathbb{P}_3$ or a $\mathbb{P}_5$ reconstruction.

Table 1
Minimum and maximum of the approximations with respect to the weight with a 2321 cells mesh (left) and a 6327 cells mesh (right). We have use the $\mathbb{P}_1$ reconstruction to compute the diffusive flux.

ω	min	max
1.7	-1.25	1.71
2.0	0.0013	1.01
4.0	0.0241	0.99

ω	min	max
1.7	-0.02	1.01
2.0	0.0004	0.99
4.0	0.0008	0.99

Table 1 provides the minimum and the maximum of the approximations in function of ω for two different meshes. We plot in Fig. 3 three simulations performed with three different values of ω. We clearly observe that the maximum principle is not fulfilled for $\omega = 1.7$ and $\omega = 2$ but, in the second case, we obtain a lower number of non-physical cells *i.e.* cells associated to a negative value or a value larger to 1. For ω large enough, (here $\omega = 4$), the inverse matrix has all the coefficients non-negative and the numerical solution ranges in $[0,1]$. In the following tests, we shall use the value $\omega = 4$ to fulfil the maximum principle and provide the stability.

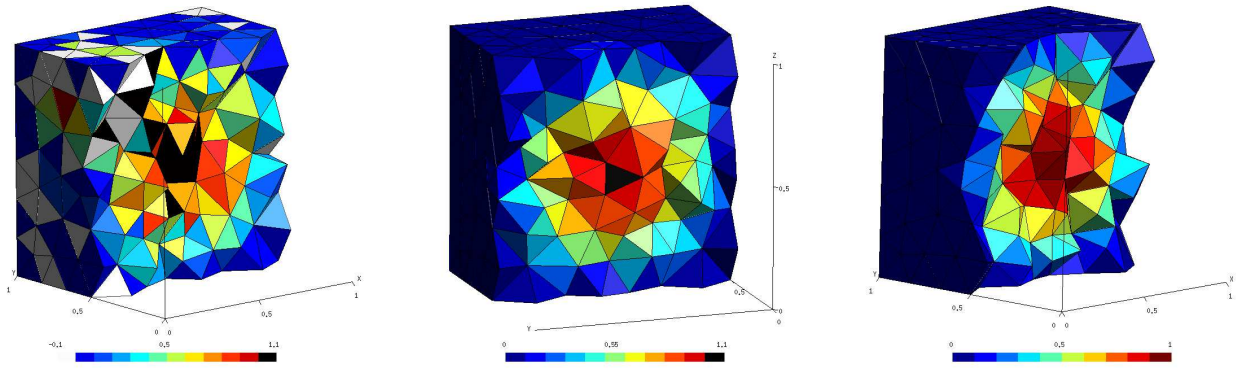

Fig. 3. Maximum principle property for different weights ω: the adjacent cells have a weight equal to 1.7 (left), the adjacent cells have a weight equal to 2 (middle) while the right panel displays the approximation for $\omega = 4$ (very close to the exact solution). White cells correspond to negative values while black cells correspond to values larger than one.

4.2 Convergence test

Convergence tests are carried out to check the order of accuracy of the scheme in the three-dimensional context. In the unit-length cube domain, we compute approximations of the solution for problem $-\Delta V = g$ with $g = 3\pi^2 \sin(\pi x)\sin(\pi y)\sin(\pi z)$ and homogeneous boundary condition $V_D = 0$ that we compare with the exact solution $V = \sin(\pi x)\sin(\pi y)\sin(\pi z)$. We have experimented three reconstructions and reported in Table 2 the error in L^1- and L^∞-norm with the associated convergence orders. We obtain an effective second-order, fourth-order and sixth-order of convergence using the $\mathbb{P}_1$, $\mathbb{P}_3$ and $\mathbb{P}_5$ reconstructions respectively. To highlight the efficiency of very high-order methods, a simple extrapolation shows that the error of $4.84e-08$ obtained with the sixth-order method with 32408 tetrahedra will require around 63 billions of cells to achieve the same error with a second-order method and a huge computational effort.

Table 2
L^1 and L^∞ errors and convergence rates for a smooth solution of the Poisson problem.

Nb of	$\mathbb{P}_1$				$\mathbb{P}_3$				$\mathbb{P}_5$			
Cells	err_1		err_∞		err_1		err_∞		err_1		err_∞	
2321	4.31e-03	—	1.93e-02	—	2.03e-04	—	1.72e-03	—	1.63e-05	—	7.61e-05	—
6371	2.28e-03	1.9	1.18e-02	1.5	4.95e-05	4.2	4.64e-04	3.9	1.52e-06	7.1	1.52e-05	4.8
16020	1.24e-03	2.0	6.09e-03	2.2	1.51e-05	3.9	1.88e-04	2.9	1.92e-07	6.7	2.90e-06	5.4
32408	7.54e-04	2.1	4.33e-03	1.5	5.41e-6	4.4	7.01e-05	4.2	4.84e-08	5.9	8.34e-07	5.3

4.3 Simulation with discontinuous dielectric coefficients

We tackle the important situation where the domain is constituted of two sub-domains with different dielectric coefficients leading to a discontinuity of the electrical field at the interface. Of course the test can be generalized to several domains with discontinuous coefficients. We consider the unit-length cube such that $\varepsilon_1 = 1$ in $\Omega_1 = \{x, y, z < 0.5\}$ and $\varepsilon_2 = 10$ in $\Omega_2 = \{x, y, z > 0.5\}$ *i.e.* the discontinuity interface Γ is located in the plane $z = 0.5$. The potential function V is composed of two functions $V_1 = V_{|\Omega_1}$ and $V_2 = V_{|\Omega_2}$ and the flux conservation condition across the interface Γ writes

$$\varepsilon_1 \nabla V_1 . n_\Gamma = \varepsilon_2 \nabla V_2 . n_\Gamma, \tag{14}$$

while we assume the continuity of the potential function $V_1 = V_2$ on Γ.

One can check that the functions

$$V_1(x,y,z) = \frac{A}{\varepsilon_1}\left(\frac{1}{\pi}\right)^2 \sin(\pi z) + \frac{a}{\varepsilon_1} z + V_{D1}, \quad V_2(x,y,z) = \frac{A}{\varepsilon_1}\left(\frac{1}{\pi}\right)^2 \sin(\pi z) + \frac{a}{\varepsilon_2}(z-1) + V_{D2}$$

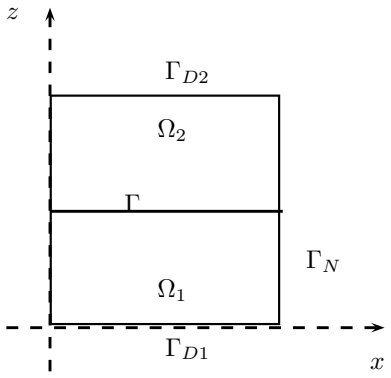

Fig. 4. The domain is constituted of two sub-domains with constant dielectric coefficients with a discontinuity at the interface Γ. We prescribe Dirichlet conditions on Γ_{D1} and Γ_{D2} and homogeneous Neumann condition on the lateral faces Γ_N.

with

$$a = \frac{1}{\frac{1}{2\varepsilon_1} + \frac{1}{2\varepsilon_2}} \left(A \left(\frac{1}{\pi} \right)^2 \left(\frac{1}{\varepsilon_2} - \frac{1}{\varepsilon_1} \right) + V_{D2} - V_{D1} \right),$$

are the solution of the electrical problem with the source term $g = A \sin(\pi z)$ with homogeneous Neumann condition on the lateral sides Γ_N (see Fig. 4) while we prescribe the Dirichlet condition $V(x, y, 0) = V_{D1}$ on Γ_{D1} and $V(x, y, 1) = V_{D2}$ on Γ_{D2}. All the numerical simulations are carried out using $A = 1$, $V_{D1} = 1$ and $V_{D2} = 10$

Table 3
Poisson problem with discontinuous coefficients. L^1 and L^∞ errors and convergence rates for the $\mathbb{P}_1$, $\mathbb{P}_3$ and $\mathbb{P}_5$ reconstructions.

Nb of	$\mathbb{P}_1$				$\mathbb{P}_3$				$\mathbb{P}_5$			
Cells	err_1		err_∞		err_1		err_∞		err_1		err_∞	
2495	4.55e-02	—	2.70e-01	—	1.23e-04	—	7.58e-04	—	1.30e-05	—	9.50e-06	—
5404	2.81e-02	1.87	1.79e-01	1.6	5.15e-05	3.38	2.78e-04	3.9	3.36e-06	5.25	1.57e-06	7.0
11546	1.59e-02	2.25	1.01e-01	2.25	1.74e-05	4.29	1.24e-04	3.19	6.25e-07	6.65	4.26e-07	5.65
24848	9.2e-03	2.14	9.58e-02	2.15	5.54e-06	4.48	3.84e-05	4.6	1.52e-07	5.55	1.08e-07	4.9

Table 3 shows that we obtain an effective second-, fourth- and sixth-order of convergence when using the $\mathbb{P}_1$, $\mathbb{P}_3$ and $\mathbb{P}_5$ reconstructions respectively. The low regularity of the solution in the vicinity of Γ is well-treated by the specific reconstruction given in section 3.1.2 and does not keep the scheme from achieving very high-order approximations.

4.4 A non-homogeneous test case

We now assess the convergence order when dealing with a non-constant smooth dielectric coefficient and non-homogeneous boundary conditions. The goal is to check that the scheme

effectively provides the optimal convergence order. To this end, we assume that the solution is $V(x,y,z) = \ln(1+x+y)z(1-z)$ and the dielectric coefficient is given by $\varepsilon(x,y,z) = 1+x+y$. Hence, one has to solve $-\nabla\cdot(\varepsilon\nabla V) = g$ with the right-hand side source term $g(x,y,z) = 2\ln(1+x+y)(1+x+y)$ and the Dirichlet condition is given by the exact solution on the whole boundary.

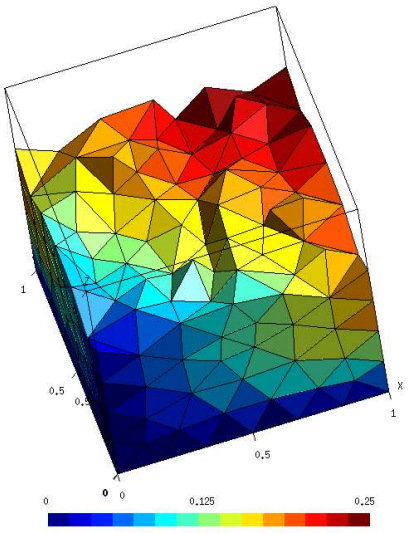

Fig. 5. A view of the solution for the Poisson equation with non-constant dielectric coefficient and a non-homogeneous boundary condition.

We display a cut of the numerical solution in Fig. 5 while we report in Table 4 the L^1 and L^∞ errors and convergence rates. Effective second-, fourth- and sixth-order of accuracy are obtained for the L^1 norm while the accuracy with the L^∞ norm is slightly worse but no instabilities are reported. To highlight the efficiency of the very high-order scheme, we observe that the sixth-order scheme cuts the error with an order of six magnitudes (in L^1-norm) in relation with the second-order one (for example with the 32048 cells mesh).

Table 4
L^1 and L^∞ errors and convergence rates using non-constant dielectric coefficient and non-homogeneous Dirichlet condition.

Nb of	$\mathbb{P}_1$				$\mathbb{P}_3$				$\mathbb{P}_5$			
Cells	err_1		err_∞		err_1		err_∞		err_1		err_∞	
2321	1.03e-03	—	4.76e-03	—	1.72e-06	—	2.01e-05	—	2.75e-08	—	6.50e-06	—
6327	5.01e-04	2.16	2.85e-03	1.53	4.41e-07	4.07	5.07e-06	4.12	3.2 e-09	6.43	1.39e-06	4.61
16020	2.7e-04	2.0	1.39e-03	2.32	1.32e-07	3.9	2.05e-06	2.92	5.7 e-10	5.57	2.49e-07	5.55
32048	1.7e-04	2.0	9.5e-04	1.65	4.82e-08	4.36	8.71e-07	3.7	1.32e-10	6.32	7.7e-07	5.08

An important goal of the present study is to achieve very high accuracy even with curved boundary. Indeed, a formal sixth-order scheme may be spoiled by a rough approximation of the boundary condition leading to, at most, a second-order method. We start with the two-dimensional case and assess the efficiency of the methods detailed in section 3.2.2 where the straightforward approximation on edge (Dirichlet condition applied on edges) is substituted by an algorithm to recover the optimal convergence order. We consider an annulus of external radius $R_1 = 10^{-6}$ and internal radius $r_2 = 10^{-7}$. We prescribe Dirichlet condition on the boundaries, namely $V = V_1$, $V = V_2$ on the external and internal circles respectively. The solution of problem $-\Delta V = 0$ is given by $V(x,y) = a\ln(\sqrt{x^2+y^2}) + b$ with $a = \dfrac{V_1 - V_2}{\ln(R_1) - \ln(R_2)}$ and $b = \dfrac{V_2 \ln(R_1) - V_1 \ln(R_2)}{\ln(R_1) - \ln(R_2)}$ (see Fig. 6).

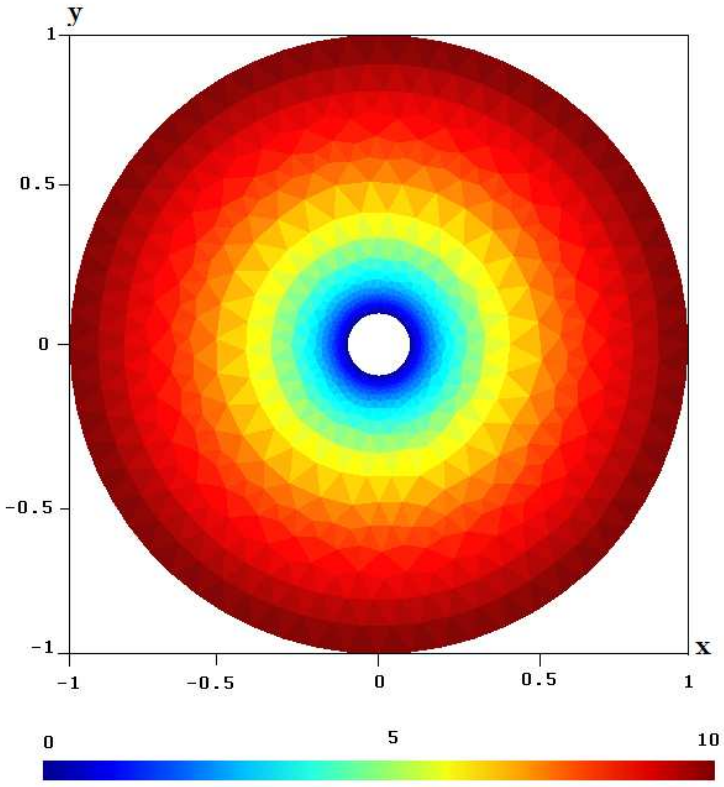

Fig. 6. Solution of the Poisson equation for the annulus problem.

Table 5 reports the errors and convergence orders using the simple approximation of the Dirichlet condition as proposed in section 3.2.1 (the Dirichlet condition is directly involved in the mean value over the face) while we report in Table 6 the errors and convergence orders when using the correction proposed in section 3.2.2.

Numerical evidences show that prescribing the Dirichlet on the edge in a straightforward man-

Table 5
L^1 and L^∞ errors and convergence rates for the two-dimensional annulus using the straightforward approximation of the Dirichlet condition.

Nb of	$\mathbb{P}_1$				$\mathbb{P}_3$				$\mathbb{P}_5$			
Cells	err_1		err_∞		err_1		err_∞		err_1		err_∞	
804	9.26E-02	—	1,95E-01	—	3,80E-02	—	1,50E-01	—	5.10E-02	—	1,38E-01	—
2178	3.09E-02	2.0	7.50E-02	1.9	9.46E-03	2.8	1.54E-02	4.1	9.60E-03	3.0	1.46E-02	4.10
8226	7.17E-03	2.2	2.14E-02	1.9	2.09E-03	2.3	4.43E-03	1.9	1.98E-03	2.4	4.19E-03	1.88
24502	2.92E-03	1.7	9.98E-03	1.1	8.05E-04	1.4	2.50E-03	0.9	7.79E-04	1.3	2.46E-03	0.75

Table 6
L^1 and L^∞ errors and convergence rates for the two-dimensional annulus using the correction method for the Dirichlet condition.

Nb of	$\mathbb{P}_1$				$\mathbb{P}_3$				$\mathbb{P}_5$			
Cells	err_1		err_∞		err_1		err_∞		err_1		err_∞	
804	6.40E-02	—	1,66E-01	—	1.98E-02	—	1.39E-01	—	2.40E-02	—	1.28E-01	—
2178	2.20E-02	2.2	6.65E-02	1.9	1.00E-03	5.1	7.48E-03	5.9	8.84E-04	6.6	4.34E-03	6.8
8226	5.19E-03	2.2	1.81E-02	2.0	4.87E-05	4.6	6.59E-04	3.7	4.99E-06	7.8	6.33E-05	6.0
24502	1.20E-03	2.1	4.35E-03	2.1	3.60E-06	3.7	7.49E-04	3.1	9.09E-08	5.9	2.26E-06	5.5

ner provides at most a second-order scheme where the limitation is clearly due to a poor evaluation of the Dirichlet condition define on a curved boundary. The technique we propose to overcome that limitation provides an excellent result since we recover the optimal sixth-order in the case of the $\mathbb{P}_5$ reconstruction (and the fourth-order with the $\mathbb{P}_3$ reconstruction). Such an example underlines that a correct approximation of the boundary condition in the case of curved boundary is crucial.

4.6 Three-dimensional problem with curved boundary

We now turn to the three-dimensional case following the strategy proposed for the two-dimensional annulus. The domain is situated between two spheres of radius $R_1 = 1$ and $R_2 = 0.1$ and we prescribe the Dirichlet condition $V = V_1$ and $V = V_2$ respectively. The solution of problem $-\Delta V = 0$ is given by $V(x, y, z) = \dfrac{a}{r} + b$ with $a = \dfrac{R_1 R_2 (V_1 - V_2)}{R_2 - R_1}$ and $b = \dfrac{R_2 V_2 - R_1 V_1}{R_2 - R_1}$ as shown in Fig. 7.

In Table 7, we report the errors and convergence orders using the straightforward approximation (the Dirichlet condition is directly involved in the mean value over the face) while we report in Table 8 the errors and convergence orders using the correction proposed in section 3.2.3. Alike the two-dimensional case, the poor approximation of the boundary ruins the efforts of

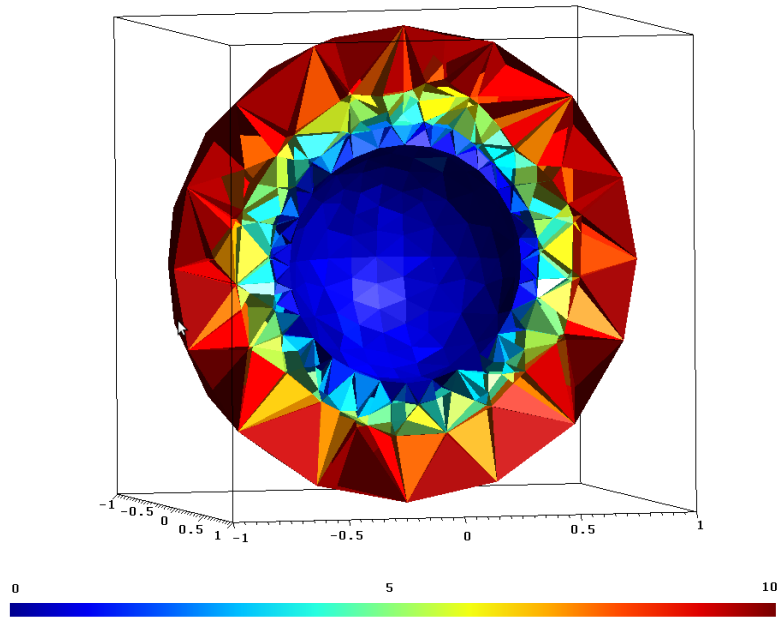

Fig. 7. Numerical approximation of the electrical potential between two spheres.

the polynomial reconstruction to achieve high accuracy whereas the corrected version recover the optimal orders.

Table 7
L^1 and L^∞ errors and convergence rates for the three-dimensional case using the straightforward method for the Dirichlet condition

Nb of Cells	$\mathbb{P}_1$				$\mathbb{P}_3$				$\mathbb{P}_5$			
	err_1		err_∞		err_1		err_∞		err_1		err_∞	
2924	5.19E-01	—	9.11E-01	—	6.41E-02	—	3.17E-01	—	2.24E-01	—	7.83E-01	—
7917	3.02E-01	1.63	5.87E-01	1.32	2.52E-02	2.81	1.59E-01	2.07	8.77E-02	2.82	2.96E-01	2.93
17901	1.96E-01	1.59	4.30E-01	1.15	1.27E-02	2.52	7.72E-02	2.66	4.39E-02	2.54	9.15E-02	4.32
31976	1.50E-01	1.38	3.06E-01	1.75	7.47E-03	2.75	4.33E-02	2.99	2.33E-02	3.28	5.90E-02	2.27

Table 8
L^1 and L^∞ errors and convergence rates for the three-dimensional case using the correction method for the Dirichlet condition

Nb of Cells	$\mathbb{P}_1$				$\mathbb{P}_3$				$\mathbb{P}_5$			
	err_1		err_∞		err_1		err_∞		err_1		err_∞	
2924	4.17E-01	—	8.34E-01	—	3.94E-02	—	2.77E-01	—	1.55E-01	—	5.60E-01	—
7917	2.06E-01	2,1	5.20E-01	1,4	1.07E-02	3.9	9.47E-02	3.2	7.33E-03	9.2	2.89E-02	8.9
17901	1.21E-01	1,9	3.27E-01	1,7	3.50E-03	4.1	2.90E-02	4.4	1.17E-03	6.8	4.54E-03	6.8
31976	8.11E-02	2,1	2.40E-01	1,6	1.72E-03	3.7	1.40E-02	3.8	4.11E-04	5.4	1.73E-03	5.0

To end the section we propose a concrete example and assess the efficiency of the new polynomial reconstruction detailed in section 3.4.1. The micro-tip is the main piece in Atomic Force Microscopy (AFM) set-up [26]. Fig. 8 left panel represents a general view of the micro-tip while the middle panel shows a zoom of the tip extremity which is the critical part of the piece. We aim at computing the electrostatic field since electrostatic interactions at the nano-scale are among those of greatest strength and longer interaction distance, compared to the other relevant forces such as van der Waals and capillarity [27]. We use a realistic model based on the Poisson equation to reproduce the electrostatic interaction in AFM working conditions [28].

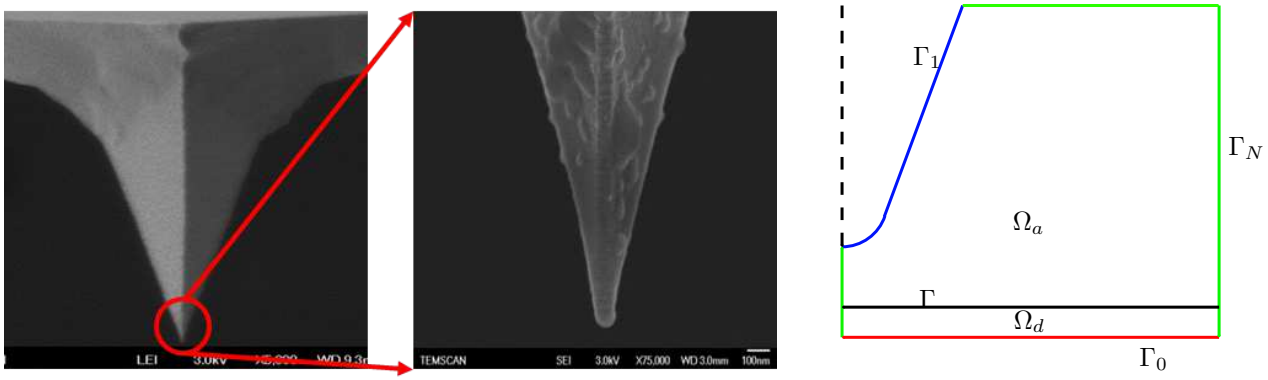

Fig. 8. The micro-tip.

Fig. 8 right panel sketches a longitudinal cut of the three-dimensional computational domain Ω constituted of air (domain Ω_a) and a dielectric (domain Ω_d). The boundary is partitioned into Γ_0 where we prescribe the null voltage condition, Γ_1 where we prescribe the bias voltage while we have an homogeneous Neumann condition on Γ_N since no current crosses the boundary. The model then writes

$$-\nabla \cdot (\varepsilon \nabla V) = \rho$$

with V the electrical potential, ρ the charge density, ε the dielectric coefficient and $E = -\nabla V$ the electric field. We prescribe

$$V = 0 \text{ on } \Gamma_0,\ V = V_b \text{ on } \Gamma_1,\ \nabla V \cdot n = 0 \text{ on } \Gamma_N,\ \varepsilon_a = 8.85\,10^{-12},\ \varepsilon_d = 10\varepsilon_a.$$

We here highlight the two main difficulties of the exercise: the dielectric coefficient is discontinuous at the interface Γ which shares the two domains and the tip extremity is spherical hence the boundary condition is not supported by the polyhedral mesh boundary. To overcome the difficulties, we have applied the techniques tested in section 4.3 and section 4.6. Numerical simulations have been carried out with a 88645 tetrahedra mesh where the tip extremity has been locally refined. Fig. 9 displays the electrical potential (left panel) and the electrical field (right panel) for a bias $V_b = 25\,V$.

To assess the performances of the improvements proposed in section 3.4, we have computed the electrical potential following two different methods. Method M1 uses the fast reconstruction

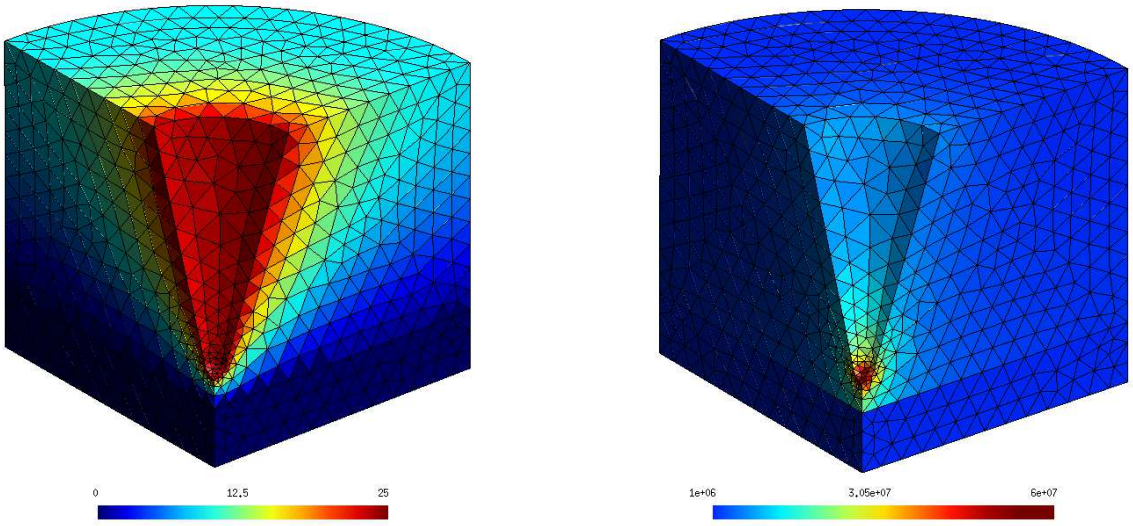

Fig. 9. The electrical potential (left) and electrical field (right) around the tip.

where polynomial functions are pre-calculated at the Gauss points (see section 3.4.1) and used the preconditioning matrix proposed in section 3.4.2. Method M2 is the original (or former) method where we recompute all the polynomial coefficients while the preconditioning matrix is based on an ersatz of Patankar matrix (see [18]). Table 9 provides the computational time (in second) and the number of iterations for both the methods using the $\mathbb{P}_1$, $\mathbb{P}_3$ and $\mathbb{P}_5$ reconstructions. Numerical simulations are carried out with three levels of meshes: very coarse, coarse and fine to evaluate the computational performances. Notice that method M2 with the P_5 reconstructions has fail (the GMRES procedure does not converge) or the computational time is too large.

Table 9
Computational time (in seconds) and number of iterations between the M1 (improved) and the M2 (original) method.

Nb of	$\mathbb{P}_1$				$\mathbb{P}_3$				$\mathbb{P}_5$			
Cells	M1		M2		M1		M2		M1		M2	
	time	iter	time	iter	time	iter	time	iter	time	iter	time	iter
3663	3	51	98	215	15	105	181	288	38	200	—	—
11339	13	75	361	237	27	147	711	340	112	237	—	—
22590	55	93	697	311	65	183	1263	512	204	321	—	—

Clearly, the new version requires three or four times less iterations to converge (the residual norm in the GMRES method is lower than 10^{-10}) and the computational time is cut by ten or more. We also observe that the computational effort of the new method is less sensitive to the polynomial degree in comparison with the original method. Notice that we do not have taken advantage of the high capacity of parallelization of the scheme (each evaluation of the flux is independent) which will provide very good speed-up.

5 Conclusions

We have proposed several extensions of the sixth-order finite volume method initially developed for two-dimensional geometries in [18]: an accurate method to prescribe the Dirichlet condition on bended boundary for the two- and the three-dimensional case which provide the optimal order; a new strategy to reduce the computational effort. Numerical tests have been carried to assess the technique efficiency in particular the question of curved boundary has been tackled. Extension of the method for the Stokes problem is the next step to demonstrate the versatility and the competitiveness of very high-order finite volume methods.

Acknowledgements

The first and third authors are supported by Region Midi-Pyrénées (France). The second author research was financed by FEDER Funds through Programa Operacional Factores de Competitividade — COMPETE and by Portuguese Funds through Fundação para a Ciência e a Tecnologia, within the project PTDC/MAT/121185/2010, the project FCT-ANR/MAT-NAN/0122/2012 and the strategic programme PEst-OE/MAT/UI0013/2014.

References

[1] C. Ollivier-Gooch, High-Order ENO schemes for unstructured Meshes based on least-squares reconstruction, AIAA Paper 97-0540, January 1997.

[2] T.J. Barth,P.O. Frederickson, Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction, AIAA 90-0013, January 1990.

[3] C. Ollivier-Gooch, M. Van Altena, A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation, Journal of Computational Physics Archive 181 (2) (2002), pp. 729–752.

[4] L. Ivan, C.P.T. Groth, High-order solution-adaptative central essentially non-oscillatory (CENO) method for viscous flows, AIAA 2011-367, January 2011.

[5] E.F. Toro, A. Hidalgo, ADER finite volume schemes for nonlinear reaction-diffusion equations, Applied Numerical Mathematics archive 59 (1) (2009), pp. 1–31.

[6] C. Michalak, C. Ollivier-Gooch Unstructured High-Order Accurate Finite-Volume Solutions of the Navier-Stokes Equations AIAA 2009-954, January 2009.

[7] L. Cea, J. Puertas, M.-E. Vázquez-Cendón Depth Averaged Modelling of Turbulent Shallow Water Flow with Wet-Dry Fronts, Arch. Comput. Eng. 14 (2007), pp. 303–341

[8] Y. Coudière, J.P. Vila, P. Villedieu, Convergence rate of a finite volume scheme for a two dimensional convection diffusion problem, Modl. Math. Anal. Numr. 33 (3) (1999), pp. 493–516.

[9] Y. Coudière, P. Villedieu, Convergence rate of a finite volume scheme for the linear convection-diffusion equation on locally refined meshes, M2AN Math. Model. Numer. Anal. 34 (6) (1999), pp. 1123–1149.

[10] R. Eymard, T. Gallouët, R. Herbin, The finite volume method, Handbook for Numerical Analysis, Ph. Ciarlet J.L. Lions eds., North Holland, (2000), pp. 715–1022.

[11] F. Hermeline, A finite volume method for the approximation of diffusion operators on distorted meshes, J. Comput. Phys. 160 (2000), pp. 481–499.

[12] J.A. Hernández, High-order finite volume schemes for the advection-diffusion equation, International Journal for Numerical Methods in Engineering, 53 (2002), pp. 1211–1234.

[13] K. Domelevo, P. Omnes, A finite volume method for the Laplace equation on almost arbitrary two-dimensional grids, M2AN Math. Model. Numer. Anal. 39 (2005), pp. 1203–1249.

[14] G. Manzini, A. Russo, A finite volume method for advection-diffusion problems in convection-dominated regimes, Comput. Methods Appl. Mech. Engrg. 197 (2008), pp. 1242–1261.

[15] Y. Coudière, G. Manzini, The discrete duality finite volume method for convection-diffusion problems, SIAM J. Numer. Anal. 47 (6) (2010), pp. 4163–4192.

[16] A. Hidalgo, M. Dumbser, ADER Schemes for Nonlinear Systems of Stiff Advection-Diffusion-Reaction Equations, J. Sci. Comput. 48 (2011) 173–189

[17] C. E. Zambra, M. Dumbser, E. F. Toro, N. Moraga, Finite Volume Schemes of High Order of Accuracy in Time and Space for Flows in Unsaturated Porous Media, International Journal for Numerical Methods in Engineering, 89 (2012) 227–240.

[18] S. Clain, G.J. Machado, J.M. Nobrega, R.M.S. Pereira, A sixth-order finite volume method for multidomain convection-diffusion problem with discontinuous coefficients Comput. Methods Appl. Mech. Engrg. 267 (2013) 43–64

[19] Z.J. Wang, Yen Liu, Extension of the spectral volume method to high-order boundary representation J. Comput. Phys. 211 (2006) 154–178

[20] T. J. Barth, Recent developments in high order k-exact reconstruction on unstructured meshes, AIAA 93-0668, January 1993.

[21] S. Clain, S. Diot, R. Loubère, A high-order polynomial finite volume method for hyperbolic system of conservation laws with Multi-dimensional Optimal Order Detection (MOOD), Journal of Computational Physics 230 (2011), pp. 4028–4050.

[22] S. Diot, S. Clain, R. Loubère, Improved detection criteria for the Multi-dimensional Optimal Order Detection (MOOD) on unstructured meshes with very high-order polynomials, Computers & Fluids 64 (2012) 43–63.

[23] S. Diot, R. Loubère, S. Clain, The MOOD method in the three-dimensional case: Very-High-Order Finite Volume Method for Hyperbolic Systems, International Journal for Numerical Methods in Fluids, 73 (2013) 362–392.

[24] Y. Saad, Iterative methods for spares linear systems, Society for Industrial and Applied Mathematics, 2003.

[25] Y. Sadd, M.H. Schultz, GMRES: a general minimal residual algorithm for solving nonsymmetric linear systems, SIAM journal of scientific and statistical computing 7 (3) (1986) 856–869.

[26] G. Binnig, C.F. Quate, and Ch. Gerber, Atomic Force Microscope, Phys. Rev. Lett. 56 (1986) 930–.

[27] J. Colchero, A. Gil, and A. M. Baro, Resolution enhancement and improved data interpretation in electrostatic force microscopy, Phys. Rev. B, 24 (2001), 245403

[28] B. Cappella, G. Dietler, Force-distance curves by atomic force microscopy, Surf. Sci. Rep. 34 (1999) 1–104.

[29] C. Geuzaine and J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities, International Journal for Numerical Methods in Engineering 79 (2009) 1309–1331.