HAL
open science

Accurate discretization of diffusion in the ls-stag cut-cell method using diamond cell techniques

B Portelenelle, Olivier Botella, Y Cheny

To cite this version:

B Portelenelle, Olivier Botella, Y Cheny. Accurate discretization of diffusion in the ls-stag cut-cell method using diamond cell techniques. ECCM ECFD 2018, Jun 2018, Glasgow, United Kingdom. hal-01874484

HAL Id: hal-01874484

https://hal.univ-lorraine.fr/hal-01874484v1

Submitted on 14 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

6th European Conference on Computational Mechanics (ECCM 6)
7th European Conference on Computational Fluid Dynamics (ECFD 7)
11-15 June 2018, Glasgow, UK

ACCURATE DISCRETIZATION OF DIFFUSION IN THE LS-STAG CUT-CELL METHOD USING DIAMOND CELL TECHNIQUES

B. PORTELENELLE[1,2], O. BOTELLA[1,2] and Y. CHENY[1,2]

[1] Université de Lorraine, LEMTA, UMR 7563, Vandoeuvre-lès-Nancy, F-54500, France
E-mail: name.surname@univ-lorraine.fr. URL: https://lemta.univ-lorraine.fr/

[2] CNRS, LEMTA, UMR 7563, Vandoeuvre-lès-Nancy, F-54500, France.

Key words: Computational fluid dynamics (CFD), Incompressible Flows, Heat Transfer, Cut-Cell Method, Gradient Discretisation, Data Interpolation.

1 INTRODUCTION

The LS-STAG method [7] is a Cartesian method for incompressible flow computations in irregular geometries which aims at discretizing accurately the flow equations in the cut-cells, *i.e.* cells of complex polyhedral shape formed by the intersection of the Cartesian mesh with the immersed boundary. In the three basic types of 2D cut-cells (pentagonal, trapezoidal and triangular, see Fig. 1(a)), the discretization is designed such as the global invariants of the flow (total mass, momentum and kinetic energy) are preserved at the discrete level. The LS-STAG discretization in the cut-cells is consistent with the MAC discretization used in Cartesian fluid cells, and has the ability to preserve the 5-point Cartesian structure of the stencil, resulting in a highly computationally efficient method. We have successfully applied the LS-STAG method to Newtonian flows at moderate Reynolds number in fixed and moving geometries [7], pseudoplastic flows [5], and viscoelastic flows [6]. In a recent paper [16], we have extended the LS-STAG discretization for 3D configurations with translational symmetry in the z direction (subsequently called *3D extruded* configurations), where are only present the 4 types of cells depicted in Fig. 1(b). In such geometries, only extruded 2D cut-cells are present, which enables us to easily extend the principles of the energy-conserving discretization of Ref. [7] to the cut-cells of Fig. 1(b) : the discretization of the continuity equation, pressure gradient, convective fluxes, normal stresses are straightforward extensions of the 2D LS-STAG formulas.

However, the discretization of cell-face fluxes involving shear stresses such as $\partial w/\partial x$ and $\partial w/\partial y$, which were absent from the 2D case, needed further attention. In effect, due to the non-orthogonality of the cut-cells, the use of 2-point formulas for computing these fluxes proves to be inaccurate. A way to improve the accuracy is to compute the whole velocity gradient at the cut-cell faces, thus decomposing the flux as an orthogonal contribution (using a standard 2-point formula) and non-orthogonal correction (using data at cell vertices). The data at cut-cell vertices are then interpolated from face-normal velocities and boundary conditions. This gradient reconstruction technique is commonly denominated “secondary gradients" [14] in the CFD community and “diamond cell method" [8] in the applied mathematics community.

This issue is also found for computing heat transfer problems with the 2D version of the LS-STAG method. Hence, for the ease of discussion the diamond cell method will first be described and analyzed for the diffusion fluxes of the heat equation, using various interpolation

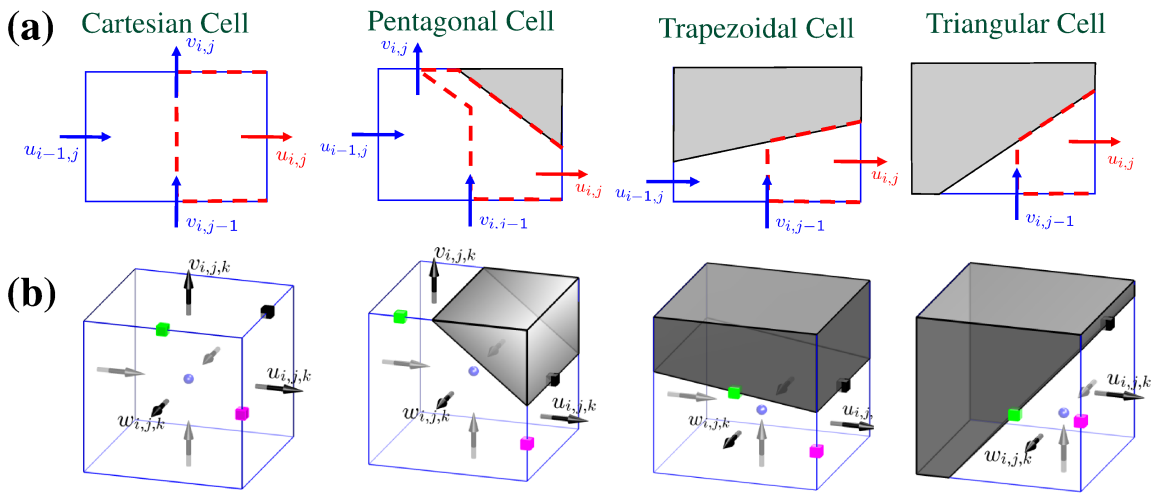

Figure 1: Cartesian cell and 3 basic types of cut-cells present in the LS-STAG mesh in 2D (a) and 3D extruded (b) geometries. For the 3D extruded mesh, the velocities are discretized at cell faces, pressure and normal stresses at cell center (●), shear stresses at edge centers : $\frac{\partial u}{\partial y}$, $\frac{\partial v}{\partial x}$ (■), $\frac{\partial u}{\partial z}$, $\frac{\partial w}{\partial x}$ (■) and $\frac{\partial v}{\partial z}$, $\frac{\partial w}{\partial y}$ (■). For heat transfer problems, the temperature is located at cell centers.

schemes (inverse distance weighting, least-squares, barycentric interpolation on triangulation). The accuracy of the discretization will firmly be assessed on a series of benchmark problems (Taylor-Couette flow, natural convection from a cylinder in an enclosure [9]) by inspecting the formal order of accuracy and the heat flux distribution at the immersed boundary. Finally, the diamond cell technique is employed for enhancing the accuracy of the shear stresses in 3D extruded geometries. Comparison with body-fitted CFD codes will be provided in terms of accuracy for benchmark flow past circular cylinder [3].

2 DISCRETIZATION OF DIFFUSION WITH THE DIAMOND CELL TECHNIQUES

The various discretization of diffusion that we consider are best detailed for the 2D temperature equation :

$$\frac{\partial T}{\partial t} + \nabla \cdot (\boldsymbol{v} T) - \kappa \boldsymbol{\nabla}^2 T = 0, \tag{1}$$

where T is the temperature, $\boldsymbol{v} = (u, v)$ is the convection velocity and κ is the thermal diffusivity. In cut-cell $\Omega_{i,j}$ depicted in Fig. 2, whose volume is denoted $V_{i,j}$, the temperature $T_{i,j}$ is located at the centroid $\boldsymbol{x}^{\mathrm{G}}_{i,j} = (x^G_{i,j}, y^G_{i,j})$, and the faces of the cut-cell are decomposed as $\Gamma_{i,j} = \Gamma^{\mathrm{w}}_{i,j} \cup \Gamma^{\mathrm{e}}_{i,j} \cup \Gamma^{\mathrm{s}}_{i,j} \cup \Gamma^{\mathrm{n}}_{i,j} \cup \Gamma^{\mathrm{ib}}_{i,j}$, where $\Gamma^{\mathrm{ib}}_{i,j}$ denotes the solid part of the cut-cell, while the fluid faces are denoted with the standard compass notations. The diffusive flux at the east face $F^{\mathrm{d}}_{\mathrm{e}} = \int_{\Gamma^{\mathrm{e}}_{i,j}} \kappa \nabla T \cdot \vec{n}_{\mathrm{e}} \, \mathrm{d}S$ is discretized with midpoint quadrature, yielding :

$$F^{\mathrm{d}}_{\mathrm{e}} \cong \theta^u_{i,j} \Delta y_j \kappa \left. \frac{\partial T}{\partial x} \right|_{\mathrm{e}}, \tag{2}$$

where $\theta^u_{i,j} \in [0, 1]$ is the fluid part of the face, such as $\theta^u_{i,j} \Delta y_j = \|\boldsymbol{x}_{\mathrm{N}} - \boldsymbol{x}_{\mathrm{S}}\|$ in Fig. 2, and $\partial T / \partial x|_{\mathrm{e}} = \nabla T|_{\mathrm{e}} \cdot \boldsymbol{n}_{\mathrm{e}}$ is the face-normal temperature gradient.

Several alternate formulas can be considered for computing $\partial T / \partial x|_{\mathrm{e}}$. The simplest is the

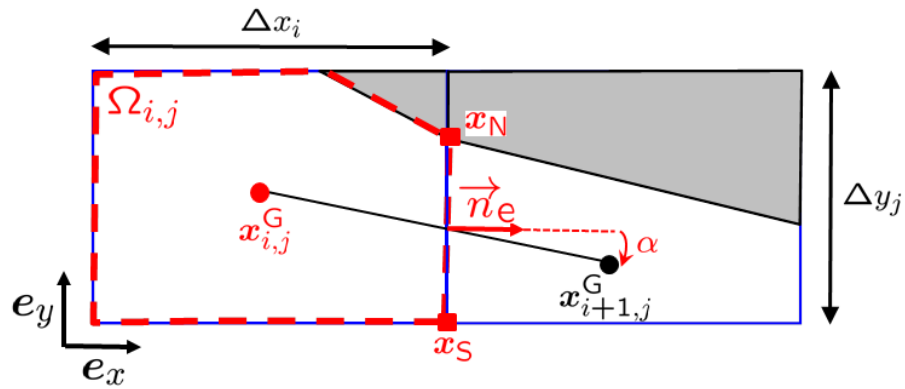

Figure 2: Relevant notations used for the discretization of the diffusion flux at the east face of cut-cell $\Omega_{i,j}$.

following two-point approximation :

$$\left.\frac{\partial T}{\partial x}\right|_{\mathrm{e}} \cong \frac{T_{i+1,j} - T_{i,j}}{\|\boldsymbol{d}\|}, \tag{3}$$

where $\boldsymbol{d} = \boldsymbol{x}^{\mathrm{G}}_{i+1,j} - \boldsymbol{x}^{\mathrm{G}}_{i,j}$ is the vector joining the cell centroids. In the case where $\boldsymbol{d}$ and the face normal vector $\boldsymbol{n}_{\mathrm{e}} = \boldsymbol{e}_x$ are colinear, formulae (2)-(3) recover the usual second-order Cartesian discretization. However, for the cut-cell depicted in Fig. 2, the colinearity is lost and formula (3) loses its accuracy as the angle $\alpha = \mathrm{mes}(\boldsymbol{e}_x, \boldsymbol{d})$ becomes large. This issue, which is analogous to the non-orthogonality of curvilinear and unstructured grids [15], has first been observed in [17] for the LS-STAG method in 3D extruded geometries. In this reference, they proposed to compute off-diagonal components of the velocity gradients of the Navier-Stokes equations with a formula consistent with the pressure gradient discretization. For the case of the temperature equation, this *ad hoc* formula amounts to :

$$\left.\frac{\partial T}{\partial x}\right|_{\mathrm{e}} \cong \frac{T_{i+1,j} - T_{i,j}}{(\frac{1}{2}V_{i+1,j} + \frac{1}{2}V_{i,j})/\theta^u_{i,j}\Delta y_j}. \tag{4}$$

However, the Navier-Stokes computations in [16] reported that the use of the above two-point formulas diminishes the accuracy of the LS-STAG method, and that only a superlinear order of convergence was obtained. The reason is that, when $\alpha \neq 0$, the face-normal gradient $\partial T/\partial x|_{\mathrm{e}}$ cannot be written as a function of $T_{i,j}$ and $T_{i+1,j}$ only, it also has a component tangential to the face also. It is thus necessary to compute the whole temperature gradient, which reads in the local basis of the east face :

$$\nabla T|_{\mathrm{e}} = \left.\frac{\partial T}{\partial x}\right|_{\mathrm{e}} \boldsymbol{e}_x + \left.\frac{\partial T}{\partial y}\right|_{\mathrm{e}} \boldsymbol{e}_y, \tag{5}$$

since $\boldsymbol{n}_{\mathrm{e}} = \boldsymbol{e}_x$ in Fig. 2.

An elegant way to proceed is to approximate the gradient in the $\boldsymbol{d}$ direction as :

$$\nabla T|_{\mathrm{e}} \cdot \frac{\boldsymbol{d}}{\|\boldsymbol{d}\|} = \cos\alpha \left.\frac{\partial T}{\partial x}\right|_{\mathrm{e}} + \sin\alpha \left.\frac{\partial T}{\partial y}\right|_{\mathrm{e}} \cong \frac{T_{i+1,j} - T_{i,j}}{\|\boldsymbol{d}\|}, \tag{6}$$

while the gradient tangential to the face is computed as :

$$\left.\frac{\partial T}{\partial y}\right|_{\mathrm{e}} \cong \frac{T_{\mathrm{N}} - T_{\mathrm{S}}}{\theta^u_{i,j}\Delta y_j}, \tag{7}$$

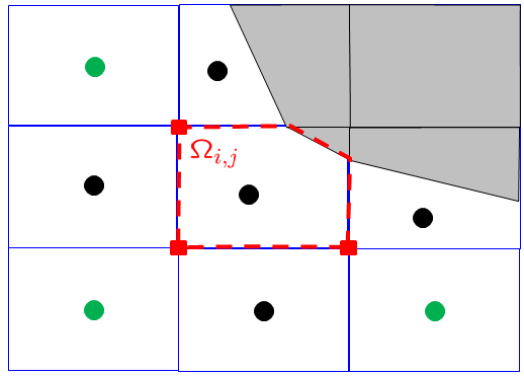

Figure 3: Stencil used for the discretization of diffusion in cut-cell $\Omega_{i,j}$. In black: 5-point stencil for 2-point discretization. For the DCM discretization with reconstruction at vertices (■), unknowns at the green centroids are added to the stencil.

where T_{N} and T_{S} are located at the face endpoints $\boldsymbol{x}_{\mathrm{N}}$ and $\boldsymbol{x}_{\mathrm{S}}$. Then, substitution of (7) in (6) gives the following approximation of the face-normal gradient :

$$\left.\frac{\partial T}{\partial x}\right|_{\mathrm{e}} \cong \frac{T_{i+1,j} - T_{i,j}}{\|\boldsymbol{d}\| \cos\alpha} - \frac{T_{\mathrm{N}} - T_{\mathrm{S}}}{\theta^{u}_{i,j} \Delta y_j} \tan\alpha, \tag{8a}$$

with

$$\|\boldsymbol{d}\| \cos\alpha = x^{\mathrm{G}}_{i+1,j} - x^{\mathrm{G}}_{i,j}, \quad \tan\alpha = \frac{y^{\mathrm{G}}_{i+1,j} - y^{\mathrm{G}}_{i,j}}{x^{\mathrm{G}}_{i+1,j} - x^{\mathrm{G}}_{i,j}}. \tag{8b}$$

The first quotient in the RHS of (8a) is the orthogonal contribution of the gradient discretization, while the second quotient is the non-orthogonal contribution that vanishes when $\boldsymbol{d}$ and $\boldsymbol{n}_{\mathrm{e}}$ are colinear ($\alpha = 0$), yielding a two-point approximation identical to (3) and (4). A similar procedure is employed to approximate diffusion at the immersed boundary. In the case of homogeneous boundary condtions $T = T_{\mathrm{ib}}$, the non-orthogonal contribution vanishes and the face normal gradient to $\Gamma^{\mathrm{ib}}_{i,j}$ is :

$$\left.\frac{\partial T}{\partial \boldsymbol{n}}\right|_{\mathrm{ib}} \cong \frac{T_{\mathrm{ib}} - T_{i,j}}{\Delta}, \tag{9}$$

where Δ is the perpendicular distance from point $\boldsymbol{x}^{\mathrm{G}}_{i,j}$ to segment $\Gamma^{\mathrm{ib}}_{i,j}$.

The gradient formula (8) can also be derived by applying the Green-Gauss theorem in the diamond-shaped control volume formed by joining the cell centers $\boldsymbol{x}^{\mathrm{G}}_{i,j}$, $\boldsymbol{x}^{\mathrm{G}}_{i+1,j}$ and the face endpoints $\boldsymbol{x}_{\mathrm{N}}$, $\boldsymbol{x}_{\mathrm{S}}$, and is known as the *diamond cell scheme*. The diamond cell scheme has been introduced in [12, 13, 11] for viscous compressible computations on unstructured meshes. Mathematical analysis has been performed in [8, 4] for diffusion equations. Furthermore, formula (8) is equivalent to the over-relaxed approach for diffusion discretization presented in [15].

3 PRACTICAL IMPLEMENTATION OF THE DIAMOND CELL SCHEME

In formula (8a), vertex value at $\boldsymbol{x}_{\mathrm{N}}$ is obtained from the boundary conditions ($T_{\mathrm{N}} = T_{\mathrm{ib}}$), while the value at $\boldsymbol{x}_{\mathrm{S}}$ need to be reconstructed from neighboring cell-centered values :

$$T_{\mathrm{S}} \cong \sum_{k,l=0}^{1} \omega_{i+k,j-1+l}\, T_{i+k,j-1+l}. \tag{10}$$

Bertolazzi & Manzini [4] reviewed various possibilities for defining the reconstruction weights $\omega_{i,j}$. Basic choices includes the inverse-volume interpolation, with weights based on the cell volumes :

$$\omega_{i,j} = \frac{V_{i,j}}{\sum_{k,l=0}^{1} V_{i+k,j-1+l}}, \tag{11}$$

and the inverse-distance interpolation, with weights :

$$\omega_{i,j} = \frac{\|\boldsymbol{x}_{i,j}^{\mathrm{G}} - \boldsymbol{x}_{\mathrm{S}}\|^{-p}}{\sum_{k,l=0}^{1} \|\boldsymbol{x}_{i+k,j-1+l}^{\mathrm{G}} - \boldsymbol{x}_{\mathrm{S}}\|^{-p}}, \tag{12}$$

where $p \geq 1$ is a given exponent ($p = 1$ corresponds to the Euclidian distance).

These two choices of weights give exact reconstruction for constant functions only, reducing thus the formal accuracy of the DCM formula to first-order only. In [4], an algorithm is discussed to reach second-order accuracy, by altering the values of the weights in order to fulfill the condition for exact reconstruction of linear functions. Note however that this algorithm involves centroid selection, solution of linear systems, and the resulting set of weights is not unique.

Second-order accuracy can also be reached by using least-square weights that minimize the reconstruction of linear functions. This least-square reconstruction is described in [11] for the solution of the compressible Navier-Stokes equations on unstructured grids, and has been analyzed in [8, 4] for model diffusion problems. However, our numerical experiments (see next section) showed that negative weights may appear for some cut-cells, giving rise to oscillatory solutions. For this reason, we have implemented a reconstruction based on triangulation [1] that ensures positivity of the weights. Out of the 4 triangles formed by the neighboring centroids of a given vertex, we select the triangle of largest minimal angle that contains the vertex. Value at the vertex is then obtained by barycentric interpolation.

In the LS-STAG code, diffusion is treated with implicit time-integration method, so the use of the DCM with vertex reconstruction has the effect to increase the stencil of the linear systems. A shown in Fig. 3, the DCM discretization needs to be applied to the four fluid faces of cut-cell $\Omega_{i,j}$, yielding a 9-point stencil instead of the standard 5-point stencil for 2-point discretization. Note that DCM needs only to be applied to neighboring cells of the immersed boundary. In the vast majority of the computational domain, mesh is orthogonal and standard 5-point stencil is recovered. The linear systems are solved with the HYPRE library (V2.10) using the semi-structured-grid system interface [10].

4 NUMERICAL RESULTS

The diamond cell method (8) (labeled herafter DCM) is implemented in the 2D and 3D-extruded LS-STAG code for the discretization of face-normal gradients in the temperature and Navier-Stokes equations. First, results are presented for the 2D temperature equation with no coupling with the Navier-Stokes equations. For interpolation at cell vertex, the following reconstruction schemes are compared :

– LS-STAG/DCM-ID1: first-order inverse distance weights (12) with $p = 2$,

- LS-STAG/DCM-ID2: second-order inverse distance weights (with $p = 2$) using the algorithm discussed in [4],
- LS-STAG/DCM-LSQ: least-square weights [11],
- LS-STAG/DCM-TRI: barycentric interpolation on triangulation.

The performance of DCM is also compared with the 2-point gradient (4), which is labeled LS-STAG/2PT. For all methods, Dirichlet boundary conditions are imposed with Eq. (9).

Subsequently, the temperature equation is coupled with the Navier-Stokes equations for computing 2D natural convection problems. The purpose is to assess the ability to compute accurately local heat flux at immersed boundaries, which is a sensible quantity since it involves the temperature gradient at the sold face of cut-cells. Once the DCM has been assessed on 2D flows, the DCM is applied to the Navier-Stokes equations in 3D-extruded geometries to enhance the discretization of isothermal incompressible flows computed with the LS-STAG method with 2-point discretization reported in [16].

4.1 TEMPERATURE CONDUCTION IN TAYLOR-COUETTE CELL

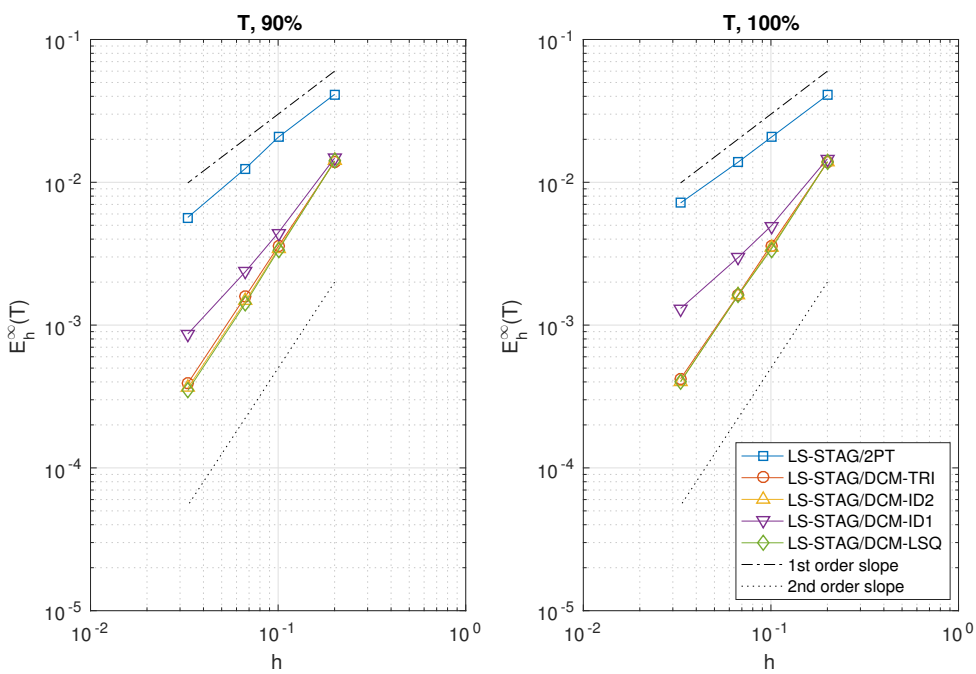

Figure 4: Maximal error on the temperature versus grid size h for the pure conduction test case. At left : on 90% of the domain, at right : on the whole domain.

First the spatial accuracy of the various variants of DCM is assessed on the test case of pure thermal conduction (Eq (1) with $\boldsymbol{v} = 0$) between concentric cylinders of diameters $r = R_1$ and $r = R_2$, where a temperature difference $\Delta T = T_2 - T_1$ is applied. The steady exact solution is purely radial :

$$T_{\text{ex}}(r) = \frac{\ln(r/R_2)}{\ln(R_1/R_2)}(T_1 - T_2) + T_2. \tag{13}$$

For the numerical experiments, we set $\Delta T = 1\text{K}$ and employ the same geometry ($R_1 = 1\,\text{m}$ and $R_2 = 4\,\text{m}$) and uniform meshes (of size $\Delta x = \Delta y = h$) considered in [5] for non-Newtonian computations with the LS-STAG method.

For the various discretization of the face-normal temperature gradient, Fig. 4 compares the maximal error :

$$E_h^\infty(T) = \frac{\max\limits_{\text{CVs } \Omega_{i,j}} \left| T_{i,j} - T_{\text{ex}}(\boldsymbol{x}_{i,j}^{\text{G}}) \right|}{\max\limits_{\text{CVs } \Omega_{i,j}} \left| T_{\text{ex}}(\boldsymbol{x}_{i,j}^{\text{G}}) \right|} \tag{14}$$

obtained on the whole computational domain and away from the immersed boundary. As expected, the 2-point method gives the worst accuracy, measured as first-order only. The errors given by all DCM variants are more than one decade lower, highlighting the benefits of this gradient computation. Note however that the errors of DCM-ID1 quickly saturates to a first-order rate when the grid size is reduced, due to the low accuracy of the vertex reconstruction scheme. All other reconstruction schemes (DCM-ID2, DCM-LSQ, DCM-TRI) gives comparable second-order accuracy. For these methods, it is worthwhile to note that the errors on 90% and 100% of the domain are of same magnitude, showing that the discretization in cut- and Cartesian cells has a comparable accuracy.

4.2 NATURAL CONVECTION FROM A CYLINDER IN AN ENCLOSURE

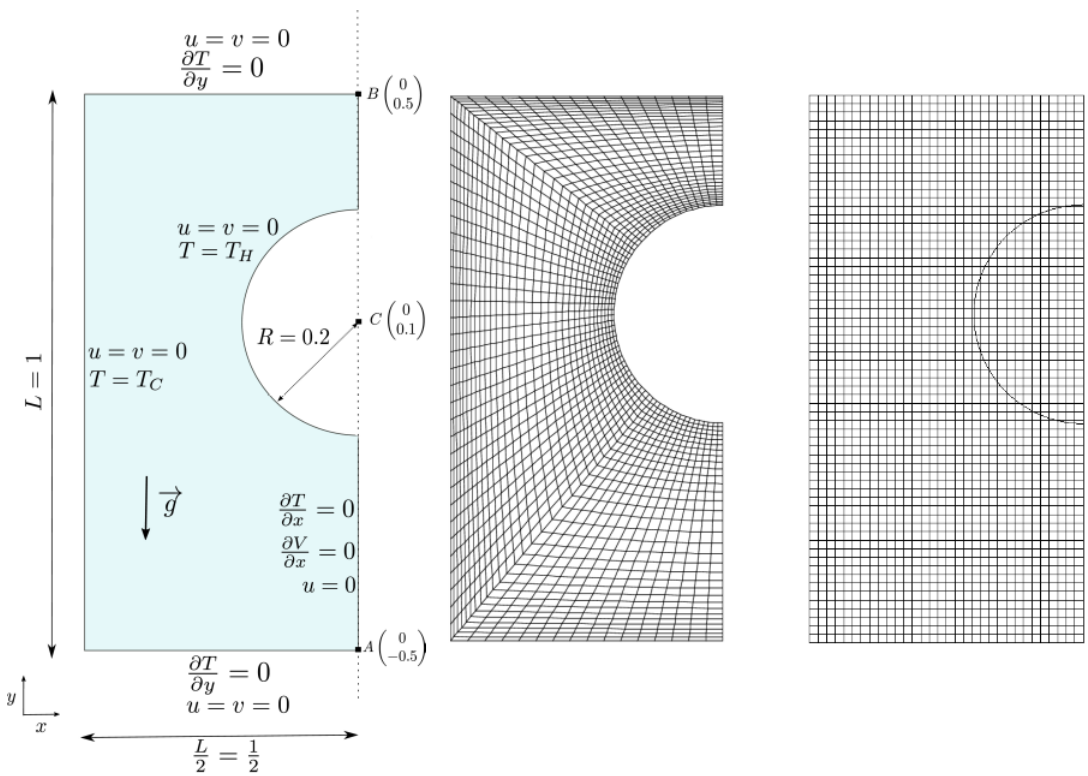

Figure 5: Sketch of the computational domain for the natural convection test-case (with $L = 1\,\text{m}$, $T_\text{H} = 1\,\text{K}, T_\text{C} = 0\,\text{K}$), curvilinear mesh L1 and Cartesian mesh M1 of size 32×64 used respectively for the FLUENT© and LS-STAG computations.

The second test-case concerns a fluid flow and heat transfer problem in complex geometries, for which Demirdžić *et al* [9] published accurate benchmark solutions. This test case concerns

natural convection of a heated cylinder placed eccentrically in an inclosure (test C4 in [9]), where the heat equation (1) is coupled with the Navier-Stokes equations in the Boussinesq approximation. The computational domain and boundary conditions are shown in Fig. 5 (left), with gravity acting in the downwards direction: since the solution possesses a vertical symmetry at $x = 0$, only half of the geometry is considered. As in [9], the dimensions and fluid properties are: enclosure length $L = 1\,\mathrm{m}$, density $\rho = 1\,\mathrm{kg}\cdot\mathrm{m}^{-3}$, gravity constant $g = 1\,\mathrm{m}\cdot\mathrm{s}^{-2}$, thermal expansion coefficient $\beta = 0.1\mathrm{K}^{-1}$, thermal diffusivity $\kappa = 10^{-4}\,\mathrm{m}^2\cdot\mathrm{s}^{-1}$, dynamic viscosity $\mu = 10^{-3}\,\mathrm{kg}\cdot\mathrm{m}^{-1}\cdot\mathrm{s}^{-1}$ and temperature difference $T_\mathrm{H} - T_\mathrm{C} = 1\,\mathrm{K}$, that set the Rayleigh and Prandtl number respectively as:

$$\mathrm{Ra} \equiv \frac{\beta\rho g L^3}{\mu\kappa}(T_\mathrm{H} - T_\mathrm{C}) = 10^6, \tag{15}$$

$$\mathrm{Pr} \equiv \frac{\mu}{\rho\kappa} = 10. \tag{16}$$

The purpose of these computations is to assess the ability of the LS-STAG/DCM method to compute accurately gradients at the immersed boundary, in the present case the average Nusselt number on the half-cylinder of length $|\Gamma^\mathrm{ib}|$:

$$\overline{\mathrm{Nu}} = \frac{1}{|\Gamma^\mathrm{ib}|\,(T_\mathrm{H} - T_\mathrm{C})/L}\int_{\Gamma^\mathrm{ib}} \nabla T\cdot\boldsymbol{n}\,\mathrm{d}S, \tag{17}$$

and the distribution of the local Nusselt number along the half cylinder, which is calculated at the center of the solid boundary of the cut-cells using formula (9).

The LS-STAG computations are performed on a series of refined meshes M1-M5 of square uniform Cartesian cells, where the coarsest mesh M1 represented in Fig. 5 (right) has 32×64 cells and the finer meshes are obtained by doubling the number of cells in each direction, such as the finest mesh M5 has 512×1024 cells. For comparison purpose, we have also computed solutions with the commercial CFD package FLUENT© (V15.07), on a series of curvilinear meshes L1-L5 (see Fig. 5 (middle)) of same number of cells, with cell clustering near the heated cylinder. The FLUENT© computations have been performed with the pressure-based solver with the steady SIMPLE algorithm, with second-order spatial discretizations: QUICK scheme for convection discretization, Green-Gauss node-based interpolation for gradient calculation, and body-force weighted pressure interpolation.

Fig. 6 shows the convergence of the average Nusselt number as the grid is refined. Irrespective of the gradient discretization scheme, all LS-STAG variants converge to the same grid-independent value at a similar rate. It is worthwhile to observe that on the 2 coarsest mesh, the FLUENT© computations are more accurate than LS-STAG. This is most certainly due to the high orthogonal quality of the curvilinear meshes near the circular cylinder.

Fig. 7 displays the distribution of the local Nusselt number along the immersed cylinder obtained for the LS-STAG and FLUENT© computations on the M4/L4 mesh, along with the reference results [9]. In order to quantify the influence of the cut-cells shape on the accuracy of the results, the LS-STAG profiles are colorized with the volume-change ratio of cut-cell $\Omega_{i,j}$:

$$RV_{i,j} = \min_{(k,l)\in N_\mathrm{b}} \frac{V_{i,j}}{V_{k,l}}, \tag{18}$$

where N_b refers to the neighboring cells that share a face with $\Omega_{i,j}$ (solid cells are excluded). For each cut-cell $RV_{i,j} \in]0,1]$ since at least one neighboring cell is totally fluid, and a value close to 0 indicates a cut-cell of small size compared to the Cartesian cells.

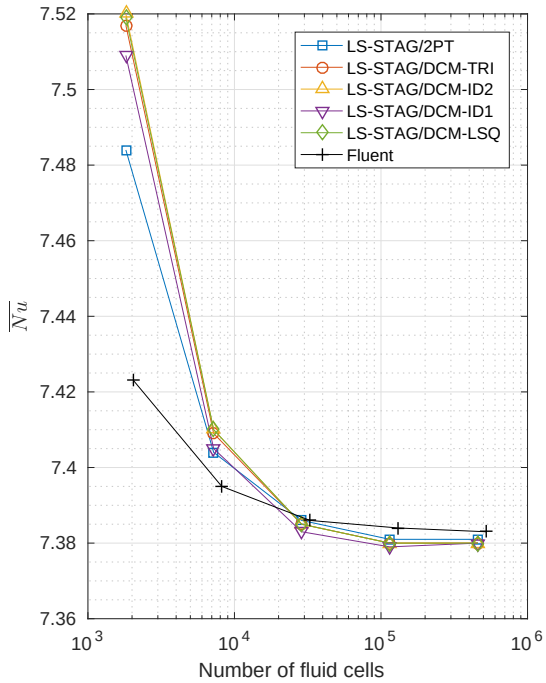

Figure 6: Convergence of the average Nusselt number with respect to the number of fluid cells.

First it can be observed that the LS-STAG method with 2-point gradients gives a highly oscillatory profile, confirming the inaccuracy of this approximation of diffusion. The use of the diamond scheme reduces the level of oscillations for the first-order method DCM-ID1 and for DCM-LSQ. For both methods, the oscillations are clearly located in cut-cells with small volume-change ratio $RV_{i,j}$, indicating inaccurate approximation in small cut-cells. In contrast, the profiles obtained with the other second-order methods (DCM-ID2 and TRI) are virtually free of any oscillations, and perfectly match the reference results.

As a conclusion, these computations show that accurate solutions can be achieved with the DCM discretization. Nonetheless, the accuracy of the DCM approximation is highly dependent on the interpolation scheme that is used, otherwise inaccurate results are encountered in the smaller cut-cells. These results show that the second-order methods DCM-ID2 and DCM-TRI stand as the best discretization methods, yielding indistinguishable results on the local Nusselt distribution. Owing to the simplicity of the triangular interpolation compared to the inverse distance interpolation of second order, the DCM-TRI variant is selected for implementation in the 3D-extruded code.

4.3 ISOTHERMAL FLOWS IN 3D EXTRUDED GEOMETRIES

The last test-case concerns the unsteady laminar flow past a circular cylinder in a confined square duct, for which accurate benchmark results with unstructured solvers are reported in [3]. The flow unsteadiness is induced by a time-varying inflow boundary condition, yielding a time-periodic Reynolds number $\mathrm{Re} \in [0, 100]$. This benchmark flow has previously been performed in [16] with the LS-STAG method for 3D extruded geometries, using 2-point formulas for the diffusive fluxes of the Navier-Stokes equations.

A sketch of the velocity staggering in extruded cut-cells is shown in Fig. 8 : it is clearly seen that diamond cell discretization is needed for enhancing the discretization of for the off-diagonal

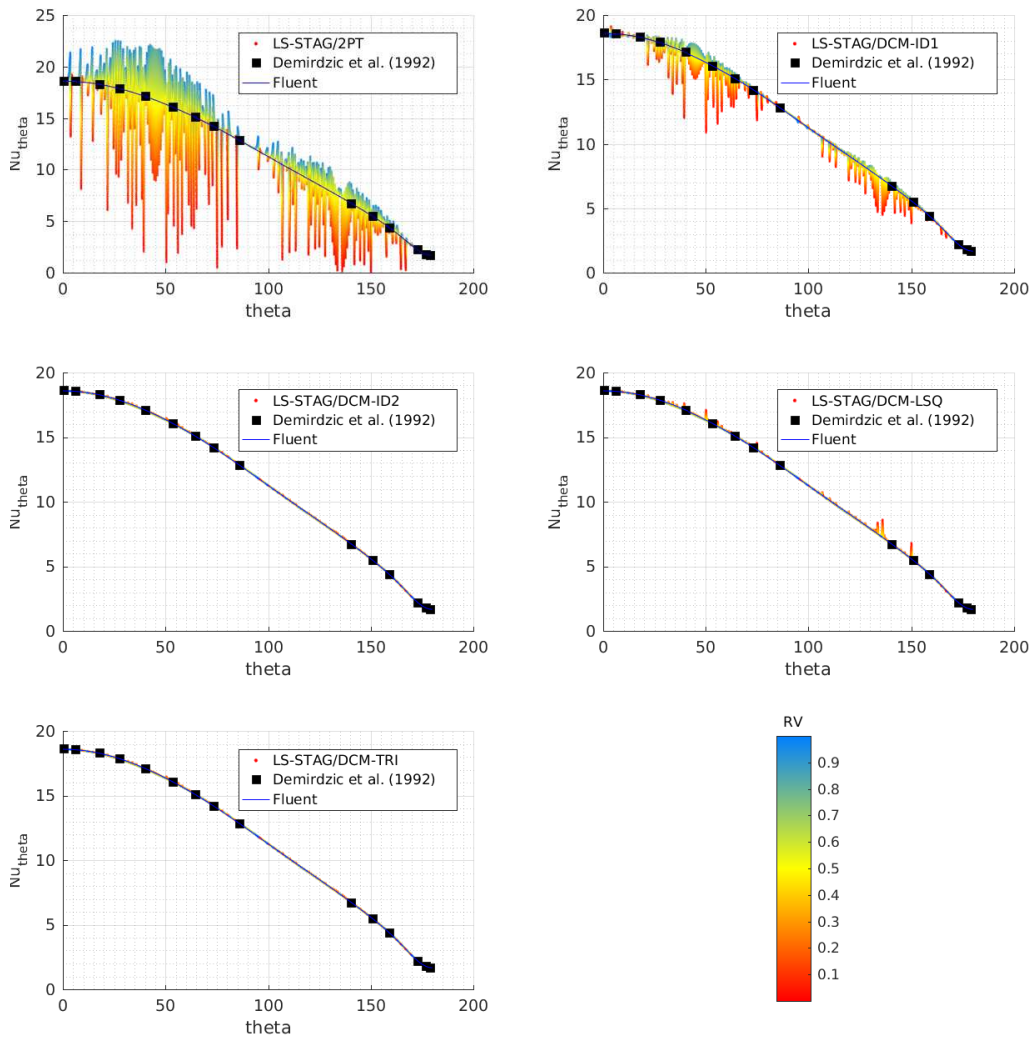

Figure 7: Profiles of the Nusselt number along the half cylinder for the various discretization of diffusion in the LS-STAG method, and comparison with the FLUENT© computation and reference results of Demirdžić *et al* [9]. The profiles are colorized with respect to the volume-change ratio $RV_{i,j}$ (18) of the cut-cells.

components $\partial w/\partial x$ and $\partial w/\partial y$ of the diffusive fluxes of the Navier-Stokes equations. Due to the extrusion in the z direction, the other velocity gradients are accurately computed with 2-point formulas [16]. Note also that due to the staggering of the discrete velocities, the discretization of $\partial w/\partial x$ and $\partial w/\partial y$ involves co-planar unknowns (on planes of constant index k), such that only the 2D DCM formula (8) is needed.

Table 1 compares the results obtained on a series of refined meshes (500,000 to 3M cells) with an unstructured CFD solver (FLUENT©), the LS-STAG solver with 2-point discretization [16] and the present DCM discretization. We refer to [16] for a complete description of the geometry, boundary conditions, meshes and discretizations used by both solvers. As observed in [16], the 2-point discretization of the LS-STAG solver gives only monotonic convergence for the lift quantities, while the drag quantities fails to convergence. In contrast, the DCM discretization show a dramatic increase in the accuracy of the LS-STAG discretization, yielding a smooth convergence for all force coefficients, with an accuracy comparable to the FLUENT© solver on all 3 grids.

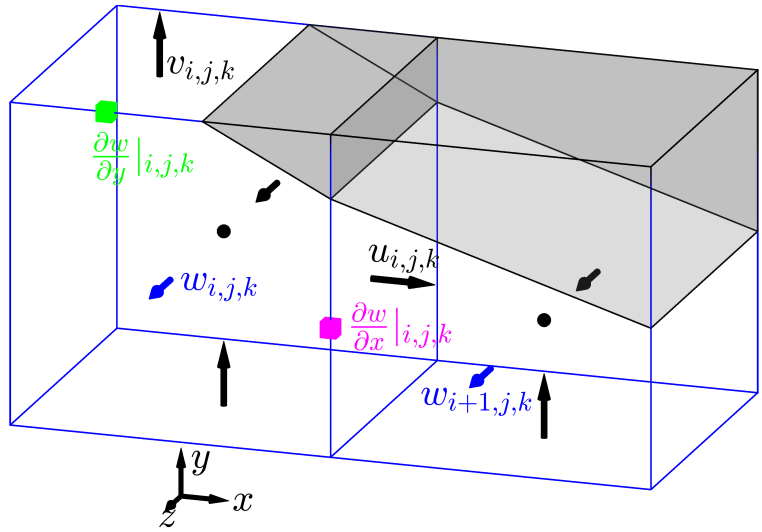

Figure 8: Staggered mesh for 3D extruded geometries in the z direction, and location of the off-diagonal components of ∇w.

Solver	Mesh	% Error			% L_2 Error	
		$C_{\mathrm{D}_{\max}}$	$C_{\mathrm{L}_{\max}}$	$C_{\mathrm{L}_{\min}}$	C_{D}	C_{L}
LS-STAG/2PT [16]	M2	3.72	11.05	36.61	3.74	48.06
	M3	2.23	2.96	18.07	2.42	16.94
	M4	2.96	0.49	10.02	3.05	10.22
LS-STAG/DCM-TRI	M2	1.31	6.99	34.00	1.48	41.91
	M3	0.63	0.72	10.78	0.65	10.06
	M4	0.24	0.06	5.42	0.22	5.46
FLUENT©	L2	2.35	4.40	16.29	2.11	17.03
	L3	0.96	2.16	10.96	0.81	10.01
	L4	0.25	0.08	2.89	0.21	2.67

Table 1: Relative errors (in percent) of the force coefficients with respect to the reference solution [2] and L_2 errors for one-half period of the simulation ($t \in [0, 8\mathrm{s}]$).

5 Concluding remarks

This paper has presented a progress report on the application of the diamond cell method (DCM) for improving the accuracy of diffusion in cut-cell methods. The DCM is employed for cell-centred data (temperature in 2D) and face-centred data (Navier-Stokes equations in 3D-extruded geometries), which illustrates the versatility of the approach, that can be applied to other cut-cell methods than the LS-STAG method. One of the major findings is the dependence of the accuracy on the reconstruction scheme used for data interpolation. The use of barycentric interpolation allowed us to obtain local accuracy in cut-cells comparable to the one reached in Cartesian cells, irrespective of the size and shape of the cut-cell. Progress is on the way to employ DCM for full 3D computations with the LS-STAG method.

References

[1] I. Amidror. Scattered data interpolation methods for electronic imaging systems: A survey. *Journal of electronic imaging*, 11:157–176, 2002.

[2] E. Bayraktar. Private communication, 2016.

[3] E. Bayraktar, O. Mierka, and S. Turek. Benchmark computations of 3D laminar flow around a cylinder with CFX, OpenFOAM and FeatFlow. *International Journal of Computational Science and Engineering*, 7(3):253–266, 2012.

[4] E. Bertolazzi and G. Manzini. On vertex reconstructions for cell-centered finite volume approximations of 2D anisotropic diffusion problems. *Mathematical Models and Methods in Applied Sciences*, 17:1–32, 2007.

[5] O. Botella, M. Ait-Messaoud, A. Pertat, C. Rigal, and Y. Cheny. The LS-STAG immersed boundary method for non-Newtonian flows in irregular geometries : Flow of shear-thinning liquids between eccentric rotating cylinders. *Theoretical and Computational Fluid Dynamics*, **29**, 93-110, 2015.

[6] O. Botella, Y. Cheny, F. Nikfarjam, and M. Stoica. Application of the LS-STAG immersed boundary/cut-cell method to viscoelastic flow computations. *Communications in Computational Physics*, 20:870–901, 2016.

[7] Y. Cheny and O. Botella. The LS-STAG method : A new immersed boundary / level-set method for the computation of incompressible viscous flows in complex moving geometries with good conservation properties. *J. Comput. Phys.*, **229**, 1043-1076, 2010.

[8] Y. Coudière, J.-P. Vila, and P. Villedieu. Convergence rate of a finite volume scheme for a two dimensional convection-diffusion problem. *ESAIM: Mathematical Modelling and Numerical Analysis*, 33:493–516, 1999.

[9] I Demirdžić, Ž Lilek, and M Perić. Fluid flow and heat transfer test problems for non-orthogonal grids: Bench-mark solutions. *International Journal for Numerical Methods in Fluids*, 15(3): 329–354, 1992.

[10] Robert Falgout, Andy Cleary, Jim Jones, Edmond Chow, Van Henson, Chuck Baldwin, Peter Brown, Panayot Vassilevski, and Ulrike Meier Yang. Center for Applied Scientific Computing (CASC). URL `http://acts.nersc.gov/hypre/`. Lawrence Livermore National Laboratory.

[11] F. Jacon and D.D. Knight. A Navier-Stokes algorithm for turbulent flows using an unstructured grid and flux difference splitting. AIAA paper No. 94-2292, 1994.

[12] L. Martinelli. *Calculations of viscous flows with a multigrid method*. PhD thesis, Princeton University, 1987.

[13] L. Martinelli and A. Jameson. Validation of a multigrid method for the Reynolds averaged equations. AIAA paper No. 88-0414, 1988.

[14] S.R. Mathur and J.Y. Murthy. A pressure-based method for unstructured meshes. *Numerical Heat Transfer*, 31:195–215, 1997.

[15] F. Moukalled, L. Mangani, and M. Darwish. *The Finite Volume Method in Computational Fluid Dynamics*. Springer, 2016.

[16] F. Nikfarjam, Y. Cheny, and O. Botella. The LS-STAG immersed boundary/cut-cell method for non-Newtonian flows in 3D extruded geometries. *Computer Physics Communications*, 226:67–80, 2018.

[17] P. van der Plas, H.J.L. van der Heiden, A.E.P. Veldman, R. Luppes, and R.W.C.P. Verstappen. Efficiently simulating viscous flow effects by means of regularization turbulence modeling and local grid refinement. Seventh International Conference on Computational Fluid Dynamics (ICCFD7), Hawaii, paper ICCFD-2503, 2012.