An efficient implementation of a 3D CeVeFE DDFV scheme on Cartesian grids and an application in image processing

Niklas Hartung, Florence Hubert

Abstract In this work we describe the implementation of a 3D Center-Vertex-Face/Edge Discrete Duality Finite Volume (CeVeFE DDFV) scheme using only the degrees of freedom (DOF) disposed on a Cartesian grid. These DOF are organised in a three-mesh structure proper to the CeVeFE DDFV setting. Reposing on a diamond structure, the approach presented here greatly simplifies the implementation, also in the case of grids topologically equivalent to the uniform Cartesian one.
The numerical scheme is then applied to a problem in image processing, where uniform Cartesian structure of the DOF is naturally imposed by the pixel/voxel structure. A semi-implicit DDFV scheme is used for solving a nonlinear advection-diffusion equation, the subjective surfaces equation, in order to reconstruct the volume of a tumour from noisy 3D SPECT images with signal intensity on the tumour boundary. The matrix of the linear system has a band structure and the method is fast and able to successfully reconstruct the tumour volume.

1 Introduction

Discrete Duality Finite Volume (DDFV) schemes, introduced in 2D for the Laplace problem by Hermeline [10], are a possible discretisation strategy applying to very general meshes and a large variety of PDE [6, 3]. A dual or "node" mesh is used in this framework and gradients are defined on a structure called the diamond mesh. A main feature of the DDFV approach is that discrete gradients and divergence operators are defined in a way that a discrete Green formula holds, called "discrete

Niklas Hartung
Aix Marseille Université, CNRS, Centrale Marseille, I2M, UMR 7373, Marseille, FRANCE, e-mail: niklas.hartung@univ-amu.fr

Florence Hubert
Aix Marseille Université, CNRS, Centrale Marseille, I2M, UMR 7373, Marseille, FRANCE, e-mail: florence.hubert@univ-amu.fr

duality".
In 3D, several methods have been inspired by the 2D DDFV methodology. The Center-Vertex (CeVe) DDFV schemes have a dual mesh with centers at the vertices of the primary mesh [11, 2, 1]. A different method, called Center-Vertex-Face-Edge (CeVeFE) DDFV scheme, features a third mesh with unknowns at the faces and edges of the primary mesh [5]; this will be our framework.
The DDFV framework can also be used for the discretisation of PDE appearing in image processing, such as level set methods, which are used for a broad spectrum of applications [15]. A curvature-driven level set equation called the subjective surfaces equation has been introduced by Sarti *et al.* [14] as a tool for the completion of missing boundaries. Along with subsequent extensions, it has been successfully applied to image processing problems [13, 12].
The nonlinearity and the non-divergent form of the curvature-driven level set equation makes particular space discretisation techniques necessary. Several Finite Volume methods have been proposed along with numerical analysis [13, 7]. Recently, stability and convergence of a semi-implicit 2D DDFV scheme was proven [9], but additional vertex unknowns were introduced.
In this work we will detail the efficient implementation of the 3D CeVeFE DDFV method for Cartesian grids. The method will then be used for discretising the subjective surfaces equation on a uniform Cartesian grid. As an application, tumour volume is reconstructed from a 3D SPECT image visualising proliferating cells, which are located at the tumour boundary.

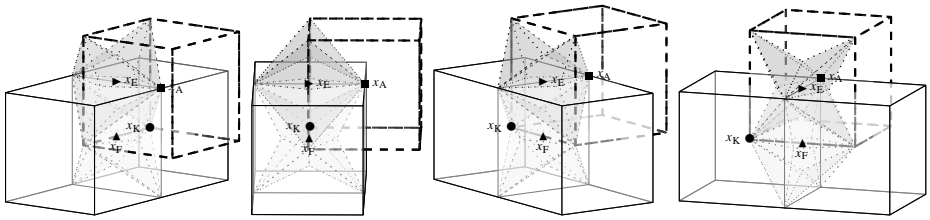
2 The 3D CeVeFE DDFV scheme with degrees of freedom on Cartesian grids

Construction of the meshes

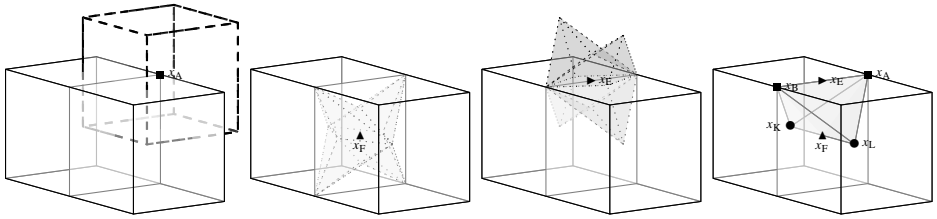
In the 3D CeVeFE DDFV scheme, three different decompositions of the computational domain Ω are used, called the primary mesh $\mathscr{M}$, the dual or node mesh $\mathscr{N}$ associated with the vertices of the primary mesh and the tertiary or "face-edge" mesh $\mathscr{FE}$ associated with the faces and edges of the primary mesh. For the detailed construction of $\mathscr{N}$ and $\mathscr{FE}$ from a general primary mesh $\mathscr{M}$, we refer to [5].

There is a canonical way to construct these three meshes if we want each gridpoint of a Cartesian structure $\mathscr{T}$ to be associated to exactly one cell of either $\mathscr{M}$, $\mathscr{N}$ or $\mathscr{FE}$. Referring to each point by its three-dimensional index (i,j,k), $1 \leqslant i \leqslant N_x \in \mathbb{N}$, $1 \leqslant j \leqslant N_y \in \mathbb{N}$, $1 \leqslant k \leqslant N_z \in \mathbb{N}$, we have the following bijections (see Fig. 1):

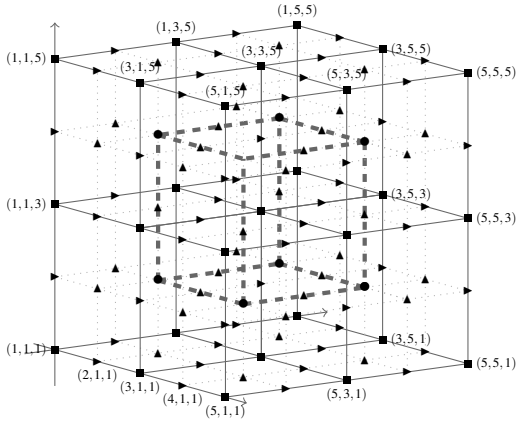
$$\{(i,j,k) \text{ with } i,j,k \text{ even }\} \mapsto \mathscr{M}, \qquad \{(i,j,k) \text{ with } i,j,k \text{ odd }\} \mapsto \mathscr{N},$$
$$\{(i,j,k) \text{ with } ijk \equiv 2 \mod 4\} \mapsto \mathscr{E}, \qquad \{(i,j,k) \text{ with } ijk \equiv 4 \mod 8\} \mapsto \mathscr{F}.$$



(a) The 3 meshes.



(b) Two primary mesh cells together with other cells appearing in the mesh construction. Extreme left: node cell, middle left: face cell, middle right: edge cell, extreme right: diamond cell.



(c) Mapping of a uniform Cartesian grid onto $\mathscr{M}$, $\mathscr{N}$ and $\mathscr{EF}$.

Fig. 1: 3D mesh views for uniform Cartesian grids. Primary mesh centers are marked by circles, nodes by squares, faces by upright and edges by sideward triangles.

We denote the control volumes of the primary mesh $\mathscr{M}$ by K or L (with centers x_K or x_L), vertices by x_A or x_B, edges by E and faces by F. Control volumes of the dual mesh $\mathscr{N}$ will be called A or B. To simplify notations, control volumes of the tertiary mesh $\mathscr{FE}$ will also be called E and F as it will be clear from the context whether the face/edge or the control volume is meant. We also define the center of gravity x_F of a face F and the midpoint x_E of an edge E.

Discrete gradients are defined on a fourth decomposition of Ω called the diamond

mesh $\mathscr{D}$. Each diamond cell $\mathrm{D} \in \mathscr{D}$ corresponds to a face-edge couple (F, E) with $\mathrm{E} \in \partial\mathrm{F}$. In our decomposition of the Cartesian grid $\mathscr{T}$, a diamond cell D will be defined by listing the indices of six points of $\mathscr{T}$: a face center x_F, the midpoint x_E of an edge $\mathrm{E} \in \partial\mathrm{F}$, the two vertices $x_\mathrm{A}, x_\mathrm{B} \in \partial\mathrm{E}$ and the two adjacent primary cell centers x_K and x_L . The diamond D is then given by $\mathrm{D} := \mathrm{hull}(x_\mathrm{A}, x_\mathrm{B}, x_\mathrm{L}, x_\mathrm{K})$. The diamond mesh can be subdevided into twelve classes of diamonds, which are listed in Table 1. This classification permits an efficient construction of the diamond mesh. The volume of D is given by $|\mathrm{D}| = \frac{\det(x_\mathrm{B}-x_\mathrm{A}, x_\mathrm{F}-x_\mathrm{E}, x_\mathrm{L}-x_\mathrm{K})}{6} > 0$. For $\mathrm{F} \in \partial\Omega$, the indices that exceed the Cartesian grid are projected onto $\mathscr{T}$, creating degenerate diamonds.

Noting $x_\mathrm{D} = \frac{1}{2}(x_\mathrm{E} + x_\mathrm{F})$, a diamond D can be decomposed into eight tetrahedra $\mathrm{D}_{\mathrm{AKE}}, \mathrm{D}_{\mathrm{ALE}}, \mathrm{D}_{\mathrm{BKE}}, \mathrm{D}_{\mathrm{BLE}}, \mathrm{D}_{\mathrm{AKF}}, \mathrm{D}_{\mathrm{ALF}}, \mathrm{D}_{\mathrm{BKF}}, \mathrm{D}_{\mathrm{BLF}}$ defined by

$$\mathrm{hull}\left(\begin{pmatrix} x_\mathrm{K} \\ x_\mathrm{L} \end{pmatrix}, \begin{pmatrix} x_\mathrm{A} \\ x_\mathrm{B} \end{pmatrix}, \begin{pmatrix} x_\mathrm{E} \\ x_\mathrm{F} \end{pmatrix}, x_\mathrm{D}\right). \tag{1}$$

This decomposition permits to define the control volumes C of any of the three meshes as the union of all tetrahedra containing the vertex x_C, e.g.

$$\mathrm{K} = \bigcup_{\mathrm{D} \in \mathscr{D} : x_\mathrm{K} \in \mathrm{D}} (\mathrm{D}_{\mathrm{AKE}} \cup \mathrm{D}_{\mathrm{BKE}} \cup \mathrm{D}_{\mathrm{AKF}} \cup \mathrm{D}_{\mathrm{BKF}}).$$

With this definition, some boundary volumes, depending on the parity of N_x, N_y, N_z, degenerate automatically. The DDFV meshes are coarser than the canonical mesh associated to the Cartesian grid.

Type	x_F	x_E	x_K	x_L	x_A	x_B
$(x,1)$	(i,j,k) with i odd	$(i,j+1,k)$	$(i-1,j,k)$	$(i+1,j,k)$	$(i,j+1,k-1)$	$(i,j+1,k+1)$
$(x,2)$	(i,j,k) with i odd	$(i,j,k-1)$	$(i-1,j,k)$	$(i+1,j,k)$	$(i,j-1,k-1)$	$(i,j+1,k-1)$
$(x,3)$	(i,j,k) with i odd	$(i,j-1,k)$	$(i-1,j,k)$	$(i+1,j,k)$	$(i,j-1,k+1)$	$(i,j-1,k-1)$
$(x,4)$	(i,j,k) with i odd	$(i,j,k+1)$	$(i-1,j,k)$	$(i+1,j,k)$	$(i,j+1,k+1)$	$(i,j-1,k+1)$
$(y,1)$	(i,j,k) with j odd	$(i+1,j,k)$	$(i,j-1,k)$	$(i,j+1,k)$	$(i+1,j,k+1)$	$(i+1,j,k-1)$
$(y,2)$	(i,j,k) with j odd	$(i,j,k+1)$	$(i,j-1,k)$	$(i,j+1,k)$	$(i-1,j,k+1)$	$(i+1,j,k+1)$
$(y,3)$	(i,j,k) with j odd	$(i-1,j,k)$	$(i,j-1,k)$	$(i,j+1,k)$	$(i-1,j,k-1)$	$(i-1,j,k+1)$
$(y,4)$	(i,j,k) with j odd	$(i,j,k-1)$	$(i,j-1,k)$	$(i,j+1,k)$	$(i+1,j,k-1)$	$(i-1,j,k-1)$
$(z,1)$	(i,j,k) with k odd	$(i,j+1,k)$	$(i,j,k-1)$	$(i,j,k+1)$	$(i+1,j+1,k)$	$(i-1,j+1,k)$
$(z,2)$	(i,j,k) with k odd	$(i+1,j,k)$	$(i,j,k-1)$	$(i,j,k+1)$	$(i+1,j-1,k)$	$(i+1,j+1,k)$
$(z,3)$	(i,j,k) with k odd	$(i,j-1,k)$	$(i,j,k-1)$	$(i,j,k+1)$	$(i-1,j-1,k)$	$(i+1,j-1,k)$
$(z,4)$	(i,j,k) with k odd	$(i-1,j,k)$	$(i,j,k-1)$	$(i,j,k+1)$	$(i-1,j+1,k)$	$(i-1,j-1,k)$

Table 1: *Construction of the diamond mesh. Diamond types are defined via their face/edge representation, noting the orientation of faces (orthogonal to x, y or z axis) and enumerating the four edges of the face.*

Discrete gradient and discrete divergence operators

For $u \in \mathbb{R}^{|\mathscr{T}|}$ and a diamond $\mathrm{D} \in \mathscr{D}$, set u_C as a notation for $u(x_\mathrm{C})$ where x_C is one of the six points defining D. The discrete gradient $\nabla^d : \mathbb{R}^{|\mathscr{T}|} \mapsto \mathbb{R}^{|\mathscr{D}|}$ is given by

$$\left(\nabla^d(u_{\mathscr{T}})\right)_{\mathrm{D}} = \frac{1}{3|\mathrm{D}|}\left((u_{\mathrm{L}}-u_{\mathrm{K}})\overrightarrow{N_{\mathrm{KL}}}+(u_{\mathrm{F}}-u_{\mathrm{E}})\overrightarrow{N_{\mathrm{EF}}}+(u_{\mathrm{B}}-u_{\mathrm{A}})\overrightarrow{N_{\mathrm{AB}}}\right)$$

for any $\mathrm{D}\in\mathscr{D}$ and with the vectors

$$\overrightarrow{N_{\mathrm{KL}}} = \frac{(x_{\mathrm{B}}-x_{\mathrm{A}})\times(x_{\mathrm{F}}-x_{\mathrm{E}})}{2}, \overrightarrow{N_{\mathrm{AB}}} = \frac{(x_{\mathrm{F}}-x_{\mathrm{E}})\times(x_{\mathrm{L}}-x_{\mathrm{K}})}{2}, \overrightarrow{N_{\mathrm{EF}}} = \frac{(x_{\mathrm{L}}-x_{\mathrm{K}})\times(x_{\mathrm{B}}-x_{\mathrm{A}})}{2}.$$

These definitions and the structure of Table 1 ensure that $\overrightarrow{N_{\mathrm{XY}}}$ points from X to Y ($(\mathrm{X},\mathrm{Y})\in\{(\mathrm{K},\mathrm{L}),(\mathrm{A},\mathrm{B}),(\mathrm{E},\mathrm{F})\}$). Note that although there are more edges than faces, u_{E} and F contribute to the gradient similarly. The discrete divergence $\mathrm{div}^d : \mathbb{R}^{|\mathscr{D}|}\mapsto\mathbb{R}^{|\mathscr{T}|}$ is defined by

$$\left(\mathrm{div}^d(\xi)\right)_{\mathrm{C}} = \frac{1}{|\mathrm{C}|}\sum_{\mathrm{D}:\mathrm{D}\cap\mathrm{C}\neq\emptyset}\xi_{\mathrm{D}}\cdot\overrightarrow{N_{\mathrm{C}}}, \tag{2}$$

with $\overrightarrow{N_{\mathrm{C}}}=\overrightarrow{N_{\mathrm{KL}}}$ if $\mathrm{C}=\mathrm{K}\in\mathscr{M}$, $\overrightarrow{N_{\mathrm{C}}}=\overrightarrow{N_{\mathrm{AB}}}$ if $\mathrm{C}=\mathrm{A}\in\mathscr{N}$, $\overrightarrow{N_{\mathrm{C}}}=\overrightarrow{N_{\mathrm{EF}}}$ if $\mathrm{C}=\mathrm{E}\in\mathscr{E}$ and $\overrightarrow{N_{\mathrm{C}}}=-\overrightarrow{N_{\mathrm{EF}}}$ if $\mathrm{C}=\mathrm{F}\in\mathscr{F}$.

These expressions simplify on uniform Cartesian grids (e.g. $|\mathrm{D}|=\frac{2}{3}h^3$ for interior diamonds, with voxel length h), but no acceleration is obtained by implementing these simplifications, which is why we only present the general case.

3 An application to the subjective surfaces equation

In image processing, uniform Cartesian grids arise naturally because image information is given on pixels or voxels. We will illustrate the performance of the numerical scheme taking an application from this field. The subjective surfaces equation reads

$$\partial_t u + |\nabla u|\mathrm{div}\left(g(|\nabla I|)\frac{\nabla u}{|\nabla u|}\right) = 0 \tag{3}$$

with $g(x)=\frac{1}{1+kx^2}$, $k>0$, I the (given) image intensity and Dirichlet boundary conditions. Numerically, the solution u of Eq. (3) evolves to a piecewise constant function delimited by regions where $|\nabla I|$ is large. The support of the initial condition u_0 is chosen in the region of which the boundary should be determined.

Discretisation of the subjective surfaces equation with CeVeFE DDFV

The meshing described in the previous section has the advantage that the unknowns correspond to the image voxels; we stress that no additional degrees of freedom, nor interpolated values, are used. Following [4, 9], we choose a semi-implicit time discretisation of a regularised form of Eq. (3), which yields a linear scheme:

$$\frac{u^{n+1}-u^n}{\Delta t} + (|\nabla^d u^n|+\varepsilon)\mathrm{div}^d\left(g(|\nabla^d I|)\frac{\nabla^d u^{n+1}}{|\nabla^d u^n|+\varepsilon}\right) = 0, \tag{4}$$

with $\varepsilon>0$. A symmetric scheme is obtained by multiplying Eq. (4) by the diagonal matrix Λ_n with entries $((|\nabla^d u^n|_{\mathrm{C}}+\varepsilon)/|\mathrm{C}|)^{-1}$:

$$\Lambda_n u^{n+1} + \Delta t |\mathrm{C}| \mathrm{div}^d \left(g(|\nabla^d I|) \frac{\nabla^d u^{n+1}}{|\nabla^d u^n| + \varepsilon} \right) = \Lambda_n u^n. \tag{5}$$

Observe that the matrix M with $Mu = |\mathrm{C}| \mathrm{div}^d \left(g(|\nabla^d I|) \frac{\nabla^d u}{|\nabla^d u^n| + \varepsilon} \right)$ is computed in the following way. Let $\lambda_\mathrm{D} := \frac{g(|\nabla^d I|_\mathrm{D})}{|\nabla^d u^n|_\mathrm{D} + \varepsilon}$, which is known from the previous iteration, and note that in order to calculate this quantity, an approximation of the norm of the full gradient is needed (which basic Finite Difference schemes and some Finite Volume schemes do not yield). Due to the uniform Cartesian grid structure,

$$\overrightarrow{N_{\mathrm{KL}}} \cdot \overrightarrow{N_{\mathrm{EF}}} = \overrightarrow{N_{\mathrm{KL}}} \cdot \overrightarrow{N_{\mathrm{AB}}} = \overrightarrow{N_{\mathrm{EF}}} \cdot \overrightarrow{N_{\mathrm{AB}}} = 0, \tag{6}$$

yielding

$$\begin{aligned} (Mu)_\mathrm{K} &= \sum_{\mathrm{D} \in \mathrm{D_K}} \tfrac{\lambda_\mathrm{D}}{3|\mathrm{D}|} (u_\mathrm{L} - u_\mathrm{K}) ||\overrightarrow{N_{\mathrm{KL}}}||^2, \quad (Mu)_\mathrm{A} = \sum_{\mathrm{D} \in \mathrm{D_A}} \tfrac{\lambda_\mathrm{D}}{3|\mathrm{D}|} (u_\mathrm{B} - u_\mathrm{A}) ||\overrightarrow{N_{\mathrm{AB}}}||^2, \\ (Mu)_\mathrm{E} &= \sum_{\mathrm{D} \in \mathrm{D_E}} \tfrac{\lambda_\mathrm{D}}{3|\mathrm{D}|} (u_\mathrm{F} - u_\mathrm{E}) ||\overrightarrow{N_{\mathrm{EF}}}||^2, \quad (Mu)_\mathrm{F} = \sum_{\mathrm{D} \in \mathrm{D_F}} \tfrac{\lambda_\mathrm{D}}{3|\mathrm{D}|} (u_\mathrm{E} - u_\mathrm{F}) ||\overrightarrow{N_{\mathrm{EF}}}||^2. \end{aligned} \tag{7}$$

Therefore, the meshes $\mathscr{M}, \mathscr{N}$ and $\mathscr{FE}$ are not coupled in the resolution of (5), only to the previous time step by $|\nabla u^n|$, accelerating the numerical resolution.

Iterating over diamonds

We stress that the quantities needed for the resolution of Eq. (5) can be computed only using the diamond structure. The following information has to be stored for each diamond: the point references explained in Table 1, the volumes of the diamond of its eight constituting tetrahedra (see (1)), and the vectors $\overrightarrow{N_{\mathrm{KL}}}$, $\overrightarrow{N_{\mathrm{AB}}}$ and $\overrightarrow{N_{\mathrm{EF}}}$.

The matrix M can then be assembled efficiently by iterating over $\mathrm{D} \in \mathscr{D}$ and by computing for each diamond the contributions at the indices corresponding to x_K, x_L, x_A, x_B, x_E and x_F via the formulas (7). Similarly, the measure of the control volumes can be assembled from the eight tetrahedra constituting the diamonds. These procedures, including the construction of the diamond mesh, are easily vectorised.

DDFV solution

DDFV solutions, defined on overlapping meshes, naturally give rise to averaged discrete solutions [3, 2]. In our case, based on the solution u of (5) at the final time T, on each mesh ($\mathscr{M}, \mathscr{N}$ and $\mathscr{FE}$) a cell-wise piecewise constant function $(u_\mathscr{M}, u_\mathscr{N}, u_\mathscr{FE})$ is defined. The DDFV solution is

$$u_{DDFV} = \frac{1}{3} \left(u_\mathscr{M} + u_\mathscr{N} + u_\mathscr{FE} \right),$$

which is constant on each tetrahedron constituting the diamond cells. In order to visualize the DDFV solution on the Cartesian grid, it is projected on the cells of $\mathscr{T}$:

$$u_{DDFV}^{cart} = \left(\frac{1}{|\mathrm{C}|} \int_\mathrm{C} u_{DDFV} \right)_{\mathrm{C} \in \mathscr{T}}.$$

This averaging is the price we pay for avoiding additional unknowns (as compared to [9], in 2D); indeed, in 3D it is crucial to reduce the number of degrees of freedom. It is important to note that the use of u^{cart}_{DDFV} is generally necessary and cannot be replaced by the evaluation of u. Indeed, the weak coupling of the three meshes due to the semi-implicit time discretisation allows u to contain local checkerboard structures caused by noise whereas u_{DDFV} is smooth.

Numerical results

The numerical scheme is illustrated on 3D SPECT images visualising proliferating tumour cells. These cells are mainly localised on the tumour boundary but do not cover the entire surface, notably due to physical constraints such as bones. We want to obtain the volume and shape of the tumour based on these images. In practice, tumour diameters are often measured manually and volume is approximated with an ellipsoid formula. The numerical method described above permits to obtain a less heuristical estimation of the volume, also indicating the shape. Voxels of the $N_x \times N_y \times N_z$-image are numbered in a classical way by $N(i,j,k) = i + N_x \cdot (j-1) + N_y \cdot N_x \cdot (k-1)$, such that $\Lambda_n + \Delta t M$ is a band matrix. Figure 2 shows the three different 2D cuts of the original image and the reconstructed tumour volume.

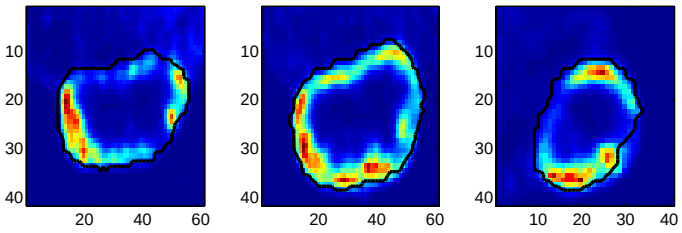


Fig. 2: *2D cuts of a 3D SPECT image (through then center of the tumour and parallel to the x, y and z axes, respectively) showing the density of proliferating tumour cells, which are localised at the boundary. The tumour reconstruction is marked by the black line.*

4 Conclusion

We have presented here the implementation of a 3D CeVeFE DDFV scheme using a Cartesian structure without introducing artificial unknowns, which is an important property in view of the high computational complexity in 3D. It comes at the cost of a mild smoothing by projecting the discrete solution on underlying voxels.

The implementation presented here finds an application in image processing, where uniform Cartesian grids naturally arise. The fact that DDFV schemes can

be used on degenerate meshes makes the implementation relevant for non-uniform Cartesian grids, for example the highly deformed Kershaw meshes appearing in porous media [8]. Similar band matrix profiles can be obtained in these cases.

We have successfully used a 3D DDFV discretisation of the subjective surfaces equation for the reconstruction of the tumour shape on an exemplary SPECT image. A subsequent step would be to test the performance of an automatised version on a large number of images and to compare it to the ellipsoid formula.

It should also be stressed that the DDFV framework is one out of many possible discretisation strategies, each with their advantages and shortcomings. It is hoped that this work permits an easy access to the DDFV approach.

Acknowledgements The authors would like to thank Boris Andreianov for fruitful discussions. The authors were partially supported by the Agence Nationale de la Recherche under grant ANR-09-BLAN-0217-01 and by the Cancéropôle PACA.

References

1. Andreianov, B., Bendahmane, M., Hubert, F.: On 3D DDFV discretization of gradient and divergence operators. Part II. Comput Method Appl M **13**(4), 369–410 (2013)
2. Andreianov, B., Bendahmane, M., Hubert, F., Krell, S.: On 3D DDFV discretization of gradient and divergence operators. Part I. IMA J Numer Anal **32**(4), 1574–603 (2012)
3. Andreianov, B., Boyer, F., Hubert, F.: Discrete duality finite volume schemes for Leray-Lions type elliptic problems on general 2D meshes. Numer Meth Part D E **23**(1), 145–95 (2007)
4. Corsaro, S., Mikula, K., Sarti, A., Sgallari, F.: Semi-implicit covolume method in 3D image segmentation. J Sci Comput **28**(6), 2248–65 (2006)
5. Coudière, Y., Hubert, F.: A 3D discrete duality finite volume method for nonlinear elliptic equations. SIAM J Sci Comp **33**(4), 1739–64 (2011)
6. Domelevo, K., Omnes, P.: A finite volume method for the Laplace equation on almost arbitrary two-dimensional grids. Math Model Numer Anal **39**(6), 1203–49 (2005)
7. Eymard, R., Handlovičová, A., Mikula, K.: Study of a finite volume scheme for the regularized mean curvature flow level set equation. IMA J Numer Anal **31**(3), 813–46 (2011)
8. Eymard, R., Henry, G., Herbin, R., Hubert, F., Kl
9. Handlovičová, A., Kotorová, D.: Numerical analysis of a semi-implicit DDFV scheme for the regularized curvature driven level set equation in 2D. Kybernetika **49**, 829–54 (2013)
10. Hermeline, F.: A finite volume method for the approximation of diffusion operators on distorted meshes. J Comput Phys **160**(2), 481–99 (2000)
11. Hermeline, F.: A finite volume method for approximating 3D diffusion operators on general meshes. J Comput Phys **288**(16), 5763–86 (2009)
12. Mikula, K., Peyriéras, N., Remešíková, M., O., S.: Segmentation of 3D cell membrane images by PDE methods and its applications. Comput Biol Med **41**, 326–39 (2011)
13. Mikula, K., Remešíková, M.: Finite volume schemes for the generalized subjective surface equation in image segmentation. Kybernetika **45**(4), 646–56 (2009)
14. Sarti, A., Malladi, R., Sethian, J.: Subjective Surfaces: A Method for Completing Missing Boundaries. Proc Nat Acad Sci **12**(97), 6258–63 (2000)
15. Sethian, J.: Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Material Science. Cambridge Monographs on Appl and Comput Math. Cambridge University Press, New York (1999)