

# Voronoi grids conforming to 3D structural features

Romain Merland · Guillaume Caumon · Bruno Lévy ·  
Pauline Collon-Drouaillet

Received: 13 December 2012 / Accepted: 24 January 2014 / Published online: 24 May 2014  
© Springer International Publishing Switzerland 2014

**Abstract** Flow simulation in a reservoir can be highly impacted by upscaling errors. These errors can be reduced by using simulation grids with cells as homogeneous as possible, hence conformable to horizons and faults. In this paper, the coordinates of 3D Voronoi seeds are optimized so that Voronoi cell facets honor the structural features. These features are modeled by piecewise linear complex (PLC). The optimization minimizes a function made of two parts: (1) a barycentric function, which ensures that the cells will be of good quality by maximizing their compactness; and (2) a conformity function, which allows to minimize the volume of cells that is isolated from the Voronoi seed *w.r.t.*, a structural feature. To determine the isolated volume, a local approximation of the structural feature inside the Voronoi cells is used to cut the cells. It improves the algorithm efficiency and robustness compared to an exact cutting procedure. This method, used jointly with an adaptive gradient solver to minimize the function, allows dealing with complex 3D geological cases. It always produces a Voronoi simulation grid with the desired number of cells.

**Keywords** Voronoi · Reservoir grid · Structural features · Conform

## 1 Introduction

In reservoir simulation, unstructured grids offer great flexibility to adapt the cells in order to improve simulation results. Typical constraints on cells are local refinement [7, 8, 18], anisotropy [12, 13], and cell faces orientation [12, 14, 20]. In many case, it is also useful to make cells conformable to 3D geological surfaces, for example:

- In layered reservoirs, cells should not be partly in one layer and partly in another. Pinch-outs are frequent and cells should conform to them as accurately as possible. Indeed, cells as homogeneous as possible from a petrophysical point of view ensure upscaling consistency.
- For the same reason, when lateral variations exist inside a layer, cells should conform to petrophysical contrasts such as geobody boundaries.
- When faults put different rock types in contact, cells should also conform to them to ensure upscaling consistency.
- From the flow point of view, faults can act as barriers or drains. Hence, they are often modeled with transmissibility multipliers defined at the cell faces relevant to the faults. Consequently, to ensure discretization consistency, cells should conform to faults.

Previous work related to conformity to geological features has been realized on CVFE-BAG grids and Voronoi grids. These both types of grid use grid points around which the grid cells are built (Fig. 1).

In control volume finite element (CVFE) grids, a tetrahedralization of the grid points must be defined. Cells

R. Merland (✉) · G. Caumon · P. Collon-Drouaillet  
Géoressources (UMR 7356), Université de Lorraine-ENSG /  
CNRS / CREGU, Vandœuvre-lès-Nancy, France  
e-mail: merland@gocad.org

G. Caumon  
e-mail: guillaume.caumon@univ-lorraine.fr

P. Collon-Drouaillet  
e-mail: pauline.collon@univ-lorraine.fr

B. Lévy  
Institut Nationale de Recherche en Informatique, Le Chesnay,  
France  
e-mail: bruno.levy@inria.fr

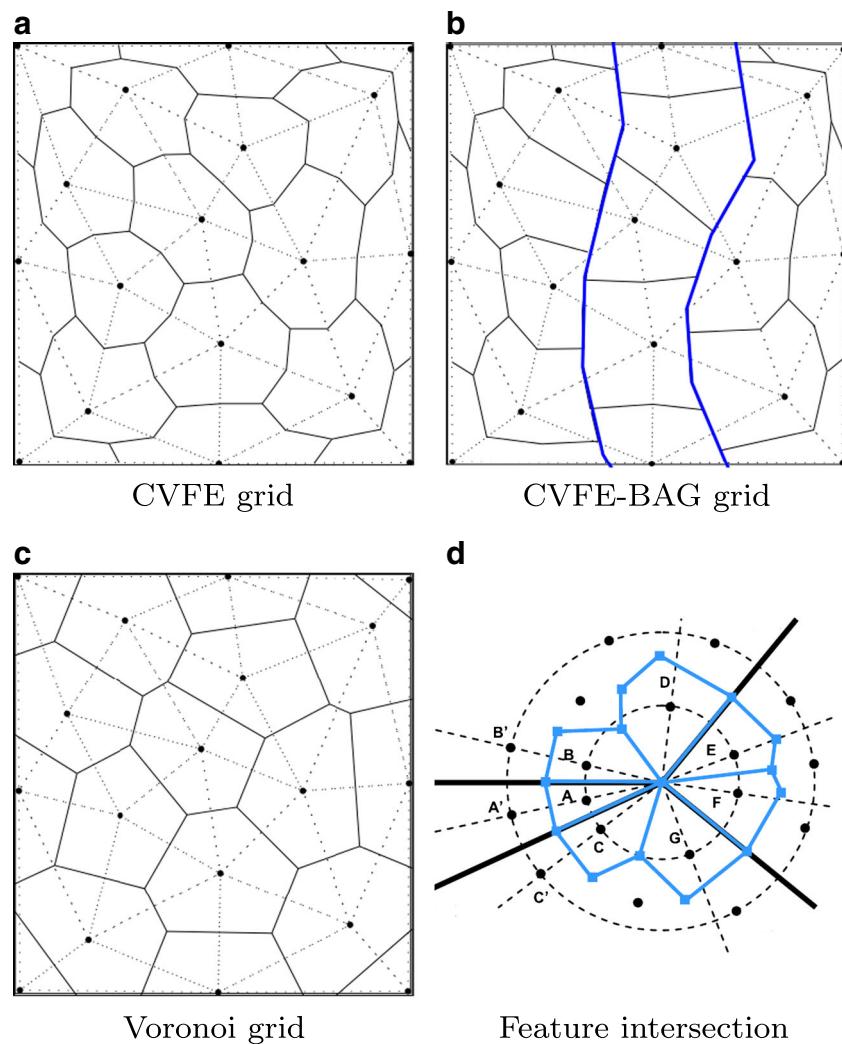
are built by joining the edge centers, the triangle barycenters, and the tetrahedron barycenters (Fig. 1a). Verma and Aziz [22] propose to modify the cell boundaries in order to conform them locally to the features, resulting in a CVFE-BAG or boundary-adapted grid (Fig. 1b). This approach cannot deal automatically with complex 3D features such as pinch-outs.

In Voronoi grids, a cell is defined as the nearest area around a grid point, also called Voronoi seed. Therefore, a face is in the perpendicular bisector plane of the segment linking the two Voronoi seeds on each side of the face (Fig. 1c). Courrioux et al. [5] propose to trace in 3D the normals to the feature, and place Voronoi seeds equidistant on each side of the feature. Consequently, the cells are conform to the feature. However, feature intersections produce contradictory disposition and need a specific optimization step by moving the cell boundaries onto the features. More recently, Branets et al. [1] propose in a similar way to define protection areas around the features by placing Voronoi seeds in mirror images. The feature intersections are treated

specifically with a circular mirror placement (Fig. 1d). The method gives good results in 2D but has not been extended to 3D to our knowledge.

All these methods allow to exactly recover the geological features. The drawback is their inability to deal with complex 3D reservoir in a robust and automatic way. In this article, rather than fixing the Voronoi seeds around the features, we choose to optimize the coordinates of a predefined number of Voronoi seeds by minimizing an objective function. This objective function allows incorporating several constraints at the same time, here a cell quality constraint and a feature conformity constraint. This function is not continuous and a specific solver is proposed to minimize the function and produce acceptable results on geological models. 3D synthetic cases are presented, and simulation results are provided for comparison: a fine Cartesian grid is used as reference, from which permeability and porosity are upscaled in coarse models. The cases have layers, faults, partly cutting faults, and pinch-outs.

**Fig. 1** Conforming polygonal grids in 2D. Triangulation of the grid points is in dashed lines. Cells around the grid points are in plain lines. **a** Control volume finite element (CVFE) grid. Cells are defined by joining segment centers and triangle barycenters. **b** Boundary-adapted grid (BAG) created by modifying a CVFE grid to honor a channel (in blue) [21]. **c** Voronoi grid. Cells are defined as the nearest area around the Voronoi seeds. Faces are in the bisector perpendicular plane of the segments joining the seeds. **d** Treatment of feature intersections with Voronoi seeds in a circular mirror disposition. Voronoi cells are in blue, features are bold lines [1]



## 2 Theory and method

### 2.1 Overview of the algorithm

The input of the methodology is made of (1) a closed boundary around the reservoir, and (2) geological features that cut partially or entirely the domain. All these surfaces are defined as 3D piecewise linear complexes (PLC), for instance triangulated surfaces.

The first step is to sample randomly the reservoir with a predefined number of Voronoi seeds inside the closed boundary. Then the seed coordinates are optimized to make the cell conform to the geological features, while guaranteeing the cell quality. This step is accomplished by minimizing iteratively an objective function  $F$  made of two parts as follows:

- A centroidal Voronoi tessellation (CVT) function  $F_{CVT}$ , which measures the inertial moment of the cells. Minimizing this function leads to a barycentric Voronoi diagram that ensures that the cells will be of good quality by maximizing their compactness.
- A conformity function  $F_{Vout}$ , which measures the volume of cells that is isolated from the Voronoi seed *w.r.t.*, a structural feature. Minimizing this function leads to cells that conform to the structural features.

These two functions are computed at each iteration from the Voronoi diagram computed efficiently on the fly from the current Voronoi seeds [24]. The minimized function  $F$  is the weighted sum of the two functions:

$$F = F_{CVT} + \alpha F_{Vout} \quad (1)$$

where  $\alpha$  is a weight chosen by the user and depends on the model. For units homogeneity, alpha is expressed in square meters. Our experiments suggest that choosing  $\alpha$  such that  $F_{CVT} \approx \alpha F_{Vout}$  allows to benefit from the effects of both functions simultaneously, i.e., to obtain a result approaching the results of each function minimized independently. This benefit is possible since the two function minimizations act differently, respectively, on the cell compactness and on the cell boundary. As we try to minimize  $F$ , we also compute its gradient, i.e., the sum of the gradients of the two parts:  $\nabla F = \nabla F_{CVT} + \alpha \nabla F_{Vout}$ . The solver used in this work moves each seed independently in the direction of the gradient, at a speed that depends on the gradient at the previous iteration. The number of iterations is chosen by the user, according to its evaluation of the result quality.

Once the seeds have been optimized, the final grid is obtained by clipping the 3D Voronoi diagram by the closed boundary of the reservoir. As a result, all the features are not necessarily honored (because it is an iterative optimization), but the error is expected to be small enough to meet discretization requirements, and the method is applicable to

any kind of geometries. An original feature is that the algorithm always converges to a solution, albeit approximate, whereas most other gridding algorithms strike for perfect match and do not always succeed.

An upscaling procedure is then performed to transfer the simulation parameters from the geological model to the grid, and a flow simulation is finally run on the grid.

### 2.2 Centroidal Voronoi tessellations

A centroidal Voronoi tessellation (CVT) is of particular interest for unstructured gridding because it provides the perpendicular bisector (PEBI) property, and guarantees the compactness of the cells and an optimal sampling of the Voronoi seeds (Fig. 2). The discretization errors due to badly-shaped cells are thus minimized.

As described by Du et al. [6], the Voronoi diagram relevant to a set of seeds  $X = \{x_i\}_{i=1}^N$  is a set of Voronoi cells  $V_i$  defined by the following (where  $\|\cdot\|$  denotes the Euclidean norm):

$$V_i = \{y \in \Omega \mid \|y - x_i\| \leq \|y - x_j\|, \text{ for } j \neq i\} \quad (2)$$

For each Voronoi cell, one can define the mass centroid  $x_i^* = \frac{\int_{V_i} y dy}{\int_{V_i} dy}$  (Fig. 2a). A CVT is a Voronoi tessellation where all seeds are placed at their cell mass centroids, i.e.,  $x_i = x_i^*$  (Fig. 2b).

The authors give the definition of a function  $F_{CVT}$  as follows:

$$F_{CVT}(X) = \sum_{i=1}^N \int_{V_i} \|y - x_i\|^2 dy \quad (3)$$

and prove that this function is minimal for Voronoi tessellations that are centroidal. Given an initial random set of seeds  $X$ , computing a CVT can then be expressed as a minimization problem.

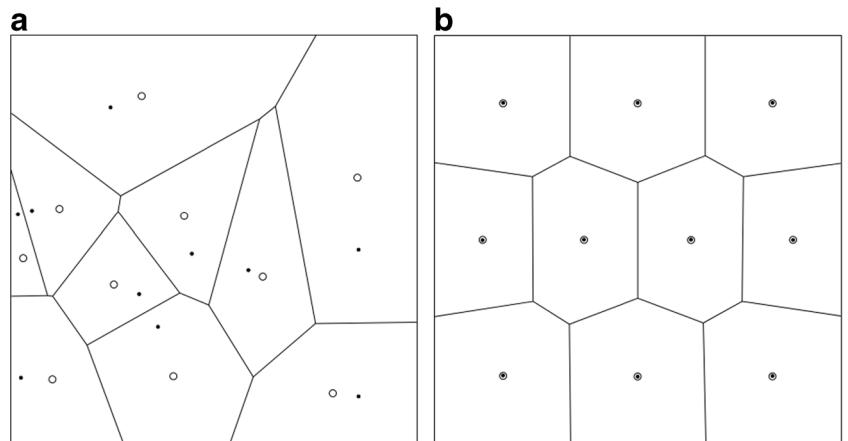
Liu et al. [10] prove that the function  $F_{CVT}$  is  $C^2$  and propose to minimize it with a fast quasi-Newton solver using the function value and its gradient, such as H-LBFGS. Considering the mass of the cell  $m_i = \int_{V_i} dy$ , the gradient of the function  $F_{CVT}$  for the seed  $x_i$  is given as follows [6]:

$$\nabla F_{CVT}|_i = 2m_i(x_i^* - x_i) \quad (4)$$

### 2.3 Honoring structural features

In a previous work [11], we proposed to modify the function  $F_{CVT}$  to make the Voronoi seeds go away from the features, resulting in a symmetrical alignment of the seeds on either side of the features. In this paper, an independent objective function  $F_{Vout}$  is rather defined and minimized. The function  $F_{Vout}$  is the sum of the outer volumes  $V_{out}$ . For a given cell,  $V_{out}$  is the volume isolated from the Voronoi seed

**Fig. 2** Voronoi tessellation and centroidal Voronoi tessellation for ten seeds in two dimensions. **a** Seeds are black dots, mass centroids are small circles. **b** Seeds are located at the mass centroids of their Voronoi regions [6]



(Fig. 3a). The function and its gradient are given as follows:

$$F_{Vout} = \sum_{i=1}^N V_{out} \quad \nabla F_{Vout} = \sum_{i=1}^N \nabla V_{out} \quad (5)$$

The gradient of the volume can be computed from the symbolic information stored in each vertex of the Voronoi cell [9].  $\nabla V_{out}$  is the gradient of the volume  $\nabla V$  computed only for the vertices in the outer part.

#### 2.4 Dealing with dangling surfaces

When a geological feature cuts partially a Voronoi cell (Fig. 3b), it is not possible to determine an outer volume. A local approximation of the feature as a 3D plane is proposed to overcome this problem. The computation of the function  $F_{Vout}$  and its gradient for a given Voronoi cell is done in four steps:

- Computation of the intersection of the Voronoi cells and the features [23]. This allows to determine which feature cuts which cell, and retrieve the part of the feature inside the cell.
- For each cell intersected by a feature, approximation of the intersected part of the feature by a mean plane, i.e., the barycenter and the mean normal vector of the intersected part of the feature.
- Determination of the inner and outer parts of the cells w.r.t the relevant mean planes.
- Computation of  $F_{Vout}$  and its gradient.

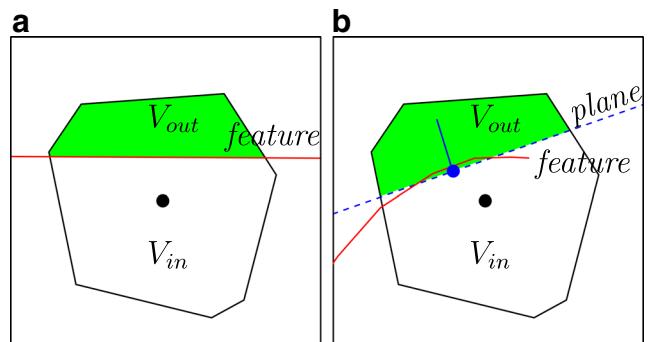
This method is robust and applicable to any kind of geometry. When several features intersect a cell, a mean plane is computed for each feature, and the computed outer volumes are summed in the function  $F_{Vout}$ . The approximation of the features can be very poor when the sampling, i.e., the number of seeds, is coarse, but improves when the sampling becomes more dense. The sufficiently high number of seeds is let at the user appreciation because the number of cells has a direct impact on the flow simulation performance.

A criterion could be the maximum curvature of the intersected parts of the features. The approximation also tends to become negligible after several iterations of the optimization process in the common case of subplanar geological features.

#### 2.5 Adaptive steepest gradient solver

The optimization process minimizes the function  $F = FCVT + \alpha F_{Vout}$ . The function  $FCVT$  is  $C^2$  [10] but the function  $F_{Vout}$  is not  $C^1$ . Indeed, when a seed crosses a feature, the inner part becomes the outer part and vice versa. Therefore, even a classical gradient descent algorithm as described in [17] cannot be applied to minimize the function. We propose to use an adaptive steepest gradient solver instead, in order to take advantage of the knowledge of the gradient, without guaranteeing its convergence.

This solver is similar to that used in image analysis [15]. It is a gradient descent solver that takes into account the direction of the gradient at the previous step to update the



**Fig. 3** Parameters taken into account for the computation of the objective function. **a** Outer volume (green), inner volume (white), w.r.t the feature (red line). **b** Approximation of a dangling feature by a 3D plane (dashed blue line) that cuts totally the cell. The plane is defined by the barycenter and the mean normal vector of the part of the feature inside the cell

step size of each seed at each iteration. For each seed  $S$ , the following steps are performed:

**Initialization:**

Step size:  $s_0$ , gradient relevant to seed  $S$ :  $\nabla F_0|_S$

**At each iteration  $i$ :**

Compute the gradient  $\nabla F_i|_S$

Direction of displacement:  $d_i = -\frac{\nabla F_i|_S}{\|\nabla F_i|_S\|}$

Step size:

If  $\nabla F_i|_S$  and  $\nabla F_{i-1}|_S$  are in opposite direction,  $s_i = (\beta_-)s_{i-1}$ ,

otherwise,  $s_i = (\beta_+)s_{i-1}$

Move the seed  $S_{i+1} = S_i + s_i d_i$

**Finalization:**

Maximum number  $N$  of iterations reached.

$s_0 (> 0)$ ,  $\beta_- (< 1)$ ,  $\beta_+ (> 1)$  and  $N$  are dimensionless constants defined by the user. Experimentally, we have observed that this solver generally gives an acceptable solution after 50 iterations when used with the function  $F$  described above.

## 2.6 Assigning flow properties to the cells

A static geological model usually integrates lots of detailed descriptions at fine scale. For computation time reasons, flow simulation is rarely run on this fine grid and a coarse grid is generally built. Effective flow parameters (porosity, permeability, and transmissibility) are computed on the coarse grid from the fine-scale parameters (the process is called upscaling), and finally the simulation is run on the coarse grid [7].

A classical upscaling technique is the volume averaging. When dealing with coarse cells that are not aligned with the underlying fine cells, the averaging generally includes those fine cells whose center falls in the coarse cell [7]. More accurate methods compute the intersection of the fine and coarse cells to get the exact volume averaged result, although they are computationally more expensive.

In this work, a volume averaging is applied for porosity and permeability upscaling. The effective porosity  $\Phi^*$  and the effective permeability  $K^*$  values are given by the following equation:

$$\Phi^* = \int_{V_i} \Phi \, dV \quad K^* = \int_{V_i} K \, dV \quad (6)$$

A fast algorithm is developed to compute the exact intersection of fine and coarse Voronoi cells, which takes advantage of Voronoi diagram properties: the Voronoi cells are convex and any point in a coarse cell is closer to its Voronoi seed than to any other seed of the diagram. The algorithm can be decomposed in three steps:

- The bounding box of the fine cell is computed and the closer seed for each corner of the box is retrieved through a kd-tree algorithm [19].
- If all the corners of the box have the same closer seed, the fine cell is entirely in the coarse cell due to the Voronoi property. Therefore, the fine cell is processed directly.
- Otherwise, the exact boundary of the fine cell is retrieved, and a fast decomposition of the fine cell is performed using the algorithm from [24].

The algorithm is further improved when the fine grid is Cartesian. In that case, the fine cell boundary is stored implicitly and the decomposition process reduces to the computation of intersections between coarse cell facets, i.e., perpendicular bisector of the segment connecting two Voronoi seeds, and segments of the fine Cartesian cell.

Isotropic permeability is assumed and TPFA discretization is applied to compute transmissibility at the  $ij$  interface as follows:

$$T_{ij}^* = \frac{2K_i^* K_j^*}{K_i^* + K_j^*} \frac{A_{ij}}{l_{ij}} \quad (7)$$

where  $A_{ij}$  is the interface area and  $l_{ij}$  is the distance between the cell centers. Other upscaling techniques can be used to get directly effective transmissibilities (see [3] for instance).

The well index is computed from the effective permeability using the so-called Peaceman formula (see [16] for details). A 3D completion cell along a well is approximated by a cube of the same volume to get the geometrical parameters needed in the Peaceman formula. Flow-based computation of the well index as described in [14] could also be considered to further improve the results.

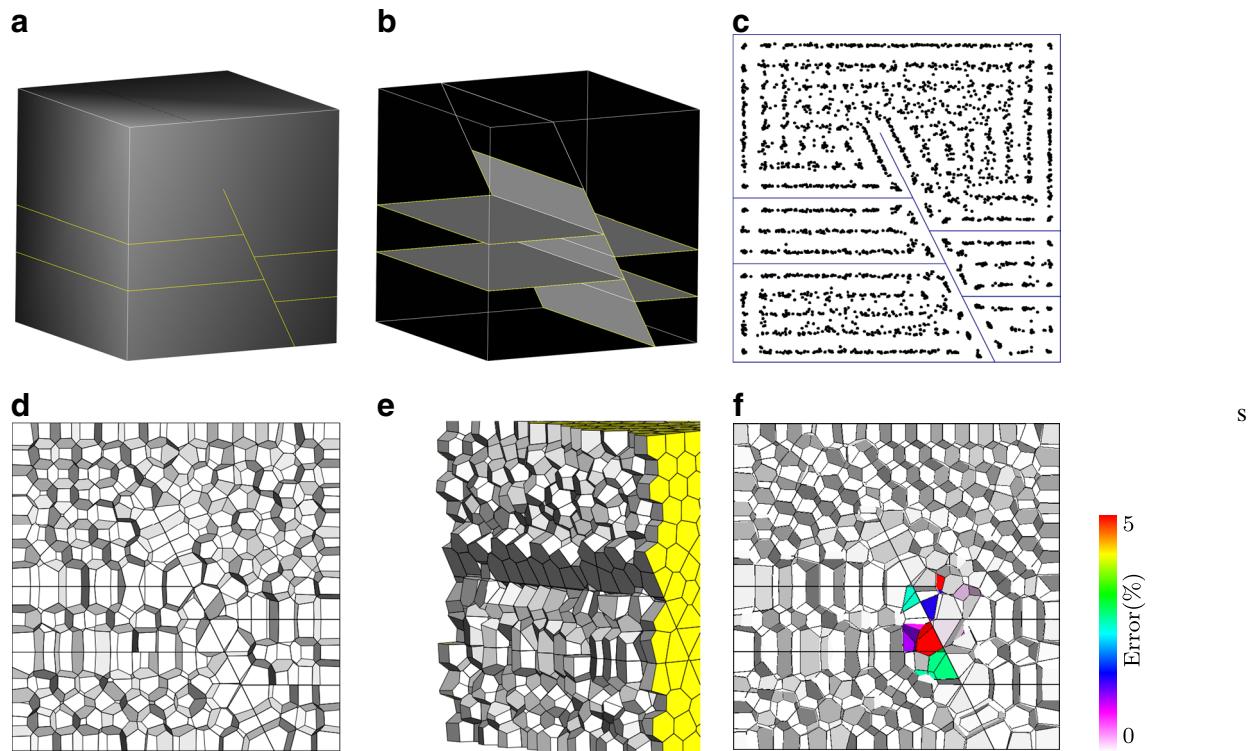
## 3 Results

### 3.1 Geometrical results

This methodology has been tested on a synthetic reservoir composed of three layers partly faulted in a cube  $100 \times 100 \times 100$  ft. The resulting mesh for 3,000 seeds obtained by minimizing (1) is presented on Fig. 4.

In this work, the quality of fit is assessed with a global error defined as  $\frac{\sum V_{out}}{\sum V_{ic}}$  where  $V_{ic}$  is the volume of the cells intersected by the features. This dimensionless metric can be used to compare various models. The errors for a reference barycentric nonconformable mesh and for the generated conformable mesh are presented in Table 1.

A local error can also be computed with the same formula for each cell to identify the problematic parts of the reservoir. The map of residual error (Fig. 4f) indicates that



**Fig. 4** Synthetic case with a partly cutting fault. **a** Outer view. **b** Inner view showing the fault, layers, and pinch-outs. **c** Slice along y-axis showing optimized Voronoi seeds. **d** Resulting grid along the same slice. **e** Slice along the x-axis. **f** Error map

the pinch-outs are the areas where the conformity error is highest.

The second model is a faulted zone (data courtesy of IFPEN [4]) with 27 surfaces, of box size  $2,840 \times 3,000 \times 570$  ft. A grid of 10,000 cells is generated and presented on Fig. 5. Global and local errors are shown in Table 1 and on Fig. 5d.

The third model is a geological model of a faulted zone with three layers (Graben, data courtesy of TOTAL), of box size  $4,000 \times 4,000 \times 1,000$  ft. A grid of 10,000 cells is

generated and presented on Fig. 6. Global and local errors are shown in Table 1 and on Fig. 6f, respectively.

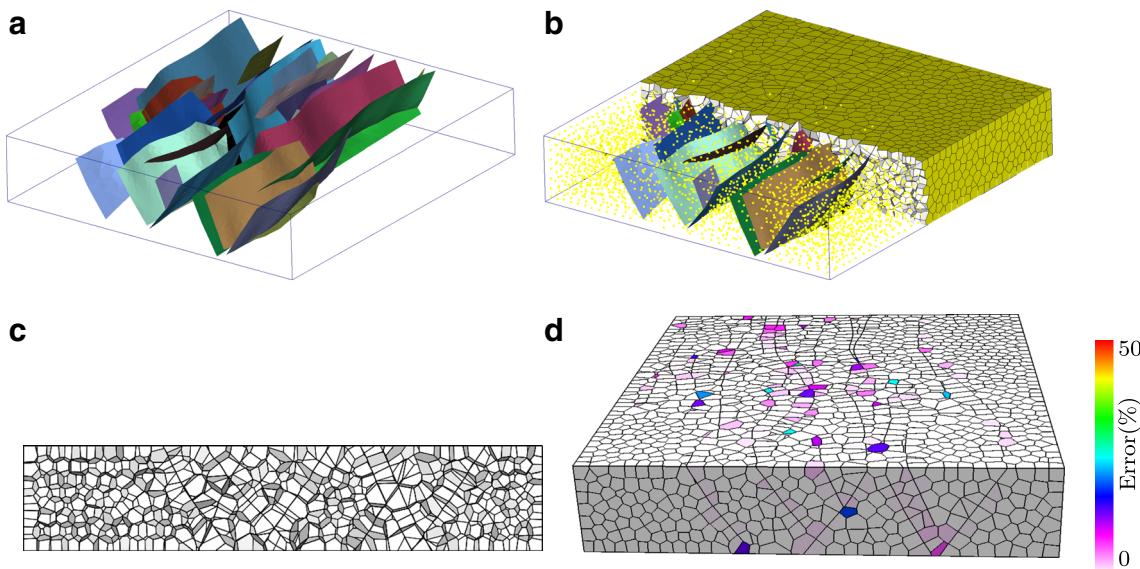
The fourth model is a faulted and folded reservoir (Fault Bend Fold, data courtesy of Harvard/Chevron). It demonstrates the capacity of the proposed method to deal with a more complex geometry (not a simple box). A grid of 20,000 cells is generated and presented on Fig. 7. Global error is shown in Table 1.

The fifth model is a geological model of the Noiraigue faulted zone with many layers (courtesy of A. Borghi,

**Table 1** Values of the global error in %  $\frac{\sum V_{out}}{\sum V_{ic}}$  where  $V_{ic}$  is the volume of the intersected cells

Model	Global error		Number of cells	Number of polygons	Optimization time (s)
	Non conformable	Conformable			
Faulted cube	18.55	0.20	3,000	11	18.24
IFPEN	17.77	1.17	10,000	17,193	97.73
Graben	17.45	1.51	10,000	7,763	84.11
FBF	17.63	0.62	20,000	17,754	230.52
Noiraigue	17.33	1.76	100,000	168,129	8,586.25
Duplex	16.18	0.33	3,000	22	24.03

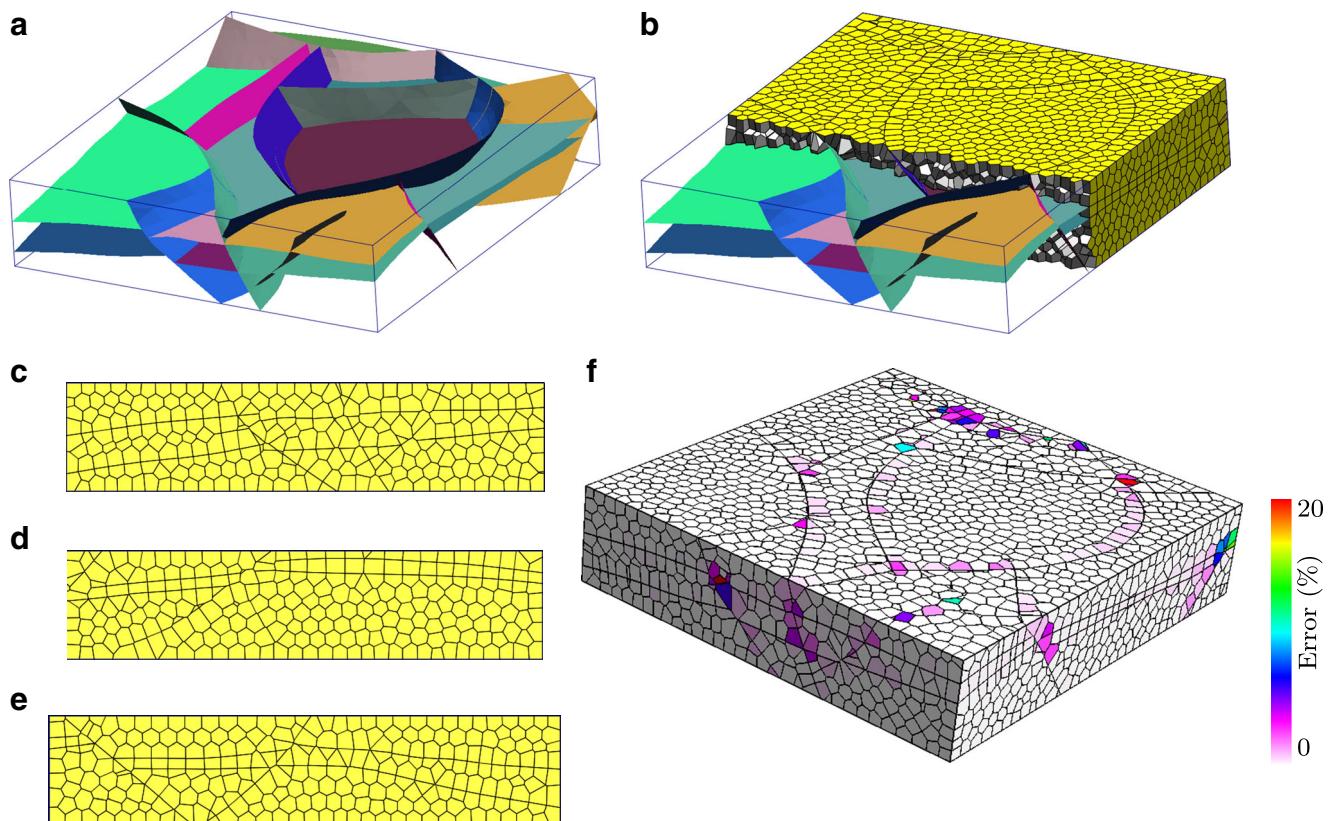
For each model, the nonconformable case and the conformable case are presented. Time performance is compared to the number of cells and the number of polygons of the geological features



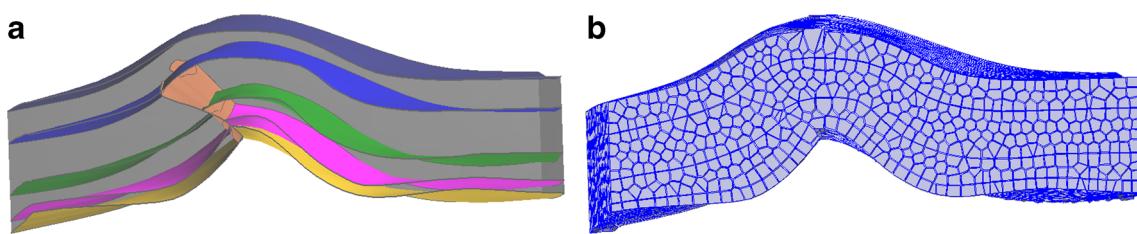
**Fig. 5** Faulted zone model (data courtesy of IFPEN) with 27 faults. A grid of 10,000 cells is generated. **a** Inner view of the faults. **b** Faults, grid, and Voronoi points. **c** Slice along y-axis. **d** Error map

Unine). This model is provided to demonstrate the robustness of the methodology to a great number of surfaces

defining thin layers with many pinch-outs. Some surfaces present degenerated triangles that are properly handled by



**Fig. 6** Graben model (data courtesy of TOTAL). A grid of 10,000 cells is generated. **a** Inner view of faults and horizons. Outer view of the grid **b** sliced along y-axis, with features displayed, **c** from south, **d** north, and **e** west, without features displayed. **f** Error map



**Fig. 7** Fault Bend fold (data courtesy of Harvard/Chevron). A grid of 20,000 cells is generated. **a** Inner view of the horizons and the fault. **b** Generated grid. Cells are slightly shrunk to optimize viewing

the algorithm. It is composed of 921 surfaces, inside a box of size 19, 100 × 10, 950 × 2, 465 ft. A grid of 100,000 cells is generated and presented in Fig. 8. Global and local errors are shown in Table 1 and on Fig. 8d, respectively.

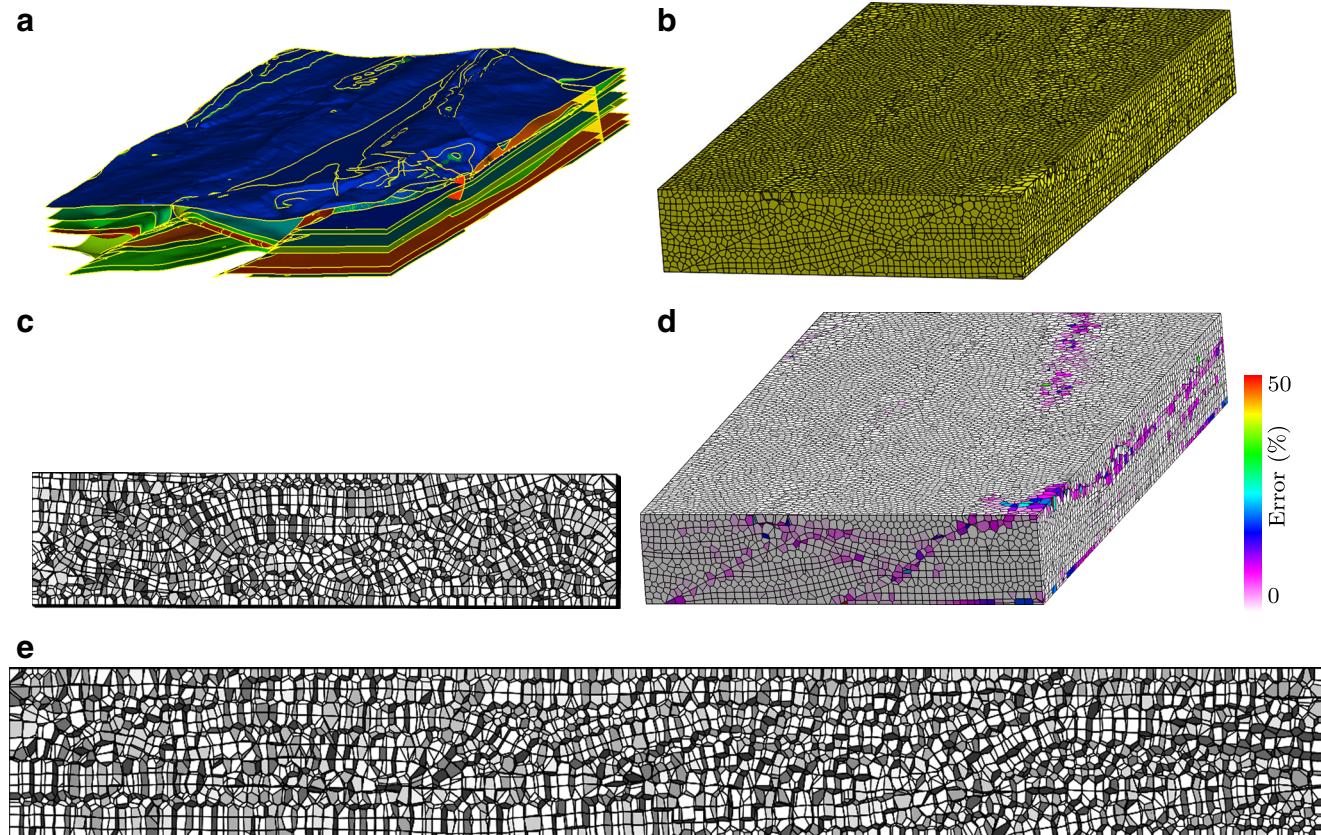
Our generated grids show a great improvement in the global error as compared to the nonconformable case. Only few cells remain with isolated parts. The algorithm fails to exactly match the features for the following two reasons:

- The algorithm is an optimization of a function made of two parts that are not perfectly perpendicular, i.e., the minimum of the conformity function does not allow to reach the minimum of the CVT function and vice versa.

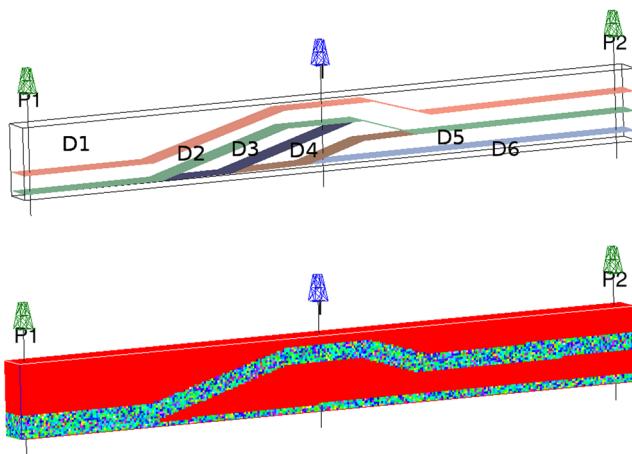
- The exact conformity in the pinch-outs is a degenerate configuration from the Voronoi diagram point of view and leads to numerical instability. We can get very close, but not exact.

A small additional effort must be performed as a post process if exact conformity is needed. However, it is generally not needed for applications such as reservoir simulation.

From a performance point of view, the number of cells has a major impact on the optimization process duration as shown in Table 1, and the number of polygons of the geological features has a minor impact.



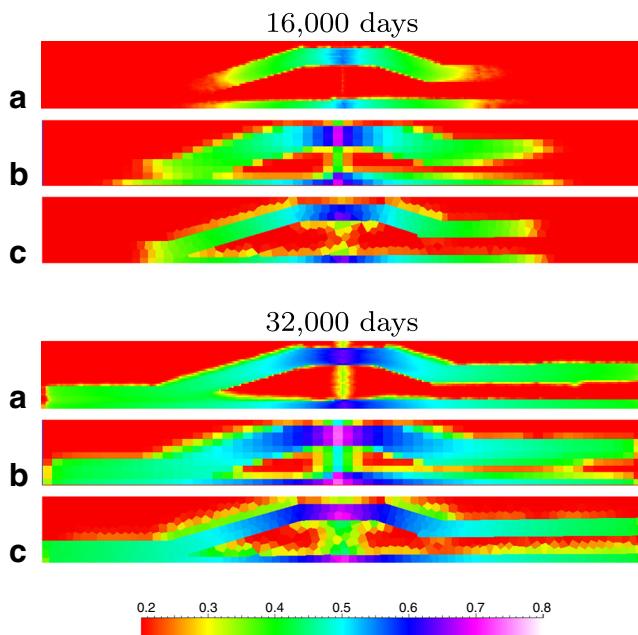
**Fig. 8** Noirague faulted zone model (data courtesy of A. Borghi, Unine). A grid of 100,000 cells is generated. **a** Inner view of the faults and horizons, **b** outer view of the generated grid, **c** slice along  $x$ -direction, **d** error map, and **e** slice along  $y$ -direction



**Fig. 9** Duplex model, fine grid with five inner surfaces, and six sub-domains D1–D6, one injector and two producer wells, I, P1, and P2, respectively. On the second picture, the permeability field is displayed

### 3.2 Flow simulation results

The methodology has been applied to a synthetic duplex of dimensions  $32,900 \times 1,600 \times 3,900$  ft with five inner surfaces, branching intersections, and pinch-outs. A fine Cartesian grid is generated with  $329 \times 16 \times 39$  cells and populated with properties according to the subdomains as seen on Fig. 9: permeability  $K$  (mD) and porosity  $\Phi$  are simulated with Gaussian distributions with  $K \sim \mathcal{N}(1, 0.15)$ ,  $\Phi \sim \mathcal{N}(0.05, 0.005)$  for subdomains D1, D3, D4, and D5,

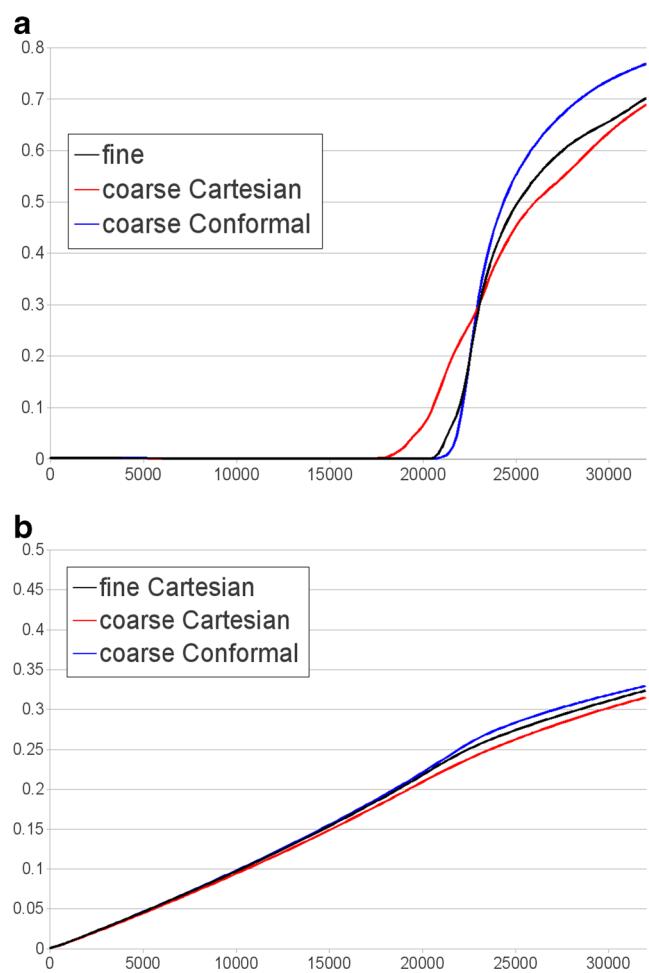


**Fig. 10** Duplex model, comparison of water saturation fronts for **a** fine grid (185,000 cells), **b** coarse Cartesian grid (3,000 cells), and **c** conformable Voronoi grid (3,000 cells), at 16,000 and 32,000 days of simulation

$K \sim \mathcal{N}(100, 15)$ ,  $\Phi \sim \mathcal{N}(0.5, 0.05)$  for subdomains D2 and D6. A two-phase flow simulation is run with GPRS [2] from the discretized reservoir using TPFA for transmissibility computation and Peaceman's formula for well index computation.

A coarse Cartesian grid and a conformable Voronoi grid of 3,000 cells each are generated. Permeability and porosity are upscaled using exact volume average and transmissibility, and well index are computed from the upscaled isotropic permeability. Simulations are run with GPRS and water saturation is compared after 16,000 and 32,000 days on Fig. 10. Water cut at well P1 and recovery factor are shown on Fig. 11.

As expected, the water saturation profile is more accurate with the conformable grid than with the Cartesian grid, and the water cut is better recovered. However, the water saturation front is globally earlier in both coarse grids than in the fine grid. It is a well-known effect of grid coarsening. However, it can also be due to the upscaling technique coupled



**Fig. 11** Duplex model, comparison of **a** water cut at producer well P1 and **b** recovery factor

with the discretization scheme which does not account for fine heterogeneities.

## 4 Conclusions

The proposed methodology shows several interesting properties:

- It is entirely automated in 3D, with a user-specified number of cells.
- It is robust *w.r.t* complex structural features such as pinch-outs and multiple intersections.
- Generated grids are strictly PEBI (Voronoi) with cells as compact as possible.
- The conformity with structural features can be combined with other constraints such as grid refinement, conformity with wells, and cell orientation through the use of additional objective functions.

Because structural features are not perfectly recovered with this optimization technique, post-processing would be needed if more accuracy is required. However, only few elements need to be post-processed, which we believe could make this step relatively easy. Grid refinement could also be applied locally, in problematic zones identified by the error maps, and a few more optimization iterations could reduce the residual error. The density can be integrated with a smooth variation through the use of a density field added to the CVT function [6]. Letting the user fix the initial number of seeds may have consequences. If a too small number of seeds is chosen, one could not generate a conforming quality grid. One solution could be to set both the initial number of seeds and the density *w.r.t.* the local curvature of the structural features. Another way could be to manually post-process these problematic zones by modifying directly the grid geometry, but the PEBI property would be locally lost.

The adaptive steepest gradient solver presented in this paper allows to obtain acceptable results when minimizing the function  $F = F_{CVT} + \alpha F_{Vout}$ . However, the convergence is not guaranteed because the function  $F_{Vout}$  is not proven to be  $\mathcal{C}^1$ . The use of a more suited solver could improve the result. Another way of development would be to find a conformity function that is at least  $\mathcal{C}^1$  or  $\mathcal{C}^2$  to be able to use gradient descent algorithms in a robust way.

The conformity function can also be minimized jointly with other functions to ease and/or speed up the optimization. For instance, a function could constrain cell facets to be parallel/perpendicular to the feature's normal in subplanar areas [9].

The application of the presented methodology on a synthetic case shows an improvement of the simulation results,

and especially on the water saturation front, in the case of TPFA discretization from the upscaled permeability. Better upscaling techniques and discretization schemes should be developed to the generated grid to better match the fine grid behavior.

**Acknowledgments** We thank Paradigm for providing the g○cad developer licenses and the g○cad Consortium for funding this research. We thank the ALICE team of LORIA in Nancy for the CVT algorithms and the CGAL team of INRIA Sophia Antipolis for the Delaunay algorithms. We thank TOTAL for providing the Graben data set. The general purpose research simulator (GPRS) developed by the Reservoir Simulation Group (SUPRI-B) at Stanford University was used in this work.

## References

1. Branets, L.V., Ghai, S.S., Lyons, S.L., Wu, X.H.: Efficient and accurate reservoir modeling using adaptive gridding with global scale up. In: Proceedings of SPE Reservoir Simulation Symposium, SPE118946, p. 11. The Woodlands (2009). doi:[10.2118/118946-MS](https://doi.org/10.2118/118946-MS).
2. Cao, H.: Development of Techniques for General Purpose Simulators. Ph.D. thesis, Stanford University, p. 202 (2002)
3. Chen, Y., Wu, X.: Upscaled modeling of well singularity for simulating flow in heterogeneous formations. Comput. Geosci. **12**(1), 29–45 (2008). doi:[10.1007/s10596-007-9059-5](https://doi.org/10.1007/s10596-007-9059-5)
4. Colletta, B., Letouzey, J., Pinedo, R., Ballard, J., Balé, P.: Computerized x-ray tomography analysis of sandbox models: examples of thin-skinned thrust systems. Geology **19**, 1063–1067 (1991)
5. Courrioux, G., Nullans, S., Guillen, A., Boissonnat, J.D., Repusseau, P., Renaud, X., Thibaut, M.: 3D volumetric modelling of cadomian terranes (northern brittany, france): an automatic method using Voronoi diagrams. Tectonophysics **331**(1–2), 181–196 (2001). doi:[10.1016/S0040-1951\(00\)00242-0](https://doi.org/10.1016/S0040-1951(00)00242-0)
6. Du, Q., Faber, V., Gunzburger, M.: Centroidal Voronoi tessellations: Applications and algorithms. SIAM Rev. **41**, 637–676 (1999)
7. Durlofsky, L.J.: Upscaling and gridding of fine scale geological models for flow simulation. In: 8<sup>th</sup> International Forum on Reservoir Simulation, p. 59. Iles Borromées, Stresa, Italy (2005)
8. Evazi, M., Mahani, H.: Unstructured coarse grid generation using background-grid approach. SPE J. **15**(2), 326–340 (2010)
9. Lévy, B., Liu, Y.: Lp centroidal Voronoi tessellation and its application. ACM Trans. Graph. **29**(4), 11 (2010). doi:[10.1145/1778765.1778856](https://doi.org/10.1145/1778765.1778856)
10. Liu, Y., Wang, W., Lévy, B., Sun, F., Yan, D.M., Lu, L., Yang, C.: On centroidal Voronoi tesselation—energy smoothness and fast computation. ACM Trans. Graph. **8**(4), 30 (2009). doi:[10.1145/559755.559758](https://doi.org/10.1145/559755.559758)
11. Merland, R., Lévy, B., Caumon, G.: Building PEBI grids conforming to 3D geological features using centroidal Voronoi tessellations. In: Proceedings IAMG, p. 12. Salzburg (2011)
12. Merland, R., Lévy, B., Caumon, G., Collon-Drouaillet, P.: Building centroidal Voronoi tessellations for flow simulation in reservoirs using flow information. In: Proceedings of SPE Reservoir Simulation Symposium, SPE141018-PP, p. 11. The Woodlands (2011). doi:[10.2118/141018-MS](https://doi.org/10.2118/141018-MS).

13. Mlacnik, M., Durlofsky, L.: Unstructured grid optimization for improved monotonicity of discrete solutions of elliptic equations with highly anisotropic coefficients. *J. Comput. Phys.* **216**(1), 337–361 (2006). doi:[10.1016/j.jcp.2005.12.007](https://doi.org/10.1016/j.jcp.2005.12.007)
14. Mlacnik, M., Durlofsky, L., Heinemann, Z.: Sequentially adapted flow-based PEBI grids for reservoir simulation. *SPE J.* **11**(3), 317–327 (2006)
15. Nivoliers, V.: échantillonnage pour l'approximation de fonctions sur des maillages ch. 4. Ph.D. thesis, Université de Lorraine (2012)
16. Palagi, C.L., Aziz, K.: Modeling vertical and horizontal wells with Voronoi grid. *SPE Reserv. Eng.* **9**(1), 15–21 (1994). doi:[10.2118/24072-PA](https://doi.org/10.2118/24072-PA)
17. Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T.: Numerical Recipes: The Art of Scientific Computing, Chap. 10, 2nd edn. Cambridge University Press, Cambridge (1992)
18. Prevost, M., Lepage, F., Durlofsky, L., Mallet, J.L.: Unstructured 3D gridding and upscaling for coarse modelling of geometrically complex reservoirs. *Pet. Geosci.* **11**(4), 339–345 (2005)
19. Samet, H.: Foundations of multidimensional and metric data structures. *Computer Graphics and Geometric Modeling*, Morgan Kaufmann Series p. 993 (2006)
20. Souche, L.: Generation of unstructured 3D streamline pressure-potential-based k-orthogonal grids. In: Proceedings of 9<sup>th</sup> ECMOR, p. 8. Cannes (2004)
21. Verma, S.: Flexible Grids for Reservoir Simulation. Ph.D. thesis, Standford University p. 247 (1996)
22. Verma, S., Aziz, K.: A control volume scheme for flexible grids in reservoir simulation. In: Proceedings of SPE Reservoir Simulation Symposium, p. 13. Dallas (1997). doi:[10.2118/37999-MS](https://doi.org/10.2118/37999-MS).
23. Yan, D.M., Lévy, B., Liu, Y., Sun, F., Wang, W.: Isotropic remeshing with fast and exact computation of restricted Voronoi diagram. *Comput. Graph. Forum* **28**(5), 1445–1454 (2009). doi:[10.1111/j.1467-8659.2009.01521.x](https://doi.org/10.1111/j.1467-8659.2009.01521.x)
24. Yan, D.M., Wang, W., Lévy, B., Liu, Y.: Efficient computation of 3D clipped Voronoi diagram. In: Advances in Geometric Modeling Processing, pp. 269–282 (2010)