Reconstruction of Geological Structures from Heterogeneous and Sparse Data

Jean-Daniel Boissonnat
Stéphane Nullans*
INRIA
BP93, 06902 Sophia-Antipolis cedex (France)
E-mail: { Jean-Daniel.Boissonnat, Stephane.Nullans} @sophia.inria.fr
http://www.inria.fr/prisme/personnel/{ boissonnat, nullans}/{ boissonnat, nullans}.html

Abstract

Given a set of sparse and heterogeneous outcrop and drilling data, we want to retrieve a complete and coherent model of the underground model, i.e. to determine the entire geometry of the surfaces and volumes of the model as well as the topology of the contacts.

The reconstruction method proposed in this article is intended to assist the geologist in his interpretation of 2-dimensional geological cross-sections. It constructs the geological formations one by one in an appropriate order. The reconstruction of each formation consists of three main steps. We construct a first model from the Voronoi diagram of the input data. Then we smooth the boundary of this formation without changing the topology. Lastly, we possibly shift part of the formation along faults.

Experimental results on several examples show that the method is very effective. The method extends without major difficulties to truly 3-dimensional reconstructions.

Keywords : Shape reconstruction, Voronoi diagrams, Deformable curves, Geological structures.

Topics : GIS software tools and environment, Novel applications in scientific and environmental domains.

1 Introduction

Besides scientific interest, a better knowledge of the subsoil is required in many applications. Let us mention the natural resource exploration (gas and oil reservoirs, ore bodies), civil engineering (tunnels with thick rock covering), environmental sciences (surface hazard linked to deep-seated excavations, underground migration of pollutants, waste storage).

The usual representations of underground geometry consist in 2D cross-sections. In the most common case of sedimentary terrains, cross-sections show geological formations, sedimentary interfaces and discontinuities of tectonic origin. A geological formation correspond to a set of homogeneous layers having a given age, a sedimentary interface to a boundary between two or more formations, a discontinuity to a fault or a thrust sheet contact, which both result from tectonic differential underground motions.

The existing and usual terrain modeling methods, are mostly based on surface oriented interpolations ([Mal89, SSMW93, RHC+92]). These methods assume dense set of data, and their principal difficulties were collected by [May93] : (1) creating surfaces with large variety of topology and geometry (2) creating volumes from surfaces (intersection of half spaces divided by these surfaces). Most of these modeling tools have been developed in the context of oil exploration, where the seismic data are very dense, and concern sedimentary terrains submitted to moderate tectonic deformations.

However, in other contexts such as civil engineering or environment , it is much more difficult to collect data and also to interpolate them since tectonic deformations can be very intense. One has then to deal with data that are rare, heterogeneous and sparse. Typical example are shown in Figures 17,24,26. In these cases, the interpolation methods cannot be applied any more and reconstructing a consistent model of the geometry of the subsoil from such data is a highly non trivial and time consuming task that requires a large expertise. The situation is even worse when truly 3D models are searched.

The goal of our work is to assist the geologist in this task. This paper considers only the reconstruction of 2D cross-sections. Our method constructs the formations one by one in an appropriate order. The reconstruction of each formation consists of several steps that successively refines the description of its reconstructed model. The first step (Section 2) defines the formation according to the nearest neighbor rule : a point P belongs to a given formation F if the nature of the soil at the data point closest to P is F. The corresponding map can be efficiently obtained from the Voronoi diagram of the input data. This allows to separate the current formation from the not yet reconstructed ones. However, additional smoothing is required to obtain a geological correct formation, without changing the topology provided by the Voronoi diagram (Section 3). Faults are considered in a last step and the formation is updated to take them into account (Section

* The work by this author has been partially supported by the Bureau de Recherches Géologiques et Minières (BRGM Orléans-France).

4). The overall reconstruction algorithm is presented in Section 5 and results are discussed.

2 Topological Reconstruction

2.1 The Voronoi Diagram

Let $S = \{s_1, \ldots, s_n\}$ be a finite set of points, called sites, of the Euclidean plane E^2. The Voronoi diagram of these sites is the partition of the plane assigning each point to its nearest site (see Figure 1). The Voronoi diagram consists of n cells, one per site. The Voronoi cell $V(s_i)$ consists of all the points at least as close to s_i as to any other site :

$$V(s_i) = \{x \in E^2, \forall q \in S \setminus s_i, \delta(x, s_i) \leq \delta(x, q)\}$$

where δ is the Euclidean distance in E^2.

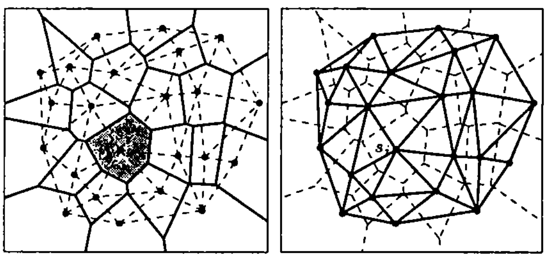

Figure 1: Voronoi diagram and Delaunay triangulation of points in the plane.

For convenience, we suppose that the sites are in general position, meaning that no four sites are co-circular. This is no real loss of generality since we can symbolically perturb the sites [Sei94].

The dual graph of the Voronoi diagram is obtained by connecting the pairs of sites that belong to adjacent Voronoi cells by line segments (see Figure 1). Under the general position assumption, the dual graph of the Voronoi diagram is a triangulation called the Delaunay triangulation. The Delaunay triangulation has the property that the circumdisk of every triangle contains no sites in its interior; such a disk will be simply called an empty disk for short.

The Voronoi diagram and the Delaunay triangulation of a set of n points in the plane can be constructed optimally in $O(n \log n)$ time [Aur91]. For reasons that will be clear in the sequel, we use an incremental algorithm that allows to insert new sites efficiently. Although incremental algorithms may be quadratic in the worst-case, the algorithm we use requires $O(\log n)$ expected[1] time for inserting a new site, which is optimal [BT93].

2.2 Data Discretization

The initial data are not only points (samples or point observations) but more generally line segments and arcs of curves (seismic interfaces, directions). The Voronoi diagram can be extended to such non punctual sites [Yap87]. However, the algorithms become more difficult to implement and to make robust. Rather than constructing a generalized Voronoi diagram, we prefer to discretize the data and to simply consider the Voronoi diagram of a discrete set of points. This is done in such a way that successive points on a discretized curve belong to adjacent cells in the Voronoi diagram (or equivalently are joined by an edge in the Delaunay triangulation). This is always possible provided that sufficiently many points are taken (see [Boi88, ET92] for details).

Moreover, the data may correspond to points inside a formation (typically drilling data) or to points belonging to the interface between two formations (mostly obtained from seismic images). Each point on a drilling line is colored according to the geological formation it belongs to, this information being part of the input. Each point on an interface is duplicated. The two copies are slightly displaced, one on each side of the interface (see Figure 2). Each copy is colored according to the formation it belongs to. Interface directions on points, are handled like local known interfaces.

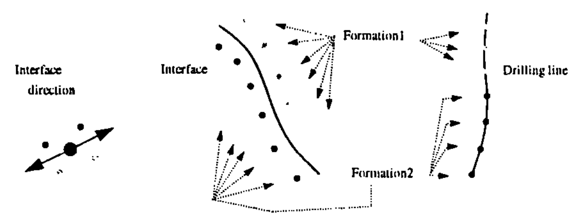

Figure 2: Data discretization.

2.3 Construction of Homogeneous Regions

Each site in S has now a color and we color each cell in the Voronoi diagram of S as its corresponding site. By merging the adjacent Voronoi cells that have the same color, we obtain a partition of the plane into colored connected regions (see Figure 3). These regions are first approximations of the geological formations obtained by applying the following rule : A point P belongs to a formation F iff the site closest to P belongs to F.

This rule does not take into account the heterogeneous spatial distribution of the data. However, it will be used in our algorithm, to provide a solution that is, in most cases, *topologically* correct, i.e. homotopic to the actual geological cross-section.

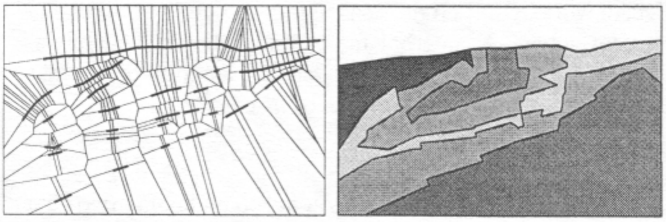

Figure 3: Voronoi diagram of the discretized data of Figure 27 (except the faults) - Homogeneous regions.

[1] Expectation is obtained by averaging over all the permutations of the entries.

However, the shapes of the interfaces are not smooth and do not look quite natural. The basic idea of our approach is to use the above rule (and the associated method based on the Voronoi diagram) to produce a first approximation of the topology and geometry of the cross-section. In the next section, we will see how it is possible to obtain smoother interfaces while keeping the same topology for the map.

3 Smoothing the Interfaces

Let us illustrate our goal on the simple example in Figure 4. The bold polygonal lines are the original data (portions of interfaces). The dashed polygonal curve is the interface between the formations A and B as reconstructed by the method above using the Voronoi diagram. The thin curve is the smooth interface we are looking for.

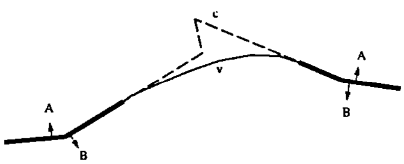

Figure 4: Two known contacts separating A from B (in bold line), the reconstructed interface (in dashed line) and the smoothed interface.

We proceed as follows: a given formation is bounded by a sequence of polygonal lines, that are alternatively original data (bold lines) and reconstructed portions (dashed lines). Each reconstructed portion c will be smoothed so that the smooth curve v satisfies the following requirements :

Condition 1 : the endpoints of v and c are the same;

Condition 2 : the tangent to v and to c are the same at their endpoints;

Condition 3 : the (possibly multi-connected) region between v and c does not contain any data points.

Condition 4 : variation of tension and bending is minimized along v.

The first two conditions will guarantee smooth transitions between the original data and the reconstructed portions. The third condition will guarantee that the new map is homotopic to the first one. The last condition insures the new curve v to be smooth.

To realize these conditions, we use deformable curves (or snakes) introduced by [KWT87] and also studied in more recent work by [NFGK]. A snake is considered as a dynamic system subject to internal forces and external constraints. The snake will deform until it reaches an equilibrium that corresponds to a local minimum of its energy.

3.1 Description of the Energies

Let $c(s)$ be a reconstructed polygonal chain joining two data points as produced by the Voronoi reconstruction. The chain is parameterized by its normalized arc length s (i.e. arc length divided by the total length of c). In order to smooth $c(s)$, we consider a deformable curve $v(s,t)$ parameterized by its normalized arc length s and by time t. We associate to the deformable curve an energy. The snake deforms itself in order to minimize its energy and reaches an equilibrium when its energy reaches a (local) minimum.

The total energy is the sum of an internal energy and of an external energy. The *internal energy* measures the resistance of the curve to tension (elasticity) and to bending (rigidity).

$$E_{intern} = E_{tension} + E_{bending}$$

$$E_{tension}(t) = \int_0^1 \frac{1}{2} w_1(s) \left| \frac{\partial v}{\partial s}(s,t) \right|^2 ds \qquad (1)$$

$$E_{bending}(t) = \int_0^1 \frac{1}{2} w_2(s) \left| \frac{\partial^2 v}{\partial^2 s}(s,t) \right|^2 ds \qquad (2)$$

In our experiments, the parameters w_1 and w_2 are kept constant along the whole curve and equal to 1.

The *external energy* E_{extern} measures the distance between $v(s,t)$ and $c(s)$. When this energy decreases, v gets closer to c.

$$E_{extern}(t) = \int_0^1 \frac{1}{2} w_3(s) D(v(s,t), c(s)) ds \qquad (3)$$

$D(v(s,t), c(s))$ denotes the distance between the two points of the curves v and c that have the same normalized arc length s. In practice, the distance D is evaluated at some discretized vertices of the curves and the integral is replaced by a sum (see 3.2).

If we increase w_3, the importance of E_{extern} with respect to E_{intern} grows, and the curve $v(s,t)$ of minimal energy gets closer to $c(s)$ (see Figure 5). This will be used to insure that condition 3 is satisfied.

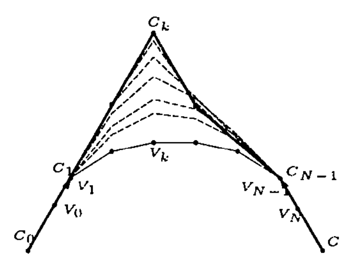

Figure 5: Evolution of $v(s)$ when increasing the parameter w_3.

3.2 Minimizing the Energy

Minimizing the energy of v amounts to solve the Euler differential equation [NFGK]:

$$-w_1 \frac{\partial^2}{\partial s^2}(v(s)) + w_2 \frac{\partial^4}{\partial s^4}(v(s)) = -\frac{\partial}{\partial s}(D(v,c)^2) \qquad (4)$$

Let $v(s,t) = (x(s,t), y(s,t))$ et $c(s) = (p(s), q(s))$. Equation 4 becomes :

$$
\begin{cases}
-w_1 \dfrac{\partial^2}{\partial s^2}(x) + w_2 \dfrac{\partial^4}{\partial s^4}(x) = -w_3 \dfrac{\partial}{\partial s}(\dfrac{1}{2}(x-p)^2) \\
-w_1 \dfrac{\partial^2}{\partial s^2}(y) + w_2 \dfrac{\partial^4}{\partial s^4}(y) = -w_3 \dfrac{\partial}{\partial s}(\dfrac{1}{2}(y-q)^2)
\end{cases}
\quad (5)
$$

In order to solve this differential equation, we discretize the curves c and v by taking $N+1$ regularly spaced points $v_i = (x_i, y_i)_{i=0}^{N}$ on v and $N+1$ regularly spaced points $c_i = (p_i, q_i)_{i=0}^{N}$ on c.

Spatial derivatives are approximated by finite differences :

$$\frac{\partial}{\partial s}(x_i) = x_i - x_{i-1} \text{ (first derivative)}$$

$$\frac{\partial^2}{\partial s^2}(x_i) = x_{i+1} - 2x_i + x_{i-1} \text{ (second derivative)}$$

The system above then becomes a linear system of $N+1$ equations in x_i and $N+1$ equations in y_i :

- for $i \in [2..N-2]$:

$$ax_{i-2} + bx_{i-1} + cx_i + bx_{i+1} + ax_{i+2} = -w_3 p_i$$
$$ay_{i-2} + by_{i-1} + cy_i + by_{i+1} + ay_{i+2} = -w_3 q_i$$

with
$$
\begin{cases}
a = w_2 \\
b = -4w_2 - w_1 \\
c = 6w_2 + 2w_1 - w_3
\end{cases}
\quad (6)
$$

- and for $i \in \{0, 1, N-1, N\}$, the initial conditions give us:

$$x_1 = p_1 \text{ et } y_1 = q_1 \text{ (same endpoints)}$$
$$x_1 - x_0 = p'_1 \text{ et } y_1 - y_0 = q'_1 \text{ (same tangents)}$$

Minimizing the energy is now reduced to solving this linear system.

3.3 Ensuring Condition 3

We initialize the external energy of v with $w_3 = 0$, and therefore we obtain a smoothed curve that satisfies Conditions 1, 2 and 4. However, Condition 3 is not necessarily satisfied as is shown in Figure 6 : differently from c (thin line), the smoothed curve v (dashed line) does not separate all the points labeled A from those labeled B.

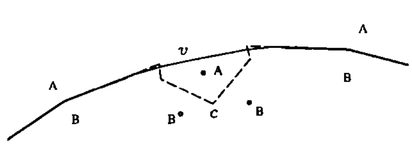

Figure 6: The region between c (in dashed line) and v (in thin line) contains a data point.

In order to ensure that Condition 3 is satisfied, we iteratively increase the parameter w_3 until all the data points are correctly separated. Let us explain in more detail how this is implemented. Since c is composed of Voronoi edges, its vertices are the centers of empty disks circumscribing some triangles of the Delaunay triangulation of the data points (see Figure 7). Let T be the set of these triangles and U the union of their circumscribing disks. As c is contained in U, Condition 3 will be satisfied if v is also contained in U. If this is not true, the parameter w_3 is increased by a fixed amount. The procedure is repeated until v meets the requirement (see Figure 8).

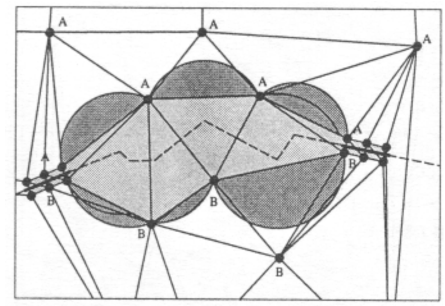

Figure 7: The union of Delaunay circumdisks.

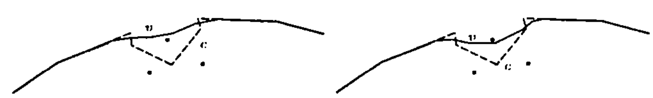

Figure 8: snake evolution with $w_3 = 0.1$ and $w_3 = 0.4$

4 Faults

Faults are accidents due to differential underground motions: geological formations are shifted along a fault as shown in Figure 9.

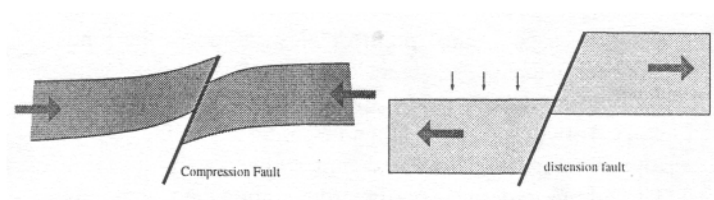

Figure 9: Example of faults.

In our method, faults are considered in a second stage, once the formation has already been reconstructed and smoothed. We then take them into account as described below, and update the formation.

For each reconstructed interface, we search all its intersections with the fault. If there is no intersection, the interface is not changed. If there is one point of intersection, we cut the interface at the intersection point leading to two pieces. We then smooth independently the two pieces. The only difference between this smoothing and the one in Section 3 is that now the end point which is on the fault is not fixed but can move along the fault. An example is shown in Figure 10.

The input data (portions of an interface and a fault) are in bold line and the reconstructed interface is in dashed line. The two pieces $I_0 I_1$ and $I_2 I_3$ of the interface after the insertion of the fault are in thin line.

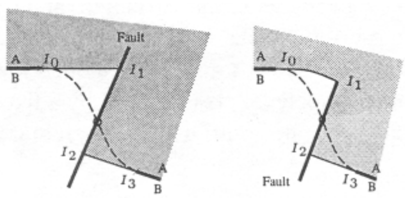

Figure 10: Insertion of a fault.

If there are more than one intersection with the fault, we cut the interface at each of the intersection points. This leads to $k+1$ pieces of interface. The first and the last ones are smoothed as in the previous case. The $k-1$ other pieces are interpolated (see Figure 11).

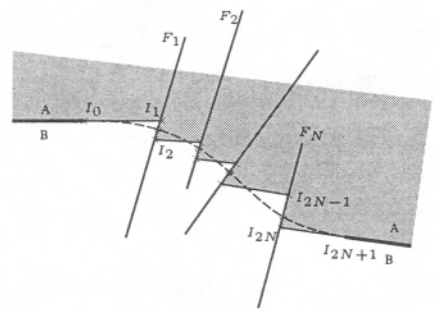

Figure 11: Insertion of a set of faults

5 Global Reconstruction

5.1 Order of Reconstruction

Smoothing is not enough to produce interfaces that are meaningful from a geological point of view. Indeed, the junctions between the interfaces are "T-junctions" that obey some syntactic rules that must be respected (see [PS95, SSMW93, Fle92]). If two interfaces S_a and S_b intersect, either S_a interrupts S_b (Figure 12 left) or S_b interrupts S_a.

This allows to define a partial order among the formations. A formation A will be said to precede a formation B if the interface between A and B is interrupting another interface bounding B. On the left of Figure 12, A precedes B and C; on the right, C precedes A and B.

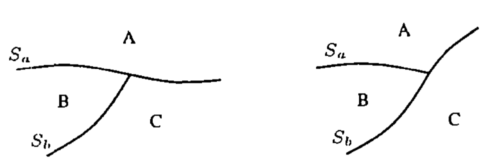

Figure 12: S_a interrupts S_b - S_b interrupts S_a.

Our algorithm will process the formations in this order, which will result in a correct reconstruction of the junctions between the interfaces. On the other hand, ignoring this order or using a bad order may result in a wrong reconstruction. This is illustrated in Figure 13. For the same initial data (in bold), we can obtain three different results, depending whether we start the reconstruction with B, A or C (from left to right).

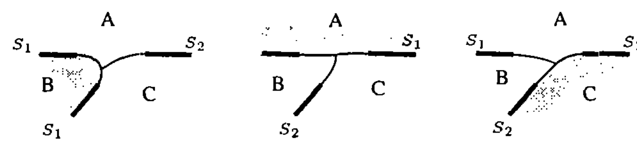

Figure 13: Reconstructions associated to different orders.

Let us consider now a more realistic example in Figure 14. $F_0, \ldots, F_7$ are eight formations bounded by the interfaces $S_0, \ldots, S_6$.

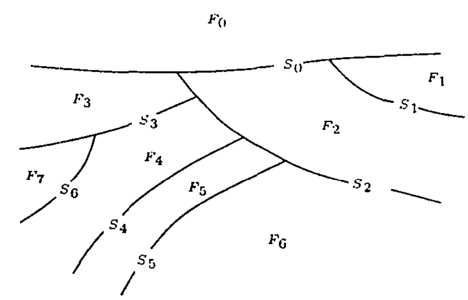

Figure 14: Example of a geological section.

We can represent the partial order relations between the formations as a tree : A formation F_j is a child of a formation F_i iff F_i precedes F_j. Figure 15 shows the tree corresponding to Figure 14.

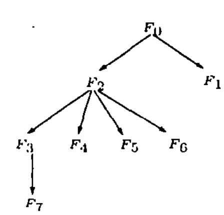

Figure 15: The tree representing the order relations between the formations of the previous figure. An arrow from F_i to F_j means that F_i precedes F_j.

Formations that are children of a given node in the tree are not sorted by the order just defined. Such formations are called *conformable*. Processing the conformable formations in an appropriate order is also important. Indeed, if formation A is documented by more data than a formation B, it is better to reconstruct A before B. This is illustrated in Figure 16, where formations A, B, C, D are more and more distant from the topographic surface and thus less and less documented. If we reconstruct formation D first, this formation appears to be erroneously divided in two disconnected parts and not the one expected (left part of the figure). On the other hand, if we process the formations in the order A, B, C, D, E, we obtain the correct result (right part of the figure). If we only consider the case of moderately deformed terrains where the geological formations lie in a normal order, the younger are closer to the topographic surface. Assuming in addition that the geological formations considered are less and less documented as they are deeper-seated, we therefore define a *total* order on the formations which, in addition to the previous partial order, sorts the children of each node in the tree according to their age. Our algorithm assumes that this order is known and given as part of the input.

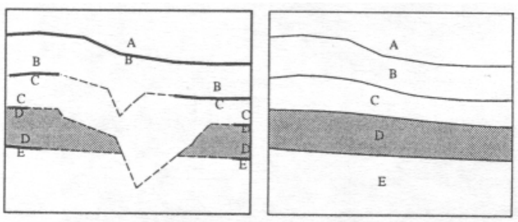

Figure 16: Importance of the order of reconstruction for comformable formations.

5.2 Incremental Reconstruction

We will now reconstruct the formations one by one in the order just defined. The reconstruction of a formation F consists of several steps. First, we reconstruct F ignoring the data points associated to the (already reconstructed) formations preceding F. The portions of the interfaces between F and the non yet reconstructed formations are then smoothed. Then we shift F along the faults that possibly intersect F. Finally, we remove the portions of F that lie inside the preceding formations.

We sum up the overall reconstruction algorithm below :

input : M a discrete set of colored data points
$C = \{1, \ldots, c\}$ the set of sorted colors
M_i the set of data points of color i
a set of faults L_i, $i = 1, \ldots, l$

output : the reconstructed formations $F_1, \ldots, F_c$

for i = 1 **to** c {

1. compute the Voronoi diagram of $\bigcup_{j=i}^{c} M_j$;
2. compute the union F_i of the cells of color i ;
3. smooth the interfaces between F_i and the F_j, $j > i$;
4. insert the faults $L_1, \ldots L_l$ and update F_i ;
5. remove the portions of F_i that are included in one of the F_k, $k < i$ }

The algorithm is illustrated on the example shown in Figure 17. Figures 18-23 show the different steps in the reconstruction of formation B, C and D. Figure 23 (right) shows the final result.

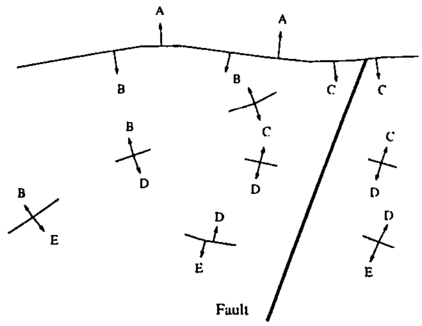

Figure 17: Initial data.

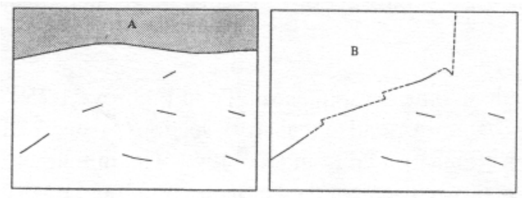

Figure 18: Formation A is known - Reconstruction of Formation B, Steps 1 and 2.

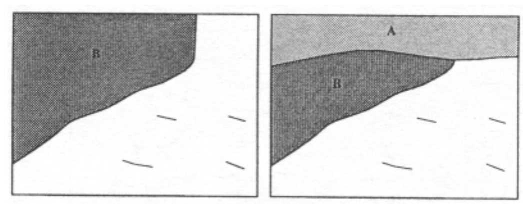

Figure 19: Reconstruction of object B, Steps 3-5.

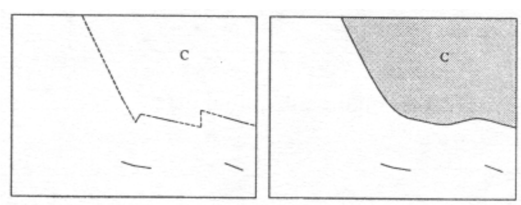

Figure 20: Reconstruction of Formation C, Steps 1-3.

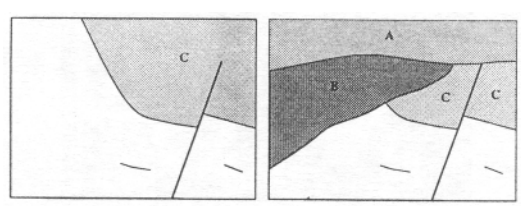

Figure 21: Reconstruction of Formation C, Steps 4 and 5.

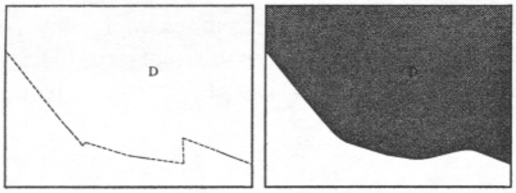

Figure 22: Reconstruction of Formation D, Steps 1-3.

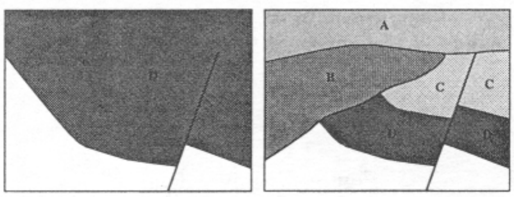

Figure 23: Reconstruction of Formation D, Steps 4 and 5.

The three other examples shown in Figures 24-25, 26-27 and 28-29, correspond to real cross-sections given by BRGM (French Research Group in Geology). The initial heterogeneous data are discretized (see Section2.2) and finally consist of 346 points in the example of Figure 24, 282 points in the example of Figure 26, and 194 points in the example of Figure 28. The reconstruction time for these instances takes about one second on a Sun Sparc 5.

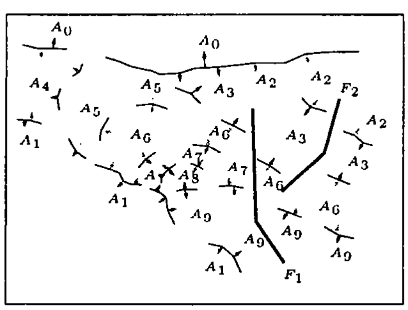

Figure 24: Initial data.

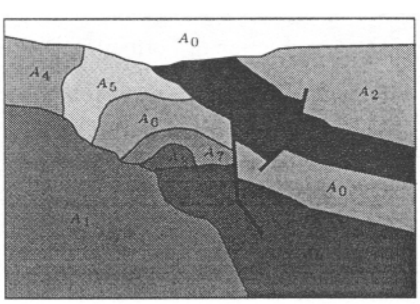

Figure 25: Reconstructed section.

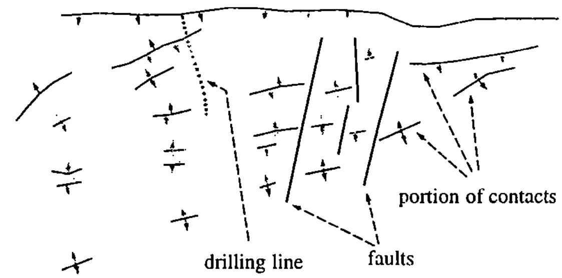

Figure 26: Initial data.

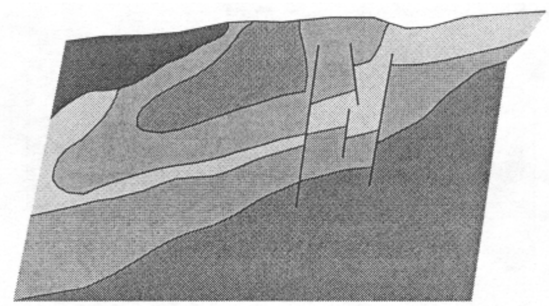

Figure 27: Final reconstruction.

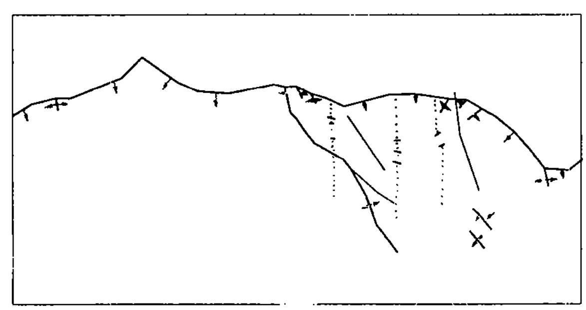

Figure 28: Initial data.

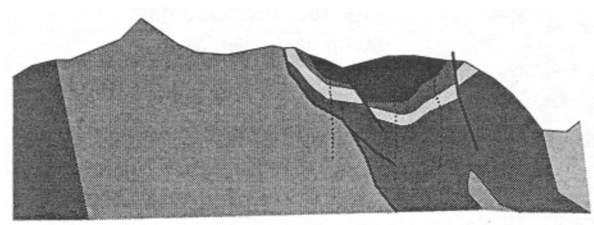

Figure 29: Final reconstruction.

6 Conclusion

We have presented a method to reconstruct a map of 2D geological sections from sparse and heterogeneous data. Our method automatically subdivides the underground into formations and produces a *volume* based representation of the underground. It has run successfully on many examples. Several extensions will be pursued.

First, we have used the usual Euclidean Voronoi diagram. However, if additional knowledge on the orientation of the strata is known at some points, we can use this information to adapt locally the metric. For instance, in Figure 16, it would be more appropriate to use a metric that reflects the horizontal orientation of the strata. This can be obtained by using a metric whose unit ball is a horizontal ellipsis instead of a circle. Voronoi diagrams can be computed for such more general metrics and the rest of the method would remain unchanged.

We are currently extending the method to 3D reconstruction. This can be done without major changes since the tools used in this paper can be extended in 3-space. Usually, the input consists of a set of 2D sections, typically along two orthogonal directions. We can first use the above method to reconstruct the sections and then extend the method to reconstruct the underground between the sections. Results will appear in a companion paper.

Acknowledgments

A. Guillen from BRGM and M. Perrin from the Ecole des Mines de Paris are gratefully acknowledged for posing this problem, giving us data and for several discussions that helped improving the method.

References

[Aur91] F. Aurenhammer. Voronoi diagrams: A survey of a fundamental geometric data structure. *ACM Comput. Surv.*, 23:345–405, 1991.

[Boi88] J.-D. Boissonnat. Shape reconstruction from planar cross-sections. *Comput. Vision Graph. Image Process.*, 44(1):1–29, October 1988.

[BT93] J.-D. Boissonnat and M. Teillaud. On the randomized construction of the Delaunay tree. *Theoret. Comput. Sci.*, 112:339–354, 1993.

[ET92] H. Edelsbrunner and T. S. Tan. An upper bound for conforming Delaunay triangulations. In *Proc. 8th Annu. ACM Sympos. Comput. Geom.*, pages 53–62, 1992.

[Fle92] Douglas M. Flewelling. Constructing geological cross sections with a chronology of geologic events. *5th Int'l Sym. on Spatial Data Handling*, pages 189–198, 1992.

[KWT87] M. Kass, A. Whitkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–331, 1987.

[Mal89] J.L. Mallet. Discrete smooth interpolation. *ACM Transactions on Graphics*, 8(2):121–144, 1989.

[May93] Raphaël Mayoraz. *Modélisation et Visualisation infographiques tridimensionnelles de structures tridimensionnelles de structures et propriétés géologiques.* Thèse de doctorat en sciences, Ecole Polytechnique Fédérale de Lausanne, Suisse, 1993.

[NFGK] W. Neuenschwander, P. Fua, G.Székely, and O. Kubler. Using boundary conditions to improve snake convergence. *Technical report, Communication Technology Laboratory, Swiss Federal Institute of Technology, Zurich, 1994.*

[PS95] M. Perrin and O. Stab. Propositions concernant la syntaxe des scènes géologiques. *ENSMP, rapport inédit.*, 1995.

[RHC+92] Pflug R., Klein H., Ramshorn Ch., Genter M., and Stärk A. 3d visualization of geologic structures and processes. *Computer Graphics in Geology, Lect. Notes in Geol. 41, Springer*, pages 28–39, 1992.

[Sei94] R. Seidel. The nature and meaning of perturbations in geometric computing. In *Proc. 11th Sympos. Theoret. Aspects Comput. Sci. (STACS)*, volume 775 of *Lecture Notes in Computer Science*, pages 3–17. Springer-Verlag, 1994.

[SSMW93] Masanori Sakamoto, Kiyoji Shiono, Shinji Masumoto, and Kiyoshi Wadatsumi. A computerized geologic mapping system based on logical models of geologic structures. *Nonrenewable Ressources*, 2(2), 1993.

[Yap87] C. K. Yap. An $O(n \log n)$ algorithm for the Voronoi diagram of a set of simple curve segments. *Discrete Comput. Geom.*, 2:365–393, 1987.