

LEVEL SET DRIVEN SMOOTH CURVE APPROXIMATION
FROM UNORGANIZED OR NOISY POINT SET *A. CLAISSE¹ AND P. FREY^{2,1}

Abstract. In this paper, we propose a curve construction method for a non uniform point data set based on a minimal curve approximation model. Numerically, the level set method is used for curve reconstruction. We represent the shape of the curve through its distance function and formulate curve reconstruction as a constrained minimization problem. We solve the minimization problem on a highly anisotropic triangulation to improve the accuracy of the numerical scheme. This method can handle complex geometries and deal with arbitrary topologies as well as with noisy data sets. Several numerical examples are provided to show the efficiency of the proposed approach.

Résumé. Dans ce papier, on propose un modèle de courbe d'approximation minimale pour construire une courbe à partir d'un nuage de points. Numériquement, la reconstruction de la courbe s'appuie sur une formulation de type ligne de niveau. On représente la forme de la courbe par sa fonction distance aux points et on exprime ce problème comme un problème de minimisation. Ce dernier est résolu sur une triangulation anisotrope qui permet d'améliorer la précision du schéma numérique. Cette méthode permet de traiter des géométries complexes et des topologies quelconques ainsi que des données bruitées. Des exemples de reconstruction sont proposés pour montrer l'efficacité de cette approche.

1. INTRODUCTION

The problem of reconstructing a regular curve from an unorganized point data set remains an important and an active area of research, despite numerous papers recently published on this topic [39, 47]. With the recent advent of sensing and scanning devices, large-scale points sampled curve and surface reconstruction algorithms have been receiving a great deal of attention in geometric modelling and reverse engineering as well as in computational workflows. There is a crucial need for a high-quality robust and efficient shape reconstruction technique for an arbitrary data set. The challenge for curve reconstruction is then to design versatile methods to handle a wide range of shapes. Furthermore, an adequate procedure should have a representation that is good for static approximation but also for dynamic deformations of the manifold.

The interpolation (resp. approximation) problem we consider can be loosely stated as follows: *Given a set of points V which are sampled from a regular curve in \mathbb{R}^2 , construct a smooth curve Γ so that all points of V lie on (resp. close to) Γ .* This formulation is obviously not very straight and since it is context dependent, it may lead to twofold interpretation. On the one hand, *data analysis* would consider that the shape of the manifold from which the data has been sampled is not known and then the aim is to find a “reasonable” solution among all possible. On the other hand, *curve reconstruction* considers that the original curve is known and that this knowledge can be suitably used to drive the algorithms.

* The first author is supported by a research grant from the Région Ile de France.

¹ UPMC Univ Paris 06, UMR 7598, Laboratoire J.-L. Lions, F-75005 Paris, France. e-mail: claisse@ann.jussieu.fr

² Universidad de Chile, UMI 2807, Centro de Modelamiento Matemático, Santiago, Chile. e-mail: frey@ann.jussieu.fr

We assume in this paper that the points in the data set V are sampled from an actual curve, possibly with noise or uncertainty in the data, although the geometry and the topology of the real shape are not known *a priori* and can be arbitrarily complex. This also means that no connection or ordering information among the data points is known. A proper reconstruction is usually possible if the sampling of the points is “sufficient”. However, finding a sufficient sampling condition is a fairly difficult problem and as such was largely neglected, if not simply ignored, in most research work up to now, at the noticeable exceptions of [7, 8].

Mathematically, the interpolation or approximation curve reconstruction as stated here is an ill-posed problem for which there is no unique solution [30]. In general, the solutions to this problem can be classified into two broad categories: depending on whether the reconstructed manifold has an explicit or an implicit representation. Explicit representations (including parametric curves and discretized curves or meshes) prescribe the exact location of the curve while implicit representations consider the curve as a specific iso-contour of a real valued function. An explicit representations present the advantage of speed and can easily handle sharp features, however it faces difficulties in tracking large deformations or processing topology changes. Recently, a lot of attention has been targeted to implicit representations. The computational cost is usually high for large data sets since the construction is global (*i.e.* requires solving a large linear system), but they offer the advantage to be almost insensitive to topological complexity and requires a simple data structure. However, they have difficulties in dealing with sharp features. Nonetheless, partial differential equation (PDE) driven approaches to solve level set problems, introduced by [38, 46], have now become a robust method to handle dynamically evolving interfaces. The algorithm starts from an initial curve (e.g. enclosing the point data set) and deforms it toward the data set by solving a PDE advection type problem until some stopping criterion is met. However, most of these approaches are not suitable for processing noisy data.

1.1. Our approach

We propose a PDE-driven, level set method to reconstruct a regular curve shape from a set of unorganized points. In this approach, the solution curve Γ is represented as the zero level set $u(x) = 0$ of a signed distance function u and the problem is reformulated as a constrained energy minimization problem. The regularity of the curve Γ is controlled by a surface tension term related to the local curvature of the manifold. A level set method is used to solve the temporal evolution of an initial curve $\Gamma(t = 0)$ toward the boundary of the point set. Actually, the problem of propagating a curve is equivalent to the problem of updating the signed distance function. Furthermore, the variational problem is solved using first-order Lagrange \mathbb{P}_1 finite elements on anisotropic unstructured triangulations that allows to improve noticeably the accuracy of the solution. We propose a three-step algorithm to perform the curve reconstruction process:

- (1) *initialization stage*: at first, a scalar value $d(x, V)$ corresponding to its distance to the point set V is assigned to each mesh vertex x of a quasi-uniform anisotropic triangulation T_h , that is highly refined in the vicinity of the point cloud using the technique described in [20];
- (2) *evolutionary stage*: an initial regular implicit curve $\Gamma(t = 0)$ is progressively deformed by solving a level set based advection equation until it fits at best through the data set;
- (3) *discretization stage*: in the third phase, a piecewise affine approximation of the regular final curve is obtained that allows for the fast visualization of the numerical solution.

Important features of our approach are the following:

- We use a variational PDE-driven approach that relies on two terms: a fitting term involving the distance to the data set and a curvature-based surface tension term to control the regularity of the curve;
- Complex topology and noisy data set can be handled with no difficulty;
- We solve the level set based minimization problem on unstructured triangulations using efficient and accurate numerical schemes;
- Our method can be extended in a straightforward manner to higher space dimension.

1.2. Related work

As pointed out, numerous research works have been published recently on the topic of curve and surface reconstruction from unorganized point clouds. Computational geometry approaches based on Delaunay triangulations or Voronoi diagrams on the one hand and volumetric techniques relying on signed distance functions on the other hand compose the main categories of published work. In general, the type of representation considered is strongly related to the domain of application. For instance, triangulations, subdivision surfaces, discretized level sets, scalar spline functions, radial basis functions or point set surfaces have been investigated [5,6,11]. It is beyond the scope of this paper to review these approaches, but we refer the reader to the paper [27,35] for a survey of the topic of surface reconstruction. We retain here only the approaches based on implicit shape representations and level set methods, because of their ability to manage topological changes and to deal with a certain degree of uncertainty in the point data sets.

Volumetric approaches aim in general at defining a discrete signed distance function to the points of the data set. Then, a piecewise affine discretization of the zero level set is constructed using efficient and straightforward heuristic algorithms, e.g. Marching polyhedra [13,32]. In most methods, the data set is used to define a discrete signed distance function on Cartesian grids and denotes the zero level of this function as the reconstructed curve or surface [8,10,26]. This approach requires the knowledge of the orientation of the manifold to distinguish between its interior and its exterior and thus the estimation of the local normals or the tangent planes [16,41]. This is known to be one of the most critical stage in the reconstruction process. Methods based on signed distance functions have usually the drawback of resulting in manifolds with not the lowest possible genus or presenting severe topological artifacts (e.g. due to local inconsistencies). Unsigned distance functions can be introduced to overcome this problem since they do not carry orientation information [27].

Following [33,46], we consider the reconstruction problem on the continuous level, by developing a continuous model based on PDEs and differential geometry properties that involves only the unsigned distance function to the data set. The level set method is used as a convenient numerical technique to deform continuously an initial curve following the gradient descent of an energy functional. However, instead of solving the PDE on structured grids, we introduce here highly stretched elements in the vicinity of the data set to considerably improve the numerical approximation of the manifold at convergence of the numerical scheme.

The remainder of this paper is organized as follows. The variational minimization problem corresponding to the curve reconstruction model is described in Section 2. Section 3 review the main numerical issues faced in the reconstruction process. In Section 4, experimental results on synthetic and real data sets are provided to illustrate the efficiency and the salient features of our approach.

2. CURVE RECONSTRUCTION MODEL

Let V denote a general data set consisting a data points sampled from a regular planar curve. Define

$$d(x) = \text{dist}(x, V) = \min_{y \in V} \|x - y\|$$

to be the distance function to V . Typically, the deformation and the evolution of an initial arbitrary regular curve Γ_0 are governed by a time-dependent system of second-order partial differential equations with the following general form:

$$\frac{\partial \Gamma(t)}{\partial t} = F(t, x, \kappa, f) \vec{n}(t, x), \quad (1)$$

where F is a velocity function, t denotes the time parameter, κ is the local curvature at point x and f is an external force; \vec{n} is an unit direction vector, e.g. the curve outward normal vector. Here $\Gamma(0) = \Gamma_0$ is the initial curve. Usually, Equation (1) can be either supplied by the user or is obtained by the Euler-Lagrange equation of some energy functional based on the calculus of variations [46]. In the latter option, the purpose is to find a local minimizer of the energy functional that corresponds to a minimal curve attached to the data set.

2.1. The level set formulation

Since the topology and the geometry of the final curve are not known *a priori*, we use a level set method for curve evolution according to [42]. In this approach, an implicit curve Γ , which is usually the zero isocontour of a suitable scalar function u , is evolved according to a law of motion similar to that prescribed by Equation (1) to the steady state. Interestingly, instead of explicitly tracking the curve at each time step, it is implicitly captured by solving the PDE for the level set function on a grid or a triangulation encompassing the set V . Hence, topological changes are easy to handle and do not need special care. Numerically, the level set formulation can be restricted to a *narrow band* surrounding the zero level set [2, 40]. But, we will see in Section 3 how we circumvented this formulation by introducing anisotropic triangulations.

The first step in the level set formulation of our reconstruction problem consists in embedding the curve Γ in a computational planar domain: we consider $\Gamma = \{x, u(x) = 0\}$ as the zero isocontour of a scalar function $u(x)$ defined on $\Omega \subset \mathbb{R}^2$. This level set function u is defined to be a smooth function that is positive in one side of the closed curve Γ and negative in the other. Now, as will be explained hereafter, geometric differential properties of the curve Γ can be computed using function u . The second step consists in introducing the time dependence allowing the motion of the curve Γ . Following [46], we derive a time-dependent PDE for the level set function u such that the curve Γ and the zero level set obey the same motion law: $\Gamma(t) = \{x, u(t, x) = 0\}$. This is equivalent to imposing

$$\forall x \in \Gamma(t), \quad u(t, x) = u(t, \Gamma(t)) = 0.$$

By differentiating this equation with respect to the time variable, we obtain:

$$\frac{du}{dt}(t, \Gamma(t)) = \frac{\partial u}{\partial t}(t, x) + \frac{d\Gamma(t)}{dt} \cdot \nabla u(t, x) = 0, \quad (2)$$

where the term $\frac{d\Gamma(t)}{dt}$ can be replaced by the velocity value of x on $\Gamma(t)$.

In our method, the motion of the curve $\Gamma(t)$ is purely geometric: its motion law is only related to the geometric features of the moving curve. Therefore, it seems pertinent to assign the same velocity law to all isolevel sets of the level set function u , leading to a so-called morphological multiscale PDE [4] of the generic form:

$$\frac{\partial u}{\partial t}(t, x) = \beta(t, \kappa(u(t, x))) |\nabla u(t, x)|,$$

where β is an arbitrary nondecreasing real function and $\kappa(u)$ is the curvature of the level set of u assing through x . In our method, this geometric motion of Γ is expressed by a Hamilton-Jacobi type of equation, for all $(t, x) \in \mathbb{R}_+ \times \mathbb{R}^2$:

$$\frac{\partial u}{\partial t}(t, x) + v_n(t, x) |\nabla u(t, x)| = 0, \quad (3)$$

where $v_n(t, x) = \frac{d\Gamma(t)}{dt} \cdot \vec{n}(u(t, x)) = \frac{d\Gamma(t)}{dt} \cdot \frac{\nabla u(t, x)}{|\nabla u(t, x)|}$ denotes the velocity field in the direction normal to the level set passing through x , \vec{n} is the unit outward normal to the level set curve at x .

We mentioned that the velocity $v_n(t, x)$ depends usually on the geometric properties of the curve $\Gamma(t)$, but it can be also related to the physical parameters of the problem. Mathematically, the viscosity solutions have been introduced to give a physical sense to the solution of the Hamilton-Jacobi equation. It is well known that singularities are often encountered in solving this equation and thus appropriate techniques must be developed to select the unique viscosity solution [9, 15]. Numerically, this solution is difficult to obtain, mainly because of the existence of singularities and, in general, requires implicit schemes. Here, we are looking for a viscosity solution that coincides with the curve $\Gamma(t)$.

2.2. The curve evolution model

The partial differential equation we used to solve our problem is the general weighted mean curvature flow:

$$\frac{\partial u}{\partial t}(t, x) = \alpha d(x) + d(x) \kappa(u(t, x)), \quad \Gamma(0) = \Gamma_0, \quad (4)$$

where $d(x)$ is the distance function to the point set, α is a real parameter and the curvature is defined as:

$$\kappa(u) = \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = |\nabla u|^{-1} \left(\Delta u - D^2 u \left(\frac{\nabla u}{|\nabla u|}, \frac{\nabla u}{|\nabla u|} \right) \right), \quad (5)$$

this term can be interpreted as the curvature of the level curve $\{y, u(y) = u(x)\}$ passing through x .

It is easy to see that our model is composed of two terms. The term $\alpha d(x)$ corresponds to the attraction by the distance field (*i.e.* a fitting term) and the term $d(x)\kappa(u)$ corresponds to a minimal surface regularization weighted by the distance function (*i.e.* a surface tension term). The parameter α balances the potential force and the surface tension. The nonlinear regularization due to surface tension combined with the attraction term provides more rigidity of the curve away from the point set and more flexibility close to the data set. Moreover, the surface tension term has a local impact as it is only related to the curve evolution and not to the data set.

2.3. The variational minimization problem

The steady state of Equation (4) corresponds to a minimum of the level set function u , when the distance function vanishes for all points of the curve $\Gamma(t)$. Hence, it yields the following elliptic equation:

$$-\operatorname{div} \left(\frac{\nabla u(t, x)}{|\nabla u(t, x)|} \right) = \alpha \quad \Leftrightarrow \quad A(u) = \alpha \quad (6)$$

where A is a second-order nonlinear elliptic operator. The calculus of variations allows us to consider this operator as the derivative of an energy functional $E : A(u) = E'(u)$ [22]. Hence, solving our problem simply requires finding a function u which minimizes the functional E . To this end, we introduce a Lagrangian operator $L : \mathbb{R}^2 \times \mathbb{R} \times \bar{\Omega} \rightarrow \mathbb{R}$ such that: $L(p, z, x) = |p| - \alpha z$, and then we write:

$$E(u) = \int_{\Omega} L(p(u(x)), z(x), x) dx = \int_{\Omega} (|p(u(x))| - \alpha z(x)) dx$$

The Euler-Lagrange equation associated to this functional E is the following:

$$\frac{\partial L}{\partial z}(p, z, x) - \operatorname{div}_x(\nabla_p L(p, z, x)) = 0.$$

Taking $p(x) = \nabla u(t, x)$ and $z(x) = u(t, x)$ leads to retrieve Equation (6) and consequently, the energy functional E to minimize is:

$$E(u(x)) = \int_{\Omega} (|\nabla u(x)| - \alpha u(x)) dx. \quad (7)$$

Actually, the integration domain Ω can be any open set that contains the zero level set of the function u . The curve $\Gamma(t)$ at steady state is given by the solution of the following minimization problem:

$$u = \min_v E(v). \quad (8)$$

The above discussion shows that if u is a solution of the elliptic problem (6), u is also a solution of the minimization problem (8) and conversely. In this respect, the curve $\Gamma(t)$ at steady state is a *minimal approximation curve*.

The minimal curve model can handle complex topologies and construct a regular curve that is smoother than piecewise affine discretization in planar domains. It can also deal with noisy data sets, to some extent, by balancing the attraction and the surface tension terms.

3. NUMERICAL ISSUES

There are several key numerical features in our curve reconstruction approach. First, we compute the distance field to an arbitrary data set on an unstructured triangulation. Second, we need to evaluate the

first and second derivatives of the level set function at the vertices of the triangulation. Third, we generate a highly anisotropic triangulation of the computational domain, to improve the accuracy of the numerical solution. Fourth, we solve the time dependent PDE for the level set function. And finally, we need to address the reconstruction of (un)signed distance functions as the gradient of the level set function may become distorted after a few time steps.

3.1. Distance function computation

The distance function $d(x)$ to an arbitrary data set V solves in principle the general Eikonal equation:

$$|\nabla u(x)| = 1, \quad u(x) = 0, \quad x \in V. \quad (9)$$

The characteristics lines of this PDE are straight lines along which the information propagates from the data set. In [46], an algorithm combining upwind differencing with Gauss-Seidel iterations of different sweeping order is used to solve Equation (9) on Cartesian grids. However, because we are dealing with an unstructured triangulation T_h , we have employed the simplest and most useful method that consists in computing the unsigned distance function as: $d(x) = \min_{y \in V} |x - y|$, for all vertices $x \in T_h$. Notice that this expensive computation is performed only at the beginning of the program, to initialize the distance function used in our model Equation (4). Moreover, since the triangulation T_h is highly anisotropic, the number of mesh vertices is quite small as will be shown in the examples. However, we have paid more attention on the procedure to reinitialize the level set function u with the signed distance function (see hereafter) as it is used much often in the numerical resolution scheme and we certainly avoided using the straightforward algorithm based on the discretized version of the equation: $d(x) = \min_{y \in V} |x - y| \operatorname{sgn}(u(x))$.

3.2. Curvature and derivative estimation

Our model Equation (4) involves the first (gradient) and second order (Hessian) derivatives of the level set function u . Hence, it is important for the efficiency and the robustness of the method to compute these derivatives as accurately as possible based on a discretized information associated with the mesh vertices. Furthermore, we need to compute the second derivatives of the unsigned distance function d in the vicinity of the data set V , in order to generate an adapted anisotropic triangulation T_h . Moreover, we would like to compute the second order derivatives of the level set function u at the mesh vertices. We have tested and used two options: a L^2 projection scheme (for the level set function) and a least-square approximation of the derivatives (for the distance function). Notice that both approaches can be easily extended to deal with three dimensional problems with simplicial triangulations.

Here, we suppose that the level set function u is known at the mesh vertices. In other words, we consider the approximation space V_h associated with the Lagrange \mathbb{P}_1 finite element: $V_h = \{v \in H^1(\Omega), v|_K \in \mathbb{P}_1\}$. Since the level set function u is considered the \mathbb{P}_1 discrete solution at each vertex of T_h , its gradient is a constant value on each element and is not continuous. This would prevent any computation of second order derivatives. The objective is to reconstruct a \mathbb{P}_1 gradient from the current level set function u .

3.2.1. A L^2 projection scheme

Setting the evaluation of the derivatives in the finite element context allows us to consider the L^2 interpolation operator Π_h . For all $v \in L^2(\Omega)$, $\Pi_h v \in L^2(\Omega)$ is such that:

$$\int_{\operatorname{Supp}(\varphi_j)} (\Pi_h v - v)(\omega) = 0, \quad \forall \omega \in \mathbb{P}_1 \quad \text{and} \quad (\Pi_h v)|_{\operatorname{Supp}(\varphi_j)} \in \mathbb{P}_1,$$

where $Supp(\varphi_j)$ denotes the support of each shape function φ_j . Considering the projection operator applied to $\nabla u \in L^2(\Omega)$, leads us to write, for all $\omega \in \mathbb{P}_0$:

$$\begin{aligned} \int_{Supp(\varphi_j)} (\Pi_h(\nabla u) - \nabla u)(\omega) = 0, &\iff \int_{Supp(\varphi_j)} \Pi_h(\nabla u) = \int_{Supp(\varphi_j)} \nabla u \\ &\iff |Supp(\varphi_j)| (\Pi_h(\nabla u))|_{Supp(\varphi_j)} = \sum_{K \in Supp(\varphi_j)} \int_K \nabla u \\ &\iff (\Pi_h(\nabla u))|_{Supp(\varphi_j)} = |Supp(\varphi_j)|^{-1} \sum_{K \in Supp(\varphi_j)} |K| \nabla(u|_K), \end{aligned}$$

where $|K|$ denotes the measure (area) of triangle $K \in T_h$.

Introducing the barycentric coordinates $(\omega_j)_{j=1,3}$ of the vertices x_j of each triangle $K \in T_h$ leads to write:

$$u|_K = \sum_{j=1}^3 u(x_j) \omega_j, \quad \text{and} \quad \nabla(u|_K) = \sum_{j=1}^3 u(x_j) \nabla \omega_j,$$

For every point $x \in K$, there exists a unique vector $(\omega_j(x))_{j=1,3}$ solution of the system:

$$x = \sum_{j=1}^3 a_j \omega_j(x), \quad \text{and} \quad \sum_{j=1}^3 \omega_j(x) = 1,$$

where the $(a_j)_{j=1,3}$ represent the vertices of element K . Indeed, this 3×3 linear system admits the following matrix M as associated matrix, where $(a_{i,j})_{i=1,2}$ denote the coordinates of point (a_j) :

$$M = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ 1 & 1 & 1 \end{pmatrix}$$

and this matrix M is invertible if and only if the simplex K is non degenerated. Having computed the coordinates ω_j , it is now possible to compute $\nabla u|_K$.

The Clément operator $\Pi_c : L^2 \rightarrow V_h$, defined for the vertices $a_j \in K$ as:

$$\Pi_c v = \sum_{j=1}^3 \Pi_h v(a_j) \varphi_j,$$

allows to reconstruct the \mathbb{P}_1 gradient of the level set function u as follows:

$$\nabla u = \Pi_c(\nabla u) := \left[\Pi_c \left(\frac{\partial u}{\partial x} \right), \Pi_c \left(\frac{\partial u}{\partial y} \right) \right].$$

Finally, the gradient of ω_i is obtained as:

$$\nabla \omega_1 = \begin{pmatrix} a_{2,2} - a_{2,3} \\ -(a_{1,2} - a_{1,3}) \end{pmatrix} \quad \nabla \omega_2 = \begin{pmatrix} a_{2,3} - a_{2,1} \\ -(a_{1,3} - a_{1,1}) \end{pmatrix} \quad \nabla \omega_3 = \begin{pmatrix} a_{2,1} - a_{2,2} \\ -(a_{1,1} - a_{1,2}) \end{pmatrix}.$$

Numerically, computing the \mathbb{P}_1 gradient of the level set function u at a vertex $a_j \in K$ requires identifying the set $B(a_j)$ of triangles that compose the support of the shape functions, *i.e.* all triangles sharing the vertex a_j . Actually, the numerical scheme is a weighted reconstruction scheme, where the weights are related to the areas of the triangles in $B(a_j)$.

After the reconstruction of the \mathbb{P}_1 gradients of the level set function u , the same L^2 projection procedure is applied to each component of the gradient vectors in order to reconstruct the \mathbb{P}_1 Hessian matrix H of u :

$H(u) = \nabla((\nabla u)^t)$, with for each component $(x_i)_{i=1,2}$:

$$\nabla \left(\Pi_c \left(\frac{\partial u}{\partial x_i} \right) \right) (x) = |Supp(\varphi_j)|^{-1} \sum_{K \in Supp(\varphi_j)} |K| \nabla \left(\left(\Pi_c \left(\frac{\partial u}{\partial x_i} \right) \right) \Big|_K \right).$$

The resulting algorithm is straightforward:

```

/* Compute the  $\mathbb{P}_1$  gradient of the level set function  $u$  */
for each vertex  $x \in T_h$  do
  construct the set  $B(x) = \{K \in T_h, x \in K\}$ 
  compute the  $\mathbb{P}_1$  gradient of  $u(x)$  as:

```

$$\nabla u(x) = \frac{\sum_{K \in B(x)} |K| \nabla u|_K}{|B(x)|}$$

```

end
/* Compute the  $\mathbb{P}_1$  Hessian matrix of the level set function  $u$  */
for each vertex  $x \in T_h$  do
  identify the set  $B(x) = \{K \in T_h, x \in K\}$ 
  compute the  $\mathbb{P}_1$  gradient of each component of vector  $\nabla u(x)$  as:

```

$$H(u)_i(x) = \frac{\sum_{K \in B(x)} |K| \nabla(\Pi_c \nabla u)_i|_K}{|B(x)|}$$

```

end

```

The local mean curvature term $\kappa(u)$ that appears in our model Equation (4) is computed using Equation (5) or, more simply, as the trace of the Hessian matrix.

3.2.2. A least squares approximation scheme

The previous approximation scheme for computing the first and second order derivatives of the level set function u worked fairly well in most applications. However, for generating an anisotropic metric tensor field suitable for mesh adaptation, we considered a least squares approximation scheme [23], that we briefly describe here.

Let x be a vertex of T_h and let consider the set $P(x)$ of all mesh vertices connected to x . We suppose that $Card(P(p)) \geq 3$, a reasonable hypothesis in two dimensions. Considering a second order Taylor expansion of the level set function u at a vertex $a_i \in P(x)$ leads to write:

$$\begin{aligned} u(a_i) &= u(x) + x\vec{a}_i \cdot \nabla u(x) + \frac{1}{2} \langle x\vec{a}_i, H(u)(x) x\vec{a}_i \rangle \\ \iff \frac{1}{2} \langle x\vec{a}_i, H(u)(x) x\vec{a}_i \rangle &= u(a_i) - u(x) - x\vec{a}_i \cdot \nabla u(x). \end{aligned}$$

This relation can be also developed as follows:

$$\frac{1}{2}(ax_i^2 + 2bx_iy_i + cz_i^2) = u(a_i) - u(x) - (\alpha x_i + \beta y_i), \quad (10)$$

where x_i, y_i denote the coordinates of vector $x\vec{a}_i$, α, β are the components of $\nabla u(x)$ and a, b, c are the components of the Hessian matrix $H(u)$. Writing this equation for all vertices in the set $P(x)$ leads to a usually over-determined system of the form:

$$AU = B, \quad \text{with } U^T = (a \quad b \quad c)$$

where A is a $(n \times 3)$ matrix function of (x_i, y_i) and B is a vector of size n given by the right-hand side of the Relation (10). This system is solved using a least-square approximation which consists in minimizing the distance between the vectors AU and B of \mathbb{R}^n by minimizing the square of the Euclidean norm of their difference, as follows:

$$”Find $U \in \mathbb{R}^3$, such that $\|AU - B\|^2 = \inf_{Y \in \mathbb{R}^3} \|AY - B\|^2.$ ”$$

It can be shown that the solution of this problem is the solution of the linear 3×3 system of normal equations [14]:

$$A^T A U = A^T B.$$

This resulting system is solved using a Gauss pivoting algorithm and the local mean curvature is computed as with the previous approximation scheme.

3.3. Anisotropic mesh generation

High-order schemes (ENO, WENO) have been proposed to resolve the curve evolution equation on structured meshes [28, 37]. However, their extension to arbitrary unstructured triangulations is difficult and only a few algorithms have been proposed [1]. In order to achieve a good numerical accuracy and to limit to some extent the diffusive effect of the numerical scheme, we heavily rely on anisotropic adapted triangulations.

Schematically, mesh adaptation consists in concentrating a large number of vertices in regions of large solution variations and in minimizing the number of vertices in other regions of the computational domain. Consequently, the number of nodes required to compute a solution with a desired accuracy can be reduced by a substantial amount, thus impacting favorably the computational cost of the simulation. While isotropic mesh adaptation only controls the element size, anisotropic adaptation adjusts the element size as well as the element shape and orientation using a metric tensor field to better match the solutions variations. The adaptation procedure involves an error estimate or error indicator to evaluate the solution variations. The theory of error estimates has been largely investigated these last years and has provided the concept of optimal triangles with respect to a metric tensor field based on the gradient and the Hessian of the solution (see [23, 24] for a survey).

The metric tensor defining the generic mesh element features is represented as an ellipsoid in which the element must fit. Hence, the notion of size, shape and orientation are related to its volume, the lengths of the semi-axes and its principal axes vectors, respectively. This metric tensor $M(x)$ is a symmetric positive definite matrix that is used to create a quasi-uniform mesh in the metric related to M . Actually, the volume of an element K is unitary:

$$\int_K \sqrt{\det(M(x))} dx = 1,$$

which corresponds to the discrete formulation:

$$|K| \sqrt{\det(M_K)} = 1$$

taking M_K as a mean value of $M(x)$ on K . The length of a segment or a curve $\gamma(t) : [0, 1] \rightarrow \mathbb{R}^2$ is defined as:

$$|\gamma_M| = \int_0^1 \sqrt{\langle \gamma'(t), M(x) \gamma'(t) \rangle} dt,$$

The spectral decomposition theorem allows to decompose M as:

$$M = R \Lambda R^t = \sum_{i=1}^2 \lambda_i \vec{e}_i \vec{e}_i^t,$$

where the normalized eigenvectors of M are the columns of the matrix $R = [\vec{e}_1, \vec{e}_2]$, Λ is the diagonal matrix of the eigenvalues $(\lambda_i)_{i=1,2}$.

In [20, 21], we proposed a method for defining the metric tensor to control the generation of anisotropic elements in the vicinity of an interface defined with a level set function. Here, we adapted this approach as follows. The main idea is to consider the local curvature of the zero level set of the distance function d to prescribe the local element size, the orientation of each element being chosen to coincide with the tangent and

the normal directions to the function d . This will automatically provide a refinement close to the data set V (where the distance function vanishes) and a desirable stretching of the element related to the curvature of the zero isoline. By doing so, we can prove that the Hausdorff distance between the zero isoline Σ of the distance function and a piecewise affine approximation Σ_h of this isoline is bounded.

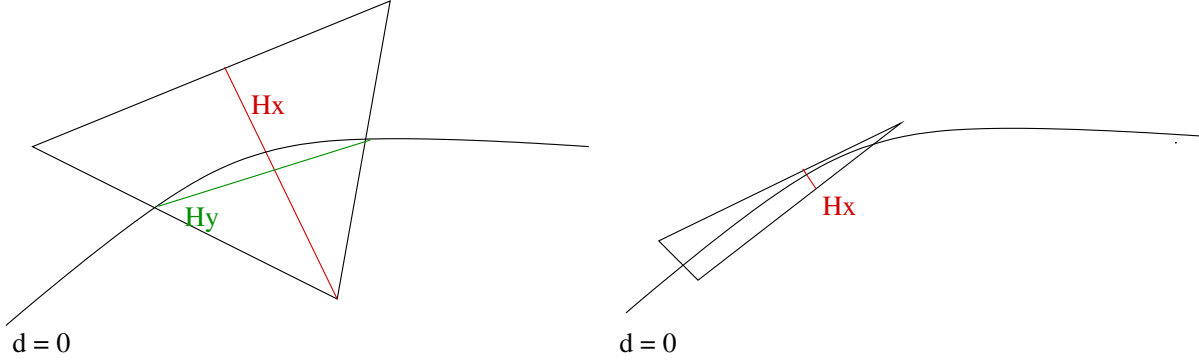


FIGURE 1. *Isotropic vs. anisotropic mesh adaptation in the vicinity of the zero level set.*

More precisely, here we define the local metric tensor at each vertex $x \in T_h$ as:

$$M(x) = \frac{\nabla d(x) \nabla d^t(x)}{h_{min}^2} + \frac{H(d)(x)}{\varepsilon},$$

where h_{min} represents the smallest (user-prescribed) element size in T_h , ε is a tolerance value such that $d(\Sigma, \Sigma_h) < \varepsilon$ and $H(d)$ is the Hessian matrix of the distance function at x . This metric tensor ensures that the mesh elements will be aligned with the tangent and the normal directions to the isoline Σ and the size $l(x)$ in the tangent direction will be related to the mean curvature by setting:

$$l(x) = \sqrt{\frac{\varepsilon}{\kappa(d)(x)}}.$$

Given an initial triangulation T_0 of the computational domain Ω , the triangulation is iteratively adapted to this metric tensor by using classical local mesh modification operations: edge split, edge flip, edge collapsing and vertex relocation [24]. In addition, a mesh gradation procedure is used to propagate the metric information inside the domain [12]. Figure 2 shows the difference between the isotropic and the anisotropic mesh refinement in the vicinity of the zero isoline Σ . The isotropic triangulation contains 41,127 vertices while the anisotropic triangulation contains 11,751 vertices corresponding to the same h_{min} and ε values.

3.4. Solving the PDE

Our numerical implementations of the numerical resolution of the nonlinear model Equation (4) are based on standard schemes for the level set method. We have first implemented a first order explicit scheme. A stable time explicit requires a restrictive time step, $\Delta t = O(h_{min}^2)$, where h_{min} is the minimal mesh element size. However, since we are working with minimal anisotropic triangulations (and not Cartesian grids), it seems worth to test first this option, that is also easy to implement.

3.4.1. A First order explicit scheme

We introduce the notations $u_i^n = u(x_i, t^n)$, $d_i^n = d(x_i, t^n)$ and $\kappa_i^n = \kappa(x_i, t^n)$. In this scheme, the time derivative is replaced by the classical differential quotient:

$$\frac{\partial u(x, t)}{\partial t} \approx \frac{u_i^{n+1} - u_i^n}{\Delta t}$$

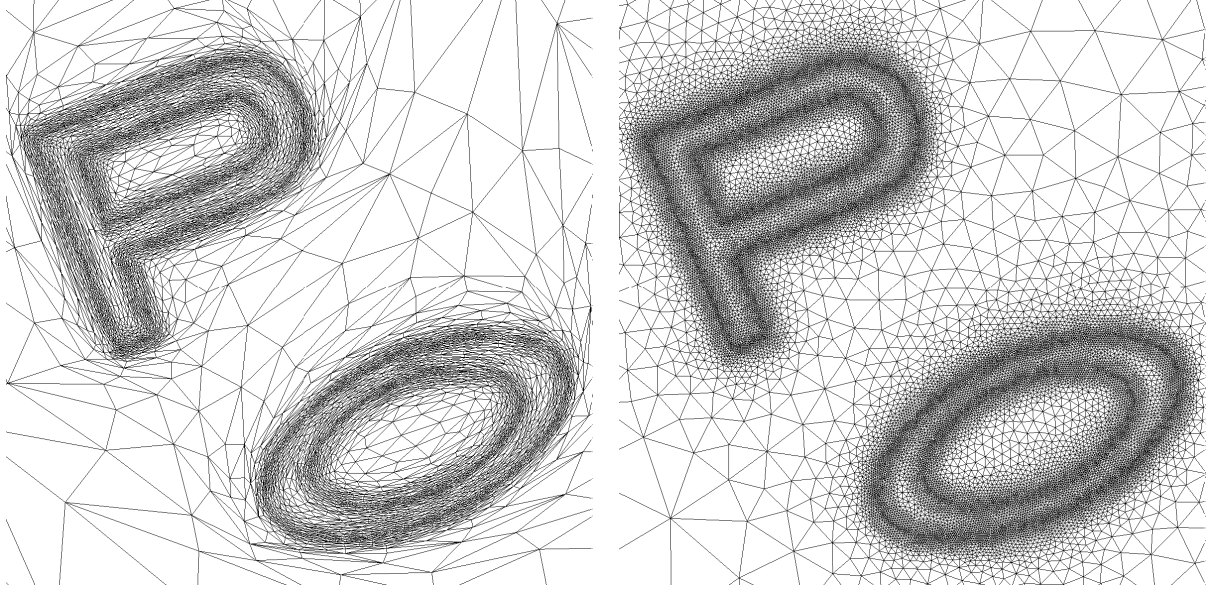


FIGURE 2. Impact of the anisotropic metric tensor (left) in mesh adaptation compared to isotropic refinement (right).

and our model Equation (4) is then solved by the explicit first order upwind scheme:

$$u_i^{n+1} = u_i^n + \Delta t (d_i^n (\kappa_i^n + \alpha))$$

with $\kappa_i^n = \operatorname{div} \frac{\nabla u_i^n}{|\nabla u_i^n|}$. Hence, knowing the values of the level set function u at the vertices of the triangulation at step t leads to easily obtain the values of u at time $t + \Delta t$ using a simple loop over the mesh vertices. As pointed out however, the simplicity of this scheme is balanced by the restrictive CFL condition on the time stepping. To overcome this problem, we have started investigating a semi-implicit time-stepping scheme on unstructured triangulations.

3.4.2. A semi-implicit scheme

The nonlinear curve evolution equation can be written under the following form:

$$\frac{\partial u(t, x)}{\partial t} = d(x)(\alpha + \kappa(u(t, x))) = d(x) \left(\alpha + \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) \right).$$

It is classical to explicit the non linear term of the equation as follows:

$$u_i^{n+1} - d_i \Delta t \nabla \cdot \left(\frac{\nabla u_i^{n+1}}{|\nabla u_i^{n+1}|} \right) = d_i \Delta t \alpha + u_i^n$$

Solving this equation requires solving a linear system. This is achieved using a GMRES algorithm. In this approach, the time step Δt can be chosen on the order of $O(h_{min})$, as no CFL condition applies.

For both schemes, a stopping criterion has been added to avoid unnecessary expensive advectations of the level set when the curve $\Gamma(t)$ is already close to the data set. To this end, we introduce the following stopping test:

$$\max_{x \in V} |(\Pi_h u)(x)| < \varepsilon \quad (11)$$

where ε is a user-specified tolerance value and $(\Pi_h u)(x)$ is obtained using the L^2 projection scheme described above.

3.5. Renormalization algorithm

The distance function d enjoys the property: $|\nabla d| = 1$, which notably simplifies the computation of the normal vector and the mean curvature. Although any level set function can be used theoretically, in practice a signed distance function is preferred to avoid inaccuracy in numerical computations and stiffness in the resolution. Our level set function u is initialized with a signed distance function corresponding the distance of the mesh vertices to an implicitly defined simple curve geometry (e.g. a circle). Unfortunately, all motion fields, but a constant velocity field, do not preserve this property. This behavior is also amplified by using a first order scheme for solving the PDE; it introduces a numerical dissipation effect resulting in $|\nabla u| \gg 1$ after a few steps. To recover it, a renormalization stage is introduced every (user-specified) n steps of the solver.

Typically, a renormalization procedure consists in solving the following equation:

$$\partial_t u - S(u)(1 - |\nabla u|) = 0$$

where $S(u)$ is a sign function taken as 0 along the zero isocontour of the function. Sethian [42] suggested to use a Fast-Marching method to evolve a front at constant speed initiated from the interface $\Gamma(t)$. The algorithm can be seen as an extension of Dijkstra’s algorithm [17] to compute shortest paths between points in the domain. It consists in solving the following equation in each element:

$$\sum_{j=1}^3 \|\nabla \varphi_j u(a_j)\|^2 = 1 \tag{12}$$

where the φ_j are the Lagrange shape functions and $u(a_j)$ is the value of the level set function u at the vertices a_j of the element. The following algorithm correspond to a first order scheme when an affine approximation is used to solve this equation. Suppose we have three groups of vertices : (A)cccepted, (C)onsidered and (F)ar.

- (1) Compute the distance values for the vertices of the elements intersected by $\Gamma(t)$, now considered as (A),
- (2) put the neighbors of all these points in the group (C) and compute the distance values using Equation (12),
- (3) loop over the set (C) until it is empty:
 - take $x \in (C)$ with the smallest absolute value, move it in set (A),
 - transfer all neighbors of x from (F) to (C),
 - update the values of all neighbors of x in (C) using (12).

We improved this approach by considering that the function u is a *signed* distance function to a curve $\Gamma(t)$. Since the sign is not modified by this procedure, it is sufficient to compute the Euclidean distance to $\Gamma(t)$ to fully determine the function u .

4. NUMERICAL RESULTS

In this section, we will present several experimental examples on both synthetic and real datasets, to show the efficiency and the robustness of our curve reconstruction method. In particular, we show that the level set formulation can handle complicated geometries and deal efficiently with topological changes during the curve evolution.

4.1. Sampled point data set

In order to illustrate the various stages of the method, we consider first a data set V where all points have been sampled from a parametric curve Γ . Here, the computational domain Ω is a disk of unit diameter centered at $(0.5, 0.5)$. The triangulation T_h contains 6,634 vertices corresponding to a minimal element size $h_{min} = 0.001$ (Figure 3, top). The evolution of the curve $\Gamma(t)$ is illustrated Figure 3 (bottom, left) on several time steps. The parameter α in the curve evolution Equation (4) has be set to $0.5 h_{min}$ and the explicit solver was used with a stability condition $\Delta t = O(h_{min}^2)$. The initial curve $\Gamma(0) = \Gamma_0$ is the implicit circle centered

at $(0.5, 0.5)$ of diameter 0.9. The final result has been obtained in 9,800 iterations with the renormalization of the level set function u every 100 time steps.

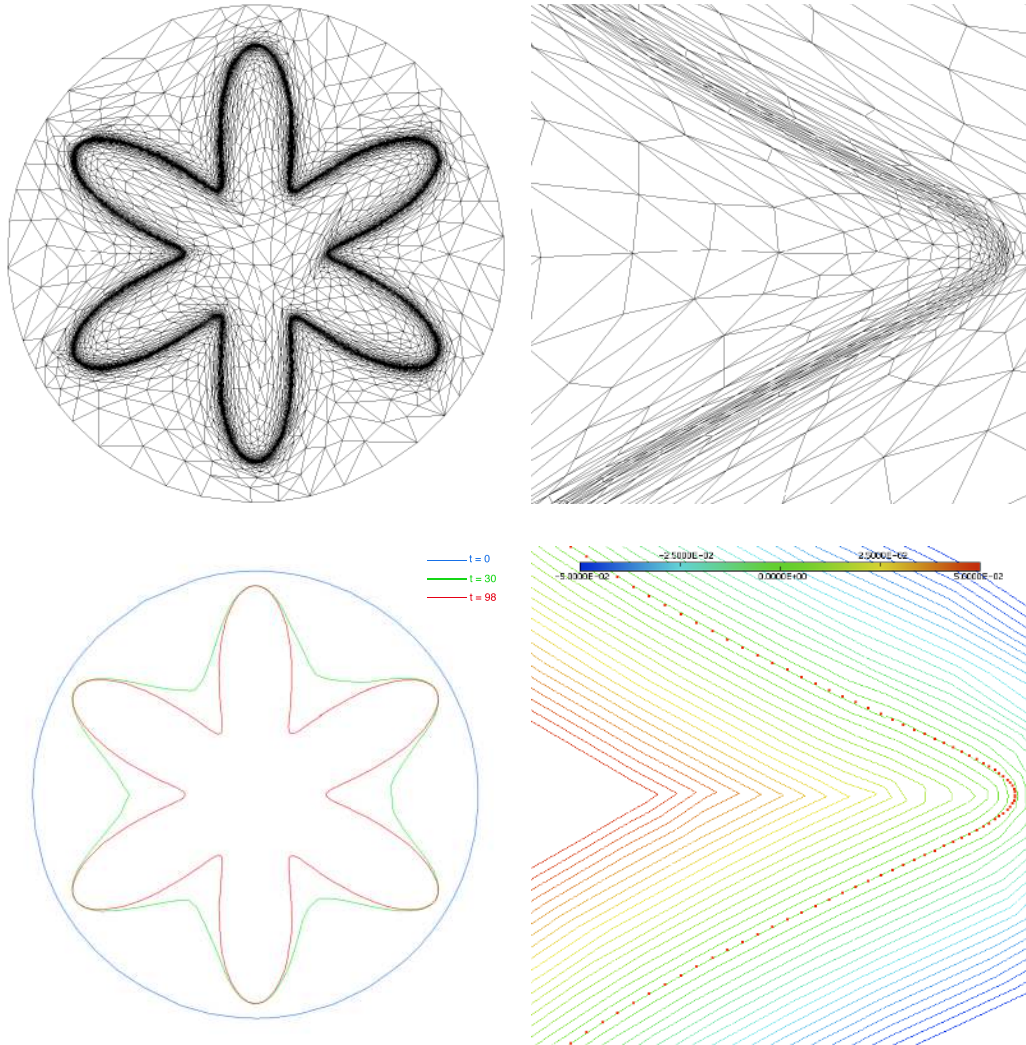


FIGURE 3. Construction of a regular curve for approximating an analytically defined point set. Anisotropic triangulation T_h (top), the curve $\Gamma(t)$ at different time steps (bottom left) and the level set function u at convergence of the method (bottom right).

In order to assess the robustness of our approach on noisy data sets, we have conducted the same numerical experiment with adding a random noise on the coordinates of the points in the data set V . The result can be seen on Figure 4 where we represented both the level set function u at convergence (left) and the unsigned distance function (right). It is easy to see that the stopping criterion (11) allows to reconstruct a smooth approximation minimal curve fitting at best through the points of the data set.

4.2. Multiple connected components data set

To illustrate the ability of our method to handle topological changes during the curve evolution, we propose the following experiment. Here, the domain Ω is defined as a black and white image where the letters $\{P, o, i, n, t, s\}$ have been written (Figure 5). A simple procedure is used to convert this image into a regular uniform mesh of a unit square. Then, the contours of each letter have been sampled defining the desired data set V . The unsigned distance function d has been computed and a quasi-uniform anisotropic triangulation

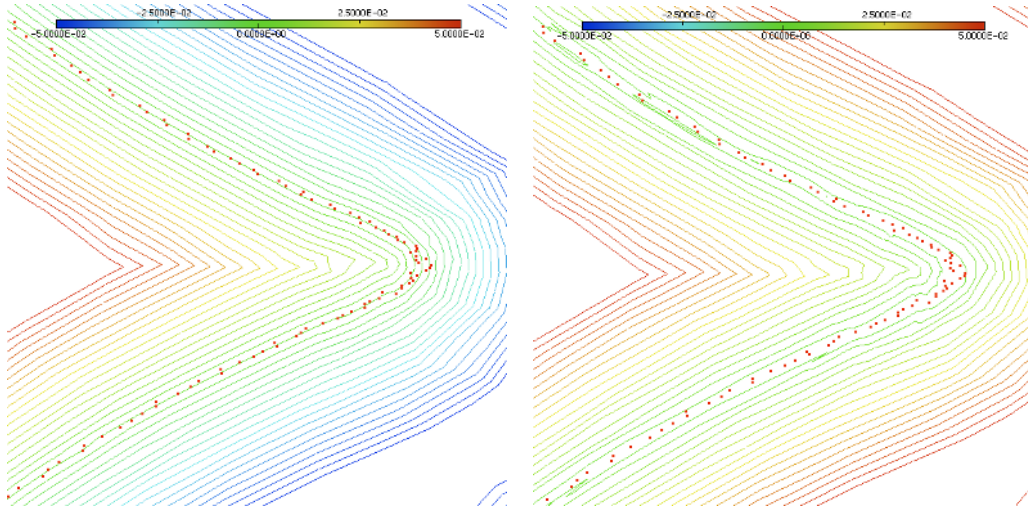


FIGURE 4. Construction of a regular approximation curve for noisy point set. Comparison between the final level set of u (left) and the unsigned distance function d (right).

similar to that represented Figure 5 has been generated based on the metric tensor field related to the local curvature of the zero isocontour of d . In this example, each letter is obviously a connected component that we would like to isolate.

Again in this example, we started from a circle enclosing the point set V (Figure 7, top right). The anisotropic triangulation contains 12,768 vertices corresponding to a minimal size $h_{min} = 10^{-4}$ and is represented on Figure 6. The explicit scheme was used to solve the problem with a stability condition such that $\Delta t = O(h_{min}^2)$ and every 100 time steps, the renormalization procedure was carried out on the data to recover a signed level set function.

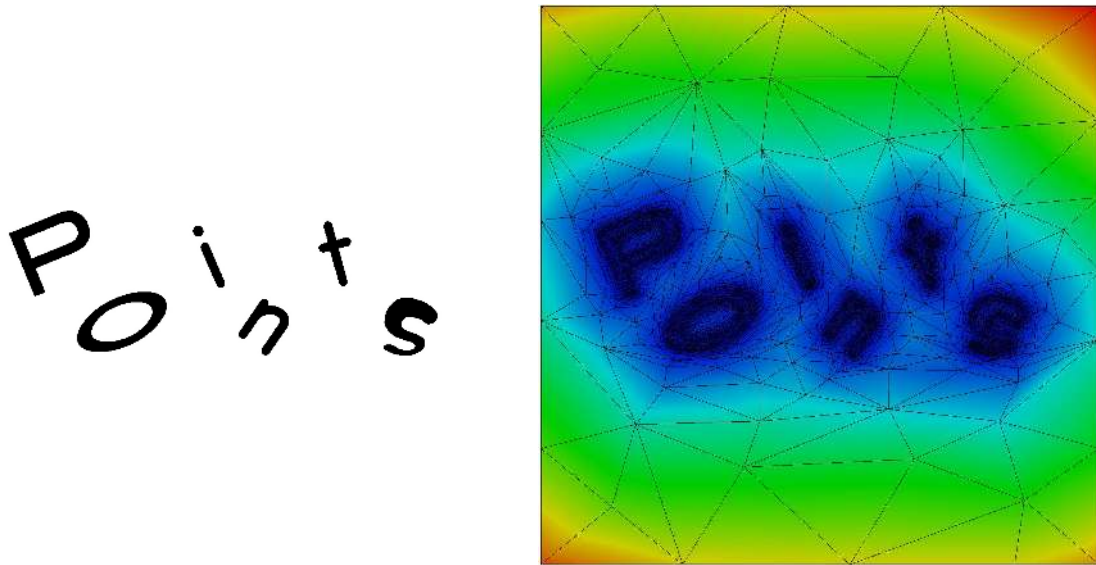


FIGURE 5. Example of a data set with multiple connected components. Original black and white image (left) and quasi-uniform anisotropic triangulation with associated distance field (right).

Figure 7 shows the evolution of the curve $\Gamma(t)$ at the initial stage (left), at intermediate (simply connected) and final stages (right). Our method captured the various connected components without apparent difficulty

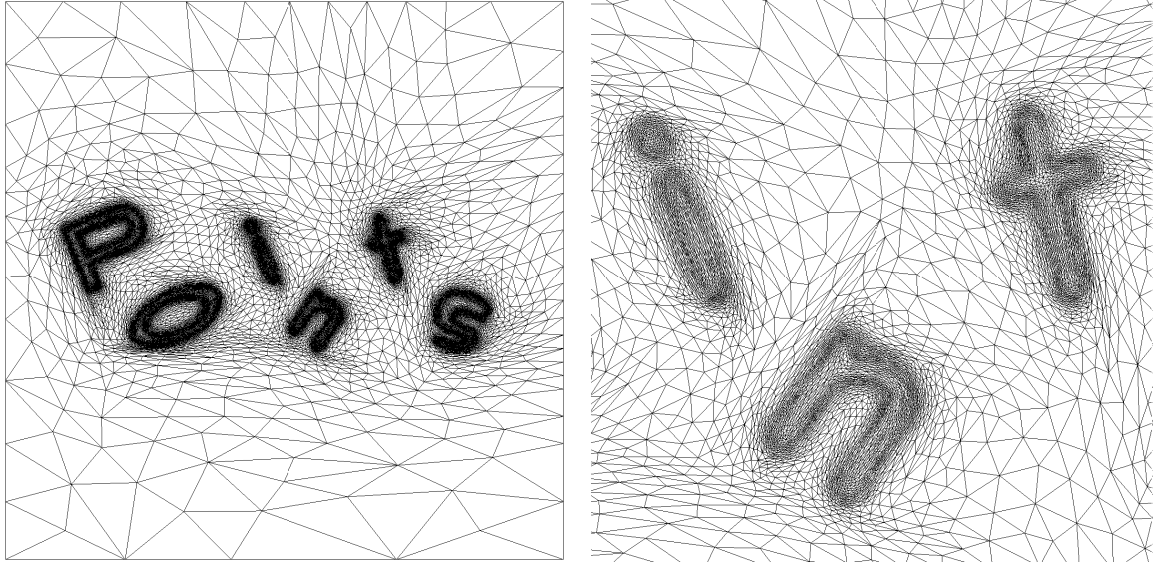


FIGURE 6. Anisotropic quasi uniform triangulation T_h used to capture the multiple connected components of the data set V and local enlargement in the vicinity of the point set (right).

on a reasonably fine triangulation. Notice that the missing embedded parts of the letters P and O have to be considered here as an artefact of the current heuristic renormalization algorithm and not as a restriction of the evolutionary process. The last Figure 8 shows a comparison between the level set function u at convergence and the unsigned distance function d in the vicinity of letter "S". Notice the irregularity of the point set related to the sampling algorithm used to convert a digital image into a mesh (no antialiasing procedure has been applied). The figure shows the regularity of the computed level set function u .

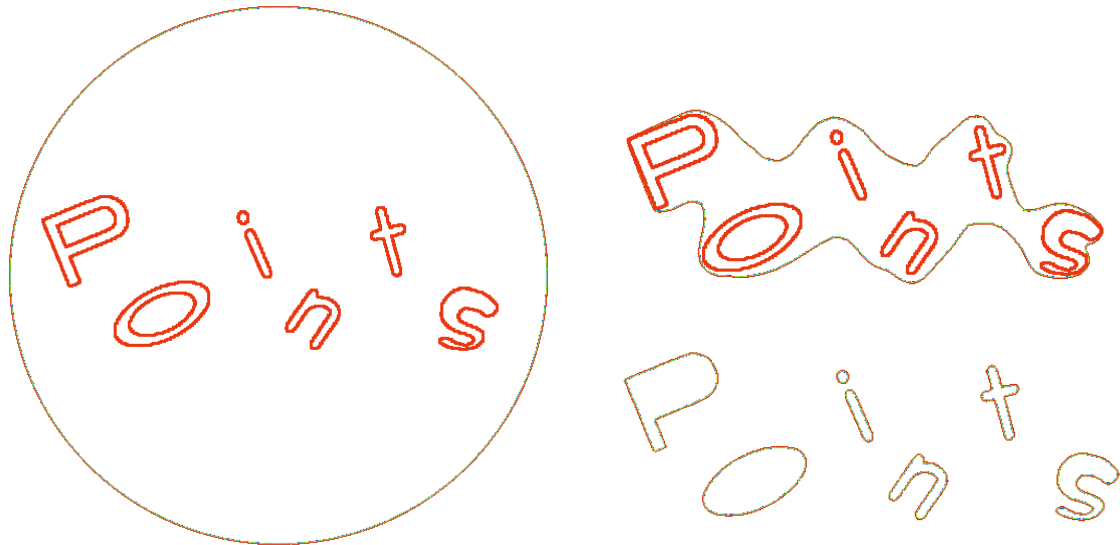


FIGURE 7. Anisotropic quasi uniform triangulation T_h used to capture the multiple connected components of the data set V and local enlargement in the vicinity of the point set (right).

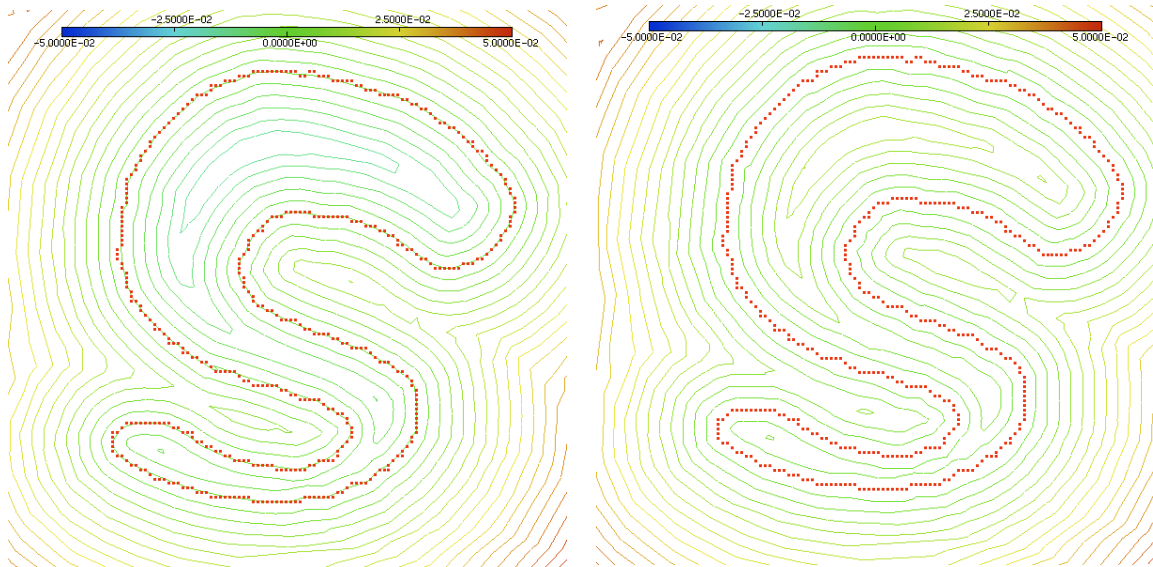


FIGURE 8. Comparison between the reconstructed level set function u at convergence (left) and the unsigned distance function d (right) near the letter "S".

4.3. First experiment in three dimensions

To conclude our presentation, we propose a preliminary numerical experiment in three dimensions. As mentioned, our approach can be extended without difficulty in three dimensions. It requires however the ability to generate anisotropic triangulations composed of tetrahedral elements. The authors have used a software developed in collaboration with C. Dobrzynski (Univ. Bordeaux I) that we are pleased to acknowledge here [18]. This example use real data sampled from a MRI data acquisition in view of a biomedical application.

The data set correspond to the sampling of a human brain. The quasi-uniform anisotropic triangulation contains 652,088 vertices corresponding to a minimal element size $h_{min} = 7 \cdot 10^{-4}$. The initial surface is an implicit sphere centered at the origin and enclosing the data set V . We used the first order explicit Euler scheme with the stability condition $\Delta t = O(h_{min}^2)$. Figure 9 illustrates the regularity of the reconstructed surface and, more importantly, that it is composed of a single connected component. This points out a potential strength of our approach in three dimensions: its ability to minimize the number of connected components when reconstructing a highly complex surface. This feature offers likely an attractive alternative to Marching Cube type algorithms which present several drawbacks regarding the topology of the reconstructed surfaces.

5. CONCLUSIONS

In this paper, we proposed a new PDE-driven curve evolution model that can deal with arbitrary point data sets. The deformation of an initial regular curve is governed by partial differential equations and using the calculus of variations we proved that this model correspond to a constrained minimization problem. Our model ensures regularity and stability of the reconstructed curve. A variational formulation of the problem is efficiently solved on anisotropic optimal triangulations with a minimal number of degrees of freedom. Next, we plan to further apply this model on three dimensional point sets as all procedures can be easily extended to higher dimensions.

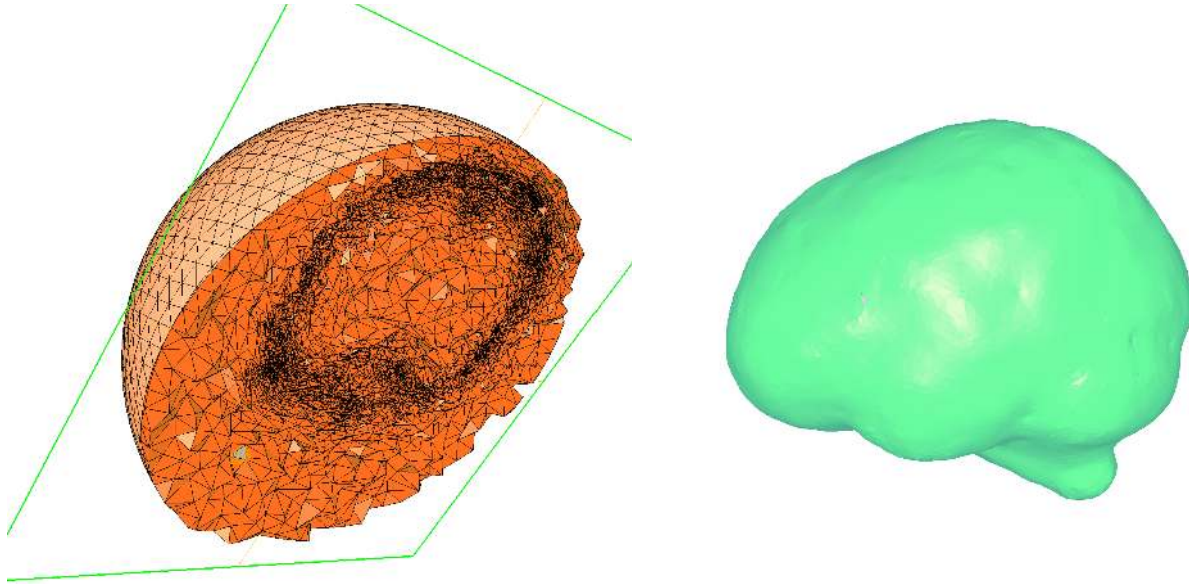


FIGURE 9. *Preliminary 3d result. Construction of a regular surface to approximate the surface of a sampled brain.*

REFERENCES

- [1] R. Abgrall, Numerical discretisation of boundary conditions for first order Hamilton-Jacobi equation on triangular meshes, *Comm. Pure Appl. Maths.*, **49**, (1996), 1339-1373.
- [2] D. Adalsteinsson and J.A. Sethian, A fast level set method for propagating interfaces, *J. Comp. Phys.*, **118**(2) (1995), 269-277.
- [3] R. Allègre, R. Chaine and S. Akkouche, A flexible framework for surface reconstruction from large point sets, *Computers & Graphics*, **31**(2), (2007), 190-204.
- [4] L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel, Axioms and fundamental equations of image processing, *Arch. Rat. Mechanics*, **123** (1993), 1992-57.
- [5] D. Apprato, R. Arcangeli, R. Manzanilla, Sur la construction de surfaces de classe C^k tirées d'un grand nombre de données de Lagrange, *M²AN*, Vol.21,n°4, (1987), 529- 2555.
- [6] D. Apprato, C. Gout, A result about scale transformation families in approximation: application to surface fitting from rapidly varying data, *Numerical Algorithms 23* (2,3), (2000), 263-279 .
- [7] D. Attali, r -regular shape reconstruction from unorganized points, *ACM Symp. Computational Geometry*, (1997), 248-253.
- [8] F. Bernardini, C. Bajaj, Sampling and reconstructing manifolds using alpha-shapes, *Proc. 9th Canadian Conf. Computational Geometry*, (1997), 193-198.
- [9] G. Barles, Remarques sur des rsultats d'existence pour les quations de Hamilton-Jacobi du premier ordre, *Annales de l'IHP, Analyse non linéaire*, **2**(1), (1985), 21-32.
- [10] J.D. Boissonnat, F. Cazals, Smooth shape reconstruction via natural neighbor interpolation of distance functions, *ACM Symposium on Computational Geometry*, (2000), 223-232.
- [11] C. de Boor, A practical guide to splines, *Springer Verlag, Berlin-Heidelberg*, (1978).
- [12] H. Borouchaki, F. Hecht and P.Frey, Mesh gradation control, *Int. j. numer. methods eng.*, **43**(6), (1998), 1143-1165.
- [13] R. Bridson, J. Teran, N. Molino, R. Fedkiw, Adaptive physics based tetrahedral mesh generation using level sets, *Engineering with Computers* **21** (2005), 2-18.
- [14] P.G. Ciarlet, Introduction to Numerical Linear Algebra and Optimization, *Cambridge University Press* (1988).
- [15] M. Crandall, J.L. Lions, Two approximations of solutions of Hamilton-Jacobi equations, *Math. Comp.*, **43**, (1984), 1-19.
- [16] T. Dey and J. Sun, Normal and feature approximations from noisy point clouds, *Proc. FST & TCS*, Lecture Notes in Computer Science, Springer, 4337 (2006), 21-32.
- [17] E.W. Dijkstra, A note on two problems in connexion with graphs, *Numerische Mathematik*, **1**, (1959), 269-271.
- [18] C. Dobrzynski and P. Frey, Anisotropic Delaunay mesh adaptation for unsteady simulations, submitted (2008).
- [19] Y. Duan, L. Yang, H. Qin, D. Samaras, Shape reconstruction from 3d and 2d data using PDE-based deformable surfaces, *Proc. European Conference on Computer Vision*, **3**, (2004), 238-251.
- [20] V. Ducrot, P. Frey, Contrôle de l'approximation géométrique d'une interface par une métrique anisotrope, *C.R. Acad. Sci. Paris*, Ser. I 345 (2007), 537-542.

- [21] V. Ducrot and P. Frey, Anisotropic levelset adaptation for accurate interface capturing, *submitted* (2008).
- [22] L.C. Evans, Partial differential equations, AMS (1997).
- [23] P. Frey, A differential geometry approach to mesh generation, in *Series in Contemporary Applied Mathematics*, CAM 9, Higher Education Press, (2007).
- [24] P. Frey and P.L. George, Mesh generation: application to finite elements, *Wiley-ISTE*, 2nd ed. (2008).
- [25] C. Le Guyader, C. Gout, Geodesic active contour under geometrical conditions: Theory and 3D Applications, *Num. Algo.*, **48**(2), (2008), 105-134.
- [26] H. Hoppe, T. deRose, T. Duchamp, J. McDonald, W. Stuetzle, Surface reconstruction from unorganized points, *Proc. Siggraph*, (1992), 71-78.
- [27] A. Hornung, L. Kobbelt, Robust reconstruction of watertight 3d models from non-uniformly sampled point clouds without normal information, *Proc. Eurographics Symposium on Geometry Processing*, ACM, (2006), 41-50.
- [28] G.S. Jiang and D. Peng, Weighted ENO schemes for Hamilton-Jacobi equations, *SIAM J. Sci. Comput.*, **21**, (2000), 2126-2143.
- [29] M.W. Jones, J.A. Bærentzen, M. Šrámek, 3D distance fields: a survey of techniques and applications, *IEEE Transactions on Visualization and Computer Graphics*, **12**(4), (2006), 581-599.
- [30] P.J. Laurent, Approximation et optimisation, *Hermann, Paris*, (1972).
- [31] C.T. Lim, G.M. Turkiyyah, M. Ganter and D. Storth, Implicit reconstruction of solids from cloud point sets, *Proc. Solid Modeling*, ACM, (1995), 393-402.
- [32] W.E. Lorensen, H.E. Cline, Marching Cubes: a high resolution 3D surface construction algorithm, *Computer Graphics*, **21**(4) (1987), 163-169.
- [33] D. Lu, H. Zhao, M. Jiang, S. Zhou and T. Zhou, A surface reconstruction method for highly noisy point clouds, *Proc. Variational, geometric, and level set methods in computer vision*, Lecture Notes in Computer Science, Springer, **3752** (2005), 283-294.
- [34] M. Marcon, L. Piccarreta, A. Sarti and S. Tubaro, Fast point-cloud wrapping through level-set evolution, *Proc. 1st Conf. Visual Media Production*, (2004) 119-125.
- [35] R. Mencl, H. Müller, Interpolation and approximation of surfaces from three-dimensional scattered data points, *Scientific Visualization*, (1997), 223-232.
- [36] O. Nilsson, D. Breen, K. Museth, Surface reconstruction via contour metamorphosis: an Eulerian approach with Lagrangian particle tracking, *Proc. IEEE Visualization*, **52**, (2005), 407-414.
- [37] S. Osher, C.W. Shu, High-order essentially nonoscillatory schemes for Hamilton-Jacobi equations, *SIAM, J. Numer. Anal.*, **28**, (1991), 907-922.
- [38] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed : Algorithms based on Hamilton-Jacobi formulations, *J. Comp. Phys.*, **79** (1988), 12-49.
- [39] S. Osher, N. Paragios, Geometric Level Set Methods in Imaging, Vision and Graphics, *Springer*, 2003.
- [40] D. Peng, B. Merriman, S. Osher, H.K. Zhao, and M. Kang, A PDE based fast local level set method, *J.Comp.Phys.*, **155** (1999), 410438.
- [41] I. Semenova, V. Savchenko and I. Hagiwara, Reconstruction of shapes based on normals analysis, *Proc. GraphiCon* (2005) 1-7.
- [42] J.A. Sethian, Level Set Methods and Fast Marching Methods, *Cambridge*, (1999).
- [43] Y. Shi, W.C. Karl, Shape reconstruction from unorganized points with a data-driven level set method, *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, (2004), 13-16.
- [44] Y R. Tsai, Rapid and accurate computation of the distance function using grids, *J. Comp. Phys.*, **178** (2002), 175-195.
- [45] H. Yang, B. Jüttler, Fitting implicitly defined curves to unorganized points with sharp features, *Curve and Surface Design: Avignon 2006*, P. Chenin, T. Lyche and L.L. Schumaker (eds.), Nashboro Press, (2007), 274-283.
- [46] H.K. Zhao, S. Osher, B. Merriman, M. Kang, Implicit and non-parametric shape reconstruction from unorganized data using a variational level set method, *Comput. Vision and Image Understanding*, **80** (2000), 295-314.
- [47] H. Zhao and S. Osher, Visualization, Analysis and shape reconstruction of unorganised datasets, in Geometric Level Set Methods in Imaging, Vision and Graphics, S.Osher and N. Paragios eds, *Springer*, 2002.