South Bohemia Mathematical Letters
Volume 18, (2010), No. 1, 29-37.

SHAPE FITTING AND NON CONVEX DATA ANALYSIS

PETRA SURYNKOVÁ, ZBYNĚK ŠÍR

Abstract. This contribution addresses the problem of curve and surface evolution. We explain a general framework for the evolution-based approximation of a given set of points by a curve. Then we apply this method to surfaces. We show the sequential evolution of curves and surfaces on some concrete examples. We apply the curve evolution as a solving method in statistical data analysis. Our aim is to use closed B-spline curves for the description of data sets related to the local density of the data sample. The result is a contour fulfilling some predefined statistical criterion. We also explain the notion of data depth. We also use the surface evolution for the surface reconstruction from point clouds. We describe the digital reconstruction problem and the methods of surface reconstruction. For the implementation of evolution algorithms we use the interactive environment MATLAB.

Introduction

Shape fitting is useful tool in many scientific branches. There is a wide range of possible applications of the shape fitting methods. In our research we deal with the digital reconstruction problem and the methods of surface reconstruction, see [8]. We explore the steps which are necessary to convert a physical model or some real object into a computer model. The surface reconstruction is a natural extension of the curve reconstruction. We want to develop some new methods of surface reconstruction especially in the context of the architectural objects. We focus on methods which are based on the sequential evolution. Both - curve and surface evolution is available. Of course, curve evolution is simpler case so that we explain this problem for curves then we apply it to surfaces.

We also apply a curve evolution as a solving method in statistical data analysis. One notion of shape estimation of a set of points in the plane can be their depth. Data depth has been used in statistics as the way to identify the centre of a bivariate distribution. This leads to a natural centre-outward ordering of the sample points. Then we can estimate important characteristics of the data. The set of depth contours of a set of points can be used in various applications.

We apply the curve evolution to finding the depth contour. The input is a finite set of points in the plane and our aim is to obtain the description of a data set by using closed B-spline curves. We have some initial closed B-spline curve. This curve moves and modifies in time such that finally we obtain depth contour fulfilling some predefined statistical criterion. One of the strength of this method is the possibility of obtaining both convex and non convex shapes of contours.

The remainder of this paper is organized as follows. The section 1 is devoted to the curve and surface fitting based on the sequential evolution. The notion of

Received by the editors 19.11.2010.
Key words and phrases. Curve and surface evolution, surface reconstruction, data depth, B-spline.

data depth and various definitions of statistical depth are given in section 2. In the section 3 we introduce our method for statistical data analysis and describe the implementation of the algorithm. Finally we present some examples of depth contours obtained using closed B-spline curve evolution – convex and non convex shapes.

1. Shape fitting

The specific task we solve is given by unorganized finite set of points in the space (point cloud). The output is the reconstructed surface so that the points of the input set lie on or close to surface. We assume that the input set corresponds with the real surface, no other information are given.

The input data can consist of a large number of points. Real data may contain over million points. Ideally, these data are precise coordinates of points on the surface of the object. But in real applications there will be measurement errors, we have to deal with. In the point cloud there may be redundant data, some points are useless, don't contain any new or important information or some points are very close to one another. For that reason these redundant data points will be removed. There exist several removal criteria which depend on the underlying application; more detailed information are in [3].

In the subsequent polygon phase, a triangle mesh is computed that approximates the given data points. This procedure is very difficult and no general method is available. In the polygon phase we obtain a first surface representation of the object. Several known algorithms for computing triangle mesh are for example alpha-shapes, crust algorithm, cocone algorithm which are based on spatial subdivision (on the dividing of the three-dimensional space). More detailed information can be found in [2, 6].

The final shape phase isn't necessary for the pure visualization but it will be crucial for architecture. We have to convert the triangle mesh into a CAD representation which is appropriate for further processing. This phase includes edge and feature line detection and decomposition into parts of different nature and geometry - for example planar parts, cylindrical patches, conical patches, freeform patches. This process is called segmentation. Then we have to approximate the data regions using surfaces of the correct type which we identified in the segmentation. For example, a region identified as being planar in the segmentation phase will be approximated by part of a plane. Computing such an approximation plane is simple task. This process is known as surface fitting. More detailed information can be found in [7].

We introduce some new methods of surface reconstruction which are based on the sequential evolution. We explain generally the principle of the curve and surface evolution.

1.1. **Curve evolution.** The principle of curve evolution is sequential modifying of planar parametric curve from some initial position and shape. The evolution will be stopped if some condition is satisfied. In our case if the final curve has minimal distance (in theory) or sufficiently small (in practical implementations) from the given data.

We identify a curve that approximates a given set of data points $\{p_j\}_{j=1\dots N}$ in the least square sense. We consider a planar parametric curve

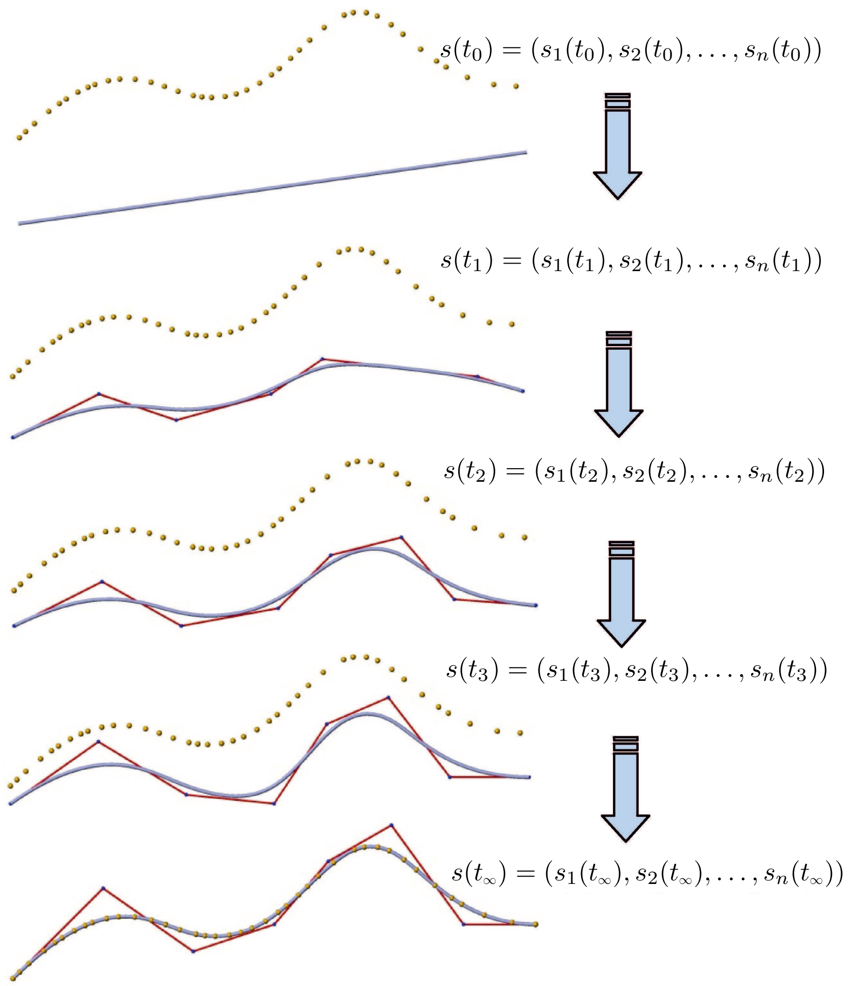

FIGURE 1. Curve evolution in time. First picture shows initial position and shape of the curve and the input set of points, last picture shows the final curve.

$$c(u) = \sum_{i=0}^{m} \beta_i(u) \cdot V_i, \tag{1.1}$$

where u is the curve parameter, V_i are control points and β_i are basis functions. We are looking for the curve such that

$$\sum_{j=i}^{N} \min_{x_j \in c} \|p_j - x_j\|^2 \longrightarrow \min, \tag{1.2}$$

where x_j are points on the curve. If we denote $c_s(u) := c(s,u)$ then two different kinds of parameters appear in the representation of the curve; the curve parameter u and a vector of shape parameters $s = (s_1, s_2, \ldots, s_n)$ where s_j denotes the coordinates of control points. It means that we are looking for the vector of shape parameters that defines the curve. We let the shape parameters s depend smoothly on an evolution parameter t, $s(t) = (s_1(t), s_2(t), \ldots, s_n(t))$. The parameter t can be identified with the time. Starting with certain initial values, these parameters are modified continuously in time such that a given initial curve moves closer to the data points. Figure 1 shows the curve evolution in time. For each point $\{p_j\}_{j=1\ldots N}$ we compute the closest points $f_j = c(u_j)$ on the initial curve. During the evolution of a curve $c_{s(t)}(u)$ each point f_j travels with the velocity

$$v_s(u_j) = \dot{c}_s(u_j) = \sum_{i=1}^{n} \frac{\partial c_s(u_j)}{\partial s_i} \dot{s}_i \tag{1.3}$$

or with the normal velocity

$$(v_s(u_j))^{\mathsf{T}} n_s(u_j) = \sum_{i=1}^{n} \left(\frac{\partial c_s(u_j)}{\partial s_i} \dot{s}_i \right)^{\mathsf{T}} n_s(u_j). \tag{1.4}$$

The dot denotes the derivative with respect to the time variable t, $n_s(u_j)$ denotes the unit normal of the curve in the point $c_s(u_j)$. We set $d_j := p_j - f_j$. If the closest point is one of the two boundary points then we consider the velocity (1.3). We can compute for each point f_j the velocity or normal velocity on the one hand and the expected velocity on the other hand. The following condition has to satisfy

$$\sum_{\substack{j=1 \\ u_j \notin \{a,b\}}}^{N} \left\| (v_s(u_j) - d_j)^{\mathsf{T}} n_s(u_j) \right\|^2 + \sum_{\substack{j=1 \\ u_j \in \{a,b\}}}^{N} (v_s(u_j) - d_j)^2 \longrightarrow \min_{\dot{s}}. \tag{1.5}$$

We compute $(\dot{s}_1, \dot{s}_2, \ldots, \dot{s}_n)$ and update the vector of shape parameters using the Euler-steps $(s_1 + \varepsilon \dot{s}_1, s_2 + \varepsilon \dot{s}_2, \ldots, s_n + \varepsilon \dot{s}_n)$. More detailed information can be found in [1].

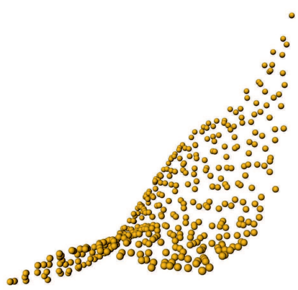

FIGURE 2. The input set of data points in the space.

1.2. **Surface reconstruction.** We can indeed the methods of shape evolution to surfaces. The principle of the surface evolution is the same. We assume that a set of data points $\{p_j\}_{j=1\ldots N}$ in the space is given, see Figure 2. We identify a surface that approximates a given set of data points in the least square sense. Surface evolution in time is shown in Figure 3.

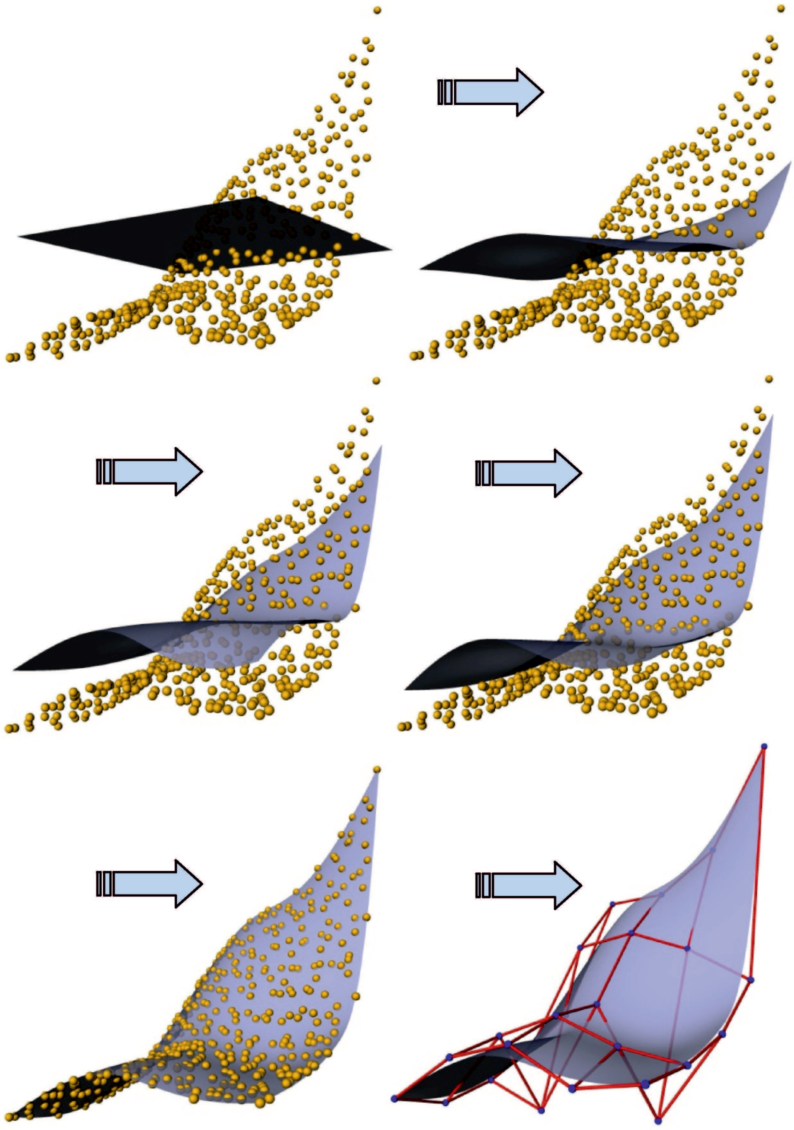

FIGURE 3. Surface evolution in time. First picture shows initial position and shape of the surface and the input set of points, last two pictures show the final surface and the final surface with control points which are modified in time.

1.3. **Example of the curve evolution.** In this part we demonstrate curve fitting based on the sequential evolution. We apply curve evolution to closed B-spline curves. We consider closed B-spline as a planar parametric curve which is modified by the evolution process. Our implementation is applicable to both convex and non convex shape. We choose MATLAB for implementation of this algorithm. Figure 4 illustrates the outputs of our algorithm.

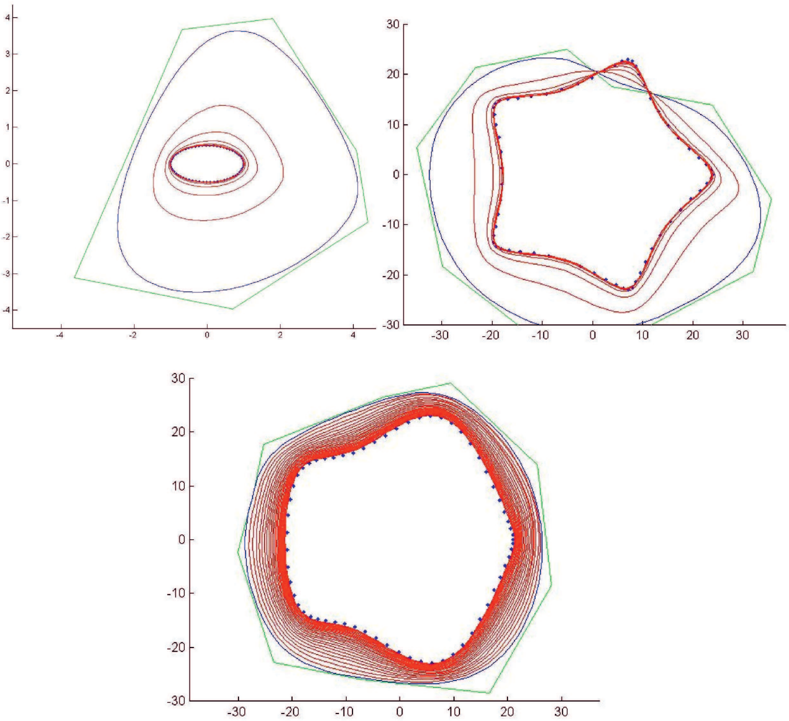

FIGURE 4. Examples of curve fitting based on the sequential evolution.

2. The notion of data depth

The notion of data depth generalizes the median to higher dimensions. The motivation and necessity in statistics to generalize the median is very natural. Various depth measures have been proposed to analyze a given theoretical distribution of a set of data points in the plane. The concept of data depth is based on ordering of points. The principle of ordering is intuitively very simple. In general, the greater the depth of a point, the more densely it is surrounded by other points of the input set. But in some examples the deep point is not itself in a dense region, but if we travel from infinity to the point we are forced cross regions with high density. We can compute over this depth measure the point in the plane with the maximum depth - the deepest point. It is analogous to the median in $\mathbb{R}$ and we will keep this name. We can also find the regions of all points with depth greater than some assigned value k. These regions are sometimes called tolerance regions. It is quite natural to suppose that their boundaries are isolines of the density function. The way we add the real numbers (the data depth) to points is described by so called *depth function*.

There are many different definitions of data depth measure, each approaching the problem in different way. These definitions vary in their efficiency and applicability to more or less general data or distributions. Typically, they restrict themselves to the cases where the depth contours are convex. Usually the following properties are required (see [4, 5, 10])

1. affine invariance,
2. maximality at centre,
3. monotonicity relative to deepest point,
4. vanishing at infinity.

While the first and the last properties are very general and acceptable, the conditions 2 and 3 suggest the convexity of the data and uniqueness of the depth centre.

The most well known are convex hull peeling, simplicial depth, half-space depth. This depth measures were introduced in our contribution last year. See [9].

3. Statistical data analysis

In this part we present our method for statistical data analysis and show some examples produced by our algorithm. We apply a curve evolution to finding the depth contours. The input is in principle a two dimensional statistical distribution, which is practical examples represented by a sampled data set.

We consider closed cubic B-spline as a planar parametric curve which will be modified by the evolution process. The evolution is guided by two forces acting simultaneously on the curve. First one forces the curve to enclose a prescribed Euclidean area. The area can be computed from the control point via the well known integral formula. It is not linear, but similarly to the shape fitting, the derivative of this formal is linear with respect to the derivatives of the control points. This force will act on the curve with a rather great weight coefficient. Consequently, the curve during it evolution will almost keep the prescribed enclosed area.

Second force is given by the principle which is very similar to the one used for the shape fitting. For a sequence of points (sensors) along the moving curve we estimate a real number which represents *the force* by which should be the corresponding point attracted to the interior of the curve. The force is evaluated in every step of curve evolution and is inverse proportional to the density function. For concrete probability statistical data we estimate the density function and the force from the number of points lying in a suitable neighborhood of the sensor. It means, the more densely the point on the curve is surrounded by other points of the input set, the smaller the force of point on the curve. The part of the curve on which the points have great data density moves less. The evolution is stopped when it reaches a stationary position, in practice when in last steps its modification was very small. As a result we obtain a region of a prescribed Euclidean area which has (locally) maximal probabilistic measure (in practice number of data points). Then we can define some new value of the prescribed Euclidean area continue the evolution towards a new depth contour.

The shape of these contours can be convex or non convex. Our method is already reliable for convex shape. For non convex shapes of the contours we obtained some promising first results. Due to the higher degrees of freedom we have to do more

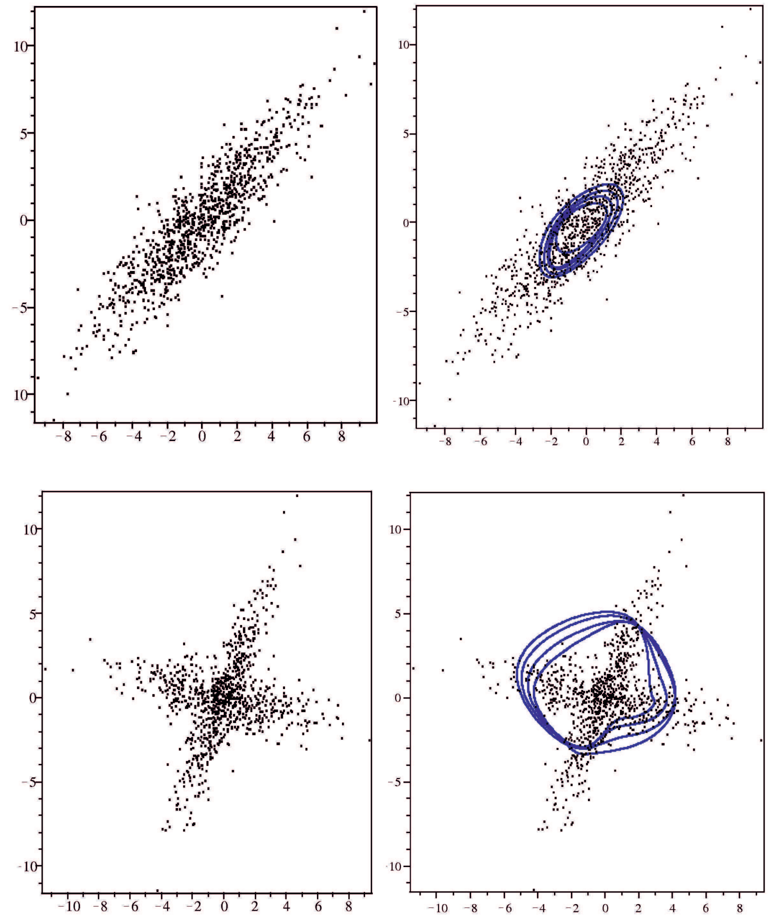

FIGURE 5. Input data (left) consisting of 1000 points and computed contours (right) which are convex (top) or non-convex (bottom).

experiments with setting the coefficients guiding the evolution in order to obtain more reliable results. The implementation was realized in the program MAPLE.

CONCLUSION

Our contribution focused on various applications of the shape evolution. We explained the general framework for the evolution-based approximation of a given set of points by a curve. Then we applied it to surfaces. We suggested the algorithm for curve fitting based on the sequential evolution and showed several examples of the outputs of our algorithm. We also applied curve evolution as a solving method to finding the depth contours in statistical data analysis. In our future work we will focus on the improvement of the implementation of our algorithms, which we apply to real data.

Acknowledgments

This work was supported by the grant no. 201/08/048 of the Czech Science Foundation. Petra Surynková was also supported by the grant SVV-2010- 261 315.

An essential part of this article has been presented at 30th Conference on Geometry and Graphics, and published in Proceedings of this conference ([11]).

References

[1] Aigner M., Šír Z., Jüttler B.: *Least-squares approximation by Pythagorean hodograph spline curves via an evolution process*, GMP, Springer LNCS, Vol. 4077, 2006

[2] Edelsbrunner, H.: *Geometry and topology for mesh generation*, Cambridge University Press, Cambridge, United Kingdom, 2001

[3] Iske, A.: *Multiresolution Method in Scattered Data Modelling*, Technische Universität München, Germany, 2004

[4] Liu R. Y., Serfling R., Souvaine D. L.: *Data depth: Robust multivariate analysis, computational geometry and applications*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. 72, 2003

[5] Liu R. Y.: *On a notion of data depth based on random simplices*, Annals of statistics, Vol. 37, 2009

[6] Mencl, R., et al.: *Interpolation and Approximation of Surfaces from Three-Dimensional Scattered Data Points, State of the Art Report*, EURORAPHICS'98, 1997

[7] Pottman H., Asperl A., Hofer M., Kilian A.: *Architectural geometry*, Bentley Institute Press, USA, 2007

[8] Surynková P.: *Surface reconstruction*, Proceedings of the 17th Annual Conference of Doctoral Students - WDS 2009, MATFYZPRESS, Faculty of Mathematics and Physics, Charles University, Prague, 2009

[9] Surynková P.: *Statistical application of curve evolution*, 29th Conference on Geometry and Graphics, Doubice, 2009

[10] Venclek O.: *Depth based classification*, ASMDA, 2009

[11] Surynková P., Šír Z.: *Shape fitting and non convex data analysis*, Proceedings of 30th Conference on Geometry and Graphics, 2010, 219–229, (available on http://www.csgg.cz/30zlenice/sbornik2010.pdf).

Department of Mathematics Education, Faculty of Mathematics and Physics, Charles University in Prague, Sokolovská 83, 186 75 Praha 8, Czech Republic

E-mail address: `petra.surynkova@mff.cuni.cz`

Mathematical Institute of Charles University, Faculty of Mathematics and Physics, Charles University in Prague, Sokolovská 83, 186 75 Praha 8, Czech Republic

E-mail address: `zbynek.sir@mff.cuni.cz`