Curve fairing using integral spline operators

F. Caliò [a] E. Miglio [b] G. Moroni [c] M. Rasella [c]

[a] *Politecnico di Milano - Dept. of Mathematics, Piazza Leonardo da Vinci 32, 20133 Milano (Italy)*

[b] *Politecnico di Milano - MOX, Modelling and Scientific Computing, Dept. of Mathematics, Piazza Leonardo da Vinci 32, 20133 Milano (Italy)*

[c] *Politecnico di Milano - Dept. of Mechanics, Piazza Leonardo da Vinci 32, 20133 Milano (Italy)*

Abstract

In this paper a local automatic planar curve fairing algorithm based parametric B-spline class is presented. In particular we employ a particular class of spline characterized by a shape parameter λ: for this family of spline it has been shown (see [9]) that the value of the parameter affects the shape of the whole spline curve. We have exploited this last property locally in order to move a subset of the control points defining the given curve. In our approach the value of λ is chosen in order to minimize a functional related to the fairness of the curve and in particular we have considered a functional involving the second derivative of the curvature. The numerical test cases we have performed showed the effectiveness of algorithm both in academic and real-world situations.

Key words: automatic fairing, B-spline curves, reverse engineering.

1 Introduction

The current trend in manufacturing industries towards the use of free-form shapes has given raise to significant research efforts in the fields of design, manufacturing and inspection of this kind of shapes. In fact, nowadays, all CAD systems support functions for modeling free-form surfaces, modern CNC machines can be adopted to create them and different digitizing devices can be used to inspect such objects. A great deal of effort has been reported in literature on free-form inspection techniques (see for example [4]). In particular many reverse engineering problems, using free form modeling, are presented in [15].

Since generated or recreated free form shapes frequently present irregularities such as aesthetic undulations, we are faced to the problem of eliminating them by means of a fairing process.

In interactive fairing [12] an operator identifies points of curve affected by irregularities and makes appropriate corrections; this process is repeated until an acceptable fairness is obtained. Interactive fairing is simple but is time consuming and can hardly be automatized, therefore research activities are directed towards automatic fairing.

Currently, automatic fairing methods are based on two alternative approaches: global fairing where the undulations are directly eliminated during the interpolation or approximation process [5],[14] and local fairing where the profile construction and the fairing process are separated.

In local automatic fairing an analytical indicator is used to identify the "bad points" (bad in the sense of a particular fairness criterion) that are moved according to a specific method minimizing some particular functional (fairing indicators). This process continues until a suitable end condition is fullfilled.

In the case of B-spline curves, a first automatic fairing algorithm, based on the movement of "bad points" by means of the minimization of the curvature functional, has been proposed by Farin and Sapidis in [13] and refined by Kjellander [6] and Poliakoff et al. [11]. Another automatic algorithm has been presented by Eck and Hadenfeld [3]: their key idea is to minimize the energy integral. In [16] is presented an algorithm that can be used to fair B-spline curves. Recently, in [7] the authors present a fairing algorithm for planar cubic B-spline. In order to identify bad points they use a target curvature plot. The corresponding control points are modified using a local constrained optimization procedure; the objective function is a weighted combination of two components: the first one related to the fairness of the curve, the second concerns the coherence to the original design. In [10] another automatic curve fairing algorithm is presented (with application to ship design). This algorithm is based on the use of optimization tools and cubic B-spline functions. The objective of the optimization algorithm is the minimization of the energy functional and some geometric constraints are imposed in order to obtain an optimum curve in terms of both fairness and closeness to the original curve.

Our paper proposes a local automatic fairing algorithm based on parametric B-spline approximating planar curves. Fairness and closeness to the original curve are the objectives of the work.

The central idea is the following: to move a "bad set" of approximating B-spline control points, the algorithm employs a particular operator (integral spline operator): this operator is characterized by a shape parameter which is chosen in order to minimize a suitable cost-functional related to the fairness of curve.

The paper is organized as it follows: section two deals with some preliminaries concerning differential geometry of curves; in section three the so-called λ-spline integral operator is introduced along with a brief description of their principal graphically interesting properties; section four describes the fairing indicators used and the fairing algorithm; finally in section five some preliminary numerical results are presented in order to assess the effectiveness of the proposed method.

2 Notations

In this section, just for the sake of completeness, we will briefly recall basic notations concerning fundamental concepts of differential geometry of curves.

Let us consider a parametric curve $\mathbf{p} = \mathbf{p}(t) = (x(t), y(t), z(t))^T$, the lenght of infinitesimal arc of curve is

$$ds = |\dot{\mathbf{p}}|dt \tag{1}$$

where the $\dot{}$ denotes the derivative with respect to t. The unit tangent vector $\mathbf{t}$ is given by

$$\mathbf{t} = \frac{\dot{\mathbf{p}}}{|\dot{\mathbf{p}}|} = \mathbf{p}' \tag{2}$$

where $'$ denotes the derivative with respect to s.

Since $\mathbf{p}' \cdot \mathbf{p}'' = 0$, *i.e.* the vector $\mathbf{p}''$ is orthogonal to $\mathbf{p}'$; hence we can define the following unit principal normal vector $\mathbf{n}$

$$\mathbf{n} = \frac{\mathbf{p}''}{|\mathbf{p}''|} \tag{3}$$

The quantity $\kappa = |\mathbf{p}''|$ is the so-called curvature and $\rho = 1/\kappa$ is the radius of curvature.

The expression of the curvature for a generic parametric planar curve can be written as follows

$$\kappa = \frac{(\dot{\mathbf{p}} \times \ddot{\mathbf{p}}) \cdot \mathbf{e}_z}{v^3} = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \tag{4}$$

where $\mathbf{e}_z$ is the unit vector normal to the xy plane and $v = \frac{ds}{dt}$ is the parametric speed.

3 Univariate integral parametric spline

In this section we recall the basics of VDS splines and of the univariate integral parametric splines proposed in [9].

Given a set of *control points* $\mathbf{P}_0, \mathbf{P}_1, \ldots, \mathbf{P}_m$ and a knots vector t

$$0 = t_{-k} = ... = t_0 < t_1 < ... < t_{n-1} < t_n = t_m = 1 \quad n = m - k$$

the expression

$$(S_m\mathbf{P})(t) = \sum_{i=0}^{m} \mathbf{P}_i B_i^k(t) \quad 0 \le t \le 1 \tag{5}$$

is called a k-order variation diminishing spline operator (VDS operator).

The basis function $B_i^k(t)$ $(i = 0, 1, \ldots m)$ are recursively defined as

$$B_i^k(t) = \frac{t - t_{i-k}}{t_{i-1} - t_{i-k}} B_i^{k-1}(t) + \frac{t_i - t}{t_i - t_{i-k+1}} B_{i+1}^{k-1}(t) \tag{6}$$

where

$$\begin{array}{ll} B_i^0(t) = 1 & t_i \le t \le t_{i+1} \\ B_i^0(t) = 0 & \text{otherwise} \end{array} \tag{7}$$

In matrix form the VDS operator can be written as follows

$$(S_m\mathbf{P})(t) = \mathbf{b}_m(t)\mathbf{p} \quad 0 \le t \le 1 \tag{8}$$

where

$$\mathbf{b}_m = (B_0^k(t), B_1^k(t), ..., B_m^k(t)), \;\; \mathbf{p} = (\mathbf{P}_0, \mathbf{P}_1, ..., \mathbf{P}_m)^T \tag{9}$$

In [9] some modifications to this class of splines are introduced: in particular a family of integral spline operators depending on a real parameter is presented.

This new family of spline will be called Univariate Integral $\lambda-$Variation Diminishing Splines.

Assuming that t_i is the value of the parameter corresponding to the given control point $\mathbf{P}_i$ we define

$$\xi_i^k = \frac{t_{i-k+1} + ...t_i}{k} \tag{10}$$

These points in the field of approximation are called Schonberg points [2]. We will call "correspondence points" such ξ_i^k values.

Let $x_j^i, (j = 1, 2, 3)$ be the generic component of vector $\mathbf{P}_i$ and $\varphi_j, (j = 1, 2, 3)$ the piecewise linear function interpolating points (ξ_i^k, x_j^i) and whose graphic is the control polygon.

The S_m operator on j-th component of $\mathbf{P}$ can then be expressed as

$$(S_m\mathbf{P})_j = (S_m\varphi_j) = \sum_{i=0}^{m} \varphi_j(\xi_i^k)B_i^k(t), \quad j = 1, 2, 3 \tag{11}$$

Substituting $\varphi_j(\xi_i^k)$ with the following integral mean

$$\mu_i\varphi_j(t) = \frac{\int_{\xi_i^{k+1}}^{\xi_{i+1}^{k+1}} \varphi_j(u)du}{\xi_{i+1}^{k+1} - \xi_i^{k+1}} \tag{12}$$

we obtain the following operator T_m (integral VDS operator)

$$(S_m\mu_i\varphi_j) = (T_m\varphi_j) = (T_mP)_j \quad j = 1, 2, 3. \tag{13}$$

The T_m operator can be used to generate a new curve model and in matrix form it can be written as

$$(T_m\mathbf{P})(t) = \mathbf{b}_m(t)(M\mathbf{P}) \qquad 0 \le t \le 1 \tag{14}$$

where matrix M has the following form

$$M = \begin{bmatrix} \beta_0 & \gamma_0 & 0 & \dots & 0 \\ \alpha_1 & \beta_1 & \gamma_1 & \dots & 0 \\ 0 & \alpha_2 & \beta_2 & \dots & 0 \\ 0 & \dots & \dots & & \gamma_{m-1} \\ 0 & \dots & \dots & \beta_m & \gamma_m \end{bmatrix} \tag{15}$$

with

$$\begin{aligned} &\alpha_0 = 0, \qquad\qquad \alpha_i = \frac{(\delta_i^l)^2}{2\Delta_{i-1}^k \Delta_i^{k+1}},\ i = 1, ...m \\ &\gamma_i = \frac{(\delta_i^r)^2}{2\Delta_i^k \Delta_i^{k+1}}, \quad i = 1, ..., m-1, \quad \gamma_m = 0 \\ &\beta_i = 1 - \alpha_i - \gamma_i,\ i = 1, ..., m \end{aligned} \tag{16}$$

$$\begin{aligned} &\Delta_i^k = \xi_{i+1}^k - \xi_i^k,\ \delta_i^r = \xi_{i+1}^{k+1} - \xi_i^k \\ &\delta_i^l = \xi_i^k - \xi_i^{k+1}, \ \ \xi_i^{k+1} < \xi_i^k < \xi_{i+1}^{k+1} \end{aligned} \tag{17}$$

Equation (14) shows that the integral spline can be regarded as the VDS operator produced by a new control points set $\tilde{\mathbf{P}}$, obtained transforming in a global way the given set $\mathbf{P}$: *e.g.* $\tilde{\mathbf{P}} = M\mathbf{P}$. It follows that

$$(T_m\mathbf{P})(t) = (S_m\tilde{\mathbf{P}})(t) \qquad 0 \le t \le 1 \tag{18}$$

The obtained curve model is characterized by the following properties:

- it is invariant under affine transformations of the coordinate system;
- the whole curve lies inside the convex hull of the control polygon (the piecewise line whose vertices are the control points);
- it is uniquely determined by its control polygon and no two polygons produce the same curve;
- it crosses an arbitrary plane no more then does the control polygon ;
- it reproduces points and lines.

A further step is the introduction of a shape parameter λ in the VDS integral operator (see [9]). The integral mean expression in (12) is replaced by

$$\mu_i^\lambda \varphi_j(t) = \frac{\int_{\zeta_i}^{\eta_i} \varphi_j(u)du}{\eta_i - \zeta_i} \tag{19}$$

where

$$\zeta_i = (1-\lambda)\xi_i^k + \lambda\xi_i^{k+1}, \quad \eta_i = (1-\lambda)\xi_i^k + \lambda\xi_{i+1}^{k+1} \tag{20}$$

with $0 \le \lambda \le 1$.

In matrix form this new operator can be written as

$$(T_m^\lambda \mathbf{P})(t) = \mathbf{b}_m(t)(M^\lambda(\lambda)\mathbf{P}) \qquad 0 \le \lambda \le 1 \tag{21}$$

where

$$M^\lambda(\lambda) = \begin{bmatrix} \beta_0^\lambda & \gamma_0^\lambda & 0 & ... & 0 \\ \alpha_1^\lambda & \beta_1^\lambda & \gamma_1^\lambda & ... & 0 \\ 0 & \alpha_2^\lambda & \beta_2^\lambda & ... & 0 \\ 0 & ... & ... & & \gamma_{m-1}^\lambda \\ 0 & ... & ... & \beta_m^\lambda & \gamma_m^\lambda \end{bmatrix} \tag{22}$$

with

$$\alpha_i^\lambda = \lambda\alpha_i, \ \beta_i^\lambda = 1 - \lambda(\alpha_i + \gamma_i), \ \gamma_i^\lambda = \lambda\gamma_i \qquad i = 0, ..., m \tag{23}$$

The operator $(T_m^\lambda \mathbf{P})$ is called integral spline VDS operator, with shape parameter. It can be shown that the λ parameter allows to control the global shape of the curve (whereas with the conventional spline only a local control can be achieved). Moreover this operator shares the same properties of the integral spline operator T_m: in particular it is worthwhile to underline that the convex hull property is valid as long as λ lies in the interval [0,1] (see the following figures).

In order to show the influence of the parameter λ on the shape of the curve we consider the following set of control points

$$\mathbf{P}_0 = \begin{bmatrix} 4 \\ 1 \end{bmatrix}, \quad \mathbf{P}_1 = \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \quad \mathbf{P}_2 = \begin{bmatrix} 1 \\ 6 \end{bmatrix}, \tag{24}$$

$$\mathbf{P}_3 = \begin{bmatrix} 3 \\ 8 \end{bmatrix}, \quad \mathbf{P}_4 = \begin{bmatrix} 4 \\ 9 \end{bmatrix}, \quad \mathbf{P}_5 = \begin{bmatrix} 5 \\ 8 \end{bmatrix}, \tag{25}$$

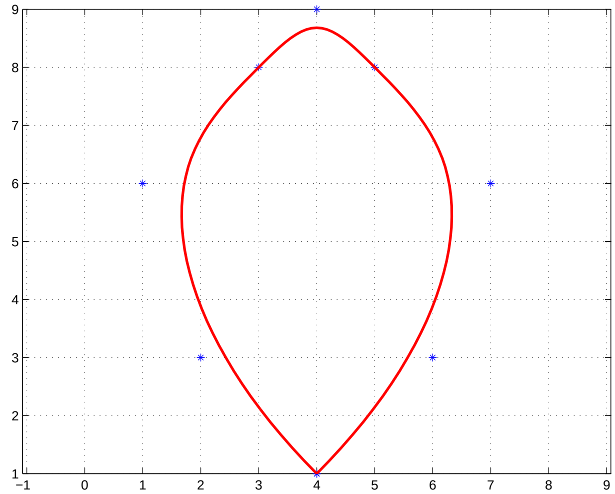

Figure 1. B-spline curve for a given set of control points ($\lambda = 0$).

$$\mathbf{P}_6 = \begin{bmatrix} 7 \\ 6 \end{bmatrix}, \quad \mathbf{P}_7 = \begin{bmatrix} 6 \\ 3 \end{bmatrix}, \quad \mathbf{P}_8 = \begin{bmatrix} 4 \\ 1 \end{bmatrix}. \tag{26}$$

using this set of control points the B-spline curve shown in figure 1 is obtained whereas figure 2 shows the influence of the λ parameter on the shape of the resulting curve for the same set of control points.

4 Fairing

A fairing process consists in detecting and removing irregularities along a curve profile.

The mathematical problem can be stated as follows. First of all, given a set of measured points $\mathbf{Q}_i \in \mathbf{R}^2$, $i = 0, \ldots, n$ we compute the corresponding control points $\mathbf{P}_i$, $i = 0, \ldots, n$ such that the following B-spline curve

$$(S_n\mathbf{P})(t) = \sum_{i=0}^{n} B_{i,k}(t)\mathbf{P}_i \tag{27}$$

interpolates points $\mathbf{Q}_i$. In (27) k is the order of the spline, $\mathbf{t}$ is the knot vector and $B_{i,k}(t)$ are the B-spline basis functions.

We assume that the measured points are obtained by means of an high-accuracy measuring system (*e.g.* CMM) and hence they can be thought to belong to the "real curve": this fact justify the first step of the algorithm in which we build an interpolating curve.

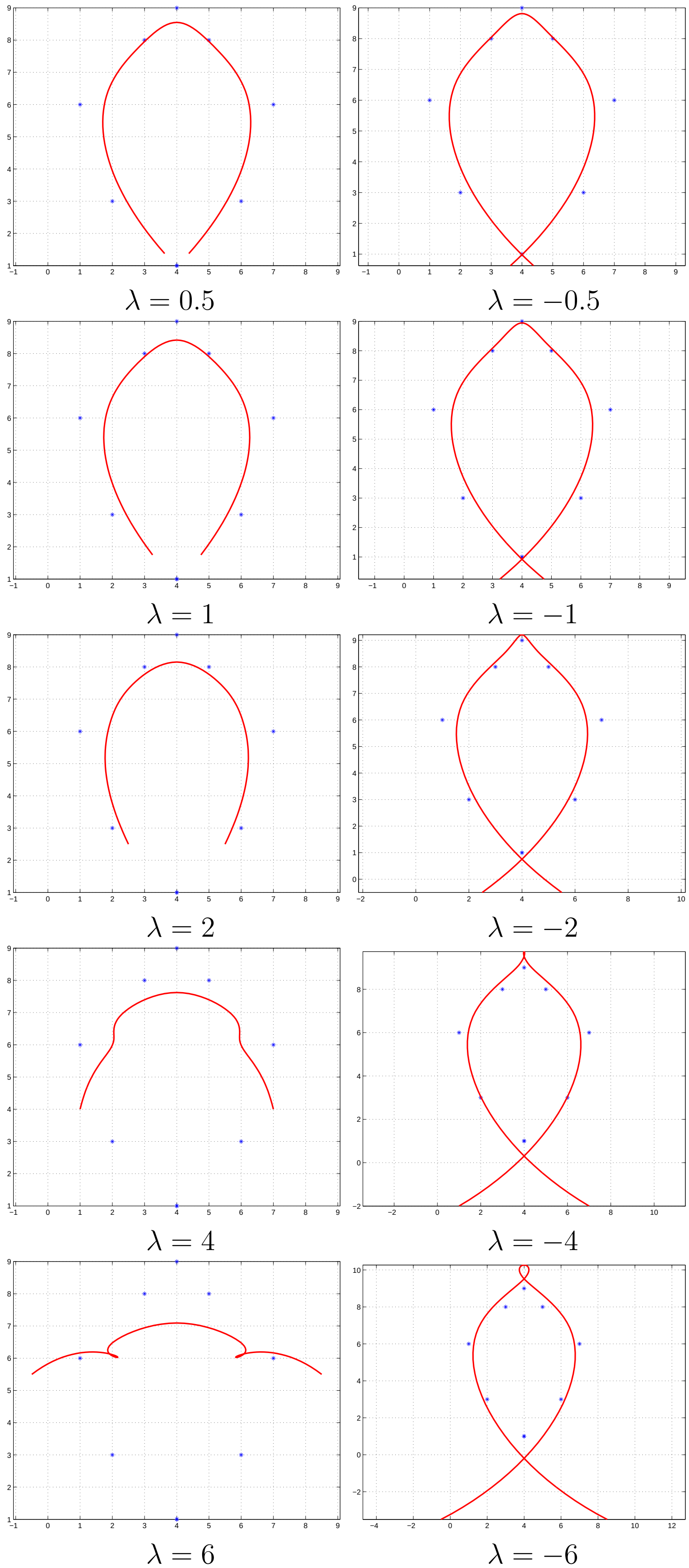

Figure 2. Shapes obtained for different values of λ.

The aim is to find a new spline curve $(S_n\mathbf{P})_{new}(t)$ that minimize a suitable cost functional related to the fairness of the curve. In addition, a shape constraint has to be considered in order to avoid large deviations from the original curve.

4.1 Fairness criteria

The first problem is to determine what fairness means. There is a consensus among authors that the fairness of a curve is related to its curvature and to the way it varies along the curve. If the curvature is monotonically increasing or decreasing the curve shape is considered good. On the other hand, if a curve has large and frequent variations of its curvature this has to be considered as a bad indication. Therefore, the curvature plot can be used to show how the curvature varies and some authors used it for interactive fairing process. For an automatic procedure it is necessary to adopt a fairness criterion that is quantifiable. The two main criteria well known and accepted in literature are:

- C1: A curve is fair if the corresponding curvature plot $k(s)$ is continuous, and is as close as possible to a piecewise monotone function with few as possible monotone pieces [13].
- C2: A curve is fair if it minimizes the integral of the squared curvature $k^2(s)$ with respect to the arc length (*i.e.* the strain energy of a thin elastic beam) [8].

First of all it must be noticed that both these criteria lead to a non-linear problem since the arc-length is also an unknown. Therefore a linearization is required and usually it is assumed that the actual parameter t of the spline curve nearly represents the arc-length.

Furthermore it has been shown (see [3]) that the adoption of criteria C2 is equivalent to considering the following Global Fairing Indicator (GFI)

$$GFI = \sum_i LFI_i, \tag{28}$$

where the Local Fairing Indicator (LFI) is defined as follows

$$LFI_i = (\kappa'')^2 \tag{29}$$

4.2 Description of the algorithm

The fairing algorithm we are advocating can be summarized as follows:

- Step 1.

  Build the initial spline curve interpolating the given points.
- Step 2.

  Evaluate the local fairing indicator (LFI_i) and the corresponding global fairing indicator (GFI).
- Step 3.

  Sort the points in descending order according to their LFI_i. Start processing point i having the highest LFI_i.
- Step 4.

  Consider the following set of five points

  $$\mathbf{L}_i = \mathbf{P}_{i-2}, \mathbf{P}_{i-1}, \mathbf{P}_i, \mathbf{P}_{i+1}, \mathbf{P}_{i+2}$$

  The local procedure creates a new set of control points

  $$\mathbf{L}_i^N = \mathbf{P}_{i-2}^N, \mathbf{P}_{i-1}^N, \mathbf{P}_i^N, \mathbf{P}_{i+1}^N, \mathbf{P}_{i+2}^N$$

  using the λ-spline operator introduced in section 2: in particular the "optimal" value of the shape parameter is computed so that the LFI_i is minimized. After this minimization procedure the new GFI (GFI_{new}) is computed: if $GFI_{new} < \tau GFI$ (where τ is a relaxation parameter) the movement is accepted and we come back to Step 2; if $GFI_{new} > \tau GFI$ we have two possibilities: if i is the last point the procedure is finished and we go to Step 5 otherwise the movement is rejected and the following highest LFI_i point is processed.
- Step 5.

  Build the final curve using the new global set of computed control points.

In figure 3 a schematic flow chart of the algorithm is presented.

It is worthwhile to notice that the local minimization procedure is a constrained procedure since λ can assume only values in the interval [0,1]: we have seen that values outside this interval do not assure the "convex hull" property. In this sense the approach we are advocating does not require the explicit imposition of the shape constraint.

5 Numerical tests

In this section both academic and real-world test are presented in order to assess the effectiveness of the proposed algorithm.

The numerical results of this section have been obtained using cubic basis function and a relaxation parameter $\tau = 0.99$.

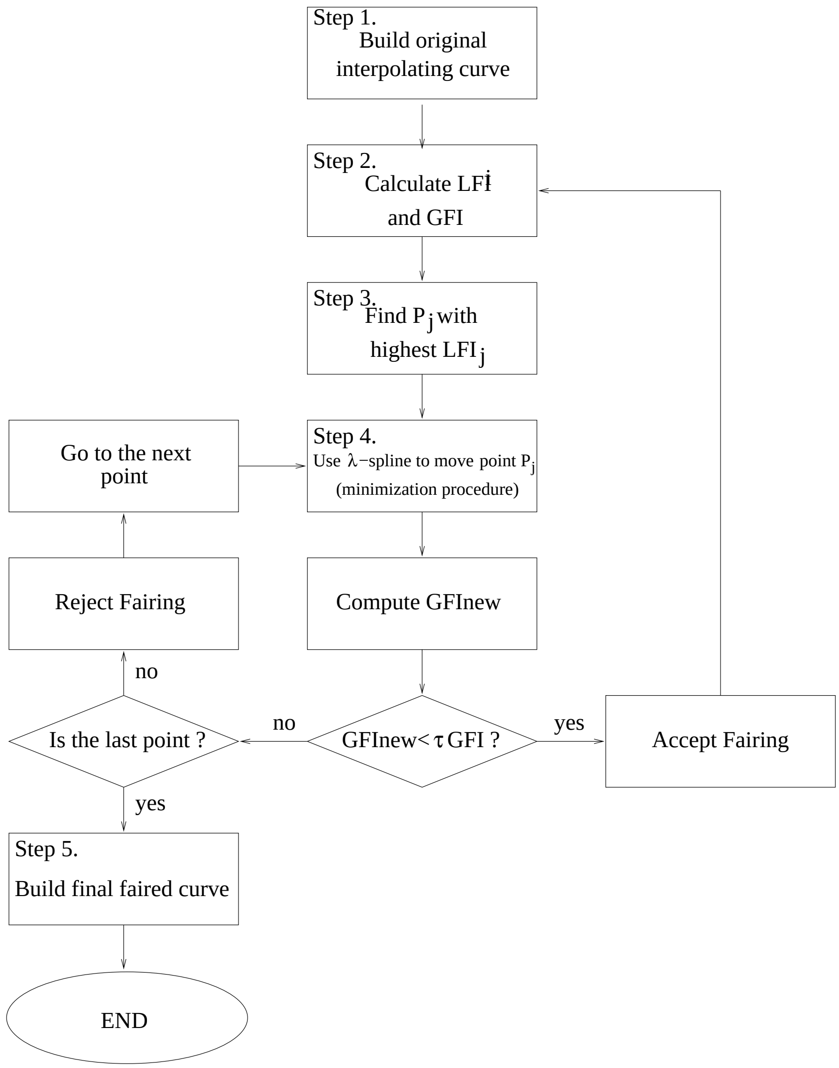

Figure 3. Flow chart of the algorithm

5.1 Strophoid

In the first case we consider the following parametric curve

$$
\begin{aligned}
x &= \frac{t^3 - t}{t^2 + 1}, \\
y &= \frac{t^2 - 1}{t^2 + 1},
\end{aligned}
\tag{30}
$$

with $-2 \leq t \leq 2$. We take 31 equally spaced points on the curve and we perturbe 25 internal nodes with a random noise characterized by a normal distribution with zero mean and variance equal to 0.15. Figure 4 shows the original noised curved in dashed line and the corresponding faired curve. In figures 5 and 6 the curvature plots of the original noised curved and of the

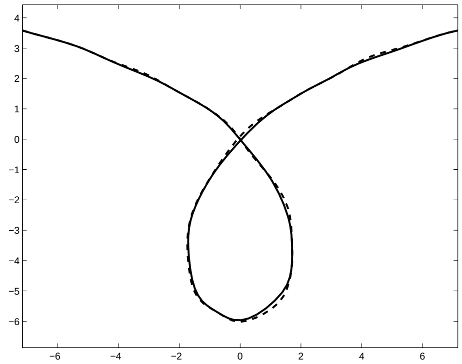

Figure 4. Original perturbed curve (dashed) and the faired one (solid).

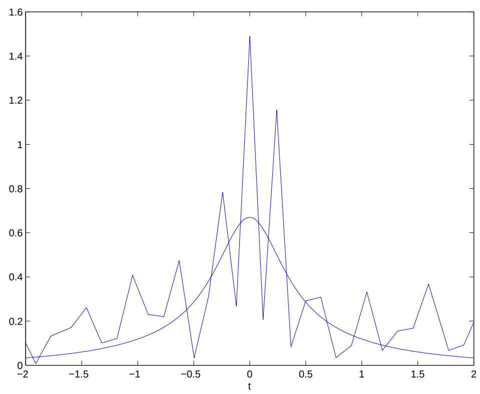

Figure 5. Curvature plot of the original noised curved (dashed) and of the analytic curve (solid).

faired curve are compared with the curvature plot of the analytic curve: we can see that, despite of the presence of some oscillations, the curvature plot of the faired curve is very close to the analytic one; moreover the starting maximum value of the unsigned curvature is 1.5 whereas after the fairing procedure this maximum value is approximately 0.8 (the analytic one is about 0.7). Finally we can see in figure 7 that the starting value of the GFI was about 75 and after the fairing it has been decreased to 10; it is worthwhile to notice that the faired curve has been obtained moving only 7 points (circles in figure 7).

5.2 Sole profile

In the second test case we consider a sole profile (see figure 8). The profile is described by 202 points which have been acquired using a CMM. Figure 9 shows the comparisons between the curvature plot before (dashed) and after (solid) the fairing procedure: we have obtained an improvement of an order of magnitude. Figure 10 shows three different details of the profile: in all these cases we can see that the fairing algorithm has performed very well in

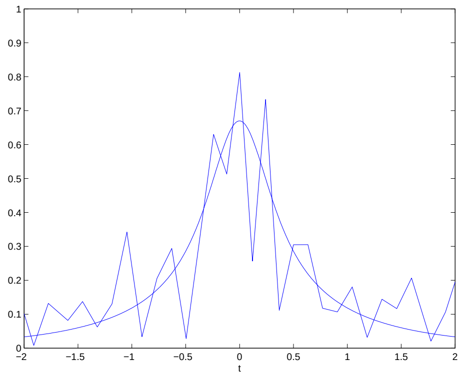

Figure 6. Curvature plot of the faired curve (dashed) and of the analytic curve (solid).

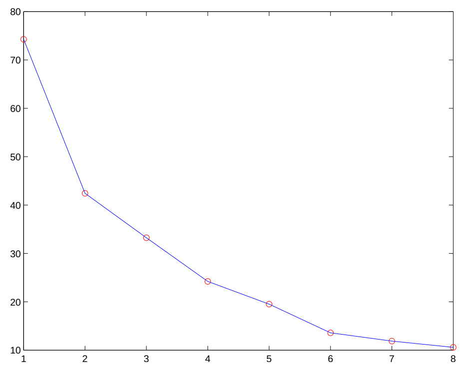

Figure 7. GFI trend during the fairing procedure for the analytic test case.

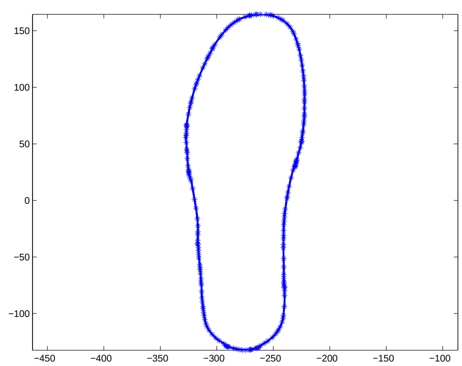

Figure 8. Sole profile.

smoothing the original profile. Finally 11 shows the trend of the GFI during the fairing procedure: in this case the final curve has been obtained moving 33 points.

The second real-world test concerns the lateral profile of the sole. As in the previous situation the points have been acquired using a CMM machine: in this case we are considering 151 points (see figure 12). Figure 13 shows the comparisons between the curvature plot before (dashed) and after (solid) the fairing procedure: we have obtained an improvement of an order of magnitude.

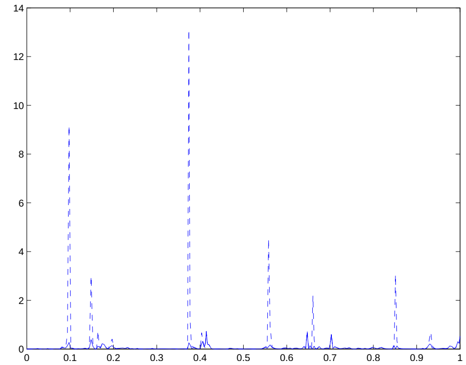

Figure 9. Comparison of the curvature plot of the original profile (dashed) and of the faired profile (dashed).

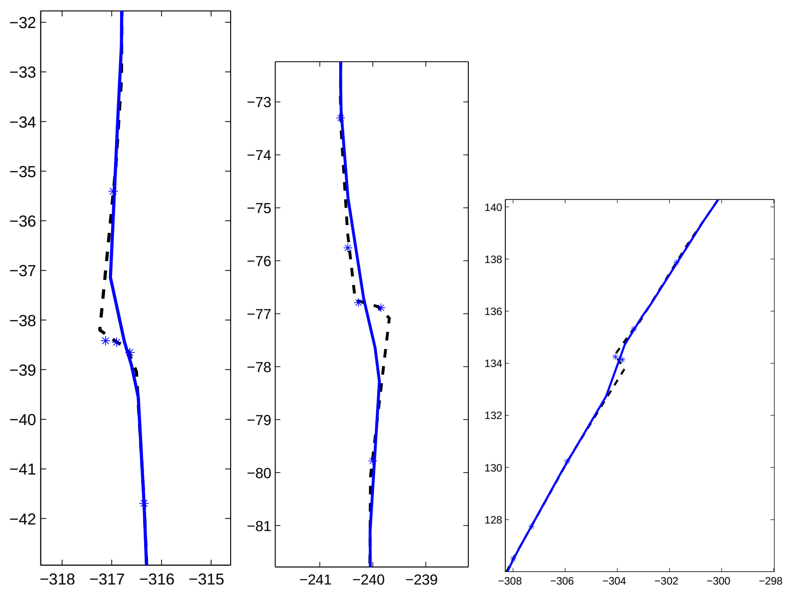

Figure 10. Three details of the faired profile (solid) with respect to the original one (dashed).

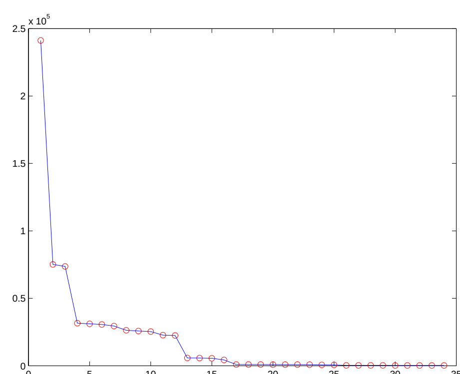

Figure 11. GFI trend during the fairing procedure for the sole profile.

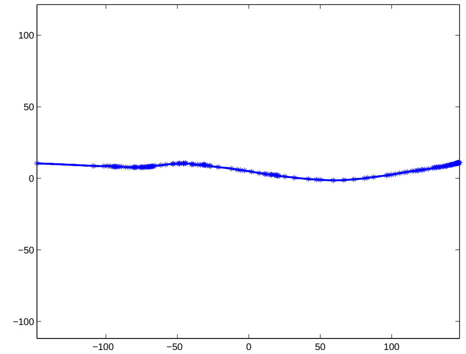

Figure 12. Lateral Sole profile.

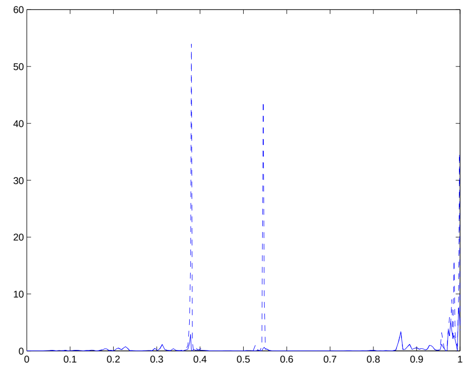

Figure 13. Comparison of the curvature plot of the original profile (dashed) and of the faired profile (dashed).

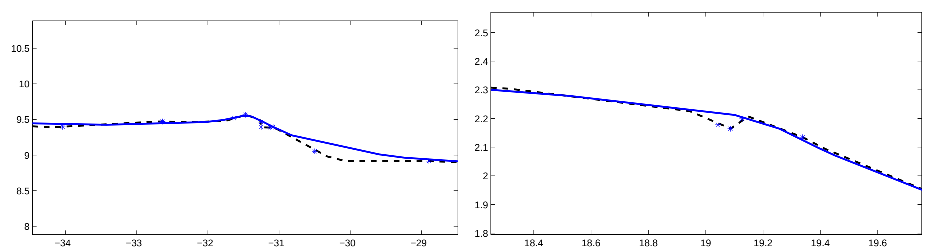

Figure 14. Two details of the faired profile (solid) with respect to the original one (dashed).

Figure 14 shows three different details of the profile: in all these cases we can see that the fairing algorithm has performed very well in smoothing the original profile. Finally 15 shows the trend of the GFI during the fairing procedure: in this case the final curve has been obtained moving 33 points.

6 Conclusion

In this paper a new automatic fairing algorithm has been introduced. The algorithm is based on the modification of the control points set using a suitable

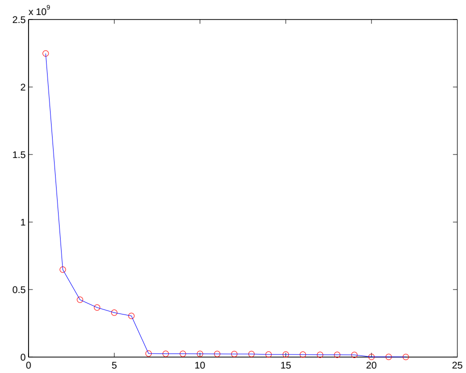

Figure 15. GFI trend during the fairing procedure for the lateral sole profile.

integral spline operator. This class of spline is characterized by a shape parameter which allows to change globally the shape of the spline curve. In the context of the fairing algorithm the ”optimal” value of the shape parameter has been computed in order to minimize a functional related to the fairness of the curve. The preliminary numerical results show the effectiveness of the proposed algorithm in terms of the ability to reduce the unwanted oscillations in the curvature plot of the curve at hand. In [1] the proposed algorithm is compared to existing fairing algorithms.

The extension to 3D curve and to surface is under development and will be the subject of a future work.

References

[1] Caliò F, Miglio E., Moroni G., Rasella M., Curve fairing using integral λ-spline: applications and comparisons with existing fairing algorithms. Submitted to JCISE (ASME Journal of Computing and Information Science in Engineering), 2005.

[2] De Boor, *A Practical Guide to Splines* Springer, New York 2001.

[3] M. Eck and J. Hadenfeld Local Energy Fairing of B-Spline Curves, *Computing Supplementum Geometric Modelling* **10** (1995) 129–147.

[4] Y. Li and P. Gu, Free-form surface inspection techniques state of the art review, *Computer-Aided Design* **36** (2004) 1395–1417.

[5] M. Hosaka, Theory of curves and surfaces synthesis and their smooth fitting *Information Processing in Japan* **9** (1969) 60–68.

[6] J. A. Kjellander, Smoothing of cubic parametric splines, *Computer Aided Design* **15** 3 (1983) 175–179.

[7] W. Li, S. Xu, J. Zheng and G. Zhao, Target curvature driven fairing algorithm for planar cubic B-spline curves, *Computer Aided Gemetric Design* **21** (2004) 499–513.

[8] Mehlum E., *Nonlinear splines* In Barnhill R.E.; Riesenfeld R.F. (eds): Computer Aided Geometric Design, Academic Press pp. 173–208, 1974.

[9] G.V. Milovanovic and L. J. M. Kocic, Integral spline operators in CAGD, *Atti Sem. Mat. Fis. di Modena* 1991, 433–454.

[10] E. Narli and K. Sarioz, The automated fairing of ship hull lines using formal optimisation methods, *Turkish J. Eng. Env. Sci.* **28** (2004) 157–166.

[11] J. F. Poliakoff, Y. K. Wong and P. D. Thomas, An automated curve fairing algorithm for cubic B-spline curves, *Journal of Computational and Applied Mathematics* **102** (1999) 73–85.

[12] W. Renz, Interactive smoothing of digitized point data, *Computer Aided Design* **14** 5 (1982) 267–269.

[13] N. Sapidis and G. Farin, Automatic fairing algorithm for B-spline curves, *Computer Aided Design* **22** 2 (1990) 121–129.

[14] T. Speer, M. Kuppe and J. Hoschek, Global reparametrization for curve approximation, *Computer Aided Geometric Design* **15** (1998) 869–877.

[15] T. Varady, R.R. Martin and J. Cox , Reverse engineering of geometric models - an introduction, *Computer Aided Design* **29** (1997) 255–299.

[16] C. Zhang, P. Zhang and FF. Cheng, Fairing spline curve and surfaces by minimizing energy, *Computer Aided Design* **33** (2001) 913–923.