# Total curvature variation fairing for medial axis regularization

Florian Buchegger<sup>a,b</sup>, Bert Jüttler<sup>a</sup>, Mario Kapl<sup>a,\*</sup>

<sup>a</sup>Institute of Applied Geometry, Johannes Kepler University, Linz, Austria <sup>b</sup>MTU Aero Engines GmbH, Munich, Germany

#### Abstract

We present a new fairing method for planar curves, which is particularly well suited for the regularization of the medial axis of a planar domain. It is based on the concept of total variation regularization. The original boundary (given as a closed B-spline curve or several such curves for multiply connected domains) is approximated by another curve that possesses a smaller number of curvature extrema. Consequently, the modified curve leads to a smaller number of branches of the medial axis. In order to compute the medial axis, we use the state-of-the-art algorithm from [1] which is based on arc spline approximation and a domain decomposition approach. We improve this algorithm by using a different decomposition strategy that allows to reduce the number of base cases from 13 to only 5. Moreover, the algorithm reduces the number of conic arcs in the output by approx. 50%.

#### Keywords:

medial axis regularization, planar domain, fairing, closed B-spline curve, total curvature variation

# 1. Introduction

The medial axis of a planar domain, which was introduced in 1967 by Blum [2], is a highly useful concept in geometry processing for describing the shape of the domain in an efficient way. The range of possible applications of the medial axis (transform) is huge and includes shape recognition, path planning, mesh generation, offset curve trimming, and font design; cf. [3-7].

Depending on the type of the boundary representation of the domain, there exist different algorithms for the construction of its medial axis. In case of curved boundaries, the computation of the medial axis is a nontrivial problem. Some existing algorithms performing this computation are [1, 8, 9, 10, 11, 12]. In this work, we will use a slightly improved version of the method described in [1], which is based on domain decomposition and an arc spline approximation of the given Bspline boundary of the domain.

The medial axis of a connected planar domain is an embedded planar graph, typically with curved edges. For such a domain with a  $C^2$ -smooth boundary (with counter-clockwise orientation for the outer boundary,

Preprint submitted to Elsevier

and clockwise orientation for boundaries of holes), each leaf (i.e., each vertex of degree 1) of the medial axis is induced by a local curvature maxima of the boundary curve. More precisely, a curvature maxima generates a leaf if and only if the curvature is positive and the osculating circle is contained in the interior of the domain. Consequently, the number of leafs can be reduced by decreasing the number of curvature maxima and by decreasing the values of the curvature at the remaining maxima. When considering a domain that is bounded by a smooth curve with a high number of local curvature maxima, the medial axis has a large number of branches. Such complicated medial axes are not well suited for certain applications, such as shape recognition. Regularization methods that lead to a simplified medial axis are therefore of interest.

For instance, when a non-regularized medial axis is used for object recognition in Computer Vision, this may falsify the result of this operation. Similarly, when it is used for "blocking" (segmentation) in the context of quadrilateral mesh generation, then the structure of the resulting mesh will be unnecessarily complicated.

The present paper addresses the following problem: Consider a connected planar domain whose boundary is given by one (or several) closed B-spline curve(s). Due to the variations of the curvature (in particular the number of extrema), the medial axis possesses a large

<sup>\*</sup>Corresponding author

*Email addresses:* florian.buchegger@jku.at (Florian Buchegger), bert.juettler@jku.at (Bert Jüttler), mario.kapl@jku.at (Mario Kapl)



Figure 1: (a) An example of a non-regularized medial axis of a simply connected domain with a large number of branches. (b) The resulting regularized medial axis obtained by applying our regularization method.



Figure 2: Direct and indirect method for medial axis regularization.

number of branches, see Figure 1 (a). We present a regularization method that simplifies this medial axis by removing unwanted branches (leafs), see Figure 1 (b).

The existing methods for regularizing the medial axis of a simply connected domain can roughly be classified into two approaches: *direct* and *indirect methods*. Indirect methods simplify the boundary of the domain to obtain a simpler medial axis, whereas direct methods deal with the initial medial axis to regularize it. The main steps of both methods are summarized in Figure 2.

A group of direct methods are the so-called pruning techniques. The amount of existing literature about these methods is large. The papers [13] and [14] present surveys of different existing pruning algorithms.

One such example is the method in [15], which describes a concept for computing a regularized medial axis of a domain, called the  $\lambda$ -medial axis. The  $\lambda$ -medial axis is a subset of the medial axis, which is homotopy equivalent to the domain, provided that the parameter  $\lambda$ is chosen sufficiently small. In some situations, however, the parameter  $\lambda$  has to be selected larger to obtain a good pruning result, which would lead to a loss of important information of the medial axis about the shape of the domain.

More generally, when using the pruning approach, the direct geometric connection between the shape of the domain and its medial axis is lost, since the simplified medial axis is no longer the medial axis of the given domain, and sometimes (depending on which pruning method is used) not even of a simplified domain derived from it.

A direct method, which can usually overcome the problem of losing important information of the medial axis, is the concept of scale axis transform [16]. First, the medial axis transform of the initial domain is computed and is used to extend the domain by enlarging the maximal disks by a factor s > 1. This extended domain possesses now a simpler medial axis compared to the initial one. By shrinking the maximal disks of the medial axis transform of the extended domain by the same factor *s*, a domain is obtained which is "close" to the initial one and having the same regularized medial axis as the extended domain. This concept has been extended to discrete three-dimensional shapes in [17].

For most direct techniques, the resulting error of the simplified approximated domain can be controlled only indirectly via simple parameters. A recently investigated direct method, which possesses a direct control of the error, is presented in [18]. The idea is to regularize the medial axis by repeatedly removing endpoints of the medial axis until the resulting one-sided Hausdorff distance is greater than some chosen tolerance. Furthermore, the obtained medial axis is approximated by several spline curves in such a way that the distance between the boundary of the initial domain and of the simplified domain is minimized.

Another disadvantage of direct techniques is that most direct methods require the computation of the full medial axis of the initial domain to be able to regularize it. This is in contrast to indirect methods, where the medial axis of the initial domain is not needed for its simplification.

The class of indirect methods mainly consists of algorithms for smoothing the boundary of the domain. A smoothing algorithm that is suitable for medial axis regularization should reduce the number of local curvature extrema, while keeping the resulting approximation error as small as possible.

The process of smoothing a B-spline curve with respect to its curvature ("fairing") has been studied thoroughly. The classical approach to fairing is based on the minimization of the integral of the squared (first derivative of the) curvature or of the integral of the squared  $\ell$ -th derivative of the given B-spline curve for computing the modified curve,  $\ell = 2, 3, 4$ . Some examples are [19-27], which include local and global fairing methods.

For instance, Eck and Hadenfeld [19] describe a very efficient iterative and local fairing algorithm. They minimize the integral of the squared  $\ell$ -th derivative of the given B-spline curve by modifying exactly one control point in each iteration step. Since the resulting minimization problem is linear, the method is fast and simple.

An example of a global fairing strategy is explained in [21], where the so-called minimum variation curves are constructed by minimizing the integral of the squared first derivative of the curvature of the B-spline curve.

A recent alternative approach for fairing a given (spline) curve is explained in [28], where the curve is approximated by piecewise clothoid curves.

The existing fairing methods, however, are not well suited for the regularization of the medial axis, since the number of remaining curvature maxima of the boundary curve remains still too high after modifying the given spline curve.

We present now a new fairing method which is especially tailored to the needs of medial axis regularization. The algorithm has been inspired by the concept of total variation regularization in image processing (cf. [29, 30]). Similar to this approach, our fairing method minimizes a linear combination of a least squares approximation term and the integral of the absolute value of the first derivative of the curvature of the B-spline curve. The latter term plays the role of the total variation term. The aim of our approach is to support the creation of fewer monotonic pieces of the curvature (i.e., spiral segments). This allows a substantial reduction of the number of local curvature extrema (maxima) while keeping the approximation error relatively small. Our method is a global optimization strategy which uses the gradient descent method for solving the minimization problem.

Variational fairing methods for level set curves are described in [31, 32, 33]. These algorithms are not directly applicable to B-spline curves and rely on the minimization of the arc length (or higher order functionals) for smoothing.

The remainder of the paper is organized as follows. We start with some basic definitions in Section 2. In particular, we introduce a class of simply connected planar domains, whose boundaries are represented by closed smooth B-spline curves. We also recall the concept of medial axes and describe a method for the medial axis computation. More precisely, we present an improved version of the algorithm from [1] that will be used in our work. In addition, we summarize the main idea of total variation regularization which has originated in image processing.

Section 3 presents a new method for fairing closed B-spline curves by reducing the number of local curvature extrema via total variation regularization. We compare our fairing algorithm with the method of Eck and Hadenfeld [19] (the "E&H method"). For this purpose we demonstrate the effect of the two different fairing algorithms by several examples. In order to make the paper self-contained, the E&H method is summarized in Appendix A.

Section 4 uses our method for fairing the boundary curve of a domain to obtain a simplified structure of its medial axis. We show a few examples of generated medial axes for several domains and compare them again with the results obtained by the E&H fairing method.

#### 2. Preliminaries

We describe the class of connected planar domains that will be considered in this work. Moreover, we give a short overview of the concept of medial axes and explain the algorithm that we will use for medial axis computation. Finally, we recall the concept of total variation regularization which has originated in image processing.

## 2.1. Basic definitions

Let  $\Omega \subset \mathbb{R}^2$  be the closure of a connected bounded open domain, whose boundary  $\partial \Omega$  is described by one (or several) regular, simple, and closed  $C^3$ -smooth Bspline curve(s) **p** of degree  $d \ge 4$ . More precisely, the curve **p** is given by the B-spline representation

$$\mathbf{p}(t) = \sum_{i=0}^{m} N_i^d(t) \mathbf{c}_i, \ t \in [0, 1],$$
(1)

with control points  $\mathbf{c}_i \in \mathbb{R}^2$  and  $m \ge 2d - 1$ . The basis functions  $(N_i^d(t))_{i=0,...,m}$  are B-splines of degree d with respect to an increasing knot sequence  $\mathcal{T} = (t_i)_{i=0,...,m+d+1}$  with  $t_d = 0$  and  $t_{m+1} = 1$ . In order to ensure a  $C^3$ -connection at the closing point  $\mathbf{p}(0) = \mathbf{p}(1)$ , i.e.

$$\frac{\mathrm{d}^{k}}{\mathrm{d}t^{k}}\mathbf{p}(t)\Big|_{t=0} = \frac{\mathrm{d}^{k}}{\mathrm{d}t^{k}}\mathbf{p}(t)\Big|_{t=1}$$

for  $k \in \{0, ..., 3\}$ , it suffices to require that the first *d* control points coincide with the last *d* control points, i.e.

$$\mathbf{c}_i = \mathbf{c}_{i+m-d+1}$$

for  $i \in \{0, \ldots, d-1\}$ , and the knots satisfy

$$t_{i+1} - t_i = t_{m+2+i} - t_{i+m+1}$$

for  $i \in \{0, ..., d-1\}$ .

An example of a domain  $\Omega$ , whose boundary  $\partial \Omega$  is given by such a closed B-spline curve **p**, is shown in Figure 3 (a).

We recall that the curvature  $\kappa$  of the (closed) B-spline curve **p** is given by

$$\kappa(t) = \frac{\dot{p}_1(t)\ddot{p}_2(t) - \dot{p}_2(t)\ddot{p}_1(t)}{(\dot{p}_1^2(t) + \dot{p}_2^2(t))^{\frac{3}{2}}}, \ t \in [0, 1]$$

where  $p_1$  and  $p_2$  are the first and second coordinate function of the curve **p**, respectively. The dots ' and " denote the derivatives with respect to the curve parameter *t*. Since the B-spline curve **p** is regular and at least  $C^3$ -smooth, the curvature  $\kappa$  is at least  $C^1$ .

Figure 4 (see also Examples 2 and 3 below) shows three examples of planar domains, their curvature distribution, the medial axis, and the curvature distribution of an arc spline approximation. (The latter approximation will be discussed later.)

#### 2.2. Medial axis

We recall the notion of the medial axis, which was introduced by Blum in [2].

A disk  $D_r(\mathbf{z}) \subset \mathbb{R}^2$  with radius r > 0 and center  $\mathbf{z} \in \mathbb{R}^2$  is the set

$$D_r(\mathbf{z}) = \{\mathbf{x} \in \mathbb{R}^2 | \|\mathbf{x} - \mathbf{z}\| \le r\}.$$

We say that a disk  $D_r(\mathbf{z}) \subseteq \Omega$ , which is inscribed into (i.e., contained in)  $\Omega$ , is maximal, if there does not exist another inscribed disk  $D_{\bar{r}}(\bar{\mathbf{z}})$  with  $D_{\bar{r}}(\bar{\mathbf{z}}) \neq D_r(\mathbf{z})$  and  $D_{\bar{r}}(\bar{\mathbf{z}}) \subseteq \Omega$ , which contains  $D_r(\mathbf{z})$ , i.e.  $D_r(\mathbf{z}) \subset D_{\bar{r}}(\bar{\mathbf{z}})$ . The set of all maximal inscribed disks  $D_r(\mathbf{z}) \subseteq \Omega$  is called the *medial axis transform* of  $\Omega$  and is denoted by MAT( $\Omega$ ). The *medial axis* of  $\Omega$ , denoted by MA( $\Omega$ ), is simply the set of centers of all disks in MAT( $\Omega$ ).

Equivalently, the medial axis MA( $\Omega$ ) can be defined as the (closure of the) set of all points in  $\Omega$ , that have at least two closest points on  $\partial \Omega$ .

Our assumptions on the domain  $\Omega$  imply that the medial axis MA( $\Omega$ ) is an embedded planar graph with finitely many edges and vertices. In detail, the edges are the bisector curves (branches) of the medial axis MA( $\Omega$ ). The internal vertices, which are also called *branching points*, are the intersections of these curves; and the non-internal vertices, which are also called *leafs*, are induced by local curvature maxima of the boundary curve **p**. Moreover, the branching points

are vertices of valency  $n \ge 3$  and the leafs are vertices of valency one. Figure 3 (b) shows a simple example of a medial axis of a simply connected domain, where the three types of components (i.e. bisector curves, branching points and leafs) are identified.

Our strategy for the regularization of the medial axis in Section 4 will be based on the approximation of the boundary curve  $\mathbf{p}$  by a curve with a smaller number of local curvature maxima to reduce the complexity of the medial axis, see Section 3. It should be noted, however, that some of the local curvature maxima of  $\mathbf{p}$  do not generate a leaf. More information about the theoretical background of medial axes is available in the rich literature on this topic, e.g. [4, 6, 34].

#### 2.3. Computation of the medial axis

A fast and simple method for the medial axis computation of a simply connected domain is explained in [1]. It has later been extended to the case of planar domains of general topology [35]. In this work, we will use a modified algorithm for generating the medial axes, which leads to a smaller number of base cases and a reduced data volume.

As a preprocessing step, the boundary of the domain, which is given by a simple closed B-spline curve **p**, is approximated by spiral biarcs, see [36]. Boundary segments with vanishing curvature (line segments) are kept unchanged. The use of spiral biarcs implies that the arc spline approximation of the boundary preserves the local curvature extrema of the initial boundary curve.

Figure 4 also shows the curvature distribution of the arc spline approximation. The curvature is now piecewise constant, but the distribution of the extremal values has been preserved due to the use of spiral biarcs.

Now, following the approach in [1], we use the randomized Divide & Conquer algorithm medial-axis for computing the exact medial axis of the approximated boundary. It is based on the Domain Decomposition Lemma from [4], which provides a tool for splitting the domain into subdomains in such a way that the union of the medial axes of the subdomains is the medial axis of the whole domain.

In the divide step, a maximal disk D is chosen that splits the boundary  $\partial \Omega$  into  $k \ge 2$  chains. This is handled by the algorithms maximal-disk for k = 2 and maximal-disk\* for k > 2. Then the chains are closed with artificial arcs that lie on  $\partial D$ , which leads to new, separate domains  $\Omega_0 \dots \Omega_k$ .

This disk D is constructed by randomly selecting a curve **b** of the boundary and finding the maximal disk D touching one of the endpoints of **b** instead of the middle point of **b** as suggested in [1]. In this way, **b** does



Figure 3: (a) The boundary  $\partial \Omega$  of the domain  $\Omega$  is a closed B-spline curve **p** of degree 4 with 24 control points **c**<sub>*i*</sub> with uniform knots. (b) The medial axis of the domain  $\Omega$  consists of several components.



Algorithm 1: medial-axis

not have to be subdivided and therefore the number of the newly introduced curves on the boundary is brought down by one in each step. To ensure that this step reduces the size of the problem, each boundary  $\partial\Omega_i$  should consist of less curves than the original boundary  $\partial\Omega$ . Disks which do not fulfill this condition are called "non-

**Input**: **b**, **y**,  $\partial \Omega$  - curve of boundary, endpoint of **b**, boundary Output: D - maximal disk 1 D = half-plane tangent to **b** in **y**; 2 k = number of curves on  $\partial \Omega$ ; **3** for i = 1...k do  $\mathbf{b}_i = i$ -th curve of  $\partial \Omega$ ; 4 if  $\mathbf{b} \neq \mathbf{b}_i \wedge D \cap \mathbf{b}_i \neq \emptyset$  then 5 D = disk at **y** tangent to **b**<sub>*i*</sub>; 6 end 7 8 end 9 return D;



reducing" and cannot be used. Hence, a different disk has to be found. The divide step is applied to the single resulting domains  $\Omega_0 \dots \Omega_k$  as long as no base case is reached.

The difference in the selection of the point and the fact that also every branching point will be found by maximal-disk\* lead to a reduction of the number of base cases from 13 (as in [1]) to only 5, see Figure 5. To verify that a given boundary chain is a base case, the following three criteria have to be fulfilled:

- The number of non-artificial curves is < 3.
- The number of artificial arcs is < 3.
- There are no concave vertices in the chain.

If all resulting domains  $\Omega_i$  are base cases, then the conquer step is applied. First, the medial axes of the



Figure 4: The boundary curves (top row) of the three different domains are described by simple closed B-spline curves. Due to the oscillating curvatures (second row), the medial axes (third row) possess a large number of branches. In addition, the curvature plots of the biarc approximations (bottom row) of the boundaries for the medial axis computation are shown. See Examples 2 and 3.



Figure 5: The five base cases of algorithm medial-axis (1 to 4 from [1]). The dashed lines represent artificial arcs.



Algorithm 3: maximal-disk\*

base cases are computed directly. Second they are glued together at the centers of the maximal disks, which were chosen in the divide step, to obtain the medial axis of the original domain  $\Omega$ .

We have implemented the algorithm medial-axis with the help of the commercial Parasolid<sup>TM</sup> kernel and used it to generate the regularized medial axes of our examples in Section 4.

By using the concept of generalized domains from [35], our implementation can also deal with domains that are multiply connected. Due to space limitations we do not present the details of this approach.

#### 2.4. Total variation regularization

Total variation regularization is a well known concept in image processing (cf. [29, 30]). The idea is to approximate the original function  $u_0(\mathbf{x})$  (e.g. a noisy image) by a function  $u(\mathbf{x})$ , which minimizes the total variation of u, i.e.

$$\int |\nabla u(\mathbf{x})| \mathrm{d}\mathbf{x}$$

In order to keep the unknown function u as close as possible to the original function  $u_0$ , the least squares (or  $L^2$ )

approximation term

$$\int \frac{1}{2} (u(\mathbf{x}) - u_0(\mathbf{x}))^2 d\mathbf{x}$$

is added to the minimization problem. Summing up, the function u is computed by minimizing the objective function

$$\min_{u} \int \frac{1}{2} (u(\mathbf{x}) - u_0(\mathbf{x}))^2 d\mathbf{x} + \alpha \int |\nabla u(\mathbf{x})| d\mathbf{x}$$

which depends on the regularization parameter  $\alpha > 0$ .

As a useful property, total variation regularization encourages the creation of larger monotonic pieces of the newly constructed functions u compared to other minimization methods, while keeping the approximation error relatively small. Indeed, if we consider the set of all functions on an interval [0, 1] with prescribed boundary values  $u(0) = u_0$ ,  $u(1) = u_1$ , then the minimum of the total variation equals  $|u_0 - u_1|$  and it is realized by any monotonic function. We will demonstrate this effect of using total variation regularization on the basis of an example, and we will compare the results with the ones obtained by three other minimization methods.

**Example 1.** We consider the periodic uniform  $C^3$ smooth B-spline function  $u_0(t)$  of degree 4 with 84 control points given in Figure 6 (blue function). We will construct periodic uniform B-spline functions u(t) of degree 4 with 24 control points  $\hat{\mathbf{c}} = (\hat{c}_0, \dots, \hat{c}_{23})$  which minimize the objective function

$$\hat{\mathbf{c}} = \arg\min \int_0^1 \frac{1}{2} (u(t) - u_0(t))^2 dt + \alpha \int_0^1 \eta(u) dt \quad (2)$$

for different smoothing terms  $\eta(u)$ , given by

(a) 
$$\eta(u) = |\frac{d}{dt}u(t)|,$$
  
(b)  $\eta(u) = (\frac{d}{dt}u(t))^2,$   
(c)  $\eta(u) = (\frac{d^2}{dt^2}u(t))^2,$  and





Figure 6: Example 1. (a-d) The resulting periodic B-spline functions u (black) for a given periodic B-spline function  $u_0$  (blue) by minimizing the objective function (2) for different choices of  $\eta(u)$  (a-d). The weight  $\alpha$  was chosen such that the approximation term satisfies v = 0.0289. (e) The periodic B-spline function u (black) obtained by pure least squares fitting (i.e.  $\alpha = 0$ ). The box in the center shows a close-up view.

(d) 
$$\eta(u) = (\frac{d^3}{dt^3}u(t))^2$$
.

The optimization problem (2) combined with the term (*a*) is now exactly the total variation regularization problem. For this minimization problem we generate a periodic B-spline function *u*, see Figure 6(a) (black function), which has an approximation term with the value v = 0.0289. In addition, we compare this function with the resulting functions obtained by minimizing (2) combined with the terms (b-d), see Figure 6(b-d) (black functions). In order to obtain comparable results we choose the parameter  $\alpha$  for the different optimization problems in such a way that we get the same values for the approximation terms. One can see that the total variation regularization creates fewer and larger monotonic segments of the function *u* compared to the other three minimization methods.

In the next section we will modify this concept for fairing planar B-splines curves. We reduce the number of local curvature extrema by reducing the number of monotonic segments of the curvature.

#### 3. Total curvature variation (TCV) fairing

We describe a method for fairing the boundary curve  $\mathbf{p}$  with respect to the number of local curvature extrema. This will be achieved by adapting the concept of total variation regularization, which originated in image processing (see [29, 30] and Section 2.4), to the curve fairing problem.

Later we will use this method in Section 4 to regularize the medial axis MA( $\Omega$ ). Using a modified boundary curve that possesses a smaller number of local curvature extrema reduces the number of leafs, branching points and edges of the medial axis MA( $\Omega$ ). The total curvature variation (TCV) fairing algorithm generates a simple closed B-spline curve  $\mathbf{q} : [0, 1] \rightarrow \mathbb{R}^2$  which approximates the given curve  $\mathbf{p}$  and possesses a reduced number of local curvature extrema compared to the original curve  $\mathbf{p}$ . The newly constructed curve  $\mathbf{q}$ should have the same degree *d*, the same number of control points *m*, the same knot sequence  $\mathcal{T}$ , and therefore also the same smoothness as the original curve.

The curve **q** is constructed as follows. Let  $\bar{\mathbf{c}} = (\bar{\mathbf{c}}_0, \dots, \bar{\mathbf{c}}_m)$  be the unknown coefficients of the curve **q**, which are computed by solving the minimization problem

$$\bar{\mathbf{c}} = \arg\min f(\bar{\mathbf{c}}),$$
 (3)

subject to the constraint

$$\bar{\mathbf{c}}_i = \bar{\mathbf{c}}_{i+m-d+1}$$

for  $i \in \{0, ..., d - 1\}$ , where

$$f(\bar{\mathbf{c}}) = \omega_1 \underbrace{\int_0^1 (\mathbf{q}(t) - \mathbf{p}(t))^2 dt}_{g(\bar{\mathbf{c}})} + \omega_2 \underbrace{\int_0^1 |\dot{\mathbf{k}}(t)| dt}_{h(\bar{\mathbf{c}})}$$

where  $\dot{k}$  is the first derivative of the curvature function of the unknown curve **q**. The parameters  $\omega_1 > 0$  and  $\omega_2 > 0$  are used to control the relative influence of the *approximation term*  $g(\bar{c})$  and of the *curvature term*  $h(\bar{c})$ in the minimization process, respectively. The curvature term  $h(\bar{c})$  extends the idea of the total variation (TV) term that is used in TV regularization in image processing.

Since the optimization problem (3) is highly nonlinear and the objective function  $f(\mathbf{\bar{c}})$  is  $C^1$ -smooth in the generic case, we use a simple gradient descent method to solve it. Starting with the initial coefficients  $\mathbf{\bar{c}}^{(0)} = \mathbf{c}$ of the given B-spline curve **p**, we iteratively compute the coefficients  $\mathbf{\bar{c}}^{(k+1)}$  by

$$\bar{\mathbf{c}}^{(k+1)} = \bar{\mathbf{c}}^{(k)} - \lambda^{(k)} \nabla f(\bar{\mathbf{c}}^{(k)})$$
(4)

where the step size  $\lambda^{(k)}$  is generated in each iteration step with the help of the backtracking line search strategy (cf. [37]). We repeat the step (4) until

- $\|\bar{\mathbf{c}}^{(k+1)} \bar{\mathbf{c}}^{(k)}\|$  is smaller than some chosen tolerance,
- the *L*<sup>2</sup> error (i.e., the approximation term of the objective function) is reduced within a user-defined threshold, or
- until the number of iterations exceeds a userspecified number.

The following example demonstrates the effect of our fairing algorithm and compares the generated boundary curves with the boundaries obtained by the E&H method (cf. [19] and Appendix A).

**Example 2.** We consider the three different domains  $\Omega$  given in Figure 4, whose boundaries are given by a simple closed B-spline curve **p** of degree 4 with a uniform knot sequence. For the first two domains, the boundary curve consists of 84 control points, for the third domain of 164 control points. For all three domains, the corresponding curvature plots show that the curvatures oscillate strongly which means that the single boundary curves have a high number of local curvature extrema.

In order to regularize the medial axis, we want to fair the boundary curves with respect to the number of local curvature extrema by using our total variation-based method and compare the resulting curves in Figure 7 -9 with the ones obtained by the E&H method. More precisely, we use

- (a) the total variation-based method,
- (b) the total variation-based method combined with the E&H method, and
- (c) the E&H method.

For all methods (a-c), the user-defined parameters are reported in Table 1.

First we apply the total variation-based fairing method (a) to the given boundary curves **p**. The resulting curves **q** possess curvature functions, having a better behavior with respect to two properties: On the one hand, for each curve the number of local curvature extrema is much smaller. On the other hand, the amplitudes of the curvature functions have been reduced significantly. The  $L^2$ -norm approximation error, i.e.

$$\boldsymbol{\epsilon} = \left(\int_0^1 (\mathbf{q}(t) - \mathbf{p}(t))^2 \mathrm{d}t\right)^2$$

| Fig.  | $\omega_1$ | $\omega_2$ | # iter. | # pts | l | δ     | # modif. |
|-------|------------|------------|---------|-------|---|-------|----------|
| 7 (a) | 1600       | 1          | 100     | 400   | - | -     | -        |
| 7 (b) | 1600       | 1          | 100     | 400   | 3 | 0.025 | 32000    |
| 7 (c) | -          | -          | -       | -     | 3 | 0.16  | 32000    |
| 8 (a) | 800        | 1          | 50      | 400   | - | -     | -        |
| 8 (b) | 800        | 1          | 50      | 400   | 3 | 0.06  | 32000    |
| 8 (c) | -          | -          | -       | -     | 3 | 0.206 | 32000    |
| 9 (a) | 8000       | 1          | 50      | 1000  | - | -     | -        |
| 9 (b) | 8000       | 1          | 50      | 1000  | 3 | 0.075 | 32000    |
| 9 (c) | -          | -          | -       | -     | 3 | 0.144 | 32000    |

Table 1: Example 2 and 3. The selected parameters for fairing the domain boundaries in Figure 7-9 by applying the total variation-based method (a), the total variation-based method combined with the E&H method (b) and the E&H method (c) to the corresponding initial domains from Figure 4. If no value is specified, then the parameter is not needed for the corresponding method.

for each curve is still relatively small, see Table 2.

But since the resulting curves (a) still possess a quite high number of "small" local curvature extrema, we continue the fairing process of these curves by means of the E&H method with the parameters reported in Table 1. This allows to reduce the number of these extrema even further, while only slightly increasing the  $L^2$ -norm approximation error, compare Table 2. We obtain modified boundary curves (b) with curvature functions having a reduced number of local curvature extrema and having a nicer and more smoothed shape in comparison with the corresponding curves (a) and especially by contrast with the original curves from Figure 4.

We compare these results (for methods (a) and (b)) with the ones obtained by fairing the boundaries *only* with the help of the E&H method to generate smoothed boundary curves (method (c)).

In order to obtain comparable results, we choose the parameters of the algorithm for the different examples in such a way that we obtain a similar  $L^2$ -norm of the approximation errors, see Table 1 and 2.

When using the E&H method (c), the curvature functions of the resulting boundary curves exhibit much stronger and higher amplitudes (2-3-times higher compared to the methods (a) and (b)). Moreover, and even more important for medial axis regularization, the curves (c) possess higher numbers of local curvature extrema compared to the curves (b).

The algorithms (a-c) were implemented in C++. All occurring integrals are computed numerically by using the trapezoidal rule with a variable number of sampling points, see Table 1, and the gradients are obtained with the help of numerical differentiation. The computing times for the construction of the different examples in Figures 7 - 9 are reported in Table 2. The computations were performed on a workstation running the Gen-



Figure 7: The "Square": Modified boundary curves and regularized medial axes (Examples 2 and 3). The three different results were produced by the methods (a-c). The  $L^2$ -norm of the approximation error satisfies  $\epsilon \le 0.06$ , where the bounding box of the domain is about  $20 \times 20$ . The curvature plots of the modified boundaries and of the biarc approximations are also shown.

| domain |           |                | comp   | . time in s | ec.  | approx. error $\epsilon$ |       |       |  |
|--------|-----------|----------------|--------|-------------|------|--------------------------|-------|-------|--|
| Fig.   | #ctrl pts | bound. box     | (a)    | (b)         | (c)  | (a)                      | (b)   | (c)   |  |
| 7      | 84        | $20 \times 20$ | 260.78 | 260.86      | 0.08 | 0.058                    | 0.06  | 0.06  |  |
| 8      | 84        | $21 \times 14$ | 127.55 | 127.64      | 0.09 | 0.068                    | 0.085 | 0.086 |  |
| 9      | 164       | $19 \times 22$ | 742.25 | 742.49      | 0.24 | 0.044                    | 0.061 | 0.061 |  |

Table 2: Example 2 and 3. The computing times (in seconds) needed to generate the modified boundary curves in Figure 7 - 9 from the initial domains in Figure 4 by using the total variation-based method (a), the total variation-based method combined with the E&H method (b) and the E&H method (c) with the parameters reported in Table 1. In addition, the rightmost columns report the corresponding resulting  $L^2$ -norm approximation errors  $\epsilon$ .

too Linux operating system (Intel Core i7-3770 CPU @ 3.40GHz, 4 cores, 32GB RAM, 64bit). The deviation of the modified boundary curves from the original ones are shown in the first rows of Figures 7 - 9. Note that the deviation has been amplified by a factor of 3 in order to make it visible.

Summing up, the combination of TCV fairing with the E&H method (b) leads to the best results with respect to the number of local curvature extrema. If one uses only TCV fairing (a), the global shape of the resulting curvature function becomes nice but some "small" local curvature extrema remain. The E&H method can easily eliminate those, while maintaining a good approximation, see Table 2.

However, using only a standard fairing algorithm (c) – such as the E&H method – does not give satisfactory results. The resulting curvature functions possess more and larger oscillations. Clearly, the better performance of TCV fairing comes at a price: the minimization of the non-linear objective function by the gradient descent method is much more costly than the simple quadratic optimization of the E&H method.

#### 4. Regularization of the medial axis

The fairing method from the previous section enables us to regularize the computation of the medial axis of planar domain  $\Omega$ , whose boundary is described by one (or several) simple closed B-spline curve *p* given by (1). More precisely, we generate a modified boundary curve, which possesses a reduced number of local curvature maxima. This significantly simplifies the structure of the medial axis of the domain, see Section 2.2. Recall that we compute the medial axes using a modified version of the Divide-and-Conquer-type algorithm of [1], see Section 2.3.

On the basis of Example 2, we will demonstrate the potential of our regularization method. Again, we will compare the results obtained by the three different approaches (a-c).

**Example 3.** We consider again the three different domains  $\Omega$  given in Figure 4, whose boundaries are given by a simple closed uniform B-spline curve of degree 4 with 84 or 164 control points, respectively. For all three domains, the medial axes of the initial domains possess a lot of small branches, which occur because of the high number of local curvature maxima of the corresponding boundary curves. Therefore we use the modified boundary curves of these domains computed in Example 2 to generate regularized medial axes of the initial domains with the help of the algorithm medial-axis, see Figure 7 - 9.

For all three domains, the total variation-based method (a) and the total variation-based method combined with the E&H method (b) lead to regularized medial axes of similar quality. Both methods perform significantly better than using solely the E&H method (c).

Since the computations of the medial axes are based on the spiral biarc approximations of the modified boundary curves (cf. Section 2.3), we have also visualized the corresponding curvature plots of the used biarc approximations. Due to the use of spiral biarcs, these approximations preserve the curvature distribution of the modified boundaries.

The computing times for biarc approximation and medial axis computation as well as the number of leafs of the resulting medial axes are reported in Table 3. The calculations were performed on a workstation running the SUSE Linux Enterprise Desktop 11 operating system (Intel Xeon E3-1240 CPU @ 3.30GHz, 4 cores, 16GB RAM, 64bit). Note that we used the Parasolid<sup>TM</sup> geometry kernel to perform the geometric operations.

The following example will demonstrate that our regularization method can also deal with multiply connected domains, whose boundaries are described by several simple closed B-spline curves  $\mathbf{p}$  given by (1). The idea is as follows. We first separately smooth the single boundary curves of the multiply connected domain. Then we compute the medial axis by means of the modified version of the algorithm from [1], described in Section 2.3, by using the concept of generalized do-





Figure 8: The "Hat": Modified boundary curves and regularized medial axes (Examples 2 and 3). The three different results were produced by the methods (a-c). The  $L^2$ -norm of the approximation error satisfies  $\epsilon \le 0.086$ , where the bounding box of the domain is about  $21 \times 14$ . The curvature plots of the modified boundaries and of the biarc approximations are also shown.

| Figure | bia   | rc approx | . time in a | sec.  | # bound. curves |     |     | medial axis time in sec. |      |      |      | # leafs |      |     |     |     |
|--------|-------|-----------|-------------|-------|-----------------|-----|-----|--------------------------|------|------|------|---------|------|-----|-----|-----|
|        | init  | (a)       | (b)         | (c)   | init            | (a) | (b) | (c)                      | init | (a)  | (b)  | (c)     | init | (a) | (b) | (c) |
| 7      | 21.81 | 21.26     | 21.37       | 22.84 | 511             | 339 | 343 | 404                      | 6.64 | 2.39 | 2.42 | 2.28    | 20   | 4   | 4   | 6   |
| 8      | 21.62 | 17.42     | 17.06       | 20.73 | 490             | 229 | 210 | 334                      | 4.44 | 1.67 | 1.30 | 2.35    | 25   | 6   | 5   | 13  |
| 9      | 27.46 | 25.71     | 23.05       | 40.57 | 914             | 622 | 552 | 3731                     | 5.06 | 4.26 | 3.82 | 20.06   | 20   | 7   | 8   | 13  |

Table 3: Example 2 and 3. The times (in seconds) needed for computing the biarc approximation of the boundary, the number of resulting boundary curves and the computing times (in seconds) needed to generate the medial axis for the initial domains ("init") from Figure 4 and of the domains with modified boundaries from Figure 7 - 9, by using the total variation-based method (a), the total variation-based method combined with the E&H method (b) and the E&H method (c). In addition, the rightmost columns report the number of leafs of the resulting medial axes.

mains from [35].

**Example 4.** We consider the multiply connected domain given in Figure 10(a), whose inner and outer boundary of the domain are represented by simple closed uniform B-spline curves of degree 4 with 164 points. The medial axis of the multiply connected domain possesses a large number of branches, which implies that the two boundary curves have a high number of local curvature extrema.

We separately fair the two boundary curves of the domain to reduce the number of local curvature extrema of each curve to obtain a regularized medial axis. More precisely, we apply the same three fairing methods as in Example 2 and 3 to smooth the two single boundary curves and compare the resulting medial axes, see Figure 10(b-d). In order to obtain comparable results we generate boundary curves with a similar  $L^2$ -norm approximation error by using the different methods. Again, the total variation-based method (b) and the total variation-based method combined with the E&H method (c) lead to significantly better results than using solely the E&H method (d).

In practice, our regularization method can be applied to any domain, provided that the boundary curves are  $C^3$  smooth. If the boundary is not described by a simple closed B-spline curve by (1), or if the smoothness of the boundary is too low, one may approximate the boundary by a quartic B-spline curve with the help of least squares fitting (cf. [38]).

In the following example we will regularize the medial axes of two simply connected domains, which are represented as point clouds. The used data has been provided by the authors of the recent paper [18], which describes an error-controlled method for medial axis regularization.

**Example 5.** We consider the two initial domains given in Figure 9(a-b) from [18]. The two domains, i.e. the example of the car and dolphin, consist of 1000 and 800 points, respectively, and possess medial axes with a large number of small branches. As first step, we

use least squares fitting to generate closed uniform Bspline curves of degree 4, which approximate the initial point clouds and which will be used later as initial curves in the TCV fairing process. Figure 11 shows that the resulting boundaries of the domains have already quite regularized medial axes, since least squares fitting produces fairly regular boundary curves. Then we continue the smoothing of the boundaries by applying the total variation-based method combined with the E&H method. By choosing appropriate parameters we obtain regularized medial axes of similar quality compared to the method in [18], see Figure 11.

For this example, we slightly modified the TCV method in such a way that the approximation term  $g(\bar{\mathbf{c}})$  in the minimization problem (3) is replaced by the least squares fitting term

$$\tilde{g}(\bar{\mathbf{c}}) = \sum_{i} (\mathbf{q}(\tau_i) - \mathbf{p}_i)^2,$$

where  $\mathbf{p}_i$  are the initial points with the corresponding parameters  $\tau_i$ . The advantage of this modification is that we still compare the approximation error of the resulting curve  $\mathbf{q}$  with the initial point cloud.

Moreover, we compare the resulting approximations errors by using the TCV + E&H method with the errors obtained by the method in [18]. Since in [18] the approximation error is measured by the one-sided Hausdorff distance  $\tilde{\epsilon}$ , i.e.

$$\tilde{\epsilon} = \max\{ \min\{||\mathbf{p}_i - \mathbf{q}(t)|| : t \in [0, 1]\} \},\$$

we will also use this distance for error comparison in this example, see Table 4. Although the TCV method is controlled by another approximation error, namely by the  $L^2$ -norm approximation error, the resulting onesided Hausdorff distance approximation errors  $\tilde{\epsilon}$  are in the same order of magnitude as for the results of the method described in [18].

### 5. Conclusion

We presented a method for the regularization of the medial axis of a domain. The boundary (or boundaries)



Figure 9: The "Hand": Modified boundary curves and regularized medial axes (Examples 2 and 3). The three different results were produced by the methods (a-c). The  $L^2$ -norm of the approximation error satisfies  $\epsilon \le 0.061$ , where the bounding box of the domain is about  $19 \times 22$ . The curvature plots of the modified boundaries and of the biarc approximations are also shown.



Figure 10: Example 4. Regularized medial axes of a multiply connected domain (a) by using the three different methods (b-d). The  $L^2$ -norm of the approximation error of the inner (outer) boundary curve satisfies  $\epsilon \le 0.022$  (0.049), where the bounding box of the domain is about 40 × 40.

| Initial poi | int cloud | meth. [18]         | Fitted de  | omain              | TCV+E&H            |  |  |
|-------------|-----------|--------------------|------------|--------------------|--------------------|--|--|
| Domain      | # pts     | $\tilde{\epsilon}$ | # ctrl pts | $\tilde{\epsilon}$ | $\tilde{\epsilon}$ |  |  |
| Car         | 1000      | 0.32%              | 254        | 0.13%              | 0.3%               |  |  |
| Dolphin     | 800       | 0.15%              | 254        | 0.10%              | 0.15%              |  |  |

Table 4: Example 5. Comparison of the one-sided Hausdorff distance approximation error  $\tilde{\epsilon}$ , here normalized with respect to the diagonal of the bounding box, by using the method from [18] and the TCV + E&H method for the computation of regularized medial axes of similar quality for the domains from Figure 9(a-b) given in [18], see Figure 11. In addition, we report the resulting approximation errors  $\tilde{\epsilon}$  of the initial domains for our method which are obtained by applying least squares fitting to the initial point clouds.

of the domain is (are) given by a simple closed  $C^3$ smooth B-spline curve. Our technique is based on the new approach of TCV fairing applied to the boundary curve. This method, which is derived from the concept of total variation regularization in image processing (cf. [29, 30]), significantly reduces the number of local curvature extrema of the boundaries and it therefore produces a simplified medial axis.

The potential of our algorithm has been demonstrated by several examples of modified boundaries and of generated regularized medial axes. These examples led us to conclude that the new approach of TCV fairing gives better results than traditional fairing techniques, such as the E&H method [19] for fairing B-spline curves.

Finally we identify three directions for future research. First, it would be interesting to explore methods that improve the computational performance of our method by reducing the computing times. For instance, it should be possible to achieve this by using more sophisticated optimization techniques, provided that the objective function allows for this. Second, one may wish to approach the problem from a more abstract viewpoint, by considering all boundary curves within a certain tolerance and asking for the one(s) that produce the "simplest" medial axis (in an appropriate sense). Finally, the extension of the TCV fairing method to threedimensional objects could be of interest.

Acknowledgment. The authors wish to thank the anonymous reviewers for their comments that helped to improve the paper. They also wish to thank the authors of [18] for providing them with the initial data for Example 5.

This work was supported by the ESF EUROCORES Programme EuroGIGA – Voronoi, Austrian Science Foundation (FWF).

# Appendix A. Eck & Hadenfeld's (E&H) method for fairing B-spline curves

In order to make this paper self-contained, we recall the E&H method [19] for fairing a B-spline curve of arbitrary degree and adapt it to closed B-spline curve **p** given by (1). The idea of the algorithm is to minimize the integral of the squared  $\ell$ -th derivative of the B-spline curve. This is done iteratively by changing *only one control point in each iteration step* and keeping the others fixed. We give a short overview of this fairing process. For more detail we refer to [19].

The algorithm starts with the initial B-spline curve  $\mathbf{p}$ , given by (1). After a certain number *k* of iterations we arrive at the partially modified B-spline curve  $\mathbf{\bar{p}}$ , which is represented by

$$\bar{\mathbf{p}}(t) = \sum_{i=0}^{m} N_i^d(t) \bar{\mathbf{c}}_i, \ t \in [0, 1],$$

with the generated control points  $\bar{\mathbf{c}}_i \in \mathbb{R}^2$ , satisfying again

$$\bar{\mathbf{c}}_i = \bar{\mathbf{c}}_{i+m-d+1}$$



Figure 11: Example 5. Medial axis regularization for the domains from Figure 9(a-b) presented in [18]. Compared to the method in [18], the TCV + E&H method provides results of similar quality having a similar one-sided Hausdorff distance approximation error  $\tilde{\epsilon}$ , too, see Table 4.

for  $i \in \{0, ..., d - 1\}$ . In the k + 1-th iteration step, we compute a new curve

$$\tilde{\mathbf{p}}(t) = \sum_{i=0}^{m} N_i^d(t) \tilde{\mathbf{c}}_i, \ t \in [0, 1],$$

with  $\tilde{\mathbf{c}}_i \in \mathbb{R}^2$ , by fixing all control points of the curve  $\bar{\mathbf{p}}$  from the *k*-th iteration step, i.e.  $\tilde{\mathbf{c}}_i = \bar{\mathbf{c}}_i$ , except one control point, say  $\tilde{\mathbf{c}}_j$ , which is obtained by minimizing the integral of the squared  $\ell$ -th derivative of the B-spline curve  $\tilde{\mathbf{p}}$ , i.e.

$$\tilde{\mathbf{c}}_j = \arg\min \int_0^1 \left(\frac{\mathrm{d}^\ell}{\mathrm{d}t^\ell} \tilde{\mathbf{p}}(t)\right)^2 \mathrm{d}t,$$
 (A.1)

subject to the constraint

$$\max\{\|\tilde{\mathbf{p}}(t) - \mathbf{p}(t)\| : t \in [0, 1]\} \le \delta$$
 (A.2)

for some given tolerance  $\delta$ . Note that if  $j \in \{0, \ldots, d-1\}$ or  $j \in \{m - d + 1, \ldots, m\}$ , then both control points  $\tilde{\mathbf{c}}_j$  and  $\tilde{\mathbf{c}}_{j+m-d+1}$  have to be changed, in order to keep them equal. This is necessary to maintain the order of continuity at the closing point  $\tilde{\mathbf{p}}(0) = \tilde{\mathbf{p}}(1)$ . Therefore we consider from now on only the construction of the new control points  $\tilde{\mathbf{c}}_0, \ldots, \tilde{\mathbf{c}}_{m-d}$ , since the *d* control points  $\tilde{\mathbf{c}}_{m-d+1}, \ldots, \tilde{\mathbf{c}}_m$  are determined by the first *d* control points  $\tilde{\mathbf{c}}_0, \ldots, \tilde{\mathbf{c}}_{d-1}$  or vice versa. The minimization problem (A.1) leads to a linear equation whose solution is

$$\tilde{\mathbf{c}}_j = \sum_{i=j-d \atop i\neq j}^{j+d} \gamma_i \bar{\mathbf{c}}_{i \mod (m-d+1)},$$

where the weighting parameter  $\gamma_i$  is given by

$$\gamma_i = -\frac{\int_0^1 \left(\frac{\mathrm{d}^\ell}{\mathrm{d}t^\ell} N_{i \bmod (m-d+1)}^d(t)\right) \left(\frac{\mathrm{d}^\ell}{\mathrm{d}t^\ell} N_j^d(t)\right) \mathrm{d}t}{\int_0^1 \left(\frac{\mathrm{d}^\ell}{\mathrm{d}t^\ell} N_j^d(t)\right)^2 \mathrm{d}t}.$$

If the the B-spline curve  $\tilde{\mathbf{p}}$  is based on a uniform knot sequence, the parameters  $\gamma_i$  are independent of the index *j* of the chosen point  $\tilde{\mathbf{c}}_j$ . E.g., we obtain for e.g. d = 4 and  $\ell = 3$ 

$$\gamma_{-4} = \gamma_4 = \frac{1}{50}, \ \gamma_{-3} = \gamma_3 = -\frac{1}{25}$$

and

$$\gamma_{-2} = \gamma_2 = -\frac{4}{25}, \gamma_{-1} = \gamma_1 = \frac{17}{25},$$

which leads to an explicit formula for the point  $\tilde{\mathbf{c}}_{j}$ .

To satisfy the side constraint (A.2), we will use the simpler sufficient condition

 $\|\tilde{\mathbf{c}}_j - \mathbf{c}_j\| \leq \delta.$ 

Thereby, if  $\|\tilde{\mathbf{c}}_j - \mathbf{c}_j\| > \delta$ , then we compute a new point  $\tilde{\mathbf{c}}_j^*$  by

$$\tilde{\mathbf{c}}_j^* = \mathbf{c}_j + \delta \frac{\tilde{\mathbf{c}}_j - \mathbf{c}_j}{\|\tilde{\mathbf{c}}_j - \mathbf{c}_j\|}$$

So far we described how to modify one control point in the *k*-th iteration step. Next, we will describe at which point the curve should be modified to obtain the smallest possible value of the integral of the squared  $\ell$ -th derivative of the B-spline curve  $\tilde{\mathbf{p}}$ , i.e.

$$\int_0^1 \left(\frac{\mathrm{d}^\ell}{\mathrm{d}t^\ell} \tilde{\mathbf{p}}(t)\right)^2 \mathrm{d}t.$$

For this purpose we compute for each control point  $\bar{\mathbf{c}}_j$ in the *k*-th iteration step a ranking number  $z_j$  given by

$$z_j = (\mathbf{\bar{c}}_j - \mathbf{\tilde{c}}_j)^2 \int_0^1 \left(\frac{\mathrm{d}^\ell}{\mathrm{d}t^\ell} N_j^d(t)\right)^2 \mathrm{d}t$$

for  $j \in \{0, ..., m - d\}$ , where  $\tilde{\mathbf{c}}_j$  is the corresponding changed control point in the k + 1-th iteration step. Then the control point  $\bar{\mathbf{c}}_j$  with the highest ranking number  $z_j$  will be modified to get the control point  $\tilde{\mathbf{c}}_j$ .

In case that the B-spline curve  $\tilde{p}$  has a uniform knot sequence, the integral

$$\int_0^1 \left(\frac{\mathrm{d}^\ell}{\mathrm{d}t^l} N_j^d(t)\right)^2 \mathrm{d}t$$

is constant for all  $j \in \{0, ..., m-d\}$  and the computation of the ranking number  $z_j$  simplifies to

$$z_j = (\bar{\mathbf{c}}_j - \tilde{\mathbf{c}}_j)^2.$$

The algorithm is repeated until all ranking numbers are smaller than some given tolerance or a maximal number of iterations is reached. Since it can happen that from some point on the algorithm tries to change the same control point all the time, one also specifies a maximal number of iterations for the single control points. If this number is reached for a control point, then this point is no longer considered for modification by the fairing algorithm.

#### References

- O. Aichholzer, W. Aigner, F. Aurenhammer, T. Hackl, B. Jüttler, M. Rabl, Medial axis computation for planar free-form shapes, Computer-Aided Design 41 (2009) 339–349.
- [2] H. Blum, A transformation for extracting new descriptors of shape, in: W. Wathen-Dunn (Ed.), Proc. Models for the Perception of Speech and Visual Form, MIT Press, Cambridge, MA, 1967, pp. 362–380.
- [3] H. Blum, R. N. Nagel, Shape description using weighted symmetric axis features, Pattern Recognition 10 (1978) 167–180.

- [4] H. I. Choi, S. W. Choi, H. P. Moon, Mathematical Theory Of Medial Axis Transform, Pacific J. Math 181 (1997) 57–88.
- [5] H. I. Choi, S. W. Choi, H. P. Moon, N.-S. Wee, New algorithm for medial axis transform of plane domain, Graphical Models and Image Processing 59 (1997) 463–483.
- [6] H. I. Choi, C. Y. Han, The medial axis transform, in: Handbook of computer aided geometric design, North-Holland, Amsterdam, 2002, pp. 451–471.
- [7] H. N. Güsoy, N. M. Patrikalakis, An automatic coarse and finite surface mesh generation scheme based on medial axis transform: Part I, Engineering with Computers 8 (1992) 121–137.
- [8] L. Cao, J. Liu, Computation of medial axis and offset curves of curved boundaries in planar domain, Computer-Aided Design 40 (2008) 465–475.
- [9] W. L. F. Degen, Exploiting curvatures to compute the medial axis for domains with smooth boundary, Comput. Aided Geom. Design 21 (2004) 641–660.
- [10] R. T. Farouki, R. Ramamurthy, Degenerate point/curve and curve/curve bisectors arising in medial axis computations for planar domains with curved boundaries, Comput. Aided Geom. Design 15 (1998) 615–635.
- [11] R. Ramamurthy, R. T. Farouki, Voronoi diagram and medial axis algorithm for planar domains with curved boundaries. II. Detailed algorithm description, J. Comput. Appl. Math. 102 (1999) 253–277.
- [12] M. Ramanathan, B. Gurumoorthy, Constructing medial axis transform of planar domains with curved boundaries, Computer-Aided Design 35 (2003) 619–632.
- [13] D. Attali, J.-D. Boissonnat, H. Edelsbrunner, Stability and Computation of Medial Axes - a State-of-the-Art Report, in: Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration, Springer, 2009, pp. 109–125.
- [14] D. Shaked, A. M. Bruckstein, Pruning medial axes, Computer Vision and Image Understanding 69 (1998) 156 – 169.
- [15] F. Chazal, A. Lieutier, The "λ-medial axis", Graphical Models 67 (2005) 304–331.
- [16] J. Giesen, B. Miklos, M. Pauly, C. Wormser, The scale axis transform, in: Proceedings of the Twenty-fifth Annual Symposium on Computational Geometry, SCG '09, ACM, New York, NY, USA, 2009, pp. 106–115.
- [17] B. Miklos, J. Giesen, M. Pauly, Discrete scale axis representations for 3D geometry, ACM Trans. Graph. 29 (2010) 101:1– 101:10.
- [18] Y. Zhu, F. Sun, Y.-K. Choi, B. Jüttler, W. Wang, Computing a compact spline representation of the medial axis transform of a 2D shape, Graphical Models 76 (2014) 252–262.
- [19] M. Eck, J. Hadenfeld, Local energy fairing of B-spline curves, in: H. Hagen, G. Farin, H. Noltemeier (Eds.), Geometric Modelling, volume 10 of *Computing Supplement*, Springer Vienna, 1995, pp. 129–147.
- [20] W. Li, S. Xu, J. Zheng, G. Zhao, Target curvature driven fairing algorithm for planar cubic B-spline curves, Computer Aided Geometric Design 21 (2004) 499–513.
- [21] H. P. Moreton, C. H. Séquin, Minimum variation curves and surfaces for computer-aided geometric design, in: Designing fair curves and surfaces, Geom. Des. Publ., SIAM, Philadelphia, PA, 1994, pp. 123–159.
- [22] J. F. Poliakoff, Y. K. Wong, P. D. Thomas, An automated curve fairing algorithm for cubic *B*-spline curves, J. Comput. Appl. Math. 102 (1999) 73–85. Special issue: computational methods in computer graphics.
- [23] P. Salvi, H. Suzuki, T. Vaárady, Fast and local fairing of B-spline curves and surfaces, in: F. Chen, B. Jüttler (Eds.), Advances in Geometric Modeling and Processing, volume 4975 of *Lecture*

Notes in Computer Science, Springer, 2008, pp. 155-163.

- [24] P. Salvi, T. Várady, Local fairing of freeform curves and surfaces, in: L. Szirmay-Kalos, G. Renner (Eds.), III. magyar számítógépes grafika és geometria konferencia. Budapest, 2005., NJSZT-GRAFGEO, Budapest, 2005, pp. 113– 118.
- [25] N. Sapidis, G. Farin, Automatic fairing algorithm for B-spline curves, Computer-Aided Design 22 (1990) 121–129.
- [26] J. Ye, R. Qu, Fairing of parametric cubic splines, Mathematical and Computer Modelling 30 (1999) 121–131.
- [27] C. Zhang, P. Zhang, F. F. Cheng, Fairing spline curves and surfaces by minimizing energy, Computer-Aided Design 33 (2001) 913–923.
- [28] S. Havemann, J. Edelsbrunner, P. Wagner, D. Fellner, Curvaturecontrolled curve editing using piecewise clothoid curves, Computers & Graphics 37 (2013) 764 – 773.
- [29] L. I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D 60 (1992) 259–268.
- [30] D. Strong, T. Chan, Edge-preserving and scale-dependent properties of total variation regularization, Inverse Problems 19 (2003) 165–187.
- [31] Y. Wang, S. Song, Z. Tan, D. Wang, Adaptive variational curve smoothing based on level set method, Journal of Computational Physics 228 (2009) 6333 – 6348.
- [32] Y. Wang, D. Wang, A. M. Bruckstein, On variational curve smoothing and reconstruction, Journal of Mathematical Imaging and Vision 37 (2010) 183–203.
- [33] L. Zhu, Y. Wang, L. Ju, D. Wang, A variational phase field method for curve smoothing, Journal of Computational Physics 229 (2010) 2390 – 2400.
- [34] F. Aurenhammer, R. Klein, D. Lee, Voronoi Diagrams and Delaunay Triangulations, World Scientific, Singapore, 2013.
- [35] O. Aichholzer, W. Aigner, F. Aurenhammer, T. Hackl, B. Jüttler, E. Pilgerstorfer, M. Rabl, Divide-and-conquer algorithms for Voronoi diagrams revisited, Comput. Geom. Theory Appl. 43 (2010) 688–699.
- [36] D. S. Meek, D. J. Walton, Spiral arc spline approximation to a planar spiral, J. Comput. Appl. Math. 107 (1999) 21–30.
- [37] J. Nocedal, S. J. Wright, Numerical optimization, Springer Series in Operations Research, Springer-Verlag, New York, 1999.
- [38] J. Hoschek, D. Lasser, Fundamentals of computer aided geometric design, A K Peters Ltd., Wellesley, MA, 1993. Translated from the 1992 German edition by Larry L. Schumaker.