HOT BLADE CUTTINGS FOR THE BUILDING INDUSTRIES

DAVID BRANDER, ANDREAS BÆRENTZEN, ANTON EVGRAFOV, JENS GRAVESEN, STEEN MARKVORSEN, TOKE BJERGE NØRBJERG, PETER NØRTOFT, AND KASPER STEENSTRUP

ABSTRACT. The constructions of advanced architectural designs are presently very labour intensive, time consuming, and expensive. They are therefore only applied to a few prestige projects, and it is a major challenge for the building industry to bring the costs down and thereby offer the architects more variability in the (economically allowed) designs - i.e., to allow them to think out of the box. To address this challenge The Danish National Advanced Technology Foundation (now InnovationsFonden) is currently supporting the BladeRunner project that involves several Danish companies and public institutions. The project aims to reduce the amount of manual labour as well as production time by applying robots to cut expanded polystyrene (EPS) moulds for the concrete to form doubly curved surfaces. The scheme is based upon the so-called Hot Wire or Hot Blade technology where the surfaces are essentially swept out by driving an Euler elastica through a block of EPS. This paper will be centered around the mathematical challenges encountered in the implementation of this idea. They are mainly concerned with the rationalization of the architects' CAD drawings into surfaces that can be created via this particular sweeping and cutting technology.

1. THE NEED FOR LOW COST PROCEDURES

A recurring theme in the architectural industry of today is a conflict between the design ambitions of the architect and the economic realities of fabrication processes. The desire to create unique and attractive designs, often motivated by the competitive industry climate, leads to the use of curved geometries and bespoke elements that can be conceived easily within modern CAD systems, but, in reality, are prohibitively expensive to build. This results in compromises at the so-called *rationalization* stage, where the design is adjusted within an engineering context for production purposes. A typical example is where the desired shape of a building leads to panels (or some other element) of perhaps 200 different shapes. Consultation with fabrication contractors then reveals that dramatic cost reductions can be achieved if the design is adjusted so that only 20 unique elements are used, with repetition, instead of the 200. Finally, budgetary considerations force a compromise of the original design.

The present project addresses this issue, in particular within the domain of the production of formwork for concrete constructions. The shape of the surface of a facade or other element is produced – possibly on location – in negative in several pieces of (easily transported and packed) expanded polystyrene (EPS) foam that is then used as a mould for concrete (in situ) casting. EPS can also be used in positive shape production for some applications by applying a coating and retaining the EPS as a structural element. For curved surfaces the currently available technology for shaping the EPS is computer numerically controlled milling, a slow, and therefore expensive, process. The *BladeRunner* project,

supported by the Innovation Fund Denmark, is presently developing new processes, *robotic Hot Wire/Blade cutting*, for carving shapes out of EPS using a robotically controlled heated wire or blade. The technology is projected to reduce production time of architectural formwork by a factor of over 100, and to bring the cost of production for advanced shapes into the domain of financial feasibility.

FIGURE 1. Robotic Hot Wire cutting in Odense, Denmark.

2. PRINCIPLES OF HOT BLADE CUTTINGS

The essential principle of both Hot Wire and Hot Blade cutting is very simple. A heated wire or blade, either of which we may think of as a "blade", is moved relative to a block of EPS, carving out a surface through the block (Figure 1). Either the block or the blade, or both, are controlled by a robot. For definiteness, we regard the block as fixed and the blade as moving.

2.1. **Hot wire cutting and its limitations.**

For the wire technology, the wire is held tight, forming a straight line, and thus sweeps out a *ruled surface*. This technology is limited in its ability to approximate general freeform surfaces. This can be seen by considering the *Gaussian curvature* of a surface, which is defined as follows: through any point p on a surface a curve is obtained by intersecting the surface with a plane perpendicular to the tangent plane at that point (Figure 2). The *normal curvature* associated to the tangent direction of this curve is the curvature of this curve at the point p, with the sign determined by a fixed choice of surface normal vector (Figure 2, left). The maximum and minimum values obtained from all possible tangent directions at p are called the *principal curvatures*, κ_1 and κ_2, and their product is the Gaussian curvature $K = \kappa_1\kappa_2$. In the saddle surface shown at Figure 2, the principal curves bend in opposite directions away from the tangent plane and so κ_1 and κ_2 have opposite signs and $K < 0$.

If the Gaussian curvature is positive, then κ_1 and κ_2 at p have the same sign, and any other tangent direction at p has normal curvature κ_n with $\kappa_1 \geq \kappa_n \geq \kappa_2$. Therefore κ_n cannot be zero in this case. Now for an arbitrary arc-length parameterized curve γ in the surface the acceleration vector decomposes as $\gamma''(s) = \kappa_g(s)\nu(s) + \kappa_n(s)N(s)$, where N is the surface normal, and κ_n is the normal curvature in the direction of γ'. It follows that, if $\kappa_n \neq 0$, then the acceleration is non-zero and thus the curve cannot be a straight line. On

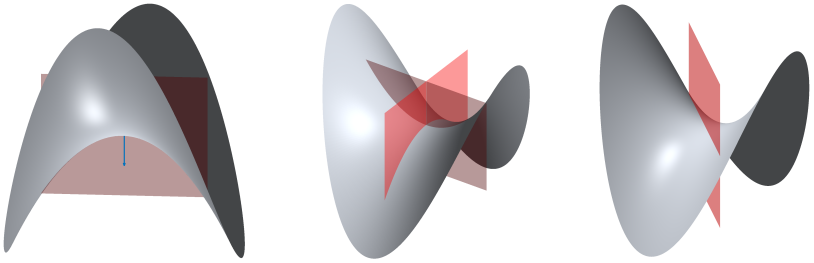

FIGURE 2. A saddle surface. Left: the normal curvature defined by this intersection curve is positive if the downward pointing surface normal is chosen. Middle: the planes defining the principal curvatures at the center. Right: this normal section is a straight line; the normal curvature is zero in this direction.

the other hand, a ruled surface is defined to be a surface swept out by a smoothly varying family of straight lines: through every point of a ruled surface there is a straight line lying in the surface. Therefore, by the discussion above, a ruled surface cannot have positive Gaussian curvature; moreover, there is no chance of obtaining a good local approximation for a positively curved surface by a ruled surface (Figure 3).

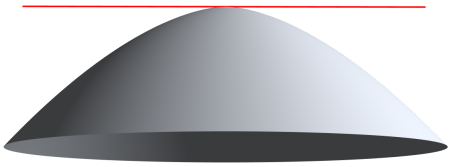

FIGURE 3. At a point of positive Gaussian curvature the surface is bowl-shaped. No straight line tangent to the surface can approximate a curve in the surface to more than first (tangential) order.

Figure 4 shows (center) an approximation of a negatively curved surface by ruled strips which can be realized by repeated hot wire cuttings. The ruling directions are chosen to be close to the *asymptotic directions*, namely directions where the normal curvature κ_n is zero. However, even for negatively curved surfaces, it is in general not possible to obtain a tangent continuous approximation - the tangent planes do not match along adjacent strips.

We refer to Section 7 for a concrete modern example of a relevant surface – a skater ramp – which clearly displays all curvatures and thence also the production challenges that we are addressing in this work.

2.2. **Hot blade cutting.** The blade concept is much more general than the wire concept illustrated above: the end points and the *tangents* at the endpoints of the blade (center-)curve can vary relative to each other during the sweeping. We will assume, however,that the curve lies in a plane, that is, that the end tangents are co-planar. This restriction makes the mechanical implementation of the process easier, both in terms of choosing the cross-sectional shape of the blade design and allowing for the possibility that only one edge of the physical blade is heated, as illustrated to the right in Figure 8 below.

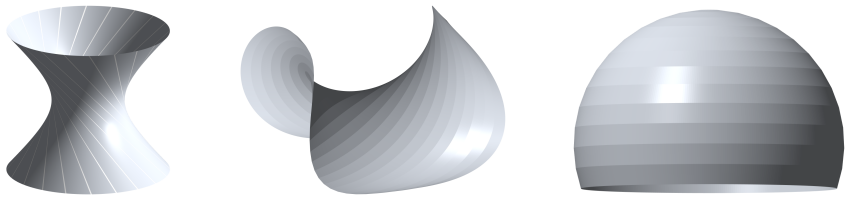

FIGURE 4. Left to right: The hyperboloid (a ruled surface with automatic tangent matching along adjacent strips); Approximation of a negatively curved surface by *strips* of ruled surfaces; Ruled strip approximation of a positively curved surface.

An elastic rod, of a fixed length and with end points and end tangents at a given position, assumes the shape of an Euler elastica (discussed below). These curves are well understood mathematically and are given in terms of elliptic functions. We refer to Section 3 below for a brief outline of the parametric representation of the family of all planar elastica.

In order to apply either of these technologies to a given CAD design, a *rationalization* of the relevant surface is necessary: the surface must first be *segmented* into pieces, each of which can be approximated within a given tolerance by a surface swept out by curves of the relevant type (lines or a family of elastica). Next, each segment is foliated by curves each of which is approximated by a curve of the type in question. Finally, the data for producing these curve sweepings is given to the robot control software.

Methods for rationalization for Hot Wire cutting have been given already in the literature (see below). Therefore, in this article, the rationalization project we are concerned with consists of both developing a segmentation algorithm for blade-cut surfaces, and an algorithm for approximating arbitrary spline curves by Euler elastica.

2.3. **Previous related works.** Pottmann and Flörey [7] developed a ruled surfaces segmentation algorithm using the fact that on ruled surfaces one of the asymptotic directions at a point must be tangent to the ruling, giving natural candidates for the ruling direction in the surface to be approximated. As such, this segmentation strategy does not generalize to the case of hot *blade* cutting, therefore a new strategy must be developed.

For the Hot Blade technology, some work has been done in the late 1990's to the early 2000's by a group at Delft: see [8, 4] and associated references. The use of the Hot Blade technology there is somewhat different, as the aim is to produce 3-dimensional solid rapid prototype models from EPS via a so-called "thick-layered fabrication" process. The solid is built up by stacking many thick slices, and the curved surface that needs to be cut is only a narrow strip around the boundary of each slice. Therefore, the segmentation problem is completely different from the surface segmentation problem that will apply to the BladeRunner process.

The work of the Delft group is concentrated on approximating the blade shape and algorithms for tool positioning. The approach they use for approximating the blade shape is to apply a numerical method to minimize the bending energy. Below we will use a different approach that takes advantage of the known analytic solutions for this problem to give an explicit parameterization of the space of solutions. This allows us to move easily in the

space of solutions, calculate gradients, and use standard optimization packages to find an elastic curve that approximates an arbitrary given curve.

3. The Euler elastica

We describe here a parameterization of the space of planar elastic curve segments. More details of this parameterization and further references can be found in [3]. An introduction to the theory of elastic curves, with historical references, can be found in [11]. Other works on the topic of elastic curves as splines are [1, 2, 5, 6].

3.1. **The Euler-Lagrange equation.** We give here a brief derivation of the differential equation that determines the solutions to the elastica problem. The reader unfamiliar with the calculus of variations could take this derivation for granted and proceed directly to the solutions given in the next subsection. Let $\gamma\colon [0,\ell] \to \mathbb{R}^2$ be a plane curve segment parameterized by arc-length, and define an angle function $\theta(s)$ by $\dot\gamma(s) = (\cos\theta(s), \sin\theta(s))$. A curve segment of length ℓ starting at (x_0, y_0) and ending at (x_ℓ, y_ℓ) satisfies

$$x_\ell = x_0 + \int_0^\ell \cos\theta \,\mathrm{d}s\,, \quad y_\ell = y_0 + \int_0^\ell \sin\theta \,\mathrm{d}s\,. \tag{1}$$

Let κ denote the curvature $\theta'(s)$. An *elastica* is a curve that minimizes the *bending energy*

$$\frac{1}{2}\int_0^\ell \kappa(s)^2 \,\mathrm{d}s\,. \tag{2}$$

The equations defining the elastica are obtained from a variational problem: suppose γ is an elastica from (x_0, y_0) to (x_ℓ, y_ℓ) with angle function $\theta(s)$. A smooth variation is given by the family γ_t with angle function $\theta_t(s) = \theta(s) + t\psi(s)$, where ψ is a differentiable function with $\psi(0) = \psi(\ell) = 0$. Applying the method of Lagrange multipliers we find that, if γ minimizes the energy among such curves, then the angle function θ satisfies:

$$\frac{\mathrm{d}^2\theta}{\mathrm{d}s^2} + \lambda_1 \sin\theta - \lambda_2 \cos\theta = 0\,. \tag{3}$$

Setting $(\lambda_1, \lambda_2) = \lambda(\cos\phi, \sin\phi)$, with $\lambda \geq 0$, this becomes $\frac{\mathrm{d}^2\theta}{\mathrm{d}s^2} + \lambda\sin(\theta - \phi) = 0$. Note that $\lambda = 0$ if and only if κ is constant, i.e the curve γ is either a straight line segment or a piece of a circle. If $\lambda \neq 0$, set

$$\widetilde\gamma(s) = \sqrt{\lambda} R_{-\phi}\gamma\left(\frac{s}{\sqrt{\lambda}}\right), \qquad R_\phi = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}. \tag{4}$$

Then $\tilde\gamma$ is a scaled and rotated version of γ and thus also an elastica. Its tangent angle is $\tilde\theta(s) = \theta(s/\sqrt{\lambda}) - \phi$ and it satisfies the normalized pendulum equation $\frac{\mathrm{d}^2\tilde\theta}{\mathrm{d}s^2} = -\sin\tilde\theta$. Hence we conclude that, up to a scaling and rotation of the ambient space, all arc-length parameterized elastica $\gamma\colon [0,1] \to \mathbb{R}^2$, with non-constant curvature κ, can be expressed as:

$$\gamma(s) = \gamma(0) + \int_0^s (\cos\theta(t), \sin\theta(t))\,\mathrm{d}t \tag{5}$$

where

$$\ddot\theta = -\sin\theta\,. \tag{6}$$

3.2. **The space of elastic curve segments.** We now find some suitable parameters to describe the space of elastic curve segments. First, it is well known that the solutions to (6) can be expressed in closed form via the elliptic functions sn, cn, and dn. These solutions can be found in Love [12]. There are two classes of solution curves: those with inflection points (i.e. points where $\kappa = \dot\theta = 0$) and those without inflections. Each class of solutions is a 1-parameter family.

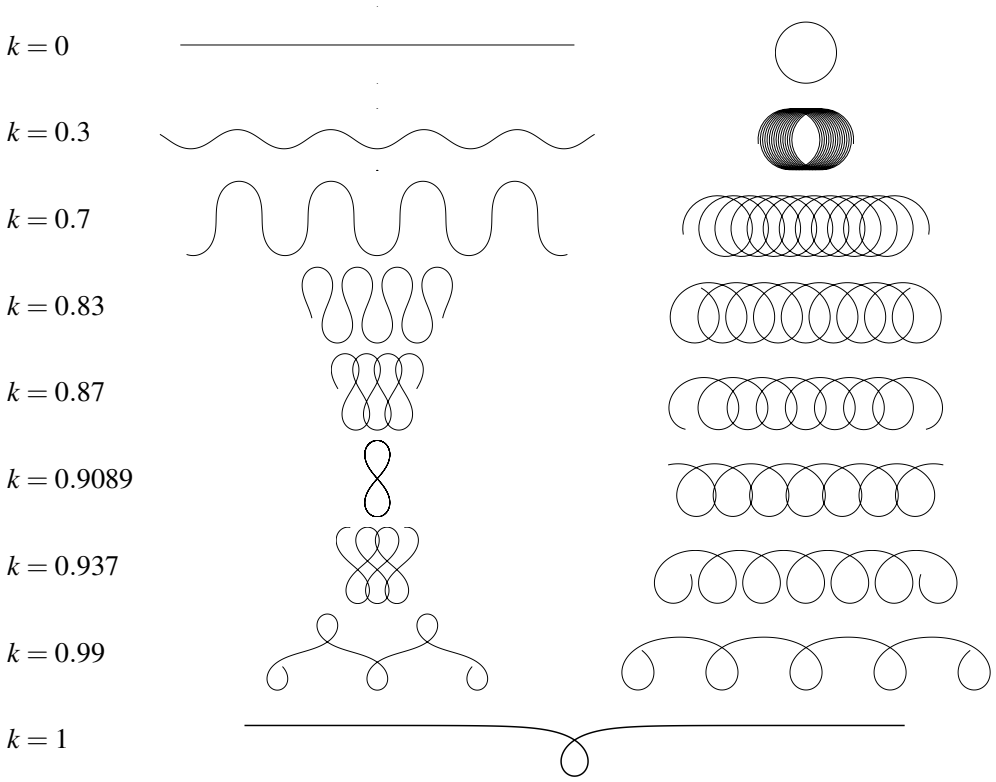

FIGURE 5. Euler elastica. Left: inflectional. Right: non-inflectional. The respective elastica – with values of k ranging from 0 at the top to 1 at the bottom – are plotted.

The *inflectional elastica* starting at $(0,0)$ with initial angle $\theta(0) = 0$ and $\dot\theta(0) > 0$ is

$$\zeta_k(s) = \zeta(s,k) = \begin{pmatrix} 2E(s,k) - s \\ 2k(1 - \mathrm{cn}(s,k)) \end{pmatrix}, \quad \text{where } k = \dot\theta(0)/2.$$

A *segment* of such a curve is determined by the value k, a starting point s_0 and a length ℓ. Finally, adding a scaling S, a rotation ϕ and a translation (x_0, y_0), we have a standard

representation $\gamma_{(k,s_0,L,S,\phi,x_0,y_0)} : [0,1] \to \mathbb{R}^2$ for a segment of an inflectional elastic curve:

$$\begin{aligned}\gamma_{(k,s_0,\ell,S,\phi,x_0,y_0)}(t) &= SR_\phi\left(\zeta_k(s_0+\ell t)\right) + \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \\ &= SR_\phi \begin{pmatrix} 2E(s_0+\ell t,k) - (s_0+\ell t) \\ 2k\left(1-\mathrm{cn}(s_0+\ell t,k)\right) \end{pmatrix} + \begin{pmatrix} x_0 \\ y_0 \end{pmatrix},\end{aligned}$$

where

$$E(s,k) := \int_0^s \mathrm{dn}^2(\tau,k)\mathrm{d}\tau\,.$$

Note that the arc-length parameter in this representation is $s = S(s_0+\ell t)$ and not t and that the length is $L = S\ell$.

Similarly, we obtain a standard representation of a *non-inflectional* elastic curve segment:

$$\gamma_{(k,s_0,\ell,S,\phi,x_0,y_0)}(t) = SR_\phi \begin{pmatrix} (1-\frac{2}{k^2})(s_0+\ell t) + \frac{2}{k}E\left(\frac{s_0+\ell t}{k},k\right) \\ \frac{2}{k}\left(1-\mathrm{dn}\left(\frac{s_0+\ell t}{k},k\right)\right) \end{pmatrix} + \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}.$$

4. Sweeping surfaces with Euler elastica

The figures in this section illustrate examples of surfaces foliated by continuously varying segments of Euler elastica. These examples are constructed by parameterizing the space of planar elastica segments, as in the previous section, choosing a small number of sample curve segments, and then interpolating the data through the parameter space. Hence each surface is swept by a family of planar elastica.

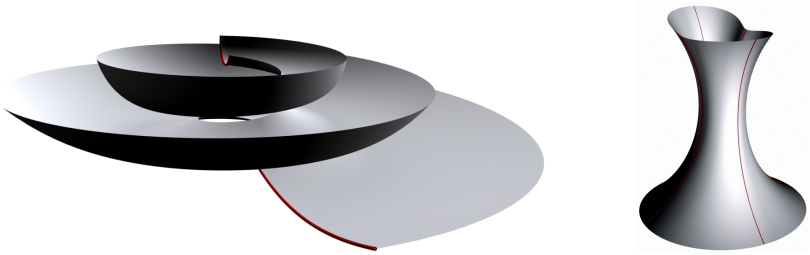

FIGURE 6. Examples of surfaces swept by continuously varying elastic curve segments.

In principle, all of these surfaces could be produced by robotic hot-blade cutting, but there are technical issues that depend on the practical implementation. For example, the surface on the left in Figure 7 is a surface of revolution, but one end of the profile curve is much closer to the axis of rotation than the other. This means that the blade moves much more slowly on the inner end resulting in too much melting of the EPS. One solution is to segment the surface into several pieces, cut separately. Another is to approximate this surface by some other, non-rotational, family of elastic segments.

Yet another restriction arises if a flat blade is used, rather than a cylindrical rod (see Figure 8). With the flat blade design, the blade is curved in a plane perpendicular to the plane of the blade. If one edge of the blade is heated, the motion of the blade should be roughly in the direction of this edge, that is, approximately perpendicular to the plane of the curve, in order to cut a path through the material. Another way to say this is that the

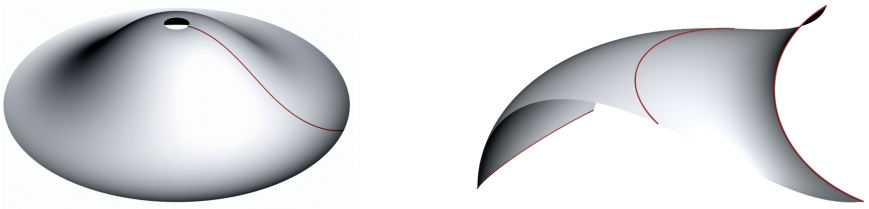

FIGURE 7. Two technically problematic situations.

elastic curves should be as close as possible to geodesic curves (which are characterized by $|\kappa_n| = \kappa = \|\gamma''\|$) on the surface under construction. To require that these planar elastic curves are true geodesics would place too large a restriction on the uses of this method; so we apply a tolerance instead. Both surfaces shown in Figure 6 are reasonable candidates for cutting with a flat blade like the blade to the right in Figure 8. The surface to the right in Figure 7 however, would be impractical with the given elastica foliation. The osculating plane spanned by γ' and γ'' of the elastic curve shown is very close to the tangent plane of the surface; thus the hot edge of the blade is pointing out of the surface, and the blade would not be able to progress in the required direction.

FIGURE 8. Left: A cylindrical rod can cut in any direction that is close to perpendicular to the tangent of the curve. Right: A flat (ribbon) blade design (with its hot edge indicated in red) moves well only in the directions given by the flat extension of the blade past the hot edge itself.

5. APPROXIMATION BY EULER ELASTICA

In this section we consider the problem of approximating a given planar spline curve $\mathbf{x} : [0,1] \to \mathbb{R}^2$ by a planar elastic curve. We present two different approaches to this problem. In the first we try to find the parameters $(k, s_0, \ell, S, \phi, x_0, y_0)$ of the elastica that has the best fit to the curve $\mathbf{x}$. This is a nonlinear optimization problem, and the final result depends crucially on a good initial guess. The second approach is purely numerical – we model the elastica with a spline on a much finer knot vector than the original curve, and then solve a constrained optimization problem minimizing the elastic energy under the constraint of being within some distance to the original curve.

5.1. **Analytic approach: finding the parameters for the elastica.** We describe here the essentials of the gradient driven analytic approach. For full details, see the article [3].

We wish to find the elastic curve segment which most closely resembles the given spline curve $\mathbf{x}$. We choose to minimize the L_2-distance between the curves. For a given set of

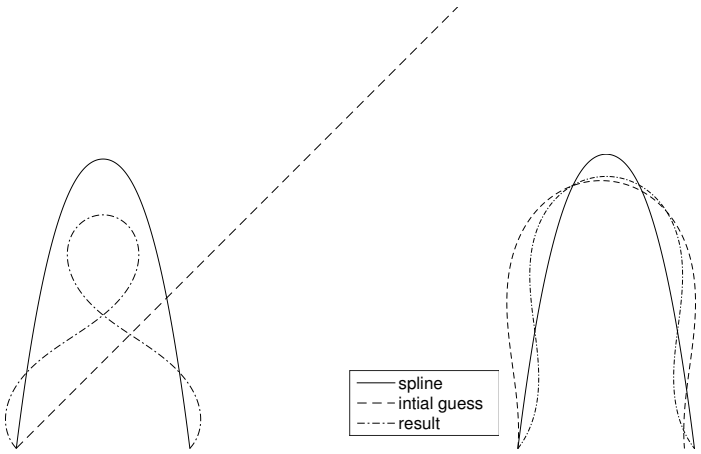

FIGURE 9. Approximating a spline by elastica. The solid line is the spline, the dashed curves are the initial guess, the dash-dotted curves are the optimized approximations. To the left an arbitrary (bad) initial guess and to the right our guess.

parameters, the elastic curve segment $\gamma_{(k,s_0,\ell,S,\phi,x_0,y_0)}$ is parameterized with constant speed ℓS over the interval $[0,1]$. The spline curve is itself also defined on $[0,1]$, but its speed is not necessarily constant. Since the L_2-norm compares points at corresponding parameter values, we need to reparameterize either the spline or the elastica for the L_2-distance to be a good measure of the curves' resemblance to each other. The simplest way is to reparameterize the elastica using the arc length s of the spline which can be calculated as

$$s(t) = \int_0^t \frac{\mathrm{d}s}{\mathrm{d}t}\,\mathrm{d}\tau = \int_0^t \|\mathbf{x}'(\tau)\|\,\mathrm{d}\tau\,, \tag{7}$$

and the length of the spline is then $L = s(1)$. We now consider the minimization problem

$$\text{minimize}_{k,s_0,\ell,S,\phi,x_0,y_0}\, \mathscr{E}(k,s_0,\ell,S,\phi,x_0,y_0), \tag{8}$$

where

$$\mathscr{E} = \frac{1}{2}\int_0^1 \left\|\mathbf{x}(t) - \gamma_{(k,s_0,\ell,S,\phi,x_0,y_0)}(s(t)/L)\right\|^2 \|\mathbf{x}'(t)\|\,\mathrm{d}t \tag{9}$$

is the square of the L_2-distance between the spline and the elastica segment.

We use a gradient driven optimization package IPOPT [9], so we need the partial derivatives of the objective function $\mathscr{E}$ with respect to the parameters $(c_1,\ldots,c_7) = (k,s_0,\ell,S,\phi,x_0,y_0)$, i.e.,

$$\frac{\partial \mathscr{E}}{\partial c_i} = -\int_0^1 \left\langle \frac{\partial \gamma_{\mathbf{c}}}{\partial c_i}(s(t)/L), \mathbf{x}(t) - \gamma_{\mathbf{c}}(s(t)/L) \right\rangle \|\mathbf{x}'(t)\|\,\mathrm{d}t\,. \tag{10}$$

The optimization problem is non-convex, so there are several local minima for $\mathscr{E}$. Therefore the optimization gives different results depending on the initial values of the parameters, cf. Figure 9. It is therefore necessary for us to have a good initial guess. We describe next our method for finding an initial guess. The full details can be found in [3].

We find the initial guess by considering the differential equation (3). If we let $u = \frac{1}{\lambda}(\lambda_2 x - \lambda_1 y)$ then the differential equation can be written as $\frac{\mathrm{d}^2\theta}{\mathrm{d}s^2} = \lambda \frac{\mathrm{d}u}{\mathrm{d}s}$, and integrating this yields

$$\kappa = \frac{\mathrm{d}\theta}{\mathrm{d}s} = \lambda u + \alpha = \lambda_2 x - \lambda_1 y + \alpha\,. \tag{11}$$

Letting θ_u denote the angle between the u-axis and the tangent, we have

$$\cos\theta_u = \frac{1}{\lambda}\begin{pmatrix}\lambda_2\\ -\lambda_1\end{pmatrix}\cdot\begin{pmatrix}\dot{x}\\ \dot{y}\end{pmatrix} = \frac{\mathrm{d}u}{\mathrm{d}s},$$

so

$$\frac{\mathrm{d}\sin\theta_u}{\mathrm{d}u} = \frac{\mathrm{d}s}{\mathrm{d}u}\frac{\mathrm{d}\sin\theta_u}{\mathrm{d}s} = \frac{1}{\cos\theta_u}\cos\theta_u\frac{\mathrm{d}\theta_u}{\mathrm{d}s} = \kappa = \lambda u + \alpha,$$

and thus

$$\sin\theta_u = \frac{1}{2}\lambda u^2 + \alpha u + \beta\,.$$

As $(\lambda_1,\lambda_2) = S^{-2}(\cos\phi,\sin\phi)$ we get estimates for the scale S and the angle ϕ by solving the first equation with respect to $\lambda_1,\lambda_2,\alpha$ in the least square sense. In a similar manner we can estimate β, and by analysing the resulting parabola we can determine whether we should use an elastica with or without inflections and estimate the parameter k. In the next step we determine which segment of the elastica we should use, i.e., estimate s_0 and ℓ. We finally determine the translation (x_0,y_0) by a least square fit. If we want end point interpolation then we can achieve that by a final scaling, rotation, and translation.

5.2. **A purely numerical approach.** We have described above a method for approximating a spline curve $\mathbf{x}:[0,1]\to\mathbb{R}^2$ by a segment of an elastic curve, represented by an analytic solution in terms of elliptic functions. An alternative approach is to approximate the spline by another spline curve $\mathbf{y}$ which is intended to be close to an elastica, in the sense that it minimizes the elastic energy. This approach could be advantageous for practical reasons. For example, existing CAD software and other mathematical software and algorithms already work with the data structure of splines.

We will use a refined knot vector for the new spline curve $\mathbf{y}$. By knot insertion we express both the target spline $\mathbf{x}$ and the elastica approximation $\mathbf{y}$ using the same basis functions (B-splines). This gives us control points $\mathbf{x}_i$ and $\mathbf{y}_i$, $i = 1,\dots,n$, that we can compare. We now seek to minimize the bending energy (2) of the new spline curve $\mathbf{y}$, with control points $\mathbf{y}_i$, while staying close to the original curve $\mathbf{x}$, with control points $\mathbf{x}_i$. The difference between these spline curves is also a spline curve, with control points $\mathbf{x}_i - \mathbf{y}_i$, and the distance between the curves is captured by the distance between the control points. These points have coordinates $(\mathbf{x}_i - \mathbf{y}_i)\cdot\mathbf{e}_i$, where $\mathbf{e}_1 = (1,0)$ and $\mathbf{e}_2 = (0,2)$. That is, we consider the constrained optimization problem:

$$\text{minimize}_{\mathbf{y}_1,\dots,\mathbf{y}_n} \qquad \frac{1}{2}\int_0^1 \kappa_{\mathbf{y}}^2\,\mathrm{d}s\,, \tag{12}$$

$$\text{such that} \qquad -\varepsilon \le \big(\mathbf{x}_i - \mathbf{y}_i\big)\cdot\mathbf{e}_j \le \varepsilon\,, \qquad i = 1,\dots,n,\, j = 1,2\,. \tag{13}$$

We need to constrain the problem additionally, e.g., by fixing the positions and tangents at the two end points. On top of this, we thus have an optimization, or sampling, over end points and tangents. For end point interpolation we simply put $\mathbf{y}_1 = \mathbf{x}_1$, $\mathbf{y}_n = \mathbf{x}_n$, and

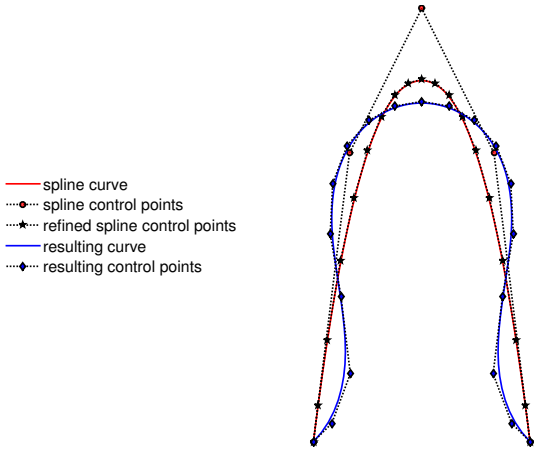

FIGURE 10. A spline approximation of an elastica (blue) constrained by a target spline (red). The end positions and tangents have been fixed.

remove these two control points from both the optimization and the constraints. The tangent constraints just specify directions along which $\mathbf{y}_2$ and $\mathbf{y}_{n-1}$ can move. The length could also be specified. In any case, we are no longer looking for the elastica that minimizes the distance to $\mathbf{x}$, but rather for an elastica that is ε-close $\mathbf{x}$. If none of the constraints are active at the end of the optimization we conclude that we have obtained an elastica which is closer to the target spline than ε. This is of course only true up to the discretization error resulting from using splines to model elastica. By refining the knot vector of the spline we obtain a smaller discretization error, and we can validate the solution by checking the differential equation (11). An example of this approach is shown in Figure 10.

A disadvantage of this method is that we cannot guarantee that our solution $\mathbf{y}$ is close to an elastica – only that it has less bending energy than the input curve $\mathbf{x}$. For this reason, we have chosen to work with the analytic approach outlined in the previous subsection.

6. SURFACE RATIONALIZATION

Before a given CAD surface can be realized as a mould in the form of a collection of EPS blocks it needs to be divided into patches. Each individual patch is approximated by a surface swept by a hot blade, i.e., a surface foliated by planar elastic curves as described in the previous sections.

In fact, we need to consider two processes: *blocking* and *segmentation*. Blocking is the process of dividing a surface into blocks such that each block can be cut individually using either a hot wire or a hot blade. Segmentation on the other hand is the process of dividing the surface into patches swept by elastica or ruled patches. If blocking is performed before segmentation, we simply divide the 3D shape into blocks and then fit the best possible ruled or elastic surface patch to each block - possibly taking constraints between block boundaries into account. Doing segmentation first is arguably harder, but has certain benefits: knowing which segments intersect a given block can be used to inform the blocking procedure.

In the following, we consider a more concrete approach to segmentation in the context where we assume that blocking has been performed first.

We first consider the problem of approximating a single surface by a surface foliated by planar elastic curves. One way to accomplish this is first to foliate the surface by planar curves and then approximate these planar curves by elastica. That is, we intersect the surface with a family of planes that are sufficiently overall transversal to the given surface, and thereby foliate the surface by planar curves. We then pick a finite number of these planes, approximate each of the corresponding planar sections with a segment of an elastica, calculate endpoints, end tangents, and lengths and interpolate this data to obtain an approximation of the original surface.

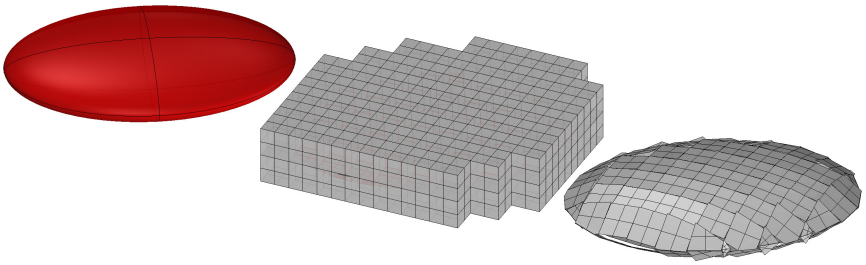

FIGURE 11. A simple approach to the rationalization of the red ellipsoid by planar surfaces. No boundary conditions are enforced in this rationalization.

For the general case we imagine our CAD surface sitting inside a collection of EPS blocks. This divides the surface into a collection of pieces each of which is the intersection between a block and the full surface, see figure 11. We now approximate each of these pieces by an elastica swept surface while demanding that neighbouring surfaces fit together in a C^1 fashion. This can be a large global optimization problem, and at the end we check to see if the result is within the required tolerance. We then pick the blocks where the tolerance is exceeded, cut these blocks in half and redo the optimization.

In the more complex approach, where segmentation (of the CAD surface) is performed first, several options can be considered. One way is to fit the largest patch that upholds the tolerance criteria to the surface and remove this part to create a reduced surface. This procedure is repeated until the whole surface is removed, i.e., the original surface is covered by patches. Another approach is a patch-growing algorithm as in [10]: A number of patches grow on the surface and whenever two patches meet a competition determines the boundary between the patches. The determining force in the competition is the improvement on the Euler elastica sweep approximation, i.e., the resulting boundary is the one with the largest combined improvement.

For fabrication, each patch needs to be divided into blocks, and this can be difficult on the boundaries; either the blocks need to be cut smaller to align with the boundaries or multiple elastica sweeps are needed, i.e., the block can be cut more than once by the blade.

A third option is a hybrid of the above mentioned methods, where the knowledge from the patch methods guides the placement of the blocks.

7. EXAMPLE

To illustrate the procedures, we consider the modelling and construction of the skater ramp shown in Figure 12. This CAD surface consists of spline surfaces, some of which are

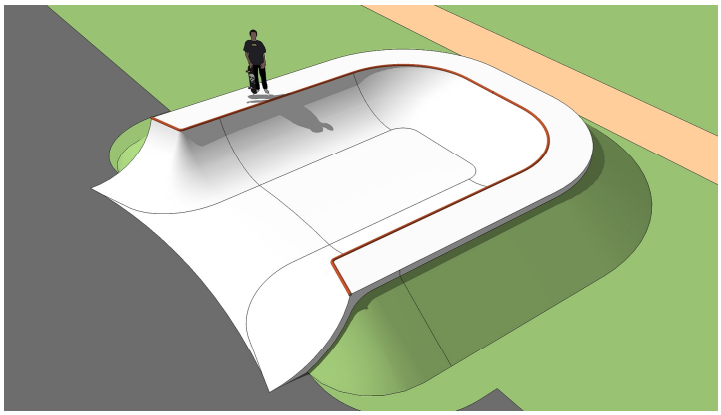

FIGURE 12. The skater ramp example.

doubly curved. The curved surfaces (see Figure 13) are the ones that need special moulds. Here there are three different types: 1) three ruled parts (the "sides"), 2) two corners with negative curvature at the front of the image and 3) two corners with positive curvature at the back. We will approximate each corner by a surface swept by elastica. The ruled parts can be cut either by the hot wire following the rulings or by the hot blade approximating the curved cross section curve by an elastic curve.

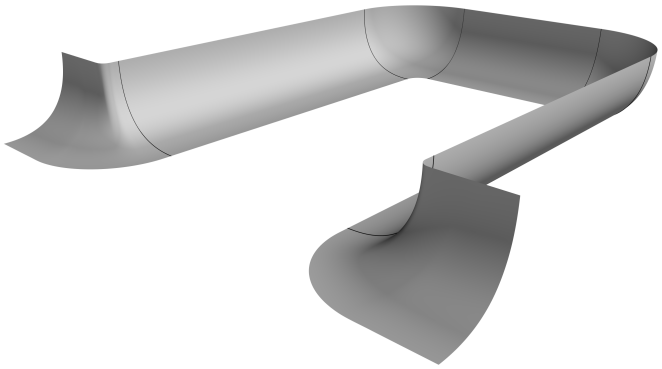

FIGURE 13. The spline surfaces of the skater ramp must be approximated by elastica swept surfaces

For each corner, the control points give rise to a set of planar spline curves which foliate the surface (see Figure 14 left). These curves can be approximated by elastica as described in Section 5.

If the splines are approximated independently, the control parameters for the resulting elastica might differ quite a lot between two adjacent curves. This is because, for a typical (uncomplicated) curve segment, there can be many different elastic curve segments that

approximate it quite closely. To avoid large jumps in the control parameters we use the elastica that approximates the first spline curve as the initial guess for the optimization at the next spline, and so forth.

The optimization is performed with constraints: the approximating elastic curve is in each case required to have the same length and the same end points as the original spline curve. The resulting elastic curves can be seen in Figure 14 right.

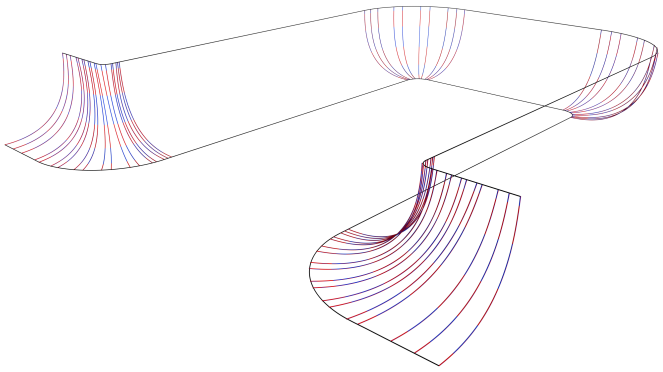

FIGURE 14. The corner surfaces are foliated by planar spline curves (blue). Each of these are approximated by an elastic curve (red).

Our optimization algorithm minimizes the square on the L_2-distance between the spline curve and the elastica, see (9). Table 1 shows some of these distance values.

	Min	Max	Average
Negative	0,838846837	0,9107882	0,868163184
Positive	5,738445432	5,788943718	5,778703296

TABLE 1. The optimized value for the L_2-distance for the two corner types. The minimal value corresponds to the elastic curve which best approximates the spline. The height of the ramp is 854.10 with the lengths of the spline curves varying between 1342.6 and 1459.8.

For a visual inspection and evaluation of the result, in Figure 15 we have plotted the spline and the approximating elastica in the worst case (i.e. highest L_2 distance). For the corner with negative curvature the curves are nearly indistinguishable. For the positively curved corner, there is clearly a difference, but the overall shape is the same, and the approximation is certainly within any conceivable tolerance for this particular application.

8. CONCLUSION

Our work on approximating arbitrary spline curves by elastic curves, illustrated here by the test case of the skater ramp, indicates that the problem of approximating most of the

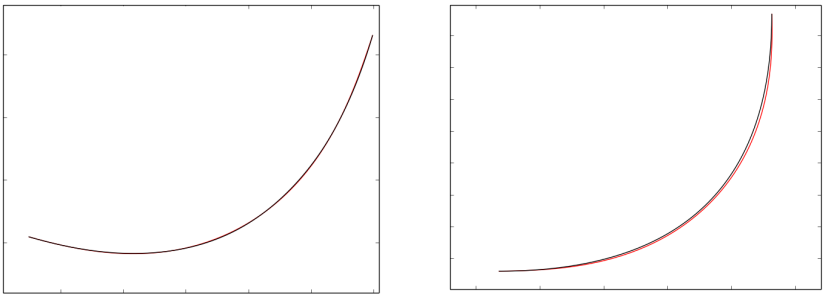

FIGURE 15. The original planar spline curve (black) on top of the approximating elastica (red). These are the worst cases for the corners with negative curvature (left) and positive curvature (right).

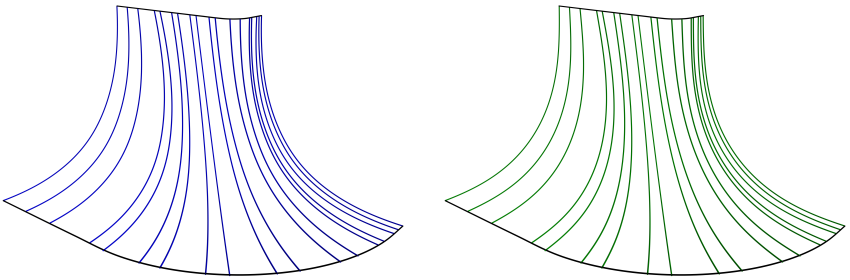

FIGURE 16. Left: The surface is foliated by planar spline curves. Right: The surfaces are foliated by elastic curves each one of which approximates the corresponding spline in the figure to the left.

CAD surfaces used in architecture by panels of surfaces swept by planar elastica is feasible, and that it can be effectively implemented into the work flow of modern robotics enhanced constructions of buildings and other manifestations of architectural design, see also the report in [13].

The utility of the technology depends now on the technical problem of designing blades that can be heated and used to cut EPS in a consistent, robust and predictable way. We have received positive experimental results and optimistic input from our project partners showing that this blade technology can indeed be developed and made operational on the scale needed.

In addition to the application to final production architecture, we anticipate that the theoretical framework described here will also have other industrial applications. For example, rapid prototyping is an important part of the innovation pipeline. Proptotypes that are currently produced in EPS using CNC milling could be produced much more quickly using hot blade cutting.

As is evident from our present description and also from the discussion of the state of the art in Subsection 2.3 the full implementation of the various assets of the robotic Hot

Wire/Blade cutting idea needs – to mention but one momentum, at least for the Building Industries – an almost paradigmatic shift of attention away from the classical use of relatively complicated scaffoldings and laths for the concrete shuttering of facade elements that are not just off-the-shelf items. In comparison with the proposed applications of the Hot Blade Cutting Technology, the classical way of facade production is often very labor intensive, and often it even demands an extra time-consuming postprocessing, a fairing by hand, in order to obtain the desired smoothness of the facades and surfaces. On a final note we should also mention that this particular MaDiFa concept, that we have presented in this chapter, offers one more genuine quality, which will also reduce transportation and logistics costs considerably, namely that in this case the digital factory in question is essentially mobile and can be set up for EPS cutting, assembling, and shaping on any location.

ACKNOWLEDGEMENT

This work was completed with the support of Innovation Fund Denmark, project number 128-2012-3.

REFERENCES

[1] Birkhoff, G., Boor, C.D.: Piecewise polynomial interpolation and approximation. Approximation of Functions, Proc. General Motors Symposium 1964, H.L. Garabedian, ed. pp. 164–190 (1965). Elsevier, Publ. Co. Amsterdam.
[2] Borbély, A., Johnson, M.: Elastic splines I: Existence. Constr. Approx. **40**, 189–218 (2014).
[3] D. Brander, J. Gravesen, and T. Nørbjerg. Approximation by planar elastic curves. Preprint, arXiv:1509.00703.
[4] J.J. Broek, I. Horváth, B. de Smit, A.F. Lennings, Z. Rusák, and J.S.M. Vergeest. Free-form thick layer object manufacturing technology for large-sized physical models. *Automation in Construction*, 11:335–347, 2002.
[5] Malcolm, M.: On the computation of nonlinear spline functions. SIAM Journal on Numerical Analysis **14**, 254–282 (1977)
[6] Mehlum, E.: Nonlinear splines. Computer aided geometric design (Proc. Conf., Univ. Utah, Salt Lake City, Utah, 1974) pp. 173–207 (1974).
[7] S. Flöry and H. Pottmann. Ruled surfaces for rationalization and design in architecture. *Proc. ACADIA*, pages 103–109, 2010.
[8] I. Horváth, Z. Kovács, J.S.M. Vergeest, J.J. Broek, and A. de Smit. Free-form cutting of plastic foams: a new functionality for thick-layered fabrication of prototypes. *Proceedings of the TCT'98 Conference, Nottingham*, pages 229–237, 1998.
[9] A. Wächter and L.T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program., Ser. A* **106**, pages 25–57, 2006.
[10] D. Julius, V. Kraevoy and A. Sheffer. D-charts: Quasi-developable mesh segmentation. *Computer Graphics Forum, Proc. of Eurographics*, pages 581–590, 2005.
[11] Levien, R.: From spiral to spline; optimal techniques for interactive curve design. Ph.D. thesis, UC Berkeley (2009)
[12] A.E.H. Love, *A treatise on the mathematical theory of elasticity*, Cambridge University Press, 1906.
[13] Søndergaard, A., Feringa, J., Nøbjerg, T., Steenstrup, K., Brander, D., Gravesen, J., Markvorsen, S., Bærentzen, A., Petkov, K., Hattel, J., Clausen, K., Jensen, K., Knudsen, L., Kortbek, J.: Robotic hot-blade cutting: an industrial approach to cost-effective production of double curved concrete structures. Proceedings of RobArch 2016, to appear.

DEPARTMENT OF APPLIED MATHEMATICS AND COMPUTER SCIENCE, TECHNICAL UNIVERSITY OF DENMARK MATEMATIKTORVET, BUILDING 303 B DK-2800 KGS. LYNGBY, DENMARK