

# Medial design of blades for hydroelectric turbines and ship propellers

M. Rossgatterer<sup>a</sup>, B. Jüttler<sup>a</sup>, M. Kapl<sup>a</sup>, G. Della Vecchia<sup>b</sup>

<sup>a</sup>Johannes Kepler University, Linz, Austria

<sup>b</sup>ANDRITZ HYDRO GmbH, Austria

---

## Abstract

We present a method for constructing blades of hydroelectric turbines and ship propellers based on design parameters that possess a clear hydraulic meaning. The design process corresponds to the classical construction of a blade using the medial surface of the blade and profile curves attached to it. The main new contribution of the paper consists in realizing this construction using B-spline techniques. In particular, it is shown how to obtain blade boundary surfaces (which describe the pressure and the suction side of the blade) which are joined with  $C^1$ -smoothness along the leading edge. Moreover, special attention is paid to the construction of propeller blades with a well-defined tangent plane at the tip, which is a singular point of the blade boundary surfaces. In order to guarantee these smoothness properties, we generate and analyze singularly parameterized medial surfaces. We contribute novel shape modeling techniques that are based on singular parameterizations and demonstrate their potential for applications in industry. Finally, it is shown how to represent the blades as B-spline surfaces with a relatively small number of control points.

*Keywords:* CAD-model, B-spline representation, hydroelectric turbine blade, propeller blade, medial axis-based design

---

## 1. Introduction

Blades of hydroelectric turbines and ship propellers are instances of functional free-form surfaces. Such blades are individually designed for each specific application scenario, taking into account various parameters controlling their performance. Thus, intuitive methods for designing blades are of vital interest.

Traditional tools for surface design in Computer-Aided Design, such as control points, are not directly suitable for blade design. Instead, the designers prefer to work with design parameters that possess a clear hydraulic meaning. For instance, the tangent direction of the blade at the leading edge – which can be derived from the angular velocity of the turbine and the velocity and direction of the incoming water – is one of the preferred design parameters.

On the other hand, the results created by the blade design process have to be compatible with the existing standard technology of Computer-Aided Design. Consequently, a B-spline representation of the blade surfaces is required, which then forms the basis for the subsequent steps in the product development pipeline, such as Computer-Aided Manufacturing. In recent years, the B-spline representation is even directly used for the nu-

merical simulation, using the novel technology of iso-geometric analysis [8].

Techniques for blade design have a long history going back to classical papers such as [17]. This paper uses the conformal mapping of a surface of revolution into the plane for designing streamlines, which will also be one of the first steps in our approach.

In more recent years, the use of Computer-Aided Design technology has opened new possibilities for blade design. Miller et al. [12] describe a methodology for interactive design of turbomachinery blades. The blade sections may be defined with respect to general surfaces of revolution which can then be adapted to the path of the fluid. The approach aims mostly at applications to gas turbines.

In [16] and [9] the authors focus on coupling CFD and Computer-Aided parametric geometry definitions to find the best design. This approach is shown by the design of Francis turbine runners coupled with 3-D Euler flow analysis.

Ye et al. [18] present a technique for designing functional surfaces by incorporating physical constraints that involve surface normal vectors. The method is illustrated by using it to design propeller blade surfaces in conjunction with hydrodynamic analysis. Anders and

Haarmeyer [1] used two different Bézier curves and two circles (or ellipses) on leading and trailing edges to describe the blade profile, using twenty parameters. Their approach, however, does not consider the use of a B-spline representation.

In [7], turbine blades are represented via lofted B-spline surfaces which are constructed from given planar profile curves in B-spline representation. The authors consider lofted surfaces with profile curves lying either in parallel planes, on concentric cylinders, or in concentric cones. While these are reasonable assumptions for gas turbines, they are generally too restrictive for hydroelectric turbines and ship propellers.

Qian and Dutta [15] present an approach for the design of a turbine blade which allows the designer to control both geometry and material composition. A heterogeneous turbine blade is made of different constituent material and possesses gradient material properties. In [10], blades are represented as NURBS surfaces using design parameters which correspond to two-dimensional blade sections. The blade surface is constructed as a single NURBS surface using a skinning procedure.

Our approach to blade design is based on the medial representation of the blades, as medial surface with profile curves attached to it. This is closely related to the concept of medial axis, see e.g. [4] and the references therein. Recently, methods for computing the medial axis of three-dimensional solid objects with free-form boundaries have been studied [14]. Also, the medial representation has been used as a design tool, e.g. in medical and biological applications [19]. A generalized medial representation has been used successfully for designing blends of canal surfaces [2].

In this paper we present a method for constructing a B-spline based geometric model for blades of turbines governed by design parameters with clear hydraulic meaning. The design process is based on the medial surface of the blade and the profile attached to it. The medial surface is represented by three-dimensional space curves which describe the rough shape of the blade. For the representation of the resulting blades we use B-spline surfaces with a relatively small number of control points compared to current models.

So far, the computer implementation of the classical medial surface-based approach to blade design uses a point-based description via streamlines, which are represented as polylines. As the main new contribution of our paper, we realize the entire construction using B-spline techniques. Using a suitable parameterization of the medial surfaces we guarantee that the two B-spline surfaces which describe the pressure and the suction

side of the blade meet with  $C^1$ -smoothness along the leading edge of the blade. In the case of ship propellers, which require a singular point at the tip, we reparameterize the medial surface using a cut-out function in order to construct a B-spline representation with a well-defined tangent plane at the tip. In order to guarantee these smoothness properties, we generate and analyze singularly parameterized medial surfaces. It is shown that  $G^1$ -smooth surfaces can be generated by suitably combining and adding several contributions, some of which possess singular parameterizations. We contribute novel shape modeling techniques that are based on singular parameterizations and demonstrate their potential for applications in industry.

The remainder of this paper is organized as follows. Section 2 describes the different design parameters which are used to generate a procedurally defined medial surface of the blade. In Section 3 and Section 4 we generate a B-spline representation of a turbine and propeller blade, respectively, from the procedurally defined medial surface. Our design scheme corresponds to the classical construction of a blade, using the medial surface and profiles which are added in normal directions to the medial surface. Within these two sections we present different examples of generated turbine and propeller blades which demonstrate the performance of the design process. Finally, we conclude this paper.

## 2. Design parameters of blades

The design of blades for hydroelectric turbines and ship propellers follows the classical approach described in [17]. First, the medial surface is designed using the conformal mapping of the projected streamlines. Second, the profile curves are added to the medial surface. The resulting surface is given by a medial surface and profiles attached to it.

This section recalls the classical approach using modern notation and explains the various design parameters in some detail.

We denote the first partial derivatives of a bivariate function  $\mathbf{f} : [0, 1]^2 \rightarrow \mathbb{R}^d$  ( $d \in \{1, 2, 3\}$ ) with respect to the first and second argument by

$$\mathbf{f}_1(u_0, v_0) = \frac{\partial}{\partial u} \mathbf{f}(u, v) \Big|_{(u,v)=(u_0,v_0)}$$

and

$$\mathbf{f}_2(u_0, v_0) = \frac{\partial}{\partial v} \mathbf{f}(u, v) \Big|_{(u,v)=(u_0,v_0)}.$$

Analogously, let  $\mathbf{f}_{11}$ ,  $\mathbf{f}_{12}$ ,  $\mathbf{f}_{22}, \dots$  be the corresponding higher partial derivatives of  $\mathbf{f}$ . In addition, the first

derivative of a univariate function  $\mathbf{g} : [0, 1] \rightarrow \mathbb{R}^d$  ( $d \in \{1, 2, 3\}$ ) is denoted by  $\mathbf{g}'$ , i.e.

$$\mathbf{g}'(t_0) = \left. \frac{\partial}{\partial t} \mathbf{g}(t) \right|_{t=t_0}.$$

### 2.1. Design of the medial surface via streamlines

The first step of the blade construction is the generation of the medial surface of the blade, which is represented by three-dimensional space curves, called streamlines, given in cylindrical coordinates  $(r, \varphi, z)$ , where the  $z$ -axis is the axis of rotation of the turbine / propeller. A direct construction of these streamlines is in general extremely difficult for the designer. Therefore it is preferable to use design parameters defined in a two-dimensional space for simplifying the generation of the medial surface of the blade.

The construction of the medial surface works as follows. We consider the projected two-dimensional  $rz$ -space, which is obtained from the  $r\varphi z$ -space by omitting the coordinate  $\varphi$ . At first, we design the channel contour of the blade by determining the contours of hub and shroud. Then we specify the trace of leading and trailing edge and obtain an area which is called meridian contour.

Depending on the axial path of the fluid flow, we obtain projected streamlines on the meridian contour, which define the blade in the  $rz$ -plane, see Figure 1. Let  $\mathbf{h} : [0, 1]^2 \rightarrow \mathbb{R}^2$  with

$$\mathbf{h}(u, v) = (r(u, v), z(u, v))$$

be a parameterization of this contour such that shroud, hub, leading edge and trailing edge are given by  $\mathbf{h}(u, 0)$  and  $\mathbf{h}(u, 1)$ ,  $\mathbf{h}(0, v)$  and  $\mathbf{h}(1, v)$ , respectively. The curve

$$u \mapsto \mathbf{h}(u, v_0) = (r(u, v_0), z(u, v_0)) \quad (1)$$

is then called a projected streamline of the blade for a constant  $v_0$ . The parameterization  $\mathbf{h}$  is chosen in such a way that the projected streamlines form a well-behaved family of curves (approximately parallel, as far as this is possible).

Each projected streamline (1) defines a surface of revolution. In the  $r\varphi z$ -coordinates, this surface is given by

$$(u, \varphi) \mapsto (r(u, v_0), \varphi, z(u, v_0)). \quad (2)$$

In Cartesian coordinates, the surface of revolution is represented as

$$(u, \varphi) \mapsto (r(u, v_0) \cos \varphi, r(u, v_0) \sin \varphi, z(u, v_0)).$$

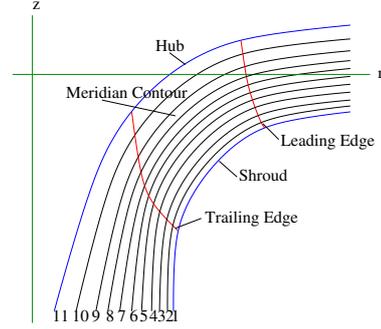


Figure 1: The meridian contour with some projected streamlines.

This surface of revolution is now mapped into the  $\mu\varphi$ -plane by

$$\begin{aligned} \Phi(v_0) : (r(u, v_0) \cos \varphi, r(u, v_0) \sin \varphi, z(u, v_0)) \\ \mapsto (\mu(u, v_0), \varphi), \end{aligned} \quad (3)$$

where  $\mu(u, v_0)$  is given by

$$\mu(u, v_0) = \int_0^u \frac{\sqrt{r_u(t, v_0)^2 + z_u(t, v_0)^2}}{r(t, v_0)} dt.$$

The function  $\mu(u, v_0)$  can be seen as the normalized arc length of the chosen segment of the projected streamline with respect to radius  $r(u, v_0)$ .

**Lemma 1.** *The mapping  $\Phi(v_0)$  given by (3) is conformal.*

**Proof.** It is sufficient to show that there exists a function  $c(u, \varphi) > 0$  such that

$$\bar{g}_{ij}(u, \varphi) = c(u, \varphi) g_{ij}(u, \varphi) \quad \text{for } i, j \in \{1, 2\}, \quad (4)$$

where  $g_{ij}(u, \varphi)$  and  $\bar{g}_{ij}(u, \varphi)$  are the coefficients of the first fundamental form of  $(r(u, v_0) \cos \varphi, r(u, v_0) \sin \varphi, z(u, v_0))$  and  $(\mu(u, v_0), \varphi)$ , respectively (see [11]). A short computation confirms that equations (4) are satisfied with  $c(u, \varphi) = \frac{1}{r(u, v_0)^2}$ .  $\square$

As next step, we construct for each surface of revolution (2) a streamline in the  $\mu\varphi$ -plane using design parameters with a clear hydraulic meaning. This step is explained in more detail in Section 2.2.

Then by applying the inverse mapping

$$\Phi^{-1}(v_0) : (\mu, \varphi) \mapsto (r(\mu, \varphi) \cos \varphi, r(\mu, \varphi) \sin \varphi, z(\mu, \varphi)),$$

we obtain from the projected streamline in the  $\mu\varphi$ -plane a streamline in the  $r\varphi z$ -space, which preserves the chosen angles in the  $\mu\varphi$ -plane. Thus, the designer can specify the angles of the streamline with respect to the tangent vectors of the circles on the surface of revolution.

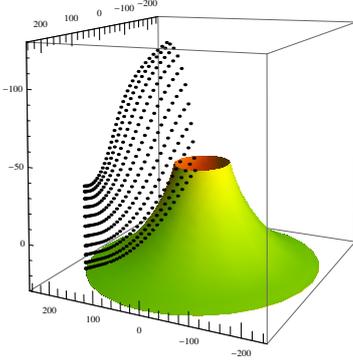


Figure 2: Point cloud of the procedurally defined medial surface of a turbine blade and a part of the hub of the blade.

This is highly important since these angles are essentially determined by the angular speed of the turbine and the velocity of the water at the inlet and outlet of the turbine.

Finally, the entire medial surface  $\mathbf{m}^{(0)}$  is obtained by

$$\mathbf{m}^{(0)}(u, v) = \Phi^{-1}(v)(\mathbf{h}(u, v)). \quad (5)$$

The inversion of the conformal mapping  $\Phi(v_0)$  and hence the construction of the streamlines are performed approximately by using a look-up table and linear interpolation. Consequently, the medial surface is described only procedurally; no closed form representation is available at this stage.

An example of a procedurally defined medial surface of a turbine blade is shown in Figure 2. In this case the medial surface is given by 11 curves each represented by 41 points.

## 2.2. Design of a single streamline

We explain the construction of the streamlines in the  $\mu\varphi$ -plane. The design of a single streamline in the  $\mu\varphi$ -plane for a given surface of revolution (2) with a constant  $v_0$  works as follows. We choose for the streamline in the  $\mu\varphi$ -plane a cubic Bézier curve, i.e.

$$\mathbf{g}(u, v_0) = B_0^3(u)\mathbf{b}_0(\varphi_0) + B_1^3(u)\mathbf{b}_1(\varphi_0, \beta_0, \zeta_0) \quad (6) \\ + B_2^3(u)\mathbf{b}_2(\varphi_1, \beta_1, \zeta_1) + B_3^3(u)\mathbf{b}_3(\varphi_1),$$

where  $(B_i^3(u))_{i=0,\dots,3}$  are the Bernstein polynomials of degree 3. The control points  $(\mathbf{b}_i)_{i=0,\dots,3}$  are defined by the 6 design parameters  $\varphi_0, \varphi_1, \beta_0, \beta_1, \zeta_0$  and  $\zeta_1$  which describe different angles of the streamlines on the leading and trailing edge.

- $\varphi_0$  and  $\varphi_1$  determine the coordinates  $\varphi$  of the streamline of the medial surface on the leading ( $\varphi_0$ )

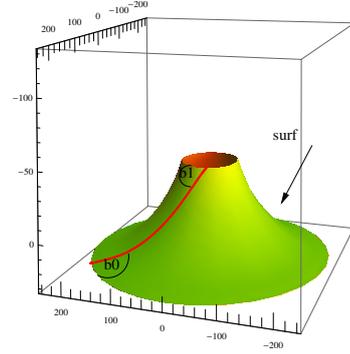


Figure 3: A streamline with the corresponding surface of revolution and the angles  $\beta_0$  and  $\beta_1$ .

and trailing edge ( $\varphi_1$ ) using cylindrical coordinates  $(r, \varphi, z)$ .

- $\beta_0$  and  $\beta_1$  are the angles between the tangent of the streamline of the medial surface and the tangent of the circle on the surface of revolution on the leading ( $\beta_0$ ) and trailing edge ( $\beta_1$ ).
- $\zeta_0$  and  $\zeta_1$  specify how long the tangent directions of the streamline of the medial surface (defined by the angles  $\beta_0$  and  $\beta_1$ ) will be kept starting from the leading ( $\zeta_0$ ) and trailing edge ( $\zeta_1$ ).

Figure 3 shows an example of a surface of revolution defined by a streamline. The angle  $\beta$  is the angle between the tangent of the medial surface and the tangent of the circle on the surface of revolution.

In detail, the four control points  $(\mathbf{b}_i)_{i=0,\dots,3}$  for a streamline (6) are constructed in the following way.

- Since the starting and end point of a streamline have to be on the corresponding circle on the surface of revolution at the leading and trailing edge, respectively, the first coordinate of the first ( $\mathbf{b}_0$ ) and last control point ( $\mathbf{b}_3$ ) is already fixed. The second coordinate of the first and last control point is given by the parameter  $\varphi_0$  and  $\varphi_1$ , respectively.
- The second control point ( $\mathbf{b}_1$ ) is defined by the parameters  $\varphi_0, \beta_0$  and  $\zeta_0$ , see Figure 4.
- The third control point ( $\mathbf{b}_2$ ) is determined by the parameters  $\varphi_1, \beta_1$  and  $\zeta_1$ , see Figure 4.

Since the mapping  $\Phi(v_0)$  is conformal, the resulting streamline in the  $r\varphi z$ -space preserves the chosen angles on the leading and trailing edge, see Subsection 2.1. This is an important feature for the blade designer, since

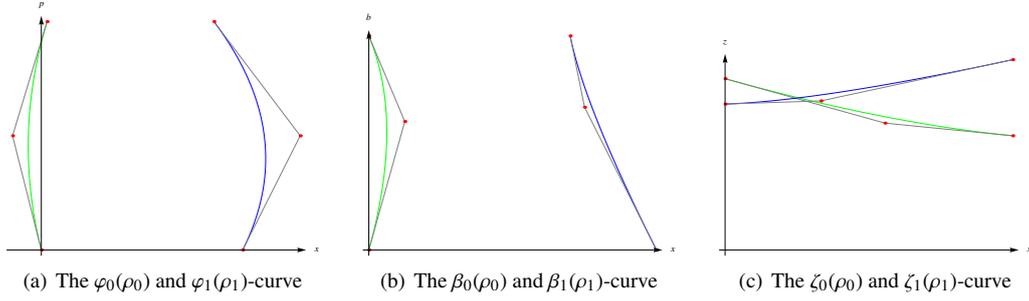


Figure 5: The 6 design parameter curves defined on the leading edge (blue) and on the trailing edge (green).

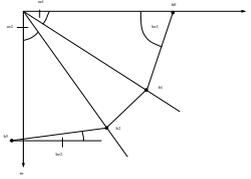


Figure 4: The effect of the parameters  $\beta_0, \beta_1, \zeta_0$  and  $\zeta_1$  on the position of the control points of a streamline in the  $(\mu, \varphi)$  plane.

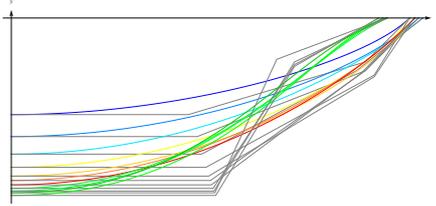


Figure 6: A few streamlines for a turbine blade in the coordinate system  $(\mu, \varphi)$ .

it allows to control the angle between the blade and the water flow and inlet and outlet.

For designing several streamlines we vary the design parameters continuously with  $v$ . Figure 5 shows the 6 parameter curves defined on the arc length parameterized leading and trailing edge. The parameter curves are chosen as quadratic Bézier curves. An example of a few streamlines for a turbine blade in the  $\mu\varphi$ -plane is shown in Figure 6.

### 2.3. Blade generation

The blade (two surfaces representing pressure and suction side) is obtained by adding a scaled profile to both sides of the medial surface.

The profile curve provides additional thickness information along the cross sections of the blade. In our case,

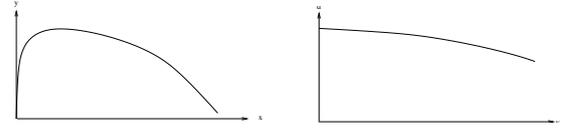


Figure 7: Example of a profile curve (left) and a scaling function (right) for a turbine blade.

the profile curve is a Bézier curve  $\mathbf{p}(t) = (x(t), y(t))$  for  $t \in [0, 1]$  with  $x(0) = 0$  and  $x(1) = 1$ . Furthermore, we have that  $x'(0) = 0$  and  $x'(t) > 0$  for  $t \in (0, 1]$ .

The  $y$ -coordinate of  $\mathbf{p}(t)$  and a scaling function  $d(v)$  determine the thickness, which is added to the medial surface along its normal to obtain the pressure and the suction side of the blade.

The scaling function  $d(v)$  depends on the streamline and may also be different for pressure and suction side. In general, it is even possible to use different profile curves for each streamline and side. In this work we consider only the case where all streamlines share the same profile curve. An example of a profile curve and a scaling function is given in Figure 7.

The pressure and the suction side of the blade are obtained by

$$\mathbf{b}(t, v) = \mathbf{m}^{(0)}(u(t), v) \pm d(v)\mathbf{n}(u(t), v)y(t), \quad (7)$$

where  $\mathbf{m}^{(0)}$  denotes the medial surface and  $\mathbf{n}$  the associated normals.  $u(t)$  is such that  $x(t)$  is proportional to the arc length of the  $u$ -parameter lines of the medial surface.

Figure 8 summarizes the main idea of this design scheme.

The only procedural definition of the medial surface is inherited by the two surfaces representing the pressure and the suction side. For various applications, which range from manufacturing to numerical simulation, it is essential to have a high-quality CAD model of

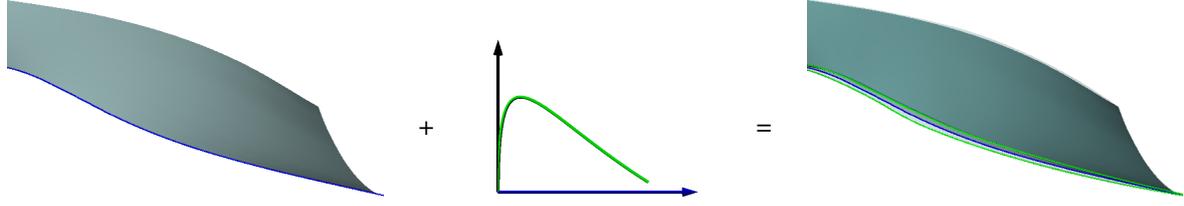


Figure 8: Schematic computation of a blade.

the blade available, representing it as a NURBS surface. This will be discussed in the next section.

### 3. B-spline representation of turbine blades

In this section we describe the different steps which have to be performed for generating a CAD-model for a turbine blade. Summing up, we have the following input for the computation of the blade:

1. A procedurally defined medial surface of the blade
2. A profile, which consists of the curve  $\mathbf{p}(t)$  and the scaling function  $d(v)$

The pressure and the suction side of the blade are obtained by adding or subtracting the scaled profile to the medial surface in the following way:

$$\mathbf{b} = \mathbf{m} \pm d\mathbf{n}_y. \quad (8)$$

Here,  $\mathbf{m}$  denotes the medial surface,  $\mathbf{n}$  the normals of the medial surface,  $y$  is the  $y$ -coordinate of the profile curve and  $d$  is the scaling function.

For the generation of a CAD-model for a blade the following steps are necessary:

1. Initial B-spline representation of medial surface
2. Reparameterization of the medial surface
3. Computation of the normals
4. Generating the blade surface

The following sections describe these steps in more detail.

#### 3.1. Initial B-spline representation of medial surface

We describe a method for the approximation of the medial surface  $\mathbf{m}^{(0)}$  of a blade by a B-spline surface.

We are interested in a B-spline surface  $\mathbf{m}(u, v)$  which approximates the surface  $\mathbf{m}^{(0)}$  best in the sense of least-squares and whose control points have the following additional property: In each  $u$ -direction the first two control points  $\mathbf{c}_{ij}$  coincide, i.e.

$$\mathbf{c}_{0j} = \mathbf{c}_{1j} \text{ for } j \in \{0, \dots, n\}. \quad (9)$$

This condition is necessary for attaching the profile curve to the medial surface, because the speed of the  $u$ -parameter lines at  $u = 0$  has to be zero since the profile curve has a vertical tangent at this point (see next section).

The B-spline surface  $\mathbf{m}(u, v)$  is computed by solving the least-squares problem

$$f(\mathbf{c}) = \int_0^1 \int_0^1 \|\mathbf{m}(u, v) - \mathbf{m}^{(0)}(u, v)\|^2, \quad (10)$$

where  $\mathbf{c}$  is the vector of the unknown coefficients (control points)  $\{\mathbf{c}_{ij}\}_{i=0, \dots, m}^{j=0, \dots, n}$ . The degrees  $(p, q)$ , the number of control points  $(m+1) \times (n+1)$  and the knot sequences  $\mathcal{S}$  and  $\mathcal{T}$  of  $\mathbf{m}(u, v)$  are specified by the user. In addition, we force the first two control points  $\mathbf{c}_{ij}$  in each  $u$ -direction to be equal, see equation (9).

For the resulting B-spline surface  $\mathbf{m}(u, v)$  we have

$$\mathbf{m}_1(0, v) = (0, 0, 0) \text{ for } v \in [0, 1], \quad (11)$$

which implies

$$\mathbf{m}_1(0, v) \times \mathbf{m}_2(0, v) = (0, 0, 0) \text{ for } v \in [0, 1]. \quad (12)$$

Nevertheless, the following lemma ensures that  $\mathbf{m}(u, v)$  has well-defined normals everywhere, despite possessing a singular curve.

**Lemma 2.** *Let  $\mathbf{m}(u, v)$  be a B-spline surface of degree  $(p, q)$  for  $p \geq 2$  with control points fulfilling condition (9) and open knot vectors  $\mathcal{S}$  and  $\mathcal{T}$  with  $(p+1)$  and  $(q+1)$ -fold boundary knots respectively. In addition, let*

$$\mathbf{m}_1(u, v) \text{ and } \mathbf{m}_2(u, v) \text{ for } (u, v) \in (0, 1] \times [0, 1],$$

and

$$\mathbf{m}_{11}(0, v) \text{ and } \mathbf{m}_2(0, v) \text{ for } v \in [0, 1]$$

be linearly independent, respectively. Then  $\mathbf{m}(u, v)$  is  $G^1$ -smooth and normals for  $u = 0$  can be obtained by

$$\frac{\mathbf{m}_{11}(0, v) \times \mathbf{m}_2(0, v)}{\|\mathbf{m}_{11}(0, v) \times \mathbf{m}_2(0, v)\|} \text{ for } v \in [0, 1]. \quad (13)$$

**Proof.** At first, we show that there exists a reparameterization of  $\mathbf{m}(u, v)$  which is  $C^1$ -smooth and regular. It suffices to consider a piecewise polynomial representation which is parameterized with respect to  $[0, 1]^2$ .

Since

$$\mathbf{m}_1(0, v) = (0, 0, 0) \text{ for } v \in [0, 1],$$

there exists a surface  $\bar{\mathbf{m}}(u, v)$  such that

$$\mathbf{m}_1(u, v) = u\bar{\mathbf{m}}_1(u, v) \text{ for } (u, v) \in [0, 1]^2.$$

The unit normal field is obtained by

$$\mathbf{N} = \frac{\mathbf{m}_1(u, v) \times \mathbf{m}_2(u, v)}{\|\mathbf{m}_1(u, v) \times \mathbf{m}_2(u, v)\|} = \frac{\bar{\mathbf{m}}_1(u, v) \times \mathbf{m}_2(u, v)}{\|\bar{\mathbf{m}}_1(u, v) \times \mathbf{m}_2(u, v)\|}. \quad (14)$$

Now, we consider the reparameterized surface

$$\tilde{\mathbf{m}}(s, v) = \mathbf{m}(\sqrt{s}, v).$$

and the associated normals

$$\tilde{\mathbf{N}}(s, v) = \mathbf{N}(\sqrt{s}, v).$$

We have to show that:

- (i)  $\tilde{\mathbf{m}}_1 \cdot \tilde{\mathbf{N}} = \tilde{\mathbf{m}}_2 \cdot \tilde{\mathbf{N}} = 0$  and
- (ii)  $\tilde{\mathbf{m}}_1$  and  $\tilde{\mathbf{m}}_2$  are linearly independent everywhere.

Since

$$\tilde{\mathbf{m}}_1(s, v) = \frac{1}{2\sqrt{s}}\mathbf{m}_1(\sqrt{s}, v) = \frac{1}{2}\bar{\mathbf{m}}_1(\sqrt{s}, v) \quad (15)$$

and

$$\tilde{\mathbf{m}}_2(s, v) = \bar{\mathbf{m}}_2(\sqrt{s}, v)$$

the first statement (i) holds. In order to prove the linear independence (ii), we consider two cases. For the case  $s \neq 0$  (ii) is obvious. For the second case ( $s = 0$ ) we obtain from (15) that

$$\tilde{\mathbf{m}}_1(0, v) = \frac{1}{2}\bar{\mathbf{m}}_1(0, v). \quad (16)$$

Furthermore,

$$\mathbf{m}_{11}(u, v) = \bar{\mathbf{m}}_1(u, v) + u\bar{\mathbf{m}}_{11}(u, v).$$

In particular we obtain that

$$\mathbf{m}_{11}(0, v) = \bar{\mathbf{m}}_1(0, v). \quad (17)$$

Combining (16) and (17) gives  $\tilde{\mathbf{m}}_1(0, v) = \frac{1}{2}\mathbf{m}_{11}(0, v)$ , thus (ii) is also fulfilled for  $s = 0$ , since we assume that  $\mathbf{m}_{11}(0, v)$  and  $\mathbf{m}_2(0, v)$  are linear independent.

So we have shown that  $\tilde{\mathbf{m}}(s, v)$  is regular and  $C^1$ . Finally, the normals (13) result from equations (14) and (17).  $\square$

Thus, the medial surface  $\mathbf{m}(u, v)$  has well-defined normals on the entire domain. Hence, by using a least-squares fit we can compute a B-spline surface  $\mathbf{n}(u, v)$  which represents the normal directions of  $\mathbf{m}(u, v)$ .

### 3.2. Reparameterization of the medial surface

We describe a method for the reparameterization of the medial surface of the blade. For adding the profile to the medial surface, the horizontal speed of the profile has to match the parametric speed of the medial surface. Otherwise, we would obtain a deformed profile along this streamline. This requirement can be formulated as the condition

$$\|\mathbf{m}_1(u, v)\| = \lambda(v)x'(u), \quad (18)$$

where  $\mathbf{m}(u, v)$  is the medial surface,  $\lambda(v)$  is the arc length of the entire  $u$ -parameter line (streamline) of  $\mathbf{m}(u, v)$  at  $v$  and  $x'(u)$  denotes the derivative of the first coordinate of the profile.

Note that  $x'(0) = 0$ , since the blade profile has a vertical tangent at the tip. This, however, is not a problem since the constructed medial surface possesses a singular edge for  $u = 0$ , i.e.,  $\mathbf{m}_1(0, v) = 0$ .

In order to fulfill the condition (18) we use a reparameterization of the medial surface.

We look for a function  $u(t) : [0, 1] \rightarrow [0, 1]$  such that the following condition

$$\left\| \frac{\partial}{\partial t} \mathbf{m}(u(t), v) \right\| = \lambda(v)x'(t) \quad (19)$$

is satisfied.

**Remark 1.** We cannot find a function  $u(t)$  which fulfills condition (19) exactly for every  $v$  since the arc length of the  $u$ -parameter lines depends on  $v$ . Therefore we compute a function  $u(t)$  which fits best for every  $v$ . One might use a reparameterization  $u(t, v)$  but experiments confirmed that using only  $u(t)$  is sufficient for applications.

For solving equation (19) numerically we formulate the condition as a minimization problem

$$\int_0^1 \int_0^1 (\|\mathbf{m}_1(u(t), v)\| |u'(t) - \lambda(v)x'(t)|)^2 dt dv \rightarrow \min_{u(t)}. \quad (20)$$

As the optimization problem (20) is in general nonlinear, we use an iterative method for solving it. Starting with an initial solution  $u^0$ , in each iteration step the new approximation  $u^+$  of the exact solution is computed by evaluating

$$u^+ = u^c + h\Delta u, \quad (21)$$

where  $u^c$  denotes the current solution,  $\Delta u$  the update and  $h \in (0, 1]$  the step size.

As an ansatz for the unknown function  $u(t)$  we take a quadratic B-spline function

$$u(t) = \sum_{i=0}^{\ell} d_i B_i^2(t) \quad (22)$$

with control points  $\mathbf{d} = (d_0, \dots, d_{\ell})^T$ ,  $d_i \in \mathbb{R}$ , and B-splines  $B_i^2(t)$  of degree 2. Thus, we rewrite (21) as

$$\mathbf{d}^+ = \mathbf{d}^c + h\Delta\mathbf{d},$$

where  $\mathbf{d}^+ = (d_0^+, \dots, d_{\ell}^+)^T$ .  $\mathbf{d}^c = (d_0^c, \dots, d_{\ell}^c)^T$  denotes the control points of the current solution and  $\Delta\mathbf{d} = (\Delta d_0, \dots, \Delta d_{\ell})^T$  is the update.

We solve the optimization problem (20) by a Gauss-Newton-type method which is an iterative method for solving nonlinear least-squares problems. Let

$$Q(t, v, \mathbf{d}) = \|\mathbf{m}_1(u(t), v)\|u'(t). \quad (23)$$

By applying first Taylor series expansion with respect to  $\mathbf{d}$  we get a linear approximation of  $Q(t, v, \mathbf{d})$  at  $\mathbf{d}^0$ . This leads to the objective function

$$F = \int_0^1 \int_0^1 (Q(t, v, \mathbf{d}^0) + \nabla Q(t, v, \mathbf{d}^0)\Delta\mathbf{d} - \lambda(v)x'(t))^2 dt dv, \quad (24)$$

where  $\mathbf{d}^0 = (d_0^0, \dots, d_{\ell}^0)^T$  denotes the initial solution and  $\nabla Q$  is a row vector given by the partial derivatives of  $Q$  with respect to the control points  $d_i$ . The objective function (24) gives the minimization problem

$$F \rightarrow \min_{\Delta\mathbf{d}}. \quad (25)$$

In order to simplify the computation we discretize the parameter intervals and replace the integrals in the objective function by sums. A necessary condition for a minimum is that the gradient with respect to  $\Delta\mathbf{d}$  of the objective function vanishes at the minimum. Since  $F$  is quadratic in the update  $\Delta\mathbf{d}$  we obtain a system of linear equations, which can be solved easily. In one step of the Gauss-Newton-type method we solve the system of linear equations for  $\Delta\mathbf{d}$  and compute the better approximation of the exact solution with  $\mathbf{d}^+ = \mathbf{d}^c + h\Delta\mathbf{d}$ . Additionally, a step-size control might be used, but this was not necessary in our examples.

As initial solution we use a quadratic B-spline curve with control points  $(d_0, d_1, d_2)$ . Obviously,  $u(t)$  has to fulfill the boundary conditions  $u(0) = 0$  and  $u(1) = 1$ , which determine the control points  $d_0$  and  $d_2$  as 0 and 1, respectively. The point  $d_1$  is chosen as 0.5.

We use knot insertion (see [5, Subsection 7.6]) to obtain more control points of the ansatz function  $u(t)$ . In

order to satisfy the boundary conditions during the iteration method we apply the Gauss-Newton-type method to the control points  $d_1, \dots, d_{\ell-1}$ .

Finally, we get a one-dimensional quadratic B-spline function  $u(t)$  fulfilling approximately condition (19).

**Example 1.** We compute the reparameterization  $u(t)$  for the medial surface given in Figure 2 and the profile curve shown in Figure 7. Figure 9 presents the resulting B-spline curve  $u(t)$  which is obtained by solving equation (19) and the ratio of the length of the  $u$ -parameter line of the reparameterized medial surface and the speed of the profile curve. The different colors correspond to different values of  $v$ . The figure shows that the ratio is approximately constant.

### 3.3. Computation of the normals

Similar to the method described in Section 3.1 we generate a B-spline approximation  $\mathbf{n} = \mathbf{n}(u, v)$  of the unit normal vector field of the initial blade surface. Note that these unit normal vectors are well-defined, despite the fact that we have a singularly parameterized medial surface.

### 3.4. Generating the blade surface

Using the previous results we generate a B-spline representation of the blade.

Let  $u(t)$  be a one-dimensional B-spline function such that  $\mathbf{m}(u(t), v)$  and  $\mathbf{p}(t)$  satisfy equation (19) approximately. Then the two sides of the turbine blade are obtained by

$$\mathbf{b}^{(\pm)}(t, v) = \mathbf{m}(u(t), v) \pm d(v)\mathbf{n}(u(t), v)y(t). \quad (26)$$

**Proposition 1.** *The two sides  $\mathbf{b}^{(\pm)}$  of the blade fit together at  $t = 0$  with  $C^1$ -continuity, more precisely*

$$\mathbf{b}_1^{(+)}(0, v) = -\mathbf{b}_1^{(-)}(0, v). \quad (27)$$

**Proof.** It is obvious that the two boundary curves  $\mathbf{b}^{(+)}(0, v)$  and  $\mathbf{b}^{(-)}(0, v)$  coincide since  $y(0) = 0$ .

Computing the tangents in  $t$ -direction of  $\mathbf{b}^{(+)}$  and  $\mathbf{b}^{(-)}$ , respectively, and using that  $\mathbf{m}_1(0, v) = (0, 0, 0)$  and  $y(0) = 0$  gives

$$\mathbf{b}_1^{(\pm)}(0, v) = \overbrace{\mathbf{m}_1(0, v)}^{\mathbf{0}} u'(0) \pm d(v)\mathbf{n}_1(0, v) \overbrace{u'(0)}^{\mathbf{0}} \overbrace{y(0)}^{\mathbf{0}} \pm \pm d(v)\mathbf{n}(0, v)y'(0)$$

which implies (27).  $\square$

We obtain a procedurally defined B-spline surface which represents a turbine blade. By choosing suitable degrees and the knot sequences, we can compute the

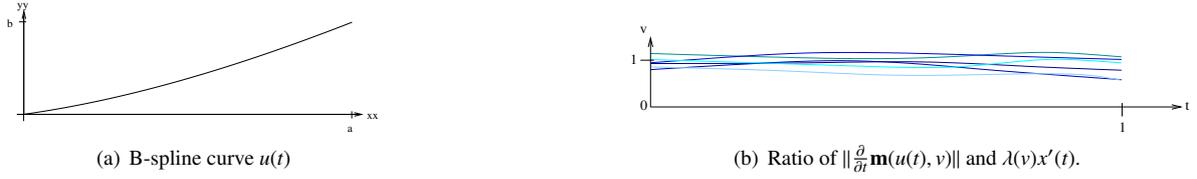


Figure 9: Example of a reparameterization function  $u(t)$ .

control points exactly, using formulas for the composition and multiplication of B-spline functions [3, 13].

In general, however, the resulting B-spline surface has high degrees and a large number of control points. Therefore, we use degree reduction and knot removal to obtain a high-quality model with lower degree and a smaller number of control points, see [6].

### 3.5. An example

We consider the point cloud of the procedurally defined medial surface of the turbine blade given in Figure 2 and the profile curve and the scaling function given in Figure 7.

We generate the pressure and the suction side of the turbine blade by adding or subtracting the profile to the medial surface, as described in Section 3. Figure 10(a) shows a point cloud which represents the turbine blade.

Figure 10(b)-(c) shows the approximated turbine blade which is described by a B-spline surface of degree (2, 2) with open uniform knot vectors and  $20 \times 10$  control points from two different views.

As can be seen in Figure 10, the blade has a high curvature at the edge where pressure and suction side fit together. In order to improve the accuracy of the fit at this edge we choose a non-uniform knot vector in  $u$ -direction. We set the knots more dense in the middle of the interval to obtain a B-spline surface with more control points in this region. We compute the average error (28) between the original point cloud and the obtained B-spline surface using different numbers of control points and different knot vectors. Table 1 shows the residual errors  $e$ ,

$$e = \frac{\sum_{i=0}^m \sum_{j=0}^n \|\mathbf{q}_{ij} - \mathbf{p}_{ij}\|}{(m+1)(n+1)}, \quad (28)$$

where  $\mathbf{q}_{ij}$  is the closest point to a point  $\mathbf{p}_{ij}$  forming a dense grid on  $\mathbf{b}(u, v)$ .

Comparing the different errors we see that not just the number of control points but also the choice of the knot vectors has an influence. Therefore we can further

Number of CP	Errors	
	uniform	non-uniform
$10 \times 5$	0.58	0.54
$20 \times 5$	0.25	0.18
$10 \times 10$	0.18	0.1
$20 \times 10$	0.08	0.06
$25 \times 10$	0.05	0.021

Table 1: The error  $e$  of resulting B-spline surfaces (turbine blades) of degree (2, 2) with different numbers of control points and different knot vectors. The bounding box is about  $115 \times 220 \times 160$ .

reduce the number of control points by choosing a suitable knot vector in  $u$ -direction. This may be a topic for future research.

## 4. B-spline representation of propeller blades

In the case of a propeller blade design we have to use a slightly different approach than for turbines. In contrast to turbine blades, a propeller blade possesses a singular point, since the last streamline collapses to one point, see Figure 11. Figure 12 shows a surface obtained via a direct fit. Note that all parameter lines with respect to one direction end in the singular point. Hence the derivative with respect to the second parameter direction is zero. As a consequence, the medial surface has no well-defined normal direction at this point. Therefore, no profile can be added at this point.

We propose the following approach to handle this situation. First we create an extended medial surface that enlarges the original one. Second, with the help of a cut-out function  $\mathbf{c}$ , this medial surface can be trimmed to the original size.

Figure 13 presents the idea of the cut-out function. The gray surface in the middle quadrilateral shows the cut-out function. The original medial surface with the singular point at the tip is described by the gray surface in the right picture. The extended medial surface is given by the entire surface (gray + red parts).

Summing up, we have to apply the following steps for the construction of a CAD-model of a propeller blade.

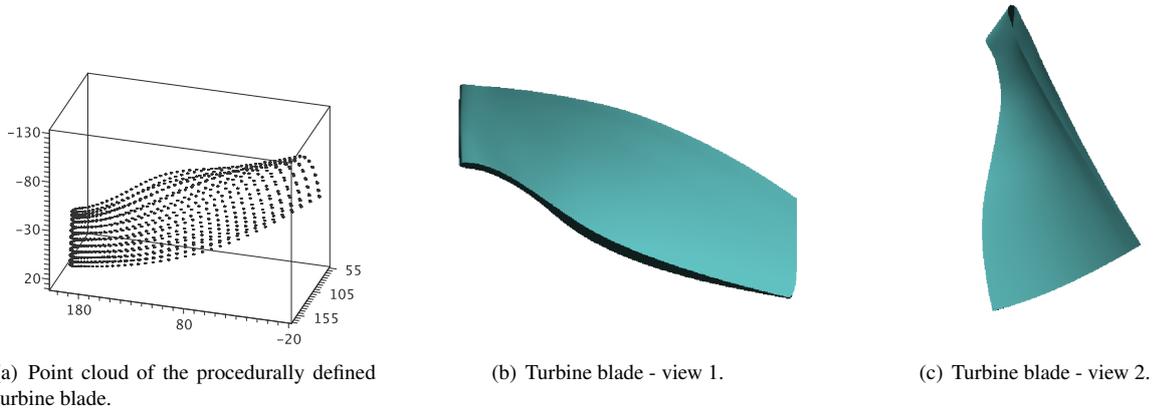


Figure 10: Turbine blade.

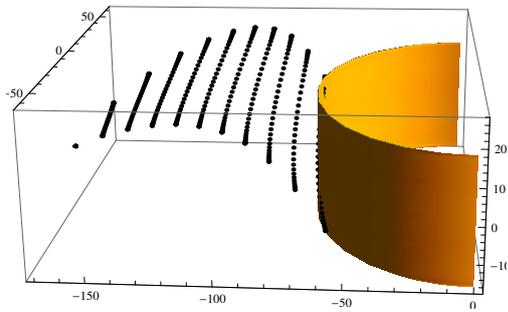


Figure 11: Point cloud of the procedurally defined medial surface of a propeller blade.

1. Create the extended medial surface and extended approximate normals.
2. Define the cut-out function.
3. Apply the reparameterization of the medial surface.
4. Generate the blade surface.

The main difference between the process used for turbine blades and the generation of a CAD-model for a propeller blade is the cut-out function  $c$ . In the following, we describe how the method is adjusted to propeller design.

#### 4.1. Extended medial surface and normals

For attaching the profile, the medial surface has to have well-defined normal directions on the entire domain. As observed before, this is not fulfilled for the medial surface of a propeller blade which is obtained by

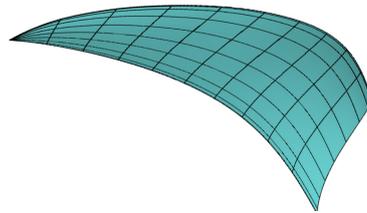


Figure 12: Parameter lines of the medial surface of a propeller blade.

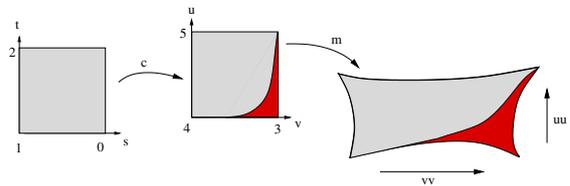


Figure 13: Sketch of a cut-out function  $c$ .

using a direct fit. To handle this situation we need to create an extended medial surface which has well-defined normals everywhere. In order to do so we compute a B-spline surface  $\mathbf{m}^*$  that enlarges the original surface  $\mathbf{m}$  and which possesses well-defined normal directions on the entire domain. Figure 14 shows an example of an extended medial surface.

#### 4.2. Cut-out function

The cut-out function  $c$  is defined on the parameter domain  $[0, 1] \times [0, 1]$ . The function  $c$  is used to “cut out” the medial surface  $\mathbf{m}$  of a propeller blade from the extended medial surface  $\mathbf{m}^*$ . Consequently, we obtain a medial surface  $\mathbf{m}$  with well-defined normals as the restriction of  $\mathbf{m}^*$ .

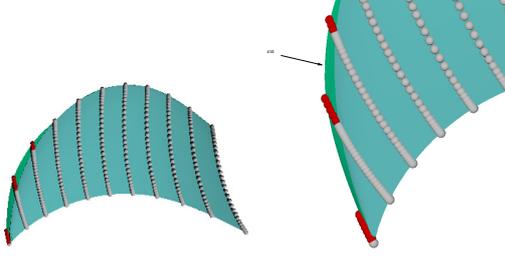


Figure 14: Extended medial surface (left) and zoom into the extension of the medial surface (right).

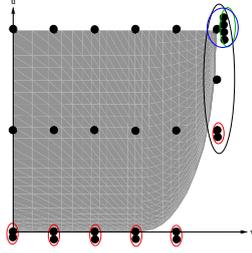


Figure 15: Example of a cut-out function  $\mathbf{c}$  fulfilling conditions (29), (30), (31) and (32) marked by green, red, blue and black loops, respectively.

For the cut-out function  $\mathbf{c}$  we use a B-spline surface of degree  $(p, q)$  with control points  $\{\mathbf{b}_{ij}\}_{j=0 \dots m}^{i=0 \dots n}$  and open knot vectors.

A typical function  $\mathbf{c}$  is shown in Figure 15. Note that the function  $\mathbf{c}$  has the following properties. Since the profile curve has a vertical tangent at  $t = 0$ , the speed of the  $u$ -parameter lines of the restricted medial surface at  $u = 0$  has also to be zero. Furthermore, we have to ensure that the medial surface of the propeller blade has a singular point. Additionally, we require that the boundary curve at  $u = 1$  of the resulting pressure and suction side of the propeller blade have collinear tangents at the tip of the blade.

Using the technique of B-splines all this properties can be achieved by choosing the control points of the cut-out function  $\mathbf{c}$  in the following way:

$$\mathbf{b}_{0n} = \mathbf{b}_{1n} = \dots = \mathbf{b}_{mn} \quad (29)$$

$$\mathbf{b}_{0j} = \mathbf{b}_{1j} \quad j \in \{0, \dots, n\} \quad (30)$$

$$\mathbf{b}_{mn} = \mathbf{b}_{m(n-1)} \quad (31)$$

$$\mathbf{b}_{mn} \text{ and } \mathbf{b}_{0(n-1)}, \mathbf{b}_{1(n-1)}, \dots, \mathbf{b}_{m(n-1)} \text{ are collinear} \quad (32)$$

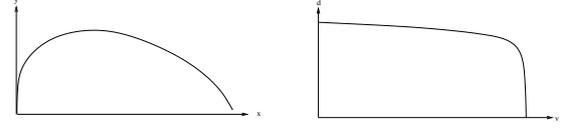


Figure 16: Example of a profile curve (left) and a scaling function (right) for a propeller blade.

#### 4.3. Reparameterization of the medial surface

For adding or subtracting the profile we have to reparameterize the medial surface. In order to do so, we use a modified version of condition (18), which we applied to turbine blades. Considering propeller blades we are interested in a function  $u(t) : [0, 1] \rightarrow [0, 1]$  such that the following condition is satisfied:

$$\left\| \frac{\partial}{\partial t} (\mathbf{m}^* \circ \mathbf{c})(u(t), v) \right\| = \lambda(v^*) x'(t), \quad (33)$$

where  $\lambda(v^*)$  is the arc length of the  $u^*$ -parameter line of  $(\mathbf{m} \circ \mathbf{c})(u, v)$  at  $v^*$ . Since the profile is added only to the trimmed medial surface we have to compose the extended medial surface with the cut-out function.

Figure 16 shows the typical form of a profile curve for a propeller blade. The scaling function for a propeller blade is zero at the end since no profile is added to the singular point of the medial surface, see Figure 16.

#### 4.4. Generating the propeller blade

Using the previous results, the pressure side and the suction side of a propeller blade can be obtained by the formula

$$\mathbf{b}^{(\pm)} = (\mathbf{m}^* \circ \mathbf{c}) \pm d(\mathbf{n}^* \circ \mathbf{c})y, \quad (34)$$

where  $\mathbf{m}^*$  and  $\mathbf{n}^*$  describe the extended medial surface and the associated normals, respectively. Furthermore,  $\mathbf{c}$  denotes the cut-out function,  $y$  is the  $y$ -coordinate of the profile curve and  $d$  is the scaling function.

That means, for computing the pressure and the suction side of the blade we have to compose the extended medial surface  $\mathbf{m}^*$  and the normals  $\mathbf{n}^*$  with the cut-out function. We add the profile to the trimmed and reparameterized medial surface and obtain a propeller blade with a well-defined behavior of the tangent plane everywhere.

**Proposition 2.** *Let  $\mathbf{c}(u, v)$  be a B-spline surface satisfying the properties given in (29)-(32). Furthermore, we assume that the scaling function  $d(v)$  is zero at  $v = 1$  and the profile curve  $\mathbf{p}(t) = (x(t), y(t))$  fulfills  $x'(0) = 0$ . Consider the B-spline surfaces*

$$\mathbf{b}^{(\pm)}(t, v) = (\mathbf{m}^* \circ \mathbf{c})(u(t), v) \pm d(v)(\mathbf{n}^* \circ \mathbf{c})(u(t), v)y(u(t)). \quad (35)$$

Then we have the following results:

1.  $\mathbf{b}^{(+)}(t, v)$  and  $\mathbf{b}^{(-)}(t, v)$  fit together with  $C^1$ -continuity at  $t = 0$ .
2. The resulting propeller has a well-defined tangent plane at the tip.
3. The boundary curves  $\mathbf{b}^{(+)}(1, v)$  and  $\mathbf{b}^{(-)}(1, v)$  have collinear tangents at  $v = 1$ .

**Proof.** 1. The proof is similar to Proposition 1.

2. In a second step we show that the propeller has a well-defined tangent plane at the tip. Let  $\mathbf{c}(u(t), v) = (c^1(u(t), v), c^2(u(t), v))$ .

First we compute  $\mathbf{b}_1^{(+)}(t, 1)$ . Using that  $d(1) = 0$  and according to the construction of  $\mathbf{c}$ , see Figure 15, we have that

$$\mathbf{b}_1^{(+)}(t, 1) = \mathbf{m}_1^*(\mathbf{c}(u(t), 1)) \underbrace{c_1^1(u(t), 1)}_{=0} u'(t) + \quad (36)$$

$$+ \mathbf{m}_2^*(\mathbf{c}(u(t), 1)) \underbrace{c_1^2(u(t), 1)}_{=0} u'(t) = \mathbf{0} \quad (37)$$

Thus, one edge collapses to a single point. It is obvious that also  $\mathbf{b}_1^{(-)}(t, 1)$  is zero, and similar for all higher derivatives,

$$\left. \frac{\partial^k}{\partial t^k} \mathbf{b}^{(+)}(t, v) \right|_{v=1} = \left. \frac{\partial^k}{\partial t^k} \mathbf{b}^{(-)}(t, v) \right|_{v=1} = \mathbf{0}. \quad (38)$$

In order to show that a well-defined tip exists, we have to prove that all derivatives with respect to  $v$  are coplanar. We get

$$\begin{aligned} \mathbf{b}_2^{(+)}(t, 1) &= \mathbf{m}_1^*(\underbrace{\mathbf{c}(u(t), 1)}_{=(1,1)}) c_2^1(u(t), 1) + \\ &+ \mathbf{m}_2^*(\mathbf{c}(u(t), 1)) \underbrace{c_2^2(u(t), 1)}_{=0} \pm \\ &\pm d'(1) (\mathbf{n}^*(\underbrace{\mathbf{c}(u(t), 1)}_{=(1,1)}) y(u(t))) \pm \\ &\pm \underbrace{d(1)}_{=0} \frac{\partial}{\partial v} (\mathbf{n}^* \circ \mathbf{c})(u(t), 1) \\ &= \mathbf{m}_1^*(1, 1) c_2^1(u(t), 1) \pm d'(1) \mathbf{n}^*(1, 1) y(u(t)). \end{aligned} \quad (39)$$

This proves that all directions  $\mathbf{b}_2^{(+)}(t, 1)$  and  $\mathbf{b}_2^{(-)}(t, 1)$  lie in the plane spanned by the vectors  $\mathbf{m}_1^*(1, 1)$  and  $\mathbf{n}^*(1, 1)$ .

3. Finally, we have to show that the two boundary curves  $\mathbf{b}^{(+)}(1, v)$  and  $\mathbf{b}^{(-)}(1, v)$  have collinear tangents at  $v = 1$ , see Figure 17. We obtain from (39) the two tangent directions

$$\mathbf{b}_2^{(+)}(1, 1) = \mathbf{m}_1^*(1, 1) \underbrace{c_2^1(1, 1)}_{=0} \pm d'(1) \mathbf{n}^*(1, 1) y(1) \quad (40)$$

which are collinear.  $\square$

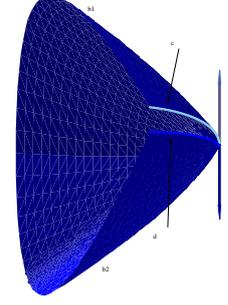


Figure 17: Having collinear tangents at  $v = 1$  for the two boundary curves  $\mathbf{b}^{(+)}(1, v)$  and  $\mathbf{b}^{(-)}(1, v)$ .

#### 4.5. An example

We consider a grid of points sampled along the streamlines of a procedurally medial surface of the propeller blade given in Figure 11 and the profile curve and the scaling function given in Figure 16. As described in the previous sections, we use the following procedure for generating a B-spline representation of the propeller. First we compute the extended medial surface and the associated normal vectors. Then we reparameterize the medial surface and generate the propeller blade by adding the profile. The resulting point cloud which describes the pressure and the suction side of the blade is given in Figure 21(a).

First we have to compute an extended medial surface which possesses well-defined normals on the entire domain. This is done by generating additional points for some of the streamlines close to the tip. For this we fit the data points of a streamline by a curve and extrapolate this curve to get additional data points. For simplicity, we take a quadratic Bézier curve, and choose the parameter values of the data points chord length parameterized. The number of additional data points depends on the shape of the blade. According to our experience with typical blades we obtain feasible results by computing about 10 additional points per streamline.

Figure 18 shows a fixed streamline (black points) which is fitted via a Bézier curve (black curve). The extrapolated points are shown in red. The associated parameter values are lying in the red regions. This extrapolation is performed for several streamlines. Clearly, the choice of the streamlines which are to be extrapolated depends on the cut-out function  $\mathbf{c}$ .

At the singular point at the tip of the blade we have to proceed differently, as no curve can be fit to a single point. Hence, we translate the Bézier curve from the last but one streamline such that it passes through the singular point.

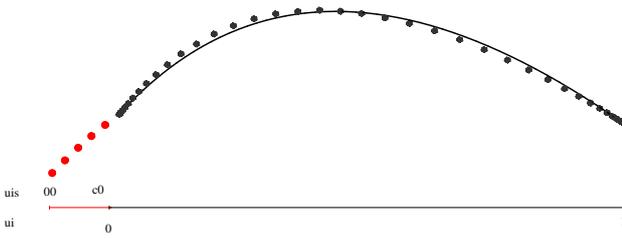


Figure 18: Streamline with additional points.

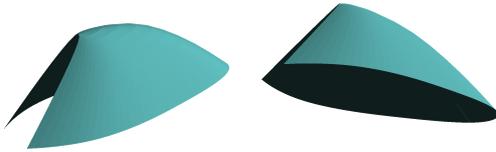


Figure 20: Zoom into the tip of the propeller blade.

We extend the original grid of points by adding the extrapolated points and apply least-squares fitting to obtain an extended medial surface  $\mathbf{m}^*$  which possesses well-defined normals on the entire domain. An example of a B-spline surface which describes an extended medial surface is given in Figure 19(b).

Now we can apply the construction described in the previous sections to generate the two boundary surfaces of the blade.

In order to analyze the behavior of the tip of the propeller blade, we visualize the point cloud given in Figure 21(a) as a surface. Figure 20 shows the zoom of the tip. The picture confirms that the resulting propeller has a well-defined tangent plane at the tip. Similar to the case of turbine blades we use least-squares approximation to obtain a B-spline surface representing the propeller blade. In order to preserve the well-defined tangent plane at the tip of the propeller we have to prescribe the normal direction at the tip in the fitting process. Figure 21 shows a B-spline surface of degree (2, 2) with  $30 \times 10$  control points which describes both the the pressure and the suction side of the propeller blade.

Analogously to the turbine blade we compute the average error  $e$  for different numbers of control points and different knot vectors. Table 2 shows the resulting errors.

## 5. Conclusion

We presented a design tool for constructing a CAD-model for blades of turbines and propellers. Our design

Number of CP	Errors	
	uniform	non-uniform
$10 \times 5$	0.32	0.21
$20 \times 5$	0.21	0.18
$10 \times 10$	0.2	0.09
$20 \times 10$	0.067	0.047
$25 \times 10$	0.059	0.04

Table 2: The error  $e$  of resulting B-spline surfaces (propeller blades) of degree (2, 2) with different numbers of control points and different knot vectors. The bounding box is about  $100 \times 60 \times 40$ .

scheme corresponds to the classical construction of a blade, using the medial surface and profiles attached to it. As the main new contribution of the paper, we realized the entire construction based on B-spline techniques. To this end we designed suitable reparameterizations and cut-out functions, which were composed with a B-spline representation of the blade’s medial surface. We contributed novel shape modeling techniques based on singularly parameterized medial surfaces and showed that these techniques are of substantial interest for industrial applications.

The main difference between the computation of a CAD-model for a turbine blade and for a propeller blade is the construction of an extended medial surface in the case of propellers. This is necessary for handling the singular point at the tip of the propeller. The single steps of the design process of the turbine and propeller blade are described in Section 3 and 4, respectively. Furthermore we presented several examples.

Obviously, it is possible to modify different steps in the construction. For instance, one may use other, more direct, methods for designing the medial surface.

Our method is fully implemented and is suitable for an interactive design process. As can be seen in the examples in Section 3.5 and Section 4.5, B-spline surfaces with about 200 control points lead to feasible results. Compared to existing models used by ANDRITZ HYDRO with approximately 1000 points, this number is very small.

**Acknowledgment.** This research was supported by the European Union through the 7th Framework programme, project no. 218536 “Exact geometry simulation for optimized design of vehicles and vessels” (EX-CITING).

## References

- [1] J.M. Anders and J. Haarmeyer. A parametric blade design system. lecture series at Karman Institute for fluid dynamics, 1999.

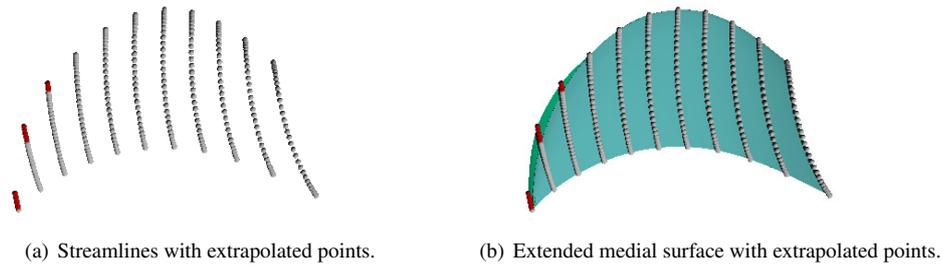


Figure 19: Extended medial surface.

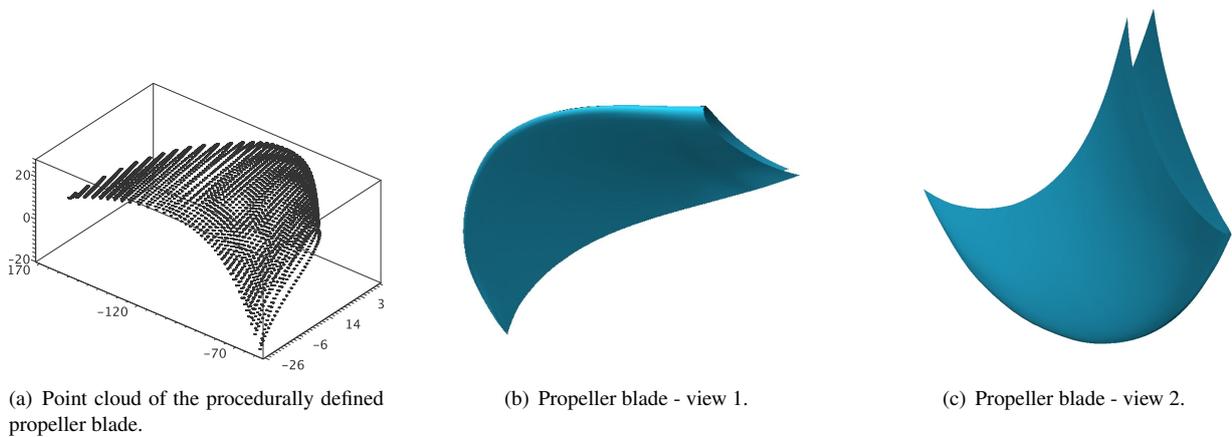


Figure 21: Propeller blade.

- [2] B. Bastl, B. Jüttler, M. Lávička, and T. Schulz. Blends of canal surfaces from polyhedral medial transform representations. *Computer-Aided Design*, 43(11):1477 – 1484, 2011.
- [3] T. D. DeRose, R. N. Goldman, H. Hagen, and S. Mann. Functional composition algorithms via blossoming. *ACM Trans. Graph.*, 12:113–135, April 1993.
- [4] P. Giblin and B.B. Kimia. A formal classification of 3d medial axis points and their local geometry. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26:238–251, 2004.
- [5] R. Goldman. *Pyramid Algorithms: A Dynamic Programming Approach to Curves and Surfaces for Geometric Modeling*. Morgan Kaufmann Publishers, San Diego, 2002.
- [6] J. Hoschek and D. Lasser. *Fundamentals of computer aided geometric design*. A K Peters Ltd., Wellesley, MA, 1993.
- [7] J. Hoschek and R. Müller. Turbine blade design by lofted B-spline surfaces. *Journal of Computational and Applied Mathematics*, 119:235–248, 1999.
- [8] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput. Methods Appl. Mech. Engrg.*, 194:4135–4195, 2004.
- [9] H. Keck, E. Goede, and J. Pestalozzi. Experience with 3-D Euler Flow Analysis as a Practical Design Tool. *International Association for Hydraulic Research*, 2000.
- [10] G.N. Koini, S.S. Sarakinos, and I.K. Nikolos. A software tool for parametric design of turbomachinery blades. *Adv. Eng. Softw.*, 40(1):41–51, 2009.
- [11] E. Kreyszig. *Differential geometry*. Dover, New York, 1991.
- [12] P. L. Miller IV, J. H. Oliver, D. P. Miller, and D. L. Tweedt. Bladecad: An interactive geometric design tool for turbomachinery blades. *NAA Technical Memorandum 107262*, 1996.
- [13] K. Mørken. Some identities for products and degree raising of splines. *Constructive Approximation*, 7:195–208, 1991.
- [14] S. Musuvathy, E. Cohen, and J. Damon. Computing medial axes of generic 3D regions bounded by B-spline surfaces. *Computer-Aided Design*, 43(11):1485 – 1495, 2011.
- [15] X. Qian and D. Dutta. Design of heterogenous turbine blade. *Computer-Aided Design*, 35:319–329, 2001.
- [16] M. Sallaberger, M. Fisler, M. Michaud, K. Eisele, and M. Casey. The design of francis turbine runners by 3d euler simulations coupled to a breeder genetic algorithm. *International Association for Hydraulic Research*, 1990.
- [17] M. Strscheletzky. Zur konformen Abbildung von Stromflächen in rotationssymmetrischen Hohlräumen. *Voith Forschung und Konstruktion*, 14:no. 2, 1965.
- [18] X. Ye, T.R. Jackson, and N.M. Patrikalakis. Geometric design of functional surfaces. *Computer-Aided Design*, 28:741–752, 1996.
- [19] P. Yushkevich, P.T. Fletcher, S. Joshi, A. Thall, and S.M. Pizer. Continuous medial representations for geometric object modeling in 2D and 3D. *Image and Vision Computing*, 21(1):17 – 27, 2003.