Highly-accurate 5-axis flank CNC machining with conical tools

Amaia Calleja[a], Pengbo Bo[*,b], Haizea González[a], Michael Bartoň[c], Luis Norberto López de Lacalle[a]

[a]*High Performance Manufacturing Group, Department of Mechanical Engineering,*
the University of the Basque Country, Plaza Ingeniero Torres Quevedo 1, 48013, Bilbao, Basque Country, Spain
[b]*School of Computer Science and Technology, Harbin Institute of Technology,*
West Wenhua Street 2, 264209 Weihai, China
[c]*BCAM – Basque Center for Applied Mathematics,*
Alameda de Mazarredo 14, 48009 Bilbao, Basque Country, Spain

Abstract

A new method for 5-axis flank computer numerically controlled (CNC) machining using a predefined set of tappered ball-end-mill tools (aka conical) cutters is proposed. The space of lines that admit tangential motion of an associated truncated cone along a general, doubly curved, free-form surface is explored. These lines serve as discrete positions of conical axes in 3D space. Spline surface fitting is used to generate a ruled surface that represents a single continuous sweep of a rigid conical milling tool. An optimization based approach is then applied to globally minimize the error between the design surface and the conical envelope. Our computer simulation are validated with physical experiments on two benchmark industrial datasets, reducing significantly the execution times while preserving or even reducing the milling error when compared to the state-of-the-art industrial software.

Key words: 5-axis CNC machining, flank milling, envelope surface, tangential movability, shape manufacturing

1. Introduction & Motivation

Free-form surfaces represented as non-uniform rational B-splines (NURBS) are popular tools for geometric modeling, engineering, and architectural curved design [1]. These curved surfaces are easily controlled by their control point network, allowing the designers and modelers a straightforward tool towards the intended shape. In the isogeometric analysis literature, free-form surfaces appear in physically based problems as approximations of solutions of partial differential equations (PDE) [2, 3]. For example, shapes such as turbine blades and impellers come as an equilibrium comforting specific physical constraints like force, pressure etc. That is, these objects possess complex free-form geometry not for aesthetic reasons but to optimally satisfy physical constraints.

Efficient manufacturing of free-form shapes has been of interest for many decades [4, 5, 6, 7, 8, 9, 10, 11]. CNC-machining is the leading subtractive technology, and even though additive manufacturing is increasing its share on the manufacturing market [12], certain types of mechanical components are, e.g. due to their stiffness properties, preferred to be manufactured from a single block by material removal rather than the other way around by additive technologies.

Our work aims at efficient, high quality flank CNC-machining of general, doubly curved, free-form surfaces. When compared to the flat-end machining, flank milling is popular due to two main reasons: i) the material removal is more efficient as the cutter touches the material block not at a single point, but along a whole curve; ii) due to the tangential contact between the material and the cutter, flank milling is, by definition, scallop free on every sweep patch. The second property makes flank milling an ideal methodology for the final polishing stage of the machining process.

The most frequently used cutting tools for flank CNC machining are cones and cylinders, see Fig. 1. Given a set of conical cutters (considering cylinder as a special case of a cone), the main focus is to detect large patches of a free-form shape that can be approximated, up to a fine machining tolerance, by a single conical envelope, see Fig. 2.

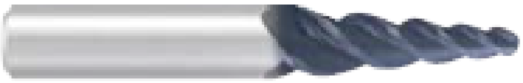

Figure 1: A conical tool for flank milling. The conical part is provided by a perforation for efficient material removal.

2. Previous work and contributions

Our work belongs to the category of flank (aka peripheral) milling where the milling tool moves along the material block tangentially, touching it along a 3D contact curve. This contact curve, known as a *characteristic* (or grazing curve) is not static and may change dynamically in time, in contrast to sweep surfaces generated by motions of rigid profiles [10].

In general, the milling tool can be arbitrary surface of revolution such as barrel [13] or general rotational surface [14].

[*]Corresponding author

Email addresses: amaia.calleja@ehu.eus (Amaia Calleja), pbbo@hit.edu.cn (Pengbo Bo), haizea.gonzalez@ehu.eus (Haizea González), mbarton@bcamath.org (Michael Bartoň), norberto.lzlacalle@ehu.eus (Luis Norberto López de Lacalle)

However, simpler shapes are preferable in practice and e.g. conical tools are frequently used for finishing machining operations of complex surfaces. In practice the conical milling tool is provided by a helicoidal perforation (teeth), see Fig. 1, that serves for material removal. For the simplicity of the argument, however, the milling tool is conceptualized as a truncated cone in our theoretical considerations.

In the computer aided design literature, a lot of work is devoted to free-form surface approximation by ruled surfaces [15, 16, 13, 17, 18, 19, 20, 21, 22, 23]. While such approach is practical as one can approximate a free-form surface by ruled surface patches within arbitrarily fine precision [17], it typically requires a large number of small patches that consequently leads to very time-demanding milling process.

Focusing on ruled surfaces is a certain restriction, especially when assumed that the contact curve is a straight line. For a general instantaneous (screw) motion [24], the contact curve between the movable cone and its envelope is a general 3D curve, see Fig. 3. Even more special case is a situation when the ruled surface becomes developable [22] as this requires the surface normal to be constant along each ruling. Therefore, approximating general free-form surfaces by these special (e.g., developable) patches is a significant restriction of the search space. In contrast, in this work the research is focused on envelope patches that correspond to a general screw motion (and therefore general characteristic) at every time instant.

The milling process involves other factors than the approximation quality between the design and manufactured surfaces. Cutting force, for example, is an entity that describes how much force has to be applied to the milling tool and is preferable to be low as high forces are typically strongly correlated with the milling tool vibrations. Mathematical modeling of the cutting force under the flank milling is also an active research area, see e.g. [25, 26, 27, 28, 29] and references cited therein. Looking for toolpaths that minimize (the mean value of) the tool deflection force are studied in [30]. Another physical factor strongly correlated to the cutting force is not only the vibration of the milling tool but also of the milled object itself. Vibration phenomenon of thin impeller blades during flank milling is discussed in [27]. An optimization-based approach that incorporates physical properties such as stiffness or vibrations is proposed in [31]. The main objective is to improve the machining process by reducing deflection and vibrations of the cutter. The shape of the cutter is optimized towards higher stiffness, while the trajectory surface is computed by interpolating the cutter axes.

Other important factors in high-performance machining are high velocities and accelerations, which both need to be controlled throughout the milling process [32, 33, 34]. An algorithm that seeks the optimal feedrate interpolation, considering both tangential and in-axis jerk of the milling tool, is presented in [32]. A trade-off between the geometric accuracy and smoothness of the motion is discussed in [35]. The reader is referred to the survey paper [36] for a detailed list of relevant literature on flank 5-axis CNC milling.

In this paper, we focus solely on the approximation quality between the design surface and the milled conical envelope and, in contrast to [31, 14], consider pre-defined conical milling tools. In all the state-of-the-art milling literature, to the best of our knowledge, the initial trajectory of the milling axis is assumed as an input or users' intervention is necessary to provide a meaningful initial trajectory. A recent research on automatic detection of conical envelopes [37] is adopted, showing that this initialization strategy, when incorporated to real manufacturable process, reduces the milling time significantly by detecting large envelopes within fine machining tolerances.

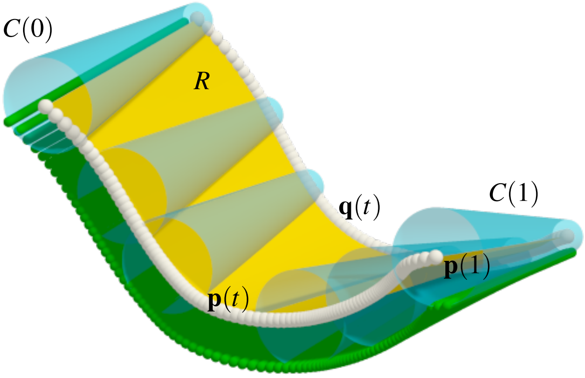

Figure 2: One-parameter motion of a truncated cone C. The motion of the conical axis $l = \mathbf{pq}$ defines a ruled surface R (yellow). A locus of all characteristics forms the (lower) envelope Ω (green).

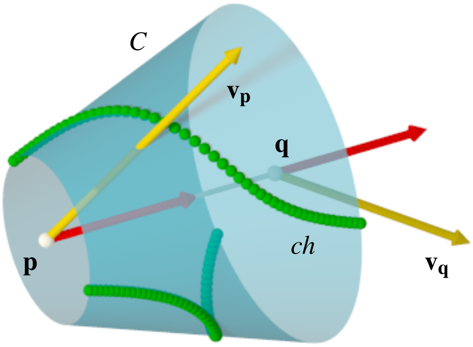

Figure 3: An instantaneous motion of a rigid conical axis $\mathbf{pq}$ is determined by two velocity vectors (yellow) that satisfy Eq. (2), i.e., their projections onto $\mathbf{pq}$ are equally oriented vectors of the same length (red). A characteristic ch (green) is the contact curve between the moving cone C and its envelope and in general consists of two branches.

The rest of the paper is organized as follows. Section 3 gives a mathematical description of the tangential movability of a truncated cone along a free-form surface and Section 4 presents the algorithm that computes conical envelopes that fit the input reference geometry. Section 5 shows our computer simulations as well as the real machining results and offers a comparison with the state-of-the-art software. Finally, Section 6 discusses the future research directions and concludes the paper.

3. Basics of kinematic geometry

We now recall several facts from kinematic and differential geometry that are fundamental pillars in our computational framework. We first start with the first order analysis of a rigid body motion.

Let $l := \mathbf{pq}$ be a finite line in 3D, see Fig. 2, considered as a function of time t. Denote $\mathbf{p}(t)$ and $\mathbf{q}(t)$ the trajectories of the endpoints, $t \in [0,1]$. Since l remains *rigid* during the motion, the distance-preserving constraint of its length $d = \|\mathbf{p}(t) - \mathbf{q}(t)\|$ reads as

$$\|\mathbf{p}-\mathbf{q}\|^2 = \langle \mathbf{p}-\mathbf{q}, \mathbf{p}-\mathbf{q}\rangle = \text{const.}, \tag{1}$$

where $\langle \cdot, \cdot \rangle$ is the scalar product. Differentiating with respect to t and denoting the velocity vectors by $\mathbf{v_p} = \dot{\mathbf{p}}(t), \mathbf{v_q} = \dot{\mathbf{q}}(t)$, the first order length-preserving constraint of l becomes

$$\langle \mathbf{v_p}, \mathbf{p}-\mathbf{q}\rangle = \langle \mathbf{v_q}, \mathbf{p}-\mathbf{q}\rangle \tag{2}$$

which is known as a *projection rule*, see Fig 3.

Further recall that a rigid body motion in 3-space requires six degrees of freedom (DoFs), and therefore a rigid motion of a line requires five DoFs (one DoF corresponds to the rotations around l). The *instantaneous motion* of a line (a motion that infinitesimally moves l) is therefore represented by a pair of vectors $\mathbf{v_p}$, $\mathbf{v_q}$ that satisfy Eq. (2), and has also, up to a multiplication factor, five degrees of freedom, see e.g. [24]. These five DoFs to control the instantaneous motion of l are encoded by a five-dimensional vector $\mathbf{x}$, where the first three coordinates determine the velocity vector $\mathbf{v_p}$, while the other two are the coordinates of $\mathbf{v_q}$ in a vector space perpendicular to $\mathbf{pq}$, see e.g. [14]. Note that any such $\mathbf{x}$ directly satisfies (2).

Consider further a truncated cone C associated to l. An instantaneous motion of a conical axis l is shown in Fig. 3. The instantaneous motion determines uniquely a *characteristic ch* that is a locus of points $\mathbf{c}$, $\mathbf{c} \in C$, such that

$$\langle \mathbf{v_c}, \mathbf{n_c}\rangle = 0, \tag{3}$$

where $\mathbf{v_c}$ is the velocity vector at $\mathbf{c}$, and $\mathbf{n_c}$ is the cone surface unit normal at $\mathbf{c}$. Observe that for a general instantaneous motion (i.e., neither translation nor rotation), ch is an algebraic curve of degree four [38]. In a special case of translation or rotation, ch consist of a pair of rulings on C, but in general

$$|\mathbf{v_p}, \mathbf{v_q}, \mathbf{d}| \neq 0, \tag{4}$$

$\mathbf{d}$ being the directional unit vector of l, the characteristic is not a straight line [24]. Observe that a family of conical envelopes is significantly broader, in the sense of inclusion, than family of ruled surfaces. Solving the approximation problem by considering only ruled surfaces is therefore a very strong restriction, because the set of instantaneous motions of a cone with linear characteristics represent a subset of Lebesgue measure zero among all admissible motions that satisfy Eq. (2) (general instantaneous motions).

3.1. Tangential movability

Let Φ be the design surface and l be a rotational axis of a truncated cone C, representing the conical milling tool, see Fig. 2. The cone is required to move tangentially to Φ. Such a constraint on an infinitesimal motion $\mathbf{x}$ is expressed by

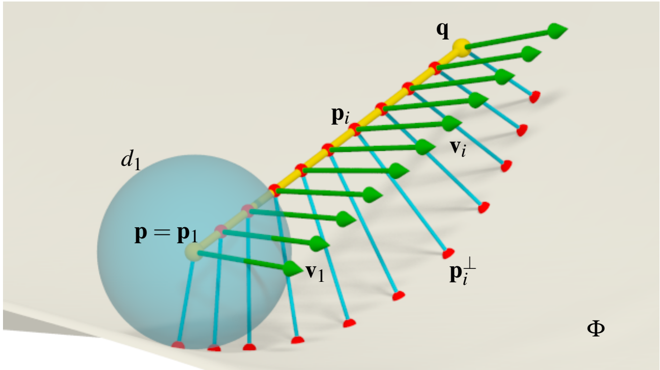

Figure 4: Tangential movability. A line $l = \mathbf{pq}$ is tangentially movable along Φ if there exists an instantanous motion (green) which moves every $\mathbf{p}_i$, $\mathbf{p}_i \in l$, such that its velocity vector $\mathbf{v}_i$ is perpendicular to $\mathbf{p}_i\mathbf{p}_i^\perp$. In our discrete setting, this corresponds to Eq. (5). The distances $d_i = \|\mathbf{p}_i\mathbf{p}_i^\perp\|$ determine the shape of a milling tool that locally admits tangential motion along Φ (here the position of l w.r.t. Φ defines a barrel-like tool [13]), i.e., d_is do not change linearly along l.

$$F(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \langle \mathbf{v}_i, \frac{\mathbf{p}_i - \mathbf{p}_i^\perp}{\|\mathbf{p}_i - \mathbf{p}_i^\perp\|}\rangle^2 = \mathbf{x}\mathbf{A}\mathbf{x}^{\mathrm{T}} \to \min, \tag{5}$$

with the constraint

$$\mathbf{v_m}\mathbf{v_m}^{\mathrm{T}} = 1, \tag{6}$$

where n is the number of sampled points $\mathbf{p}_i$ on l, $\mathbf{p}_i^\perp$ are their orthogonal projections onto Φ, $\mathbf{v}_i$ are the velocity vectors associated to $\mathbf{p}_i$, and $\mathbf{v_m}$ is the velocity at the midpoint $\mathbf{m}$ of l, see Fig. 4.

Observe that the objective function F is quadratic and has a trivial minimizer ($\mathbf{x} = [0,0,0,0,0]$). The purpose of the constraint (6) is to eliminate this trivial minimizer, that geometrically represents a degenerate, zero vector field that keeps l static. Minimization (5) with (6) yields a generalized eigenvalue problem that has a unique non-trivial minimizer $\mathbf{x}^*$.

Note that if Φ is an exact envelope, (3) holds along the whole characteristic and therefore $F(\mathbf{x}^*) = 0$. If Φ is a general free-form surface that is not an envelope, the minimization (5) seeks the instantaneous motion that is as tangential as possible. The lines with low values of F are said to be *tangentially movable* along Φ.

4. Fitting free-form surfaces by conical envelopes

The algorithm to detect parts of Φ such that Φ can be well approximated by conical envelopes consists of an initialization stage that seeks initial trajectories of the milling tool axis, and a sequential optimization. These two main parts are now described in a detail.

4.1. Initial ruled surface

Our approach is built upon the previous research on approximating free-form surfaces by envelopes of general surfaces of

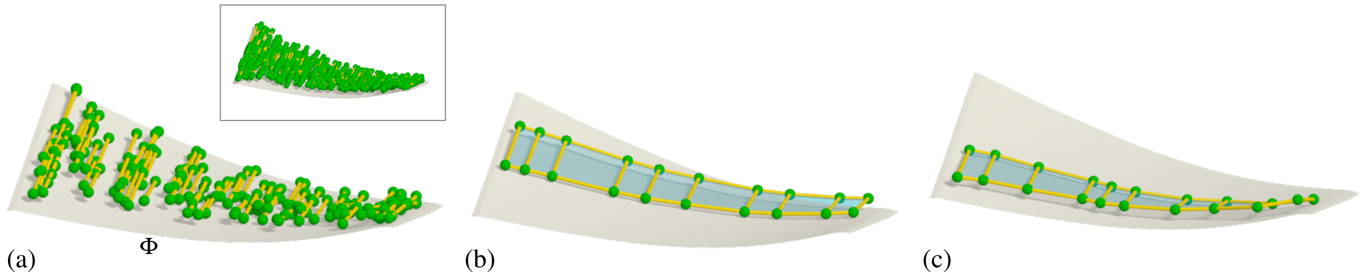

Figure 5: Initial ruled surfaces. Using [37], (a) movable lines that admit tangential motion of the prescribed conical tool along Φ are computed. Depending on the sampling density, this search for movable lines yields thousands of candidates lines (framed). These lines are clustered (bottom (a)) and ordered (b,c) to form discrete ruled surfaces as initial motions of the conical cutter.

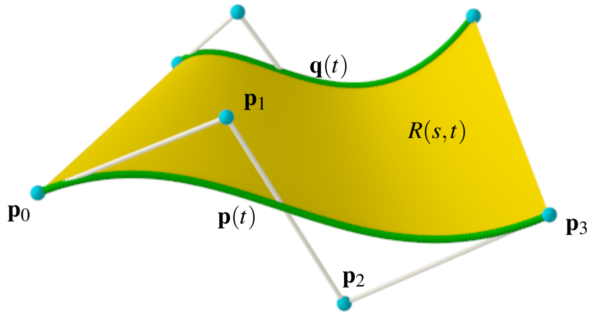

Figure 6: Ruled surface $R(s,t)$ is realized as a tensor product B-spline patch, linear in the s-direction (rulings) and of a polynomial degree $p = 3$ in the t-direction (time); here only a polynomial segment in Bézier representation is shown. The boundary curves $\mathbf{p}(t)$ and $\mathbf{q}(t)$ (green) and their control polygons are shown. The control points (blue) are the unknowns in the optimization.

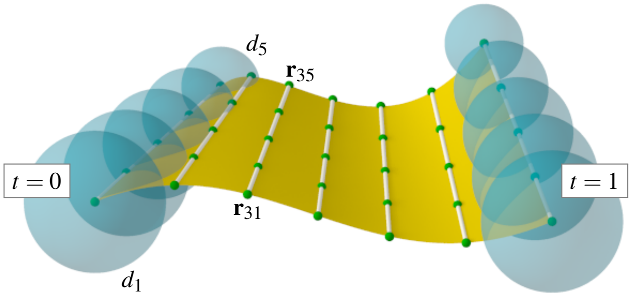

Figure 7: Motion optimization. The ruled surface (yellow) is sampled by $m \times n$ points $\mathbf{r}_{ij}$ (here $m = 7$, $n = 5$). The cone C is realized as a one-paremeter family of spheres (transparent), centered along the axis with a linear constraint on their radii d_i, $i = 1,\ldots,5$. This linear constraint has to be preserved throughout the motion ($t \in [0,1]$). The radii are known from the meridian of the input milling tool.

revolution [14]. Good candidate lines that correspond to discrete positions of a conical tool with prescribed slope and size. are found first. This search is based on the second order analysis of a surface distance function and the fact, that for conical cutters, the surface distance function changes linearly along the conical axis [37]. In Fig. 5, the (clustered) rulings that can serve as axes of the corresponding conical cutters are shown. That is, the distance functions along these lines change accordingly to a particular linear meridian, and the lines are tangentially movable along Φ. The set of candidate lines that admit tangential motion of the associated cone along Φ are sequentially clustered to form ordered sequences of lines, see Fig. 5(b,c). For each such an ordered sequence of lines, a ruled surface fitting is applied to define an initial ruled surface. The reader is referred to [37] for a more detailed explanation of this automatic initialization algorithm.

4.2. Motion optimization

From the initialization stage, see Section 4.1, an initial ruled surface is given

$$R(t,s) = (1-s)\mathbf{p}(t) + s\,\mathbf{q}(t), \quad [t,s] \in [0,1] \times [0,1], \tag{7}$$

where $\mathbf{p}(t)$ and $\mathbf{q}(t)$ are the two boundary cubic B-spline curves, see Fig. 6. During the motion, each point of the moving cutter axis l must keep its linear distance from Φ. This distance is given as an input (by knowing the meridian of the conical milling tool), see Fig. 7.

The ruled surface R is uniformly sampled both in t and s parametric directions to obtain $\mathbf{r}_{ij} := R(t_i,s_j)$, $i = 1,\ldots,m$, $j = 1,\ldots,n$. At every time instant t, the distance between the points on l and the reference surface Φ have to be as close as possible to the distances prescribed by the linear meridian of the cone. Denote these distances by d_j, $j = 1,\ldots,n$. The approximation objective is formulated as a minimization problem

$$F_{\text{prox}}(\mathbf{p},\mathbf{q}) = \frac{1}{mn}\sum_{j=1}^{n}\sum_{i=1}^{m}(\text{dist}(\mathbf{r}_{ij},\Phi) - d_j)^2 \rightarrow \min \tag{8}$$

subject to the rigidity constraints

$$F_{\text{rigid}}(\mathbf{p},\mathbf{q}) = \langle \mathbf{p}(t_i) - \mathbf{q}(t_i), \mathbf{p}(t_i) - \mathbf{q}(t_i)\rangle - L^2 = 0, \tag{9}$$

where dist(,) is a point-surface distance and L is the length of l. The unknowns in the minimization are the control points of the two B-spline curves $\mathbf{p}(t)$ and $\mathbf{q}(t)$, see Fig. 6. If not stated differently, $m = 100$ and $n = 30$ are used in our computations.

Further denote by $\mathbf{r}_{ij}^{\perp}$ the footpoints of $\mathbf{r}_{ij}$ on Φ, and $\mathbf{n}_{ij}$ the unit normals at $\mathbf{r}_{ij}^{\perp}$ oriented towards $\mathbf{p}_{ij}$, see Fig. 8. The point-surface proximity constraints then reads as

$$\begin{aligned} F_{\text{point}}(\mathbf{a},\mathbf{b}) &= \frac{1}{mn}\sum_{j=1}^{n}\sum_{i=1}^{m}\|\mathbf{r}_{ij} - (\mathbf{r}_{ij}^{\perp} + d_j\mathbf{n}_{ij})\|^2, \\ F_{\text{plane}}(\mathbf{a},\mathbf{b}) &= \frac{1}{mn}\sum_{j=1}^{n}\sum_{i=1}^{m}(\langle \mathbf{r}_{ij} - \mathbf{r}_{ij}^{\perp}, \mathbf{n}_{ij}\rangle - d_j)^2, \end{aligned} \tag{10}$$

which correspond to a point-point and point-plane distance constraints, respectively.

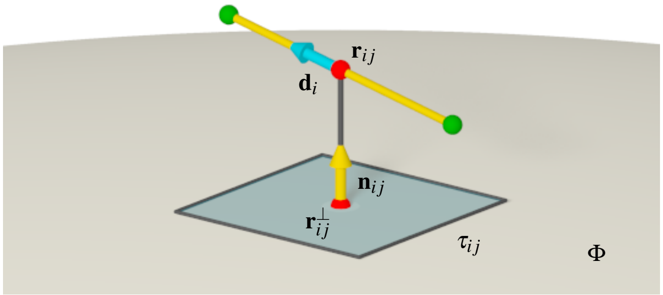

Figure 8: Point-surface distance optimization. The point $\mathbf{r}_{ij}$ is required to have a known distance d_j from Φ. τ_{ij} is the tangent plane of Φ at $\mathbf{r}_{ij}^{\perp}$, the footpoint of $\mathbf{r}_{ij}$ on Φ. The optimization functional consists of two components, see Eq. (10), that correspond to point-point and point-plane constraints, respectively.

To achieve a fair motion, fairness of the two boundary curves is expressed by

$$\begin{aligned} F_{\text{fair}}(\mathbf{p},\mathbf{q}) &= \frac{1}{m}\sum_{i=2}^{m-1}(\mathbf{p}(t_{i-1})-2\mathbf{p}(t_i)+\mathbf{p}(t_{i+1}))^2 \\ &+ \frac{1}{m}\sum_{i=2}^{m-1}(\mathbf{q}(t_{i-1})-2\mathbf{q}(t_i)+\mathbf{q}(t_{i+1}))^2. \end{aligned} \tag{11}$$

Moreover, milling paths that are most efficient are highly desirable. Such flank milling paths correspond to motions that are as orthogonal to the conical axis as possible

$$\begin{aligned} F_{\text{ortho}}(\mathbf{p},\mathbf{q}) &= \frac{1}{m}\sum_{i=1}^{m-1}\left\langle \frac{\mathbf{p}(t_{i+1})-\mathbf{p}(t_i)}{\|\mathbf{p}(t_{i+1})-\mathbf{p}(t_i)\|},\mathbf{d}_i\right\rangle \\ &+ \frac{1}{m}\sum_{i=1}^{m-1}\left\langle \frac{\mathbf{q}(t_{i+1})-\mathbf{q}(t_i)}{\|\mathbf{q}(t_{i+1})-\mathbf{q}(t_i)\|},\mathbf{d}_i\right\rangle. \end{aligned} \tag{12}$$

where $\mathbf{d}_i$ is the unit directional vector of $\mathbf{pq}$, see Fig. 8.

The final objective function becomes

$$\begin{aligned} F_{\text{motion}}(\mathbf{p},\mathbf{q}) &= \mu_1 F_{\text{plane}}(\mathbf{p},\mathbf{q}) + \mu_2 F_{\text{fair}}(\mathbf{p},\mathbf{q}) \\ &+ \mu_3 F_{\text{point}}(\mathbf{p},\mathbf{q}) + \mu_4 F_{\text{rigid}}(\mathbf{p},\mathbf{q}) \\ &+ \mu_5 F_{\text{ortho}}(\mathbf{p},\mathbf{q}), \end{aligned} \tag{13}$$

adding the rulings rigidity (9) as a soft-constraint. The optimization problem is solved using the Gauss-Newton method in all examples shown in the paper. If not stated differently in Section 5, the default values $\mu_1 = 1$, $\mu_2 = \mu_4 = \mu_5 = 0.1$, and $\mu_3 = 0.001$ are used.

5. Results and validation

Our algorithm is validated on two industrial benchmark models, the blades of the blisk and impeller shown in Fig. 9. Both blades are general, doubly-curved free-form objects represented as NURBS surfaces. The proposed tool path algorithm is imported into a state-of-the-art commercial CAM software for its simulation. The obtained results are compared against a default CAM algorithm. In this paper, the very finishing stage of the milling process is considered. Blade finishing operations require a special attention because they are the last operation of high added value aeronautic components such as blisk or impellers. These are expensive material components that require several foregoing manufacturing steps.

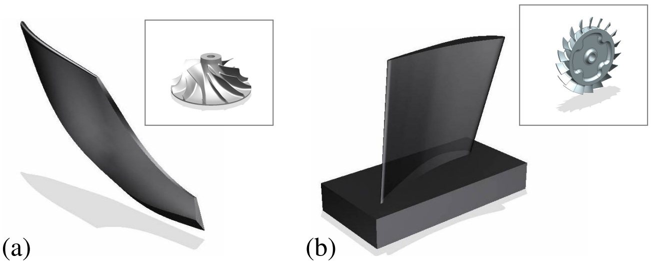

Figure 9: Milling benchmarks; free-form blades of impeller (a) and blisk (b).

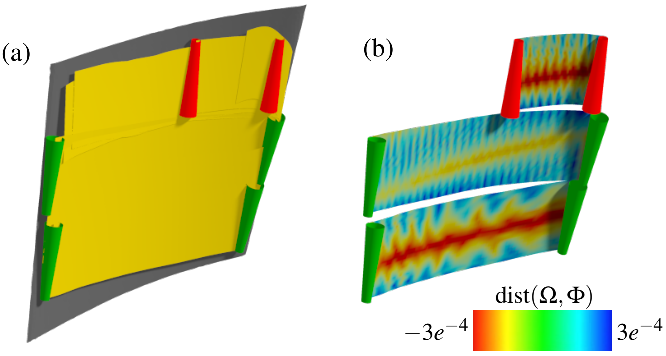

Figure 10: Surface coverage. (a) Front side of the blisk blade is covered by eight large sweeps of the conical milling tool $T_1 = (r_{min}, r_{max}, L) = (1.5, 3, 24)$ (in mm). The ruled surfaces (yellow) represent the motions of the tool axis. The initial and end positions of physically admissible (green) and gobally penetrating (red) motions are shown. (b) Three envelope approximation color-coded by the one-sided distance error dist(Ω, Φ), see Eq. (14).

The whole milling process consists of four main episodes: (i) rough material removal, (ii) roughing operations, (iii) semi-finishing operations, and finally (iv) the finishing operations. Our computer simulations and comparisons focus solely on the final finishing part.

5.1. Computer simulations

Given a conical milling tool, our algorithm returns a set of patches that approximate the input surface within a prescribed tolerance, see Fig. 10. The color coding represents the one-sided signed error ε between the envelope (Ω) and the designed surface (Φ), i.e.,

$$\varepsilon = \min_{i,j}(\text{dist}(\mathbf{r}_{ij},\Phi) - d_j), \tag{14}$$

measured over a discrete set of samples $\mathbf{r}_{ij}$ of the ruled surface R, see Fig. 7. Observe that, additionally to a highly-accurate approximation, more than 90% of the surface area is covered by only eight large conical envelopes. Observe that all these eight patches correspond to a single milling tool. Some patches, however, do not allow physical realization as the tool-holder would collide with the component base, see Fig. 9(b), and therefore the conical tool must be oriented with its tip downwards as shown in Fig. 10(a).

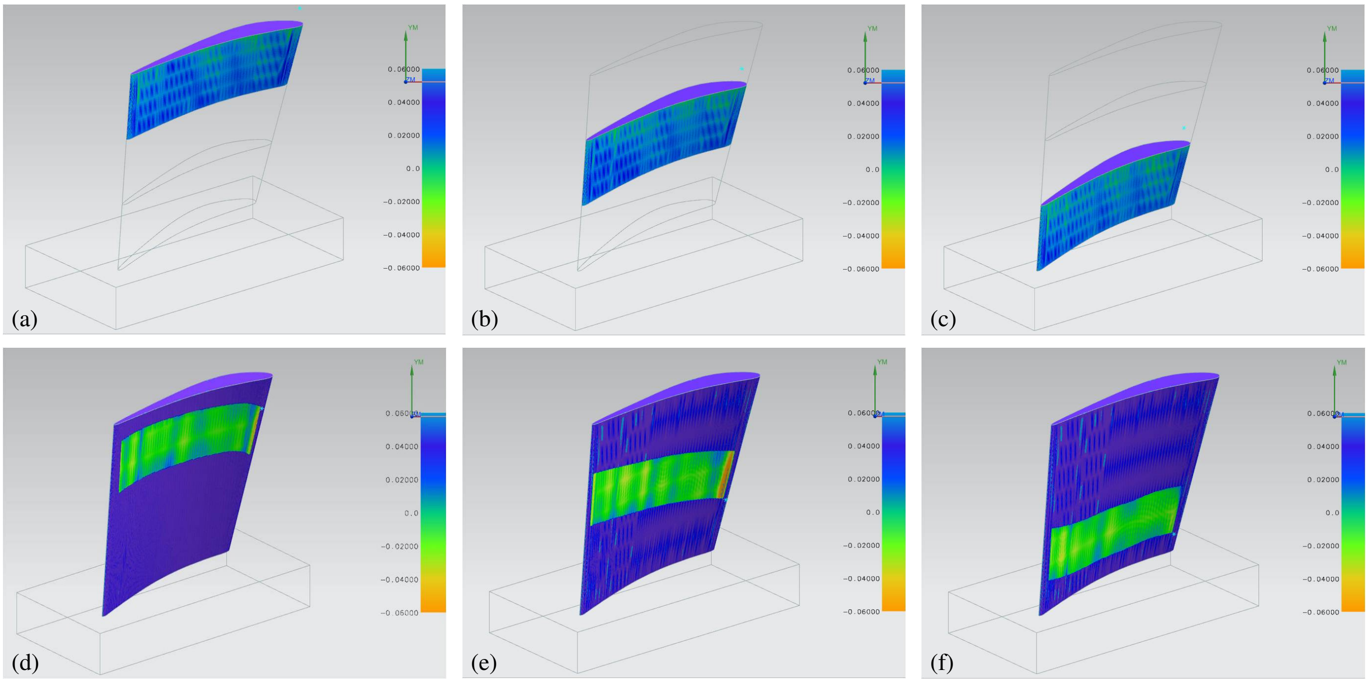

Figure 11: Approximation quality. Top: the blisk blade is segmented to three material blocks and the commercial CAM software is used to navigate the flank milling motion of the conical tool from the Fig. 12. Bottom: three largest patches detected by our algorithm in each particular region are shown. The color-coding reflects the distance error between the envelope approximation and the reference surface. The error bar range is set equally for all the patches and represents zero error (green), undercut (blue), and overcut (red).

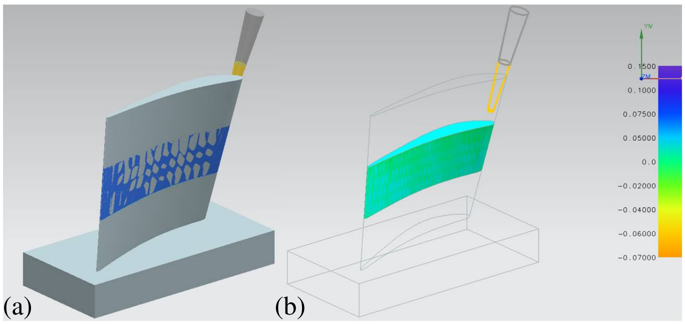

Figure 12: Blisk. (a) A blade is segmented into three material blocks (the middle one shown in blue). (b) Its manufacturing is realized by a single sweep of a conical milling tool T_1 (yellow) and the envelope is color coded by the distance error from the reference surface.

To minimize the milling time, large patches that cover a significant area of the input surface are sought after. From the set of envelope patches that meet the fine approximation quality, three large ones that share approximately the same milling direction and have minor overlap were selected, see Fig. 11. To qualitatively compare our results with the commercial software, the input surface was segmented into three material blocks, see Fig. 12, and the commercial software was used to compute three single-sweep approximations. The approximation quality is visualized using the color map in Fig. 11; the blue regions correspond to undercutting while the red ones represent overcutting.

The patches induce collision-free motions of the tool, see Fig. 11 (d-f) where the dark blue color on the complementary parts of the reference surface signals that the tool does not interfere globally the surface outside the patches. Observe that the distance error of our patches is in average by an order of magnitude smaller than the results by the commercial software. For example, the average error (computed over 30 × 100 sample points uniformly distributed in $s-$ and $t-$directions) is $\varepsilon = 0.0027$mm in Fig. 11(d), while the average error of the commercial software over the same number of samples is $\varepsilon = 0.0341$mm.

Three conical milling tools $T_1 = (1.5, 3, 24)$, $T_2 = (3, 5, 31)$, and $T_3 = (2, 5, 30)$ were tested. The first two parameters are the radii of the truncated cone at the top and bottom, respectively, and the third value is the axial length (all values are in mm). Fig. 13 shows the results when applying the tool T_1 on the concave side of the blisk blade. Observe that our approximations offer error that is significantly smaller, however, the overcutting is not fully avoided. A simulation with T_3 is shown in Fig. 14. In terms of absolute error, our envelopes still outperform the traditional software, yet some patches possess overcutting errors at the very limit of the numerical tolerance (Fig. 14(c,f)) and were therefore not used in physical experiments.

In our second type of virtual experiment, the approximation error was tested as a function of the number of passes needed to cover a certain region. While the large parts of Φ were covered by one or two patches (accomplished by one or two sweeps of the tool) using our algorithm, the commercial software required an excessive number of sweeps and yet did not achieve machining accuracy obtained by our algorithm, see Fig. 15.

The results of the second tested object, the impeller, are shown in Fig. 16. In this test, our algorithm also outperformed the commercial software in terms of approximation quality, however, unlike the blisk model, to cover a major portion (90%)

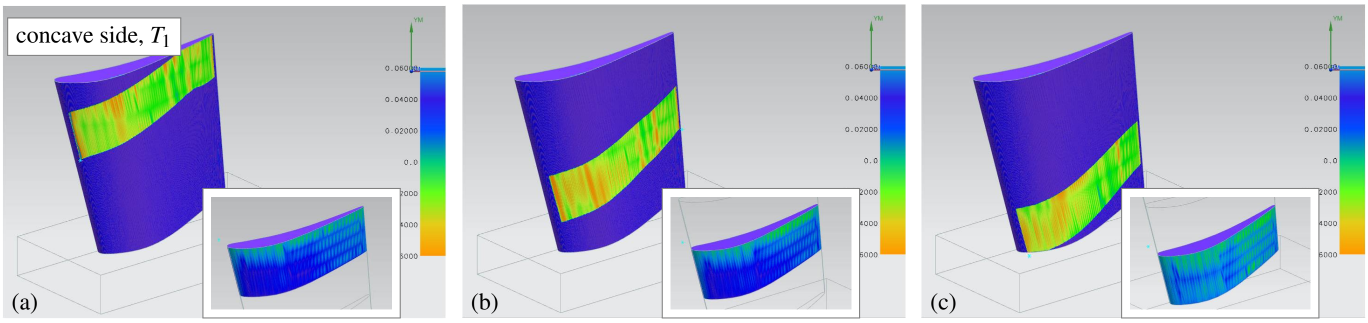

Figure 13: Blisk blade, concave side, milling tool T_1. Three large patches detected by our algorithm in the upper, middle, and lower part of the blade and the evnelopes simulated by commercial software (framed). All six envelopes are color-coded by one-sided error, see Eq. (14).

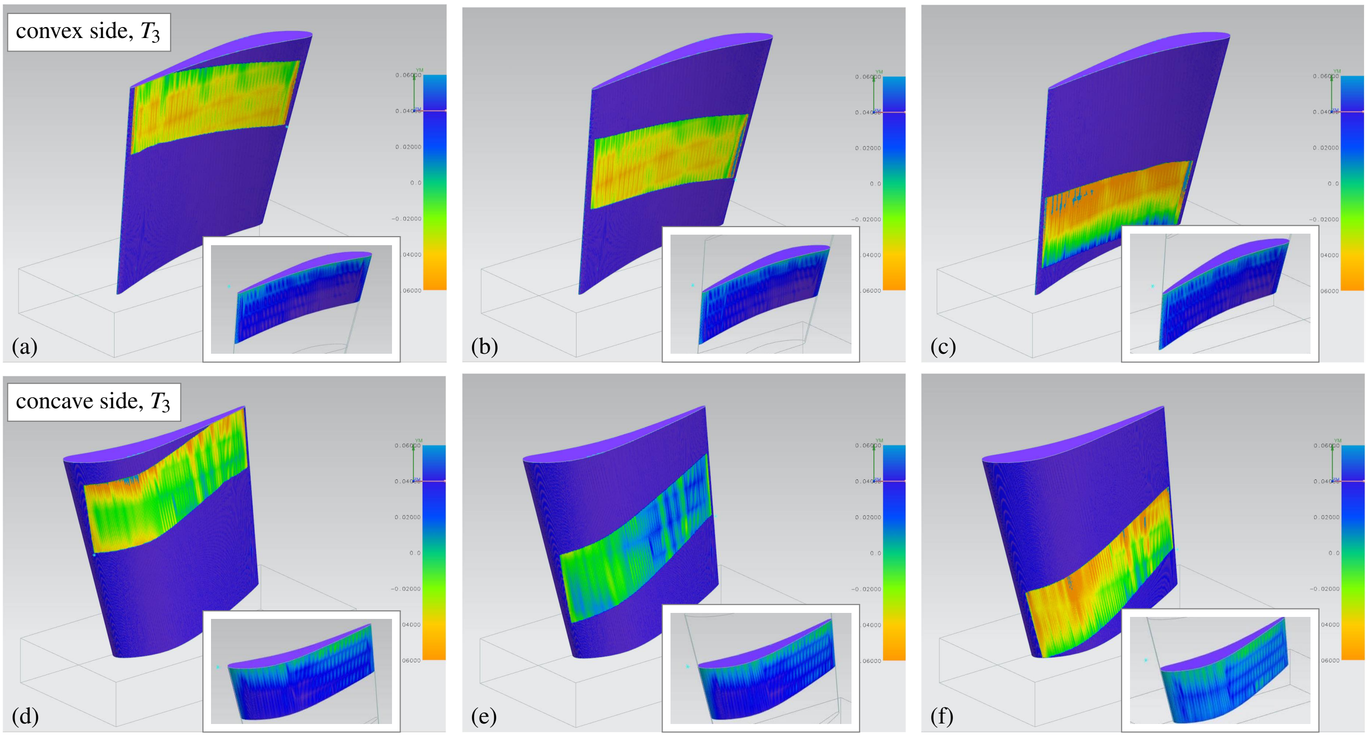

Figure 14: Blisk blade approximations with the tool $T_3 = (2,5,30)$. Six large patches detected by our algorithm, three on either side of the blade, and their counterparts simulated using the commercial software (framed) are shown.

of the surface, many small patches were needed. Fig. 16(b,c) shows two largest patches computed by our algorithm. Therefore, this workpiece was not manufactured.

5.2. *Physical experiments and validation*

Our computer simulations were validated by conducting physical experiments using KONDIA HS1000, being numerically controlled by Heidenhain iTNC530, see Fig. 18. The original material blocks are aluminium cylinders, see Fig. 17, that are pre-processed to obtain an input for the finishing flank-milling stage, accomplished by HS1000. The input for the finishing stage is shown in Fig. 17, middle.

In all the machining experiments, the parameters were set as: cutting feed $V_c = 157m/min$, feed speed $V_f = 1000/min$, and spindle speed $N = 3000rpm$. Machining time of one patch as those shown in Fig. 11 is $20-25s$, with marginal difference between our approach and the commercial software. The machining times became excessive for the commercial software in the scenario where the extreme accuracy was required, see Fig. 15. The final manufactured blades using the state-of-the-art commercial software and our algorithm are shown in Fig. 19. For physical realization, the patches modeled in Fig. 11 were used. Due to higher accuracy of our approach, the neighboring patches have minimal difference of distance error which results in perfectly smooth solution. In contrast, the lower accuracy of the commercial software results in tiny (yet visible) imperfections of the blade.

5.3. *Discussion and Limitations*

Our results approximate the input geometry within tolerances that meet the fine industrial requirements. In our examples, see e.g. Figs. 11, only the largest patches detected by our algorithm are shown. However, the output of our algorithm is a set of conical envelopes of various patch sizes and milling directions.

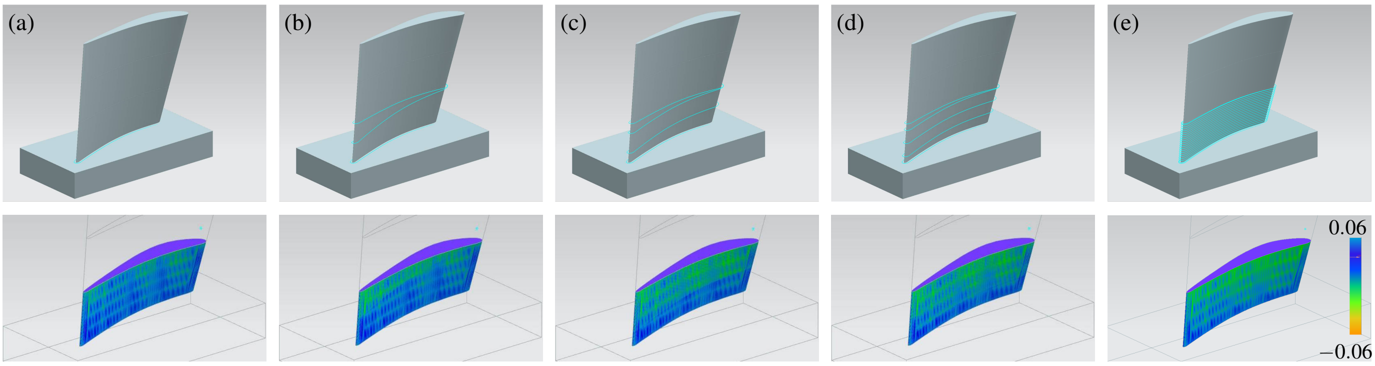

Figure 15: Improving accuracy. Top row from left to right: Increasing the number of sweeps of the milling tool from one (a) to four (d) using the commercial software (the cyan curves are the trajectories of one specific point of the tool), the approximation quality improves (bottom). The color-coding reflects the distribution of the error. (e) Using twenty milling sweeps does not improve the approximation quality considerably, see Fig. 11(f) for a comparison with our single-sweep patch.

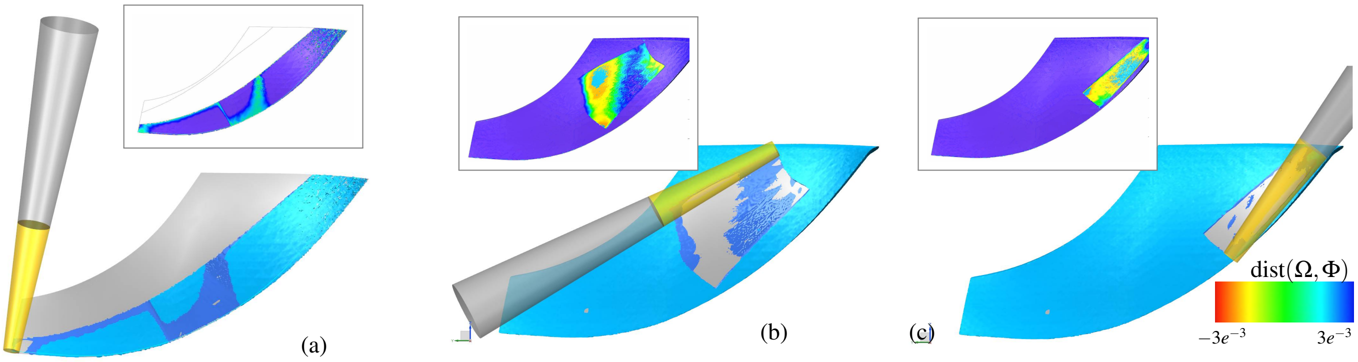

Figure 16: Impeller. (a) One sweep using default algorithm (b,c) Two best patches using our algorithm using T_2.

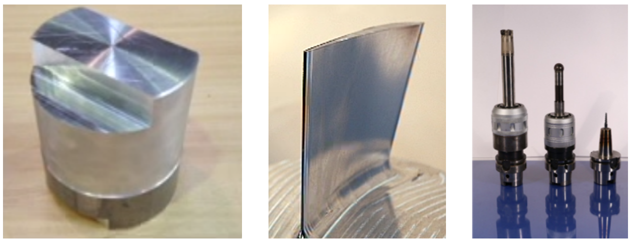

Figure 17: Real object manufacturing. Left: an input aluminimum cylinder. Middle: the blisk blade after semifinishing operations. Right: the milling tools for various stages (from left to right): roughing, semi-finishing, and finishing.

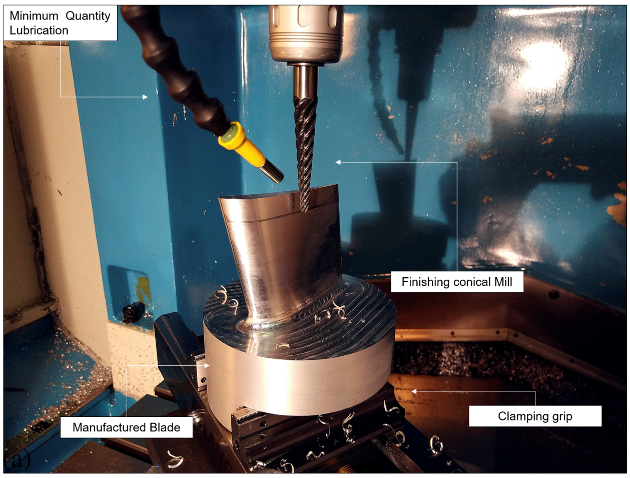

Figure 18: The interior arrangement of KODIA HS1000.

A simple greedy algorithm to select a representative subset is used. The patches are ranked according to the surface area, the largest one is selected, and all the triangles of the underlying mesh are marked as ‘visited’. Sequentially, the largest patch that covers the ‘unvisited’ area is selected, and the selection continues until the majority (90%) of the surface is marked as ‘visited’. For certain regions, our algorithm returns a set of, typically overlapping, patches. This overlapping can be considered, due to very low approximation error, as an advantage rather than a drawback as one can choose one from several milling directions for each such a region. An optimized algorithm that selects globally a subset of patches and, e.g., maximizes the coverage of the input surface, goes beyond the scope of this paper, but it is an interesting venue for future research.

A limitation is that our method does not guarantee to cover the *whole* surface. In all our experiments, the residual areas of the surface that are not covered by our patches form less than 10% of the surface and were machined using the default algorithm. To cover the whole surface, one need to complement our algorithm, e.g., by [17].

6. Conclusion and Future Work

An efficient algorithm for flank CNC machining that, for a given set of conical milling tools, computes patches that well

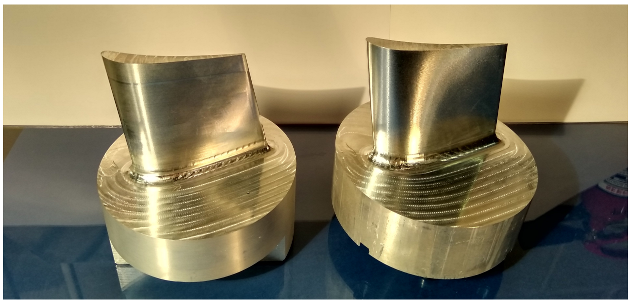

Figure 19: Comparision of the final objects using the default (left) and our algorithm (right). Additionally to higher accuracy (not visible), observe the smooth light reflection in our result, that is due to negligible distance error between the neighboring patches.

approximate the input free-form geometry is proposed. Our algorithm uses the automatic initialization introduced in [37] and computes conical envelopes that approximate the input industrial benchmark datasets within fine milling tolerances. The computer simulations were tested in real machining scenarios and the results were validated against the state-of-the-art commercial milling software. Our results outperform the commercial software in terms of the approximation error in cases where the patches are roughly of the same area. In the test cases of prescribed tolerance on the machining error, our large patches significantly reduce the total milling time as the commercial software requires tens of sweeps to cover the same area but our patches are accomplished by a single sweep.

Our current approach considers only the surface distance as the quality measure, therefore our future research points to incorporating physical machining objectives such as speed or vibrations. Moreover, testing three existing tools is a limitation as none of them may offer the optimal approximation properties and therefore one may to seek optimal, custom-shaped tools in the future research.

Acknowledgments. We thank Naiara Ortega who help with post-process measurements. The first and third authors have been supported by the Spanish Ministry of economy (MIMECOR) Implantacin de una solucin integral para la fabricacin y reparacin de componentes de turbo maquinaria (Turbo) and by DPl2016-74845-R Estrategias avanzadas de definicin de fresado en piezas rotativas integrales, con aseguramiento de requisito de fiabilidad y productividad. The second author has been partially supported by The National Natural Science Foundation of China (Grant No. 61672187) and Natural Scientific Research Innovation Foundation in Harbin Institute of Technology (HIT.NSRIF-.201711). The fourth author has been partially supported by the Basque Government through the BERC 2014-2017 program and by Spanish Ministry of Economy and Competitiveness MINECO: BCAM Severo Ochoa excellence accreditation SEV-2013-0323.

References

[1] Helmut Pottman. Geometry and new and future spatial patterns. *Architectural Design*, 79(6):60–65, 2009.

[2] Jian J Zhang and Lihua You. PDE based surface representation–vase design. *Computers & Graphics*, 26(1):89–98, 2002.

[3] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs. *Isogeometric Analysis: Toward Integration of CAD and FEA*. John Wiley & Sons, 2009.

[4] B. H. Kim and B. K. Choi. Guide surface based tool path generation in 3-axis milling: an extension of the guide plane method. *Computer-Aided Design*, 32:191–199, 2000.

[5] J. Senatore, S. Segonds, W. Rubio, and G. Dessein. Correlation between machining direction, cutter geometry and step-over distance in 3-axis milling: Application to milling by zones. *Comput. Aided Des.*, 44:1151–1160, 2012.

[6] S. A. Ivanenko, S. S. Makhanov, and M. A. Munlin. New numerical algorithms to optimize cutting operations of a five-axis milling machine. *Appl. Numer. Math.*, 49(3–4):395–413, 2004.

[7] J. Senatore, F. Monies, Y. Landon, and W. Rubio. Optimising positioning of the axis of a milling cutter on an offset surface by geometric error minimisation. *Int. J. Adv. Manuf. Technol.*, 37:861–871, 2008.

[8] C. Y. Wu. Arbitrary surface flank milling of fan, compressor, and impeller blades. *Transactions on ASME. Journal of Engineering for Gas Turbines and Power*, 117:534–539, 1995.

[9] Zhou-Long Li, Jin-Bo Niu, Xin-Zhi Wang, and Li-Min Zhu. Mechanistic modeling of five-axis machining with a general end mill considering cutter runout. *International Journal of Machine Tools and Manufacture*, 96:67–79, 2015.

[10] M. Bartoň, H. Pottmann, and J. Wallner. Detection and reconstruction of freeform sweeps. *Computer Graphics Forum*, 33(2):23–32, 2014.

[11] Robson Bruno Dutra Pereira, Lincoln Cardoso Brandão, Anderson Paulo de Paiva, João Roberto Ferreira, and J Paulo Davim. A review of helical milling process. *International Journal of Machine Tools and Manufacture*, 2017.

[12] Wei Gao, Yunbo Zhang, Devarajan Ramanujan, Karthik Ramani, Yong Chen, Christopher B Williams, Charlie CL Wang, Yung C Shin, Song Zhang, and Pablo D Zavattieri. The status, challenges, and future of additive manufacturing in engineering. *Computer-Aided Design*, 69:65–89, 2015.

[13] C. Li, S. Bedi, and S. Mann. Flank millable surface design with conical and barrel tools. *Computer- Aided Design and Applications*, 5:461–470, 2008.

[14] Pengbo Bo, Michael Bartoň, Denys Plakhotnik, and Helmut Pottmann. Towards efficient 5-axis flank cnc machining of free-form surfaces via fitting envelopes of surfaces of revolution. *Computer-Aided Design*, 79:1–11, 2016.

[15] S. Flöry and H. Pottmann. Ruled surfaces for rationalization and design in architecture. In *LIFE in:formation. On Responsive Information and Variations in Architecture*, pages 103–109, 2010.

[16] C. Li, S. Bedi, and S. Mann. Flank milling of a ruled surface with conical tools – an optimization approach. *Int. J. Adv. Manuf. Technol.*, 29:1115i–1124, 2006.

[17] G. Elber and R. Fish. 5-axis freeform surface milling using piecewise rule surface approximation. *ASME Journal of Manufacturing Science and Engineering*, 119(3):383–387, 1997.

[18] K. Sprott and B. Ravani. Cylindrical milling of ruled surfaces. *Int. J. Adv. Manuf. Technol.*, 38:649–656, 2008.

[19] H. Y. Chen and H. Pottmann. Approximation by ruled surfaces. *J. Comput. Appl. Math.*, 102:143–156, 1999.

[20] H. Gong, C. Li-Xin, and L. Jian. Improved positioning of cylindrical cutter for flank milling ruled surfaces. *Computer-Aided Design*, 37:1205–1213, 2005.

[21] J. Senatore, Y. Landon, and W. Rubio. Analytical estimation of error in flank milling of ruled surfaces. *Computer-Aided Design*, 40:595–603, 2008.

[22] CH Chu, WN Huang, and YY Hsu. Machining accuracy improvement in five-axis flank milling of ruled surfaces. *International Journal of Machine Tools and Manufacture*, 48(7):914–921, 2008.

[23] C.C.L. Wang and G. Elber. Multi-dimensional dynamic programming in ruled surface fitting. *Computer-Aided Design*, 51:39–49, 2014.

[24] H. Pottmann and J. Wallner. *Computational Line Geometry*. Springer, 2001.

[25] Yujing Sun, Jie Sun, Jianfeng Li, Weidong Li, and Bin Feng. Modeling of cutting force under the tool flank wear effect in end milling ti6al4v with solid carbide tool. *The International Journal of Advanced Manufacturing*

Technology, 69(9-12):2545–2553, 2013.

[26] A. Larue and Y. Altintas. Simulation of flank milling processes. *International Journal of Machine Tools and Manufacture*, 45(4):549–559, 2005.

[27] Kawin Sonthipermpoon, E Bohez, H Hasemann, and M Rautenberg. The vibration behavior of impeller blades in the five-axis CNC flank milling process. *The International Journal of Advanced Manufacturing Technology*, 46(9-12):1171–1177, 2010.

[28] Xing Zhang, Jun Zhang, Bo Pang, and WanHua Zhao. An accurate prediction method of cutting forces in 5-axis flank milling of sculptured surface. *International Journal of Machine Tools and Manufacture*, 104:26–36, 2016.

[29] Ke Xu, Jiarui Wang, Chih-Hsing Chu, and Kai Tang. Cutting force and machine kinematics constrained cutter location planning for five-axis flank milling of ruled surfaces. *Journal of Computational Design and Engineering*, 2017, in press.

[30] LN Lopez De Lacalle, A Lamikiz, JA Sanchez, and MA Salgado. Toolpath selection based on the minimum deflection cutting forces in the programming of complex surfaces milling. *International Journal of Machine Tools and Manufacture*, 47(2):388–400, 2007.

[31] Y.A. Lu, Q.Z. Bi, and L.M. Zhu. Five-axis flank milling of impellers: Optimal geometry of a conical tool considering stiffness and geometric constraints. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 228:1226–1236, 2014.

[32] Xavier Beudaert, Sylvain Lavernhe, and Christophe Tournier. Feedrate interpolation with axis jerk constraints on 5-axis NURBS and $g1$ tool path. *International Journal of Machine Tools and Manufacture*, 57:73–82, 2012.

[33] LN López De Lacalle, C Angulo, A Lamikiz, and JA Sanchez. Experimental and numerical investigation of the effect of spray cutting fluids in high speed milling. *Journal of Materials Processing Technology*, 172(1):11–15, 2006.

[34] Knut Sorby, Kjell Tonnessen, Jan Erik Torjusen, and Finn Ola Rasch. Improving high speed flank milling operations in multi-axis machines. *CIRP Annals-Manufacturing Technology*, 49(1):371–374, 2000.

[35] P. Y. Pechard, C. Tournier, C. Lartigue, and J. P. Lugarini. Geometrical deviations versus smoothness in 5-axis high-speed flank milling. *International Journal of Machine Tools and Manufacture*, 49:453–461, 2009.

[36] R. F. Harik, H. Gong, and A. Bernard. 5-axis flank milling: A state-of-the-art review. *Computer-Aided Design*, 45(3):796–808, 2013.

[37] Pengbo Bo, Michael Bartoň, and Helmut Pottmann. Automatic fitting of conical envelopes to free-form surfaces for flank cnc machining. *Computer-Aided Design*, 91:84–94, 2017.

[38] Helmut Pottmann and Martin Peternell. Envelopes-computational theory and applications. In *Proceedings of Spring Conference on Computer Graphics*, pages 3–23, 2000.