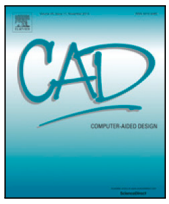
Computer-Aided Design 189 (2025) 103952

Contents lists available at ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad



On tool wear optimized motion planning for 5-axis CNC machining of free-form surfaces using toroidal cutting tools

Kinga Kruppa [a,b,*], Juan Zaragoza Chichell [a], Michal Bizzarri [d], Michael Bartoň [a,c]

[a] BCAM – Basque Center for Applied Mathematics, Alameda de Mazarredo 14, 48009 Bilbao, Basque Country, Spain
[b] Faculty of Informatics, University of Debrecen, Kassai út 26, 4028 Debrecen, Hungary
[c] Ikerbasque – Basque Foundation for Sciences, Maria Diaz de Haro 3, 48013 Bilbao, Basque Country, Spain
[d] Department of Mathematics, Faculty of Applied Sciences, University of West Bohemia, Univerzitní 8, 301 00 Plzeň, Czech Republic

ARTICLE INFO

Keywords:
5-axis CNC machining
Tool wear
Motion planning
Free-form shape manufacturing
Toroidal cutter

ABSTRACT

We propose a computational framework for motion planning for 5-axis CNC machining of free-form surfaces. Given a reference surface, a set of contact paths on it, and a shape of a toroidal cutting tool as input, the proposed algorithm designs the tool motions that are by construction locally and globally collision-free, and offers a trade-off between approximation quality and tool wear using an optimization-based framework.

The proposed algorithm first quickly constructs 2D time-tilt configuration spaces along each contact path, detecting regions that are collision-free. The configuration spaces are then merged into a single time-tilt configuration space to find a global tilt function to control the overall motion of the tool. An initial collision-free tilt function in B-spline form is first estimated and then optimized to minimize the machining error while distributing the tool wear as uniformly as possible along the entire cutting edge of the tool while staying in the collision-free region. Our algorithm is validated on both synthetic free-form surfaces and industrial benchmarks, showing that one can considerably reduce the tool wear without degrading the machining accuracy.

1. Introduction

Manufacturing of free-form surfaces is a fundamental task in many industries, such as automotive, naval, or aeronautical, to name a few. Especially in aeronautics, there are workpieces made from thermally resistant materials, such as Inconel 718 alloy, which are extremely hard-to-cut materials. Computer Numerically Controlled (CNC) machining is a dominant manufacturing technology, where the cutting tool is navigated by a program/software to move in space and carve the desired shape from a solid material block.

Tool wear is a fundamental problem of CNC machining as the worn tool is less accurate and/or the worn cutting edge may cause marks and artifacts on the to-be-manufactured workpiece. For workpieces with machining tolerance of a few tens of microns, a worn tool whose wear reaches up to hundred of microns clearly introduces a severe problem in terms of accuracy (aka dimensional deviation in engineering literature), see Fig. 1.

To avoid problems related to a damaged/worn tool, in fact its insert, one has to replace the insert which takes time and is also a non-negligible economical factor of the whole manufacturing cost. To maximize the lifespan of the tool, in an ideal scenario, the wear should be distributed equally along the whole cutting edge; however, such a uniform distribution of the tilts, in the case of a toroidal tool, decreases the approximation capabilities of the tool [1]. Moreover, equal distribution of tilts brings another challenge for the path- and motion-planning stage, namely global collision avoidance.

Our research aims at path-planning strategy for 5-axis CNC machining with toroidal cutters that is collision-free by construction, and designs motions of a given toroidal tool such that the tool wear is minimized, yet the approximation quality between the surface and the tool's motion is not degraded. We propose an optimization-based framework that optimizes both the tool wear and the approximation quality, with the priorities easily controlled using optimization weights.

The proposed work is, to the best of our knowledge, the first of the kind that in the motion planning stage distributes the wear of the tool along the cutting edge without compromising the machining accuracy. By incorporating the collision detection method of Chichell et al. [3] into our framework, the designed motions are also collision-free.

The rest of the paper is organized as follows. A related prior work is surveyed in Section 1.1, and some fundamentals on toroidal cutters and

* Corresponding author at: BCAM – Basque Center for Applied Mathematics, Alameda de Mazarredo 14, 48009 Bilbao, Basque Country, Spain.
E-mail addresses: kkruppa@bcamath.org (K. Kruppa), jzaragoza@bcamath.org (J.Z. Chichell), bizzarri@kma.zcu.cz (M. Bizzarri), mbarton@bcamath.org (M. Bartoň).

https://doi.org/10.1016/j.cad.2025.103952
Received 31 May 2025; Accepted 11 August 2025
Available online 29 August 2025

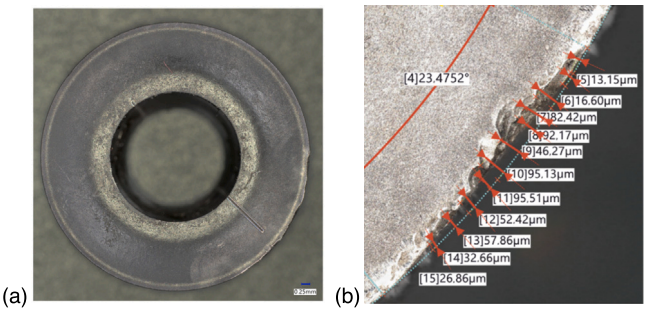


Fig. 1. Tool wear. (a) A circular insert of a toroidal cutting tool is worn in the bottom right part. (b) A magnified image of the cutting edge with tool wear reaching almost a hundred of microns. The images are courtesy of [2].

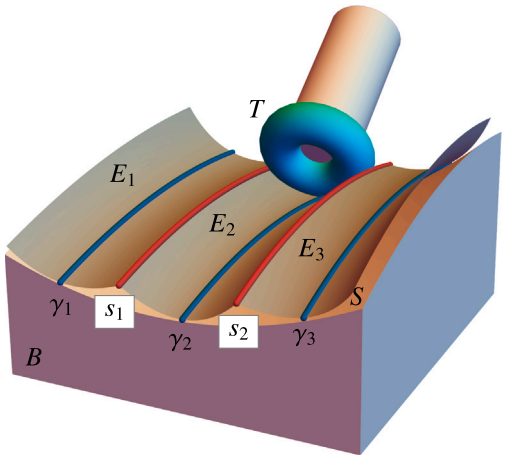


Fig. 2. Scalloping phenomenon. Three contact paths γ_i, $i = 1, 2, 3$ of a toroidal milling tool T through a material block B are shown. The tool moves along the reference surface S, and neighboring envelopes E_i of the tool's motion intersect in the *scalloping curves* s_i (red), where the milling error is typically the maximum. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

their positioning on free-form surfaces are recalled in Section 2. The main contribution, namely the motion planning algorithm is proposed in Section 3. The results are presented in Section 4 and are further discussed in Section 5. Finally, the conclusions and future research directions are drawn in Section 6.

1.1. Related work

Computer Numerically Controlled (CNC) machining methods can be categorized in several ways; the three most common classifications are: (i) The *degrees of freedom* of the milling machine: 3-, 4-, and 5-axis machines are the most frequently used [4,5]. (ii) The shape of the *milling tool*: flat-end [6,7], ball-end [8,9], toroidal (aka bull-nose) [10,11], conical or cylindrical [12,13] and general [14,15] cutting tools. (iii) The *type of contact* between the milling tool and the reference surface: single- (multi-) point milling in the case where the tool touches the surface at a single (several) point(s) [16], and *flank* milling where the tool is in contact with the target surface along a curve (see e.g. [17–20]). Our research can be classified as 5-axis point milling with toroidal tools.

5-axis single-point contact approaches offer the highest flexibility to adapt the tool to the surface [21]. The tool moves on a to-be-machined surface along a *contact path*, and the motion of the tool forms a surface, called *envelope*, which is the boundary of the volume that is removed by a single sweep of the tool. Two neighboring envelopes intersect along the so-called *scalloping curve*, and the distance between this curve and the surface de facto determines the machining error, see Fig. 2. To achieve accuracy within a few tens of microns, one requires numerous contact paths to machine an object of the size of tens of centimeters, as point milling techniques provide high precision primarily in the very neighborhood of the contact curves.

To increase the machining accuracy in the neighborhood of the contact path and consequently decrease the number of paths needed to machine the whole surface, a lot of research is devoted to *curvature matched machining*, which uses curvature analysis to compute tool orientations that are in optimized, *higher order contact* with the reference surface [1,4,22–27]. Our work belongs to this family, however, we extend these approaches towards minimized tool wear.

A subdivision-based algorithm that computes collision-free motions of a ball-end cutter along a free-form surface is proposed in [8]. The accessibility problem for ball-end cutters is converted, via the offsetting argument, to a ray-accessibility problem, and conservative normal bounding cones of non-colliding (aka accessible) tool axes are symbolically computed to construct a 4-dimensional configuration space. Such a construction avoids sampling the contact paths, which is typically expensive.

A path and motion planning algorithm for 5-axis machining of free-form surfaces using toroidal cutters is proposed in [3]. The motion planning is based on an efficient collision detection test that does not require a costly construction of the configuration space or high-resolution sampling of the cutter's axis. Instead, it uses the information of the neighboring points to efficiently prune away points of the axis that cannot cause collisions.

To make machining more efficient, multi-point milling approaches are considered [16,24]. A double-tangential positioning method for toroidal cutters on free-form surfaces is presented in [16]. To get the double-tangential configuration, one first moves the tool vertically until the first contact point is found and then tilts the tool with the first contact point fixed, until the second contact point is found. This search leads to a non-linear system of equations, and the root is computed using numerical methods. In the context of flat-end cutting, one can look even for higher order of contact between a circle and a free-form surface, namely for so-called *hyperosculating circles*, i.e., circles of the contact of order three. These circles are the limit cases of double-tangential contact when the two double-tangential points coincide [24]. The limitation is that there are at most 8 tangent directions at a given point that admit hyperosculation, and in concave regions of the surface, it leads to the internal contact of the circle with the surface, which leads to penetration.

In the engineering literature, there are works on tool wear considering physical factors such as the anti-wear coating, cutting force, or the factor cooling and/or lubricating fluids [2,28–30]. An optimization method for face milling that focuses on global collision, cutting force, the angular displacement of a rotating shaft, and angular speed is proposed in [28]. In contrast to our work, it considers only ball end cutters (which is a simpler problem from the point of view of local collisions).

A research that studies tool wear of toroidal and ball end cutters, depending on the cutting speed is proposed in [29]. It compares tools provided with anti-wear coating, namely coated carbide and boron nitride, when cutting hardened steel. In contrast to our work, the motions of the cutter are given, i.e., they are not optimized towards minimum tool wear.

The relationship between cutting force and tool flank wear of solid carbide tool during the wet end milling of the Ti6Al4V alloy is studied in [30]. The study shows that only the cutting force affects the tool wear, but also, the other way around, the worn tool requires a substantially higher cutting force.

Another relevant research that studies tool wear of toroidal cutters and their wear measurement is proposed in [31]. An effective and automated inspection method for measuring the wear of the cutting

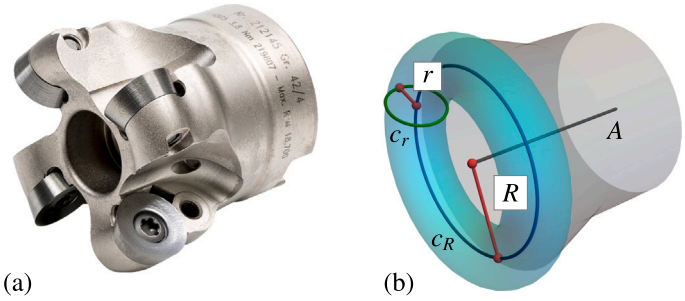


Fig. 3. Toroidal cutting tool. The real cutting tool is complicated, but under rotation about its axis it becomes a rotational solid whose boundary is a surface of revolution. Rotating a cutting circle c_r of radius r around the axis yields a torus T. The center of c_r creates a medial circle c_R of T having radius R.

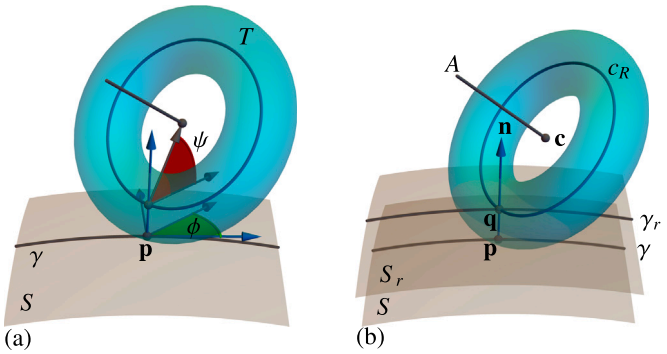


Fig. 4. (a) The position of T on S with tangent contact at a point $\mathbf{p} = \gamma(t^*)$ is uniquely given by the rotational angle ϕ (green) and the tilt angle ψ (red). (b) The motion planning of a torus T along a contact curve γ on a surface S can be translated to the motion planning of a medial circle c_R of T along an offset curve γ_r on an offset surface S_r. The contact point $\mathbf{p} = \gamma(t^*)$ of S and T corresponds to the contact point $\mathbf{q} = \gamma_r(t^*) = \mathbf{p} + \mathrm{r}\,\mathbf{n}$ of S_r and c_R, where $\mathbf{n}$ is the unit normal vector of S at $\mathbf{p}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

edge is developed, with the accuracy below 1.5 μm, which meets the industrial standards.

The closest research related to the proposed work is [2], where the tilt angle is being varied to distribute the wear uniformly along the cutting edge. In contrast to our work, however, positioning the toroidal cutter in highly accurate manner (second order line contact) is not considered nor is the global collision test as a part of the motion-planning stage.

2. Geometry of toroidal cutters

In this section, we describe toroidal cutting tools and show how such tools can be placed tangentially on a target surface. A real cutting tool is a complicated object containing cutting inserts designed to remove material from a solid block, see Fig. 3(a). However, as it rotates around its axis, it creates a rotational solid. A boundary of such a solid is a surface of revolution, which we call the tool. It consists of two main parts: The head of the tool, which is a torus T, and the shank of the tool, which is typically conceptualized as a cylinder, recall Fig. 2, although it is a bit more complicated, see Fig. 3(b). The head of the tool T is uniquely defined by two parameters: The radius r of the cutting circle c_r and the radius R of the medial circle c_R of T. The medial circle is formed by rotating the center of the cutting circle around the tool axis A. To get conservative bounds, we use a cylinder of radius $R+r$ as the shank later when building the configuration spaces.

Consider a surface $S(u,v)$, $S : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^3$ and a contact curve $\gamma(t)$, $\gamma : [t_0, t_1] \to \mathbb{R}^3$ on it. The toroidal cutter is placed to touch S along $\gamma(t)$, and at each contact point $\gamma(t^*)$, $t^* \in [t_0, t_1]$, the orientation and inclination of the tool are uniquely determined by two angles, the rotation $\phi \in [0, 2\pi]$ and the tilt angle $\psi \in [0, \frac{\pi}{2}]$, see Fig. 4(a).

At a contact point $\mathbf{p} := \gamma(t^*)$, we consider an orthonormal frame and describe the alignment of the tool with respect to it. Using the tangent vector of $\gamma(t)$ and the surface normal at $\mathbf{p}$, we consider the Darboux frame $\{\mathbf{t}, \mathbf{b}, \mathbf{n}\}$, where

$$\mathbf{t} = \frac{\gamma'(t^*)}{\|\gamma'(t^*)\|}, \quad \mathbf{n} = \frac{S_u \times S_v}{\|S_u \times S_v\|}, \quad \mathbf{b} = \mathbf{n} \times \mathbf{t},$$

and

$$S_u = \frac{\partial S(u,v)}{\partial u} \quad \text{and} \quad S_v = \frac{\partial S(u,v)}{\partial v}.$$

Planning the motion of the torus T along the contact curve γ on the surface S is equivalent to planning the motion of the medial circle c_R along the offset contact curve γ_r on the offset surface S_r, where the offset distance is equal to r, see Fig. 4(b). The center $\mathbf{c}$ of the medial circle of T is initially defined as

$$\mathbf{c} = \mathbf{q} + R\,\mathbf{t},$$

where $\mathbf{q} = \gamma_r(t^*)$, see Fig. 5(a). Such a canonical configuration puts circle c_R to the tangent plane of S_r and assumes $(\phi, \psi) = (0, 0)$.

We now apply the rotation and tilt angles. We rotate the tool around the normal vector, and the rotation angle ϕ is with respect to the tangent direction. That is, the center of the medial circle can be computed as

$$\mathbf{c}(\phi) = \mathbf{q} + R(\mathbf{t}\cos\phi + \mathbf{b}\sin\phi), \tag{1}$$

see Fig. 5(b). The tilt of the tool is defined as the angle between the plane that carries the medial circle c_R and the tangent plane of S at $\mathbf{p}$. For a given tilt angle ψ, we obtain the center as

$$\mathbf{c}(\psi) = \mathbf{q} + R(\mathbf{t}\cos\psi + \mathbf{n}\sin\psi),$$

see Fig. 5(c).

Combining the rotation and the tilt of the tool, we can construct the medial circle defined by its center $\mathbf{c}(\phi, \psi)$ as

$$\begin{aligned}\mathbf{c}(\phi,\psi) &= \mathbf{q} + R((\mathbf{t}\cos\phi + \mathbf{b}\sin\phi)\cos\psi + \mathbf{n}\sin\psi) \\ &= \mathbf{q} + R(\mathbf{t}\cos\psi\cos\phi + \mathbf{b}\cos\psi\sin\phi + \mathbf{n}\sin\psi),\end{aligned}$$

where ϕ is the corresponding rotation angle and ψ is the tilt angle, see Fig. 5(d). Analogously, one gets the tool axis vector as

$$\mathbf{a}(\phi,\psi) = -\mathbf{t}\sin\psi\cos\phi - \mathbf{b}\sin\psi\sin\phi + \mathbf{n}\cos(\psi).$$

The position of the medial circle (and therefore of the tool) is then uniquely determined by $\mathbf{c}$ and $\mathbf{a}$.

Remark 1. The surface S is considered to be given in a parametric representation to precisely construct the touching tori along a contact path $\gamma(t)$. However, we also use a fine mesh when creating the configuration spaces, as the collision detection method of Chichell et al. [3] operates faster on a discrete representation.

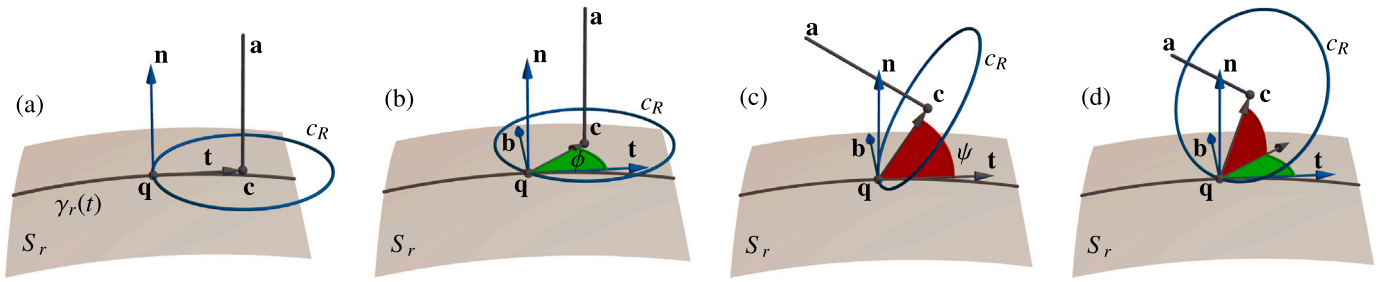


Fig. 5. Positioning the medial circle c_R on the offset surface S_r at the contact point $\mathbf{q} = \gamma_r(t^*)$. (a) Initial position of c_R on S_r. (b) Rotation of c_R around the normal at the contact point by the rotation angle ϕ (green). (c) Rotation of c_R around the **qb** line by the tilt angle ψ (red). (d) Combination of both rotations yields a general tangential position of c_R on S_r at **q**. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3. Tool motion planning

Let us suppose that we have a target surface S with $n+1$ contact paths $\Gamma = \{\gamma_0, \gamma_1, \dots, \gamma_n\}$ on it. As shown in Section 2, at a given fixed contact point, the position of the tool is fully determined by the rotation and tilt angles. Considering a whole contact path $\gamma_i(t)$, $t \in [t_0^i, t_1^i]$, to define the motion of the tool, one needs to define the tilt and the rotation functions. That is, one looks for

$$\psi_i(t) : [t_0^i, t_1^i] \to \left[0, \frac{\pi}{2}\right]$$

to define the tilt, and

$$\phi_i(t) : [t_0^i, t_1^i] \to [0, 2\pi].$$

to define the rotation of the tool along γ_i.

It is reasonable to postulate that the rotation angle is zero, recall Eq. (1). Such an approach is in accordance with the engineering praxis as it results in motions where the projection of the axis of the cutting tool to the tangent plane of S at **p** is **t**. Even though not provably optimal, it forms wide envelope strip paths when compared to the other extreme rotation, i.e., $\phi = \frac{\pi}{2}$, and reduces the number of the milling paths [1].

3.1. Tool wear measurement

Let us consider a tilt function $\psi(t) : [t_0, t_1] \to [0, \frac{\pi}{2}]$, which assigns the tilt value for each point on a given contact path $\gamma(t)$. We would like to measure how much a specific point on the insert circle is used and demonstrate how the tool would deteriorate with prolonged usage. This means that we measure how frequently the tool is tilted with specific angles during the machining along a path $\gamma(t)$, see Fig. 6. For this, we define the regions $\Psi_0, \Psi_1, \dots \Psi_m$, $m \in \mathbb{N}^+$ as non-overlapping intervals fully spanning the entire range of the tilt function $\psi(t)$. In other words, we divide the range R_ψ into smaller regions $\Psi_0, \Psi_1, \dots \Psi_m \subset R_\psi$ where

- $\Psi_i \cap \Psi_j = \emptyset, \quad i \neq j$, and $i, j = 0, 1, \dots, m$,
- $\bigcup_{i=0}^{m} \Psi_i = R_\psi$.

We measure the "duration" for which each tilt region is utilized throughout the machining process. To do this, we calculate the accumulated parametric length of the tilt function whenever it is within the range Ψ_i. One does that by projecting the intersection points of the boundary of Ψ_i (horizontal lines) with $\psi(t)$ to the t-axis, recall Fig. 6. Then, for each region Ψ_i $(i = 0, 1, \dots, m)$ one obtains a set $\mathcal{I}_i$ of intervals where

$$\mathcal{I}_i = \left\{ I \subseteq [t_0, t_1] \mid t \in I \Leftrightarrow \psi(t) \in \Psi_i \right\},$$

so that we define the tool wear for Ψ_i as

$$\omega_i = \sum_{I \in \mathcal{I}_i} |I|. \tag{2}$$

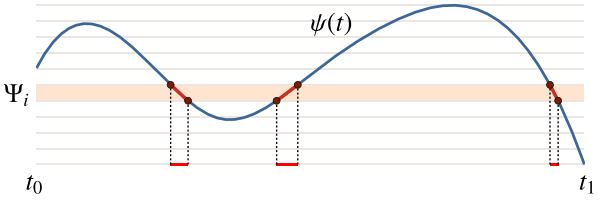


Fig. 6. Tilt range. Given a tilt function $\psi(t)$ of a contact path, the parametric intervals (red) that correspond to the engagement of the tilt from the ith range Ψ_i (yellow strip) are computed by intersecting $\psi(t)$ with the boundary of Ψ_i and projecting the intersection points to the parametric domain $[t_0, t_1]$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

This approach measures tool wear by quantifying the time spent in specific tilt regions as the total parametric length when it moves through those ranges. As a result, for any tilt function $\psi(t)$ we can obtain the tool wear values ω_i corresponding to each tilt region Ψ_i $(i = 0, 1, \dots, m)$. We postulate here that the arc length of the curve γ where the tool uses tilt value from the range Ψ_i is proportional to the tool wear ω_i, which is in accordance with the engineering praxis.[1] We also assume that the contact paths are arc-length parametrized and the tilt functions use this parametrization, i.e., the sums in Eq. (2) correspond to the arc-lengths of the contact paths where tilts from the range Ψ_i are used. We also note here that choosing m depends on how finely we would like to model the wear along the tool. When the tool is tilted at a given angle, we can determine which part of the tool will be used during the machining. Consequently, the tilt angle region Ψ_i corresponds to a specific part on the cutting edge. That is, m can be seen as a resolution parameter that controls the level of detail up to which we measure the wear across the cutting edge.

Therefore, the value of ω_i is considered as the tool wear value for the tilt region Ψ_i. Once we obtain all ω_i data $(i = 0, 1, \dots, m)$, we can visualize the distribution of the tool wear, see Fig. 7.

Further, we need to define a quality measure of the tilt function with respect to the wear. We consider that tool wear is equally distributed, i.e., the defined tilt ranges are equally used during the machining along the path $\gamma(t)$ if there is no deviation from the average of the tool wear values. Therefore, we define the measure as

$$\mathrm{MSE}_{tw} = \frac{1}{m+1} \sum_{i=0}^{m} (\omega_i - \bar{\omega})^2, \text{ where } \bar{\omega} = \frac{1}{m+1} \sum_{i=0}^{m} \omega_i, \tag{3}$$

which penalizes the deviation, in the least square sense, from the uniform distribution. An optimal solution regarding tool wear is when

[1] There are other factors, e.g., variable feed rate, that affect the relationship between the tool wear and the arc length of the contact path machined with a specific tilt angle; we assume constant feed rate in our setup.

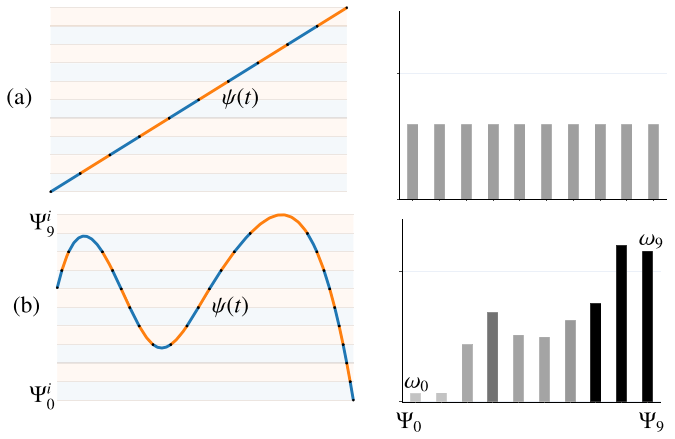


Fig. 7. Tool wear distribution. Left: a tilt function $\psi(t)$ is sliced by $m = 9$ horizontal lines to get $m+1$ tilt ranges $\Psi_0, \dots, \Psi_9$. Right: the parametric intervals for each range are summed to get the tool wear distribution ω_i. Top row shows the uniform distribution for a linear tilt function, while the bottom row shows a general tilt function.

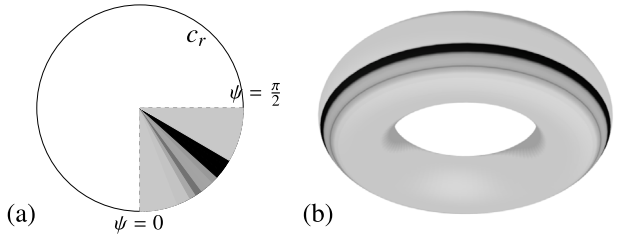


Fig. 8. Tool wear visualization for a tilt function $\psi(t)$. (a) The wedges of the meridian circle c_r, c.f. Fig. 3, are color-coded by the use of the tilt angle, with no tool wear (light) up to maximum wear (dark). (b) The same information shown on the head of the tool. See Fig. 7(b) for the tool wear distribution of $\psi(t)$.

$\psi(t) = pt + q$, $p, q \in \mathbb{R}$, $p \neq 0$, recall Fig. 7(a). In such a case, $\text{MSE}_{tw} = 0$. Additionally, it is preferable to look for maximum value of $|p|$ since a large slope of Ψ corresponds to using a wider range of the tilts.

Conversely, choosing a generic function may result in an unbalanced (non-uniform) use of specific tilt angle ranges, as demonstrated in Fig. 7(b). The worst-case scenario arises when the tilt function is constant, i.e., $\psi(t) = c$, $c \in \left[0, \frac{\pi}{2}\right]$. This means that only one region was used during the machining, so $\omega_i = t_1 - t_0$, but for all other regions $\omega_j = 0$ $(i, j = 0, 1, \dots, m, j \neq i)$.

The quality measure in Eq. (3) allows us to quantify how much a specific range of tilt angles is utilized during the machining process. We can rank the tilt function according to that measure and also use it later as a term for the optimization.

If we would like to visualize the tool wear, a straightforward solution is displaying the tool wear data in a histogram, as shown in Fig. 7. In addition to this, when a tool is tilted at a specific angle, we can accurately determine which part of the tool is in contact with the surface S. This means that we can parametrize the meridian circle in a way that corresponds with the tilt and visualize the tool wear by assigning the color values to the points on the meridian circle for each tilt range. Furthermore, since the torus is a surface of revolution formed by rotating the meridian circle around its axis, we can apply the same color scheme to the torus to clearly indicate areas of the tool that are overused. See Fig. 8 displaying the division into colors in both 2D and 3D.

3.2. Collision-free path planning

Let us now describe how we design the motion of the tool, i.e., how we create tilt functions $\psi_i(t)$ for each contact path γ_i. We have three main objectives:

(O1) design motions that are both locally and globally collision-free,
(O2) minimize the machining error,
(O3) distribute the tool wear among the feasible tilt angles as much as possible.

To achieve (O1), we adapt the method proposed by Chichell et al. [3], which performs efficient collision tests to generate a configuration space that distinguishes between colliding and collision-free cutter positions. This approach considers fixed rotation angles and employs discrete sampling of a two-dimensional (time-tilt) configuration space. This allows us to quickly determine whether the tool would be in collision at a specific position on a contact path and a specific tilt value, considering both collision of the head (torus) and a shank (conceptualized as a cylinder), recall Fig. 2.

The collision detection tests are performed along each contact path using the method of [3], resulting in collision information matrices. Each of them is a binary matrix where each cell indicates whether the time-tilt configuration is colliding (0) or collision-free (1). Specifically, the constructed information matrix M_i corresponds to path $\gamma_i : \left[t_0^i, t_1^i\right] \to \mathbb{R}^3$. At each position of γ_i we test whether the tool would be colliding if it was tilted by an angle $\psi^* \in \left[0, \frac{\pi}{2}\right]$. The method works in a discretized manner, i.e., we uniformly sample the test positions $\mathbf{p}_{i_0}, \mathbf{p}_{i_1}, \dots, \mathbf{p}_{i_{N_i}} \subset \gamma_i$ and the tilt angles $\psi_0, \psi_1, \dots, \psi_M \in \left[0, \frac{\pi}{2}\right]$ to be tested, resulting in a matrix with dimensions $(N_i+1)\times(M+1)$. If the tool positioned at $\mathbf{p}_{i_j}$ (corresponding to $\gamma_i(t_j)$, where $t_j = j(t_1^i - t_0^i)/(N_i+1)$), tilted by $\psi_k = k\pi/(2(M+1))$ collides with S, then the (j,k) element of M_i, denoted as $M_i(j,k)$, equals to 0. If there is no collision, $M_i(j,k) = 1$. To visualize the collision information matrix M_i, we color code the position-tilt pair by red color if there is a collision and green if it is non-colliding, see Fig. 9(a).

If one aims to design motions of the tool only to avoid collision, it would be sufficient to consider the collision information matrix for each path γ_i (as it was done in [3]) and select ψ_i that parses the green region, recall Fig. 9(a). However, the objective of our study is to construct the tilt function of the tool such that it is not only globally collision-free, but also results in motions that are accurate approximations of S and T has a uniform tool wear across S. Therefore, one cannot analyze the paths and their corresponding tilt functions separately, but has to handle them in a global manner.

When we obtain the collision information matrix M_i for γ_i, $i = 0, 1, \dots, n$, with dimensions $(N_i + 1) \times (M + 1)$, we merge them into a single matrix $\mathcal{M}$ with dimensions $\sum_{i=0}^{n}(N_i + 1) \times (M + 1)$, containing all collision information from all paths. Our aim is now to determine a single tilt function $\Psi(t)$ which must lie in the non-colliding region of $\mathcal{M}$, see Fig. 9(b). This function will be later optimized for the other objectives (e.g. machining accuracy).

Remark 2. For the simplicity of implementation, the global tilt function $\Psi(t)$ is represented as a cubic B-spline over a uniform clamped knot vector with single knots. Note that, at the very end of the process, $\Psi(t)$ is split again into $n + 1$ separate tilt functions to assign them to the corresponding paths. Therefore $\Psi(t)$ does not need to be even C^0 continuous at the merging points as the tilt angle at the end of one contact path can be completely different from the tilt at the beginning of another. To get a larger solution space, one could consider $\Psi(t)$ discontinuous at the merging points.

As the second objective (O2), we aim to minimize the machining error. Therefore, when constructing the merged, global collision information matrix $\mathcal{M}$, $\Psi(t)$ should have as low values as possible (being

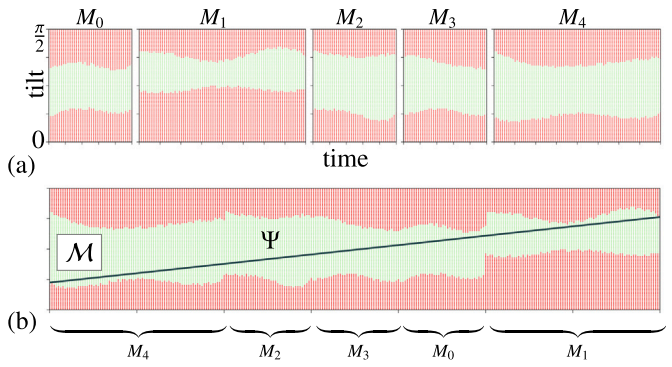


Fig. 9. Configuration space. (a) Configuration spaces M_i for $i = 0, 1, \ldots, 4$ contact paths indicating the colliding (red) and collision-free (green) time-tilt pairs. Each M_i is obtained by using the collision detection method of [3]. (b) The merged configuration space $\mathcal{M}$ for all 5 paths; We sort the configuration spaces of the particular paths in the ascending order according to the colliding (normalized) area close to the bottom (0-tilt) boundary. The initial global tilt function Ψ (black) corresponds to uniform tool wear across the whole surface. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

close to the tilt angles that provide second order line contact of the tool head with the surface [1]). This constraint may not be in accordance with the final objective, namely to uniformly distribute the tool wear (O3).

We initialize $\Psi(t)$ as a linear function with maximum possible slope. For that purpose, we sort the configuration spaces $M_0, M_1, \ldots, M_n$ of all contact paths and create a global $\mathcal{M}$. We accomplish this by computing the (normalized) area of the lower colliding regions and ordering the configuration space in ascending order, see Fig. 9(b). This sorting is a matter of choice, but we opt for it to increase the chance of having a linear initial guess of $\Psi(t)$. As a result, the domain of $\Psi(t)$ is

$$D_\Psi = \left[0, t_\Psi\right], \text{ where } \quad t_\Psi = \sum_{i=0}^{n} t_1^i - t_0^i.$$

Once we obtain all collision information, we construct a preliminary guess of $\Psi(t)$ as follows. As we have shown in Section 3.1, if $\Psi(t)$ is a linear function, it yields perfectly distributed tool wear as each point of the circular insert will be used, theoretically, only once.[2]

Therefore, we look for a linear $\Psi(t)$, that is,

$$\Psi(t) = pt + q, \quad p, q \in \mathbb{R}$$

to use it as an initial guess in the optimization framework. To find p and q such that $\Psi(t)$ is always in the non-colliding region of $\mathcal{M}$ for all $t \in D_\Psi$, we proceed with the following heuristic algorithm.

Denote by ψ_i^0 the smallest tilt angle which is collision-free in the configuration space at the ith position, i.e.,

$$\psi_i^0 = \min_j \left\{ \psi_j \mid \mathcal{M}(i, j) = 1 \right\}$$

and, analogously, denote by ψ_i^1 the largest collision-free tilt angle at the ith position.

The slope of $\psi_i(t)$ should be maximized to use a wide range of tilt values. To find the optimal value for p and q, we proceed with an iterative method as follows. As a preliminary value, we set q and then p as:

$$q = \psi_0^0 \quad \text{and} \quad p = \frac{\frac{\pi}{2} - q}{t_\psi}.$$

[2] In CNC machining practice, the situation is more complicated because of the feed rate, but here we assume a constant feed rate.

Then, we check if there exists a $t^* \in D_\psi$ value at the ith position in D_ψ, such that

$$pt^* + q < \psi_i^0.$$

If it does, we iteratively increase q as $q = q + \Delta\psi$, until the graph of the resulting $qt + p$ function is always above the bottom colliding region of $\mathcal{M}$. The second step is to find the maximum slope that does not cause collision. For this, at each position of $\mathcal{M}$ we check what is the maximum possible slope without colliding, and then choose the minimum value to assure that the graph of the function does not intersect the upper red colliding area, i.e.,

$$p = \min \left\{ \max \left\{ \frac{\psi_i^1 - q}{t_\psi} \right\} \right\}.$$

If the tilt function with slope p causes collision by intersecting the lower colliding region, we iteratively repeat the process from the first step by increasing q and finding the value of p.

Summing up, if the configuration space $\mathcal{M}$ allows it (i.e., there exists a possibility to create a linear function without collisions), our proposed algorithm finds a linear function $\Psi(t) = pt + q$ that uses the maximum possible range of tilts. In the case of a more complex configuration space $\mathcal{M}$ that does not allow such a linear Ψ, see later the discussion in Section 5.

3.3. Optimizing the tilt function

The linear function $\Psi(t)$ constructed in Section 3.2 serves as the initial guess in the optimization process. Recall that we have three objectives (O1)–(O3): locally and globally collision-free motions (O1) being the hard constraint, while machining error (O2) and tool wear (O3) are treated as soft constraints.

We represent the tilt function as a cubic B-spline function in the form of

$$\Psi(t) = \sum_{i=0}^{\dim - 1} b_i B_i^3(t), \quad t \in [0, t_\Psi],$$

where $B_i^3(t)$ are the standard B-spline basis functions of a spline space of dimension dim over a clamped knot vector on $[0, t_\Psi]$. The control coefficients b_i, $i = \{0, \ldots, \dim - 1\}$ serve as the optimization variables.

As a hard constraint, we require the motions of the tool to be always collision-free. As described in Section 3.2, we are able to determine the collision-free regions bounded by ψ_i^0 and ψ_i^1 at each position of $\mathcal{M}$. Therefore, to satisfy the first objective (O1), we require

$$\psi_i^0 \le \psi\,(i\ \Delta\psi) \le \psi_i^1, \quad i = 0, \ldots, M.$$

To achieve high accuracy (O2) and equal distribution of tool wear (O3), we prescribe the following objective function to be minimized:

$$F\left(b_0, \ldots, b_{\dim - 1}\right) = w\ F_{\text{approx}}\ + (1 - w)\ F_{\text{wear}}\ , \tag{4}$$

which consists of two quadratic terms F_{approx} and F_{wear}, where

$$F_{\text{approx}}\ = \frac{1}{M+1} \sum_{i=0}^{M} \left(\psi\left(t_i\right) - \psi_i^0\right)^2$$

measures how close we are to the lower boundary of the collision-free region. As the second order line contact is always below ψ_i^0 for all $i = 0, \ldots, M$, this also means that we aim for the highest accuracy possible while assuring collision-free motion.

The term F_{wear} will be responsible for achieving equally distributed tool wear. That is,

$$F_{\text{wear}}\ = \frac{1}{m+1} \sum_{i=0}^{m} \left(\omega_i - \bar{\omega}\right)^2,$$

where

$$\bar{\omega} = \frac{1}{m+1} \sum_{i=0}^{m} \omega_i.$$

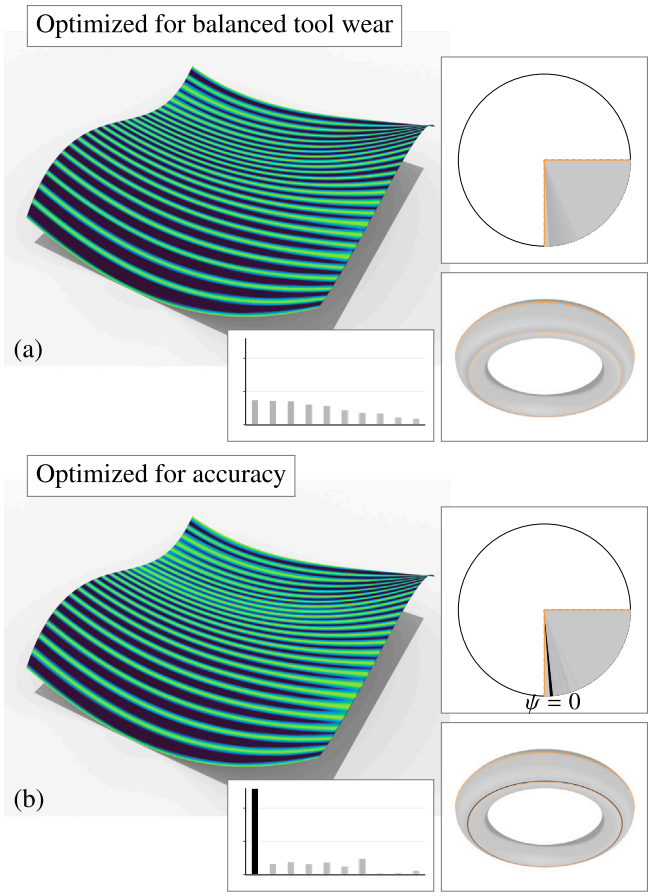


Fig. 10. Free-form surface. A free-form surface S is approximated by 25 contact paths and color-coded by the one-sided distance to the envelopes. Along the contact paths, the error is minimum (green), while along the scallop curves it is above the 50 μm threshold (blue). (a) While setting $w = 0.5$ in Eq. (4) results in smaller tool wear (right-framed), (b) by setting $w = 1$ one optimizes only the approximation quality, but wears the tool close to the bottom part (see the black narrow wedge close to $\psi = 0$). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Note that the weight $w \in [0, 1]$ in Eq. (4) is a free parameter that prioritizes the accuracy importance vs. the tool wear, see an example in Fig. 10. As one can expect, minimizing tool wear ($w = 0$) leads to reduced machining accuracy compared to focusing entirely on accuracy ($w = 1$). Our method enables a compromise between the tool wear and the machining accuracy. Later in Section 4, several examples demonstrate that choosing w in the range $[0.5, 0.8]$ yields a favorable balance and achieves almost the same accuracy as $w = 1$ while providing substantially better wear distribution. For specific error measurements on the examples, see later Table 1.

After the optimization process, we obtain the final tilt function $\Psi(t)$, which describes the motion of the tool globally in the configuration space $\mathcal{M}$. The final step of our method is to divide $\Psi(t)$ into $n + 1$ separate tilt functions and assign them back to the corresponding paths as an inverse procedure described in Section 3.2.

3.4. Error measurements

In Section 3.1, we discussed how to measure the distribution of tool wear if we machine along a contact path $\gamma(t)$ with a tilt function $\psi(t)$. Regarding our approach of optimizing tool wear for the whole machining process, we must consider the error measurements for the

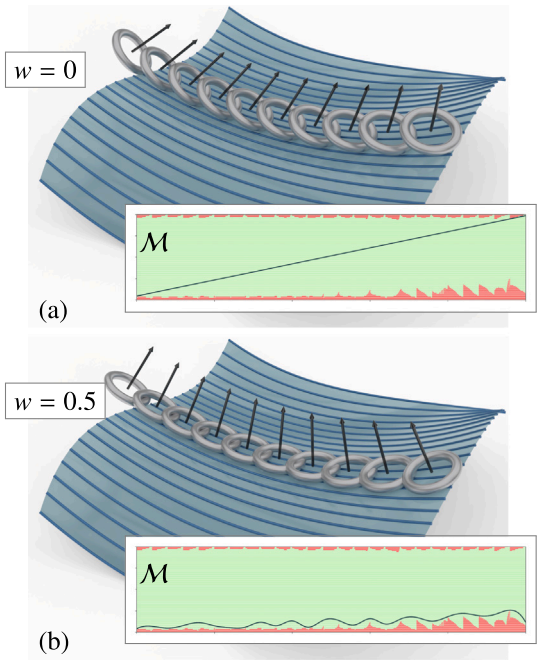


Fig. 11. Initial vs. optimized motions. (a) An initial motion of a tool along one contact path is represented by a linear tilt function (framed). (b) The motion gets optimized via Eq. (4) using $w = 0.5$ which corresponds to equal priorities of the machining accuracy and the tool wear.

global tilt function $\Psi(t)$. We use the metric MSE_{tw} introduced in Section 3.1 with respect to $\Psi(t) : D_\Psi \to \left[0, \frac{\pi}{2}\right]$. For visualizing how the tool wear accumulates and degrades the tool, we utilize a gray-to-black color scale. To determine automatically which wear values are high enough to be considered excessive usage, we gather all wear values for the given geometry across all the optimizations with distinct weights. From this data, we use $0.75\omega^*$ as the limit of our color scale, where ω^* is the maximum of all calculated wear values.

To validate the results, we need to measure the machining accuracy as well. Once we have all motions designed, i.e., the $\psi_i(t)$ tilt functions for each contact path $\gamma_i(t)$, $i = 0, 1, \ldots, n$, we then measure the distance between two surfaces: one is the ringed surface generated by the movement of the medial circle c_R of the touching torus, and the other is the offset surface S_r. Visually, we use these error measurements as color values on the surface S to signal machining accuracy.

Additionally, we use the following metric to quantify the error regarding accuracy. Based on the industrial standards, if the accuracy at a given point is less than 50 μm, we consider that it is machined accurately enough. Thus, we can use the *machined surface area* as a metric, which is the ratio between the number of points of S that are within the tolerance and the total number of points on the surface.

4. Results and experiments

In this section, we show several examples of the proposed approach to optimize the motion of the tool balancing accuracy of the approximation and the wear of the tool.

General free-form surface. We first consider an example of motions on a bicubic Bézier surface S with 25 contact paths. The bounding box of S is of size $10 \times 10 \times 3.43$ cm, and we used a toroidal tool with $R = 2$ cm and $r = 0.4$ cm. Fig. 11 shows the optimized $\Psi(t)$ function with weights $w = 0.5$ and $w = 1$, together with the visualization of the motion of the tool through one path. Recall also Fig. 10, which demonstrates the machining error and the tool wear for this geometry. Observe that when assigning the same weight to accuracy and equal distribution of

Table 1
Quantitative comparison of the optimization results across different geometries. The table presents the error of the tool wear distribution (MSE_{tw}), recall Eq. (3) and the percentage of the surface area that meets the industrial machining tolerance of 50 μm. For each model, various weight values (w, c.f. Eq. (4)) and number of paths, $n+1$, are compared. The empty lines indicate that we ran the experiment, but for the sake of space did not include the image.

Example	$n+1$	w	MSE_{tw}	% of mach. area
Bicubic Bézier patch				
Fig. 12(a), first	10	0	0.0000	10.0856%
Fig. 12(a), second	10	0.5	0.1305	19.5478%
Fig. 12(a), third	10	0.8	0.4611	21.0611%
Fig. 12(a), fourth	10	1	2.6408	23.3078%
Fig. 11(a)	25	0	0.0000	27.1078%
Fig. 10(a), Fig. 11(b)	25	0.5	1.0347	49.7222%
–	25	0.8	3.2126	51.8033%
Fig. 10(b)	25	1	12.7422	54.9878%
Fig. 12(b), first	30	0	0.0000	32.3722%
Fig. 12(b), second	30	0.5	1.4352	58.8167%
Fig. 12(b), third	30	0.8	4.2661	62.9267%
Fig. 12(b), fourth	30	1	18.0847	65.8644%
Fig. 12(c), first	60	0	0.0000	63.9167%
Fig. 12(c), second	60	0.5	5.9076	95.2200%
Fig. 12(c), third	60	0.8	12.2868	95.7811%
Fig. 12(c), fourth	60	1	22.9109	95.9767%
Bicubic Bézier patch: paths diagonal to the parametric domain				
–	28	0	0.0000	21.6611%
Fig. 13(a)	28	0.5	0.8687	50.5278%
–	28	0.8	1.7935	51.5567%
–	28	1	2.6137	52.6156%
Constant tilt: Fig. 13(b)	28	–	70.5600	27.3211%
Cavity				
Fig. 14(a), Fig. 15(a)	20	0	0.0000	40.1489%
Fig. 14(b)	20	0.5	0.1940	42.6867%
Fig. 14(c), Fig. 15(b)	20	0.8	2.0732	43.6000%
Fig. 14(d), Fig. 15(c)	20	1	10.5054	44.1622%
Industrial dataset: propeller				
Fig. 16(a)	28	0	0.0000	21.7701%
–	28	0.5	4.0129	47.7552%
Fig. 16(b)	28	0.8	6.6491	51.4433%
Fig. 16(c)	28	1	12.1544	54.8232%

tool wear ($w = 0.5$), we get an almost evenly distributed usage of the tool while the machining accuracy is only marginally deteriorated, Fig. 10(a), compared to the case when optimizing only accuracy, see Fig. 10(b).

Additionally, we ran several quantitative experiments on the same free-form surface with various weight values ($w = 0, 0.5, 0.8, 1$ in Eq. (4)) for an increasing number of contact paths ($n+1 = 10, 25, 30, 60$). Fig. 12 shows the gallery of the error heatmaps along with the corresponding histograms for the tool wear. One can observe that, as expected, optimizing towards the optimal tool wear ($w = 0$) results in much lower machining accuracy than when fully prioritizing the accuracy ($w = 1$), independent of the number of paths. However, there is a significant difference in the distribution of the wear. Our approach offers a trade-off between the tool wear and the machining accuracy and one can conclude that, e.g., for 60 contact paths, choosing $w = 0.5$ one gets almost the same accuracy while the tool wear is distributed considerably better. For the error measurements, see Table 1.

We also tested a constant-tilt approach during the whole machining process, as it is a common practice in ball-end milling, see e.g. [32]. While for ball-end milling, changing the tilt brings no gain in terms of accuracy, for toroidal cutters, it is a significant optimization variable. To get the best machining accuracy, we set the tilt function $\Psi(t)$ to the smallest value that is globally collision-free, and compared it to our approach, optimizing both the machining accuracy and the tool wear, see Fig. 13. One can see that, beyond the clear advantage of better distributed tool wear, optimized tilting offers a lot better approximation compared to a fixed tilt.

Cavity. Fig. 14 displays a more complex geometry, which would be much more challenging to machine. Here, the bounding box of the geometry is of size $24.47 \times 21.96 \times 14.05$ cm, and the radii of the tool are $R = 1$ cm and $r = 0.2$ cm. Notice that the tool can be moved without collisions when tilted only with relatively large angles, but as we approach the end of the surface, much smaller angles can also be used as the shank would not be colliding with the upper part of the cavity anymore. Observe that for real machining, one would have to consider larger shanks. We designed the shanks intentionally shorter as this leads to the formation of narrow "valleys" in the configuration space, see the collision matrices of $\mathcal{M}$ in Fig. 14, in order to test if the optimization can reach the bottom boundary of the non-colliding region when looking for highly accurate solutions ($w \cong 1$). Fig. 15 and Table 1 showcase the error measurements for machining along 20 paths. Observe that by selecting $w = 0.8$ instead of $w = 1$, one achieves a 5-fold reduction in tool wear with only a 0.56% decrease in the machined surface area, i.e., area where the machined error is below the given (industrial) threshold of 50 μm. Also, observe that on this cavity geometry one cannot use all tilts due to collisions. Yet, our algorithm distributes the tool wear among those tilts that are admissible (see the orange range in Fig. 15-bottom).

Propeller. We also evaluated our method on an industrial dataset, as seen in Fig. 17, to design the motions for machining of one blade of the propeller surface. The bounding box of the blade is $10.5 \times 12.2 \times 4.6$ cm, and the tool radii are $R = 1$ cm and $r = 0.25$ cm. Fig. 16 shows the results after the optimizations with different weights. The configuration also contains peaks that are mostly due to the global collisions with the cylindrical axis of the propeller. Table 1 shows the error measurements for the blade with 28 paths. Again, we are able to reduce tool wear without compromising machining accuracy, e.g., choosing $w = 0.8$ rather than $w = 1$ reduces tool wear by $\sim 45\%$ while only decreasing the machined surface area by $\sim 3.4\%$.

4.1. Implementation details

Our method has been implemented in Python 3.11.7 using numpy 1.26.4 and scipy 1.11.4 for numerical calculations and spline representations. Figures have been created using Wolfram Mathematica, Rhino, and in the Python environment trimesh 4.4.9 and plotly 5.9.0 were used. For the optimization processes, we used pymoo [33], an open-source optimization framework. The optimization employs pymoo's Genetic Algorithm (GA), a population-based evolutionary method. In our implementation, a population of 200 candidate solutions evolves over 150 to 500 generations, depending on the complexity of the geometry.

Regarding computation times, one can observe a clear correlation between the computational time and the complexity of the configuration space $\mathcal{M}$, that is, the more complex the configuration space $\mathcal{M}$, the slower the algorithm is. If $\mathcal{M}$ contains spikes or valleys, e.g., as the one shown in Fig. 14, it is more difficult and time-consuming to achieve our objectives (especially regarding the machining accuracy). Furthermore, due to the innate properties of the genetic algorithm, the evaluation of the objective function should be as fast as possible since using a large population size and a high number of generations leads to evaluating F tens of thousands of times. Thus, reducing the calculation times to evaluate the objective function is essential. Consequently, the total run time of our algorithm depends on the combination of the three following factors:

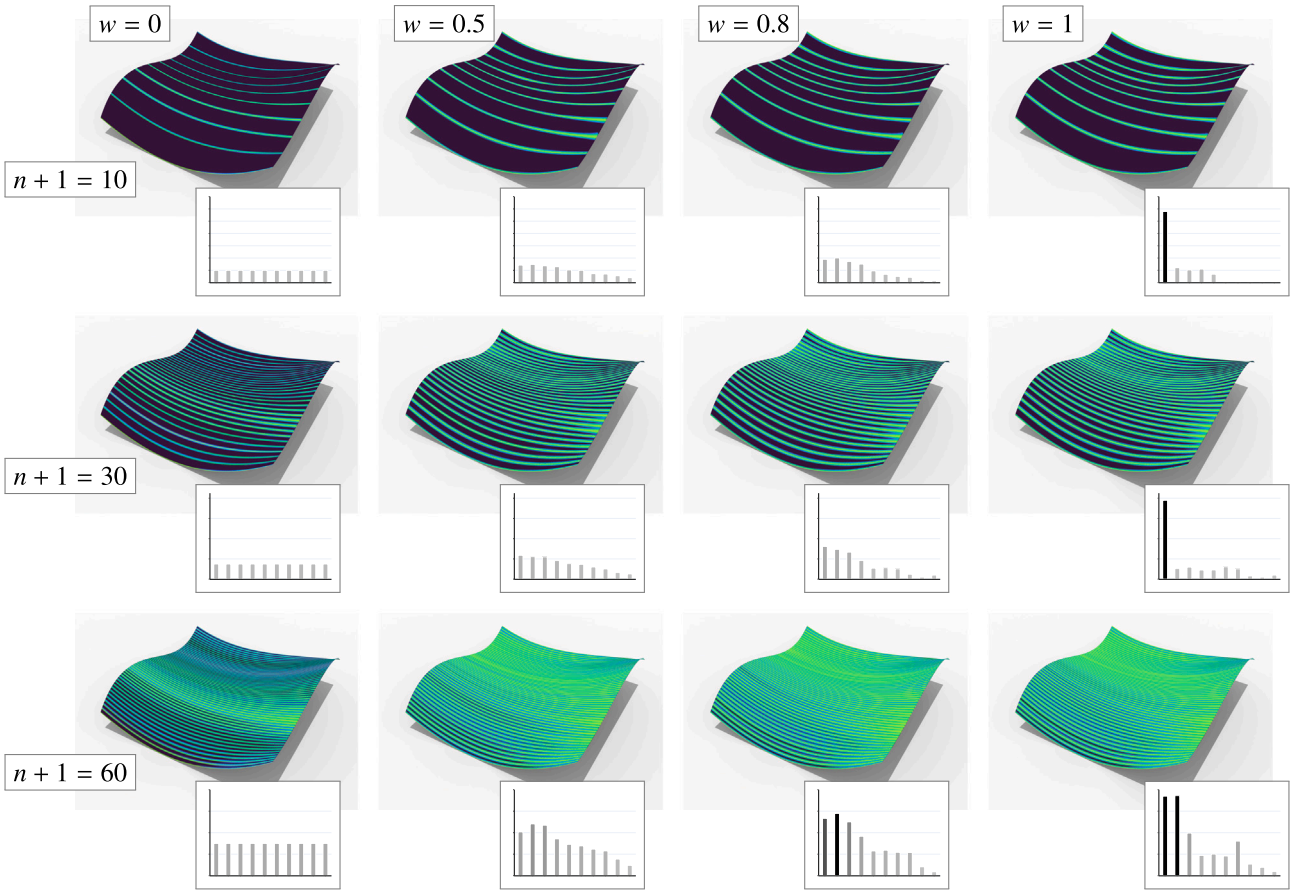


Fig. 12. A gallery of the error maps and the tool wear histograms (framed) depending on the number of contact paths $n+1$ (rows) and weights w (columns), recall Eq. (4). The green color shows a perfect machining error close to zero, while dark blue represents undercut over 50 microns. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

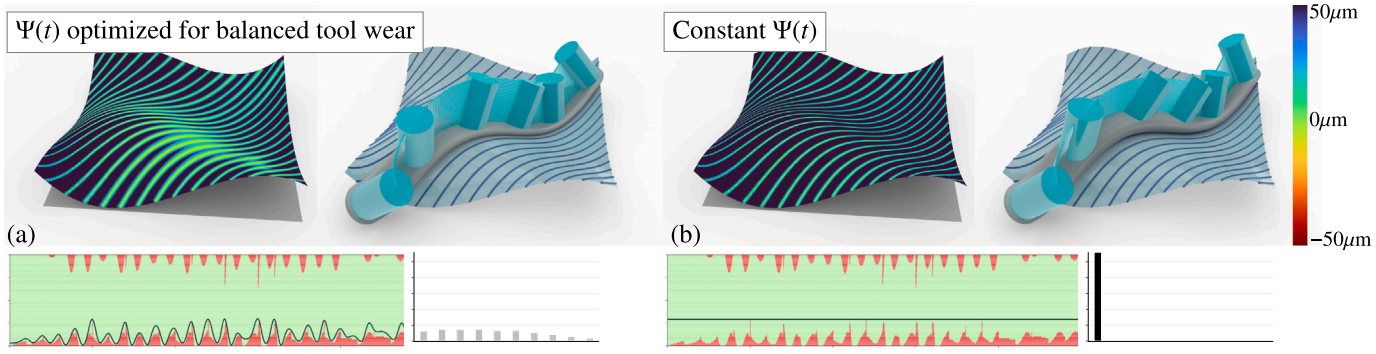


Fig. 13. Comparison of optimizing the tilt vs. applying constant tilt angle for a free-form surface with 28 diagonal-like contact paths. Top: Heatmaps demonstrating the machining accuracy and the designed motion displayed along one path. Bottom: The configuration spaces with the optimized Ψ functions and the distribution of the tool wear. (a) Optimizing for tool wear ($w = 0.5$ in Eq. (4)) results both in high accuracy and well-balanced usage of the tool, while (b) machining with a constant tilt angle is the worst case scenario regarding the tool wear distribution.

The number of variables of the optimization. The variables of optimization in the objective function F in (4) are $b_0, \ldots, b_{\dim-1}$, which are the control coefficients of the $\Psi(t)$ cubic B-spline function. With a higher number of coefficients, $\Psi(t)$ has more degrees of freedom and can get closer to the colliding region of the configuration space (leading to higher accuracy), yet it can oscillate to achieve balanced tool wear. It is especially important when the configuration spaces are more complex, with valleys and spikes, recall Fig. 14. However, the more variables we have in the optimization, the slower the method is. Furthermore, overfitting can also happen, which is also not favorable. As a general rule of thumb, we used 4 control coefficients for each M_i configuration space, resulting in $3n+4$ variables in the optimization for $n+1$ paths. However, for more complex configuration spaces (e.g., see Fig. 14), we used 10 coefficients per each M_i, as with many thin valleys one requires more degrees of freedom to control the shape of the tilt function. As an example, Fig. 18(a) shows that it is not sufficient to use only 4 control coefficients in the case of the cavity model.

Settings of the genetic algorithm. The running time is fundamentally determined by the intrinsic parameters of the genetic algorithm, that is, the selection of population size and number of generations. This

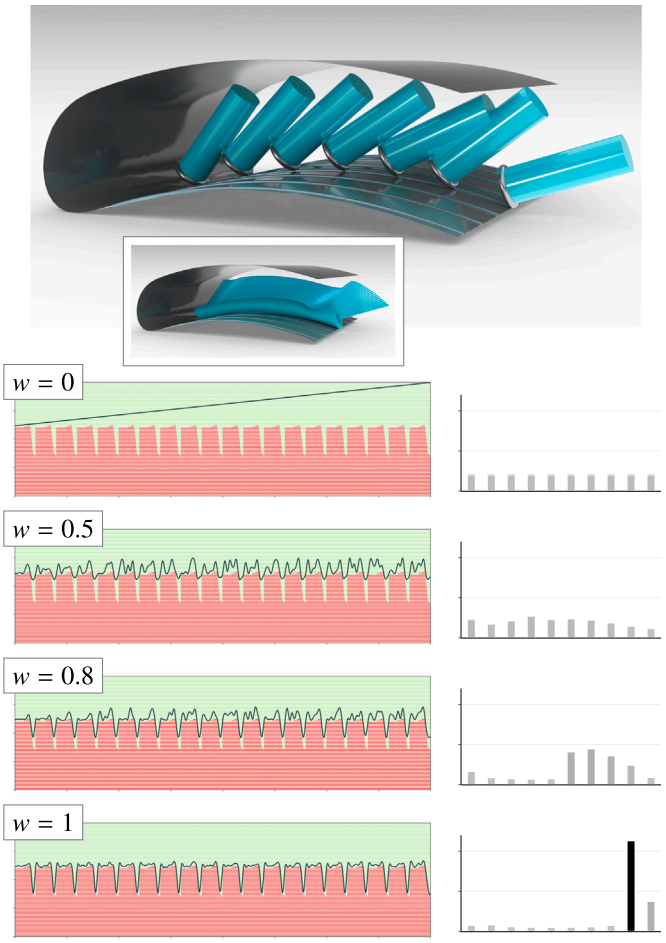


Fig. 14. Cavity. Top: A toroidal tool with a cylindrical shank moves inside a narrow cavity. Bottom: The configuration space for 20 contact paths inside the cavity is built, and global tilt functions Ψ are computed for various values of w, c.f. Eq. (4). Right: the histograms show the distribution of the tool wear.

essentially determines how many iterations will be used to explore variations of the optimization variables. While increasing these parameters may yield better results, they dramatically impact the computational time required. Moreover, increasing the number of generations beyond a certain threshold does not give significantly better results, and there are only minor improvements in further solutions. In all examples, we used a population of 200 candidate solutions, and when determining the number of generations, we tried to achieve a balance between the computation time and the resulting accuracy. A relatively low number of generations is adequate for simpler configuration spaces (in the case of Fig. 12, we used 150); however, this may not be sufficient when dealing with more complex collision maps as Fig. 18(b) shows. In such cases we must increase the number of generations (for Figs. 13, 14 and 16 we used 500) to achieve our objective of machining accuracy. The settings we used and the running times are listed in Table 2.

Computation times for the accuracy and tool wear. When evaluating F, the most time-demanding operation is calculating F_{wear}, specifically, determining the intersection points of the graph of the function $\Psi(t)$ with the boundaries of the tilt regions (see Fig. 6). To have a generic code (and support higher degree B-splines), we use a dichotomy algorithm to compute the intersection points. In the cubic and quartic case, one could eventually use the Cardano's formula to get analytic expression and to speed up the computation of the intersection points. Furthermore, choosing the number of regions when we divide the range of the tilt function also affects the running time. When $\Psi(t)$ is analyzed for tool wear, we divide its range into $m + 1$ non-overlapping regions and calculate all intersection points with all the regions. Unless stated differently, we set $m = 9$ in our examples.

Table 2
Running times for several example geometries. For each model we measured the computation time of the optimization. In the experiments, we used various models with $n + 1$ contact paths, and the tilt function $\Psi(t)$ was created by having dim control coefficients. During the optimization, the weight w was used in Eq. (4). The next two columns show the population size and number of generations employed in the genetic algorithm. The last column displays the running time in seconds.

Example	n+1	w	dim	pop. size	# of gen.	running time
Bicubic Bézier patch						
Fig. 12 (a)	10	0.5	31	200	150	318
Fig. 12 (b)	30	0.5	91	200	150	610
Fig. 12 (c)	60	0.5	181	200	150	1273
Bicubic Bézier patch: paths diagonal to the parametric domain						
Fig. 13(a)	28	0.5	85	200	500	3141
Cavity						
Fig. 18(a)	20	0.8	61	200	500	3198
Fig. 18(b)	20	0.8	181	200	250	1305
Fig. 14(c)	20	0.8	181	200	500	3880
Industrial dataset: propeller						
Fig. 16(b)	28	0.8	85	200	500	2917

The numerical experiments were run on an Apple MacBook Pro notebook with M1 Pro processor of 10 cores and 16 GB of RAM. The execution times for the optimization process ranged from several minutes up to an hour (the cavity being the most demanding one, see Fig. 14). For details, see Table 2.

5. Discussion & limitations

Ball-end cutters. One could eventually generalize the current approach towards ball-end cutters, however, varying the tilt would have no effect on the approximation quality and therefore we pursued the more challenging variant with this research thread.

Tool-path design & optimization. In the current setup, the contact paths are given as an input. For toroidal cutters, however, it was shown in [1] that one can achieve higher accuracy when optimizing the contact paths as well. Improving the machining accuracy and tool wear via optimizing the contact paths seems like a challenging task that one can tackle in the future.

Non-circular toroidal cutters. In the current work, we considered only toroidal cutters with circular inserts as they are the most common toroidal tools. One could consider toroidal cutters with curvature-varying inserts, however, for this type of cutters, the construction of the configuration space would be a lot more complex as one could not use the offsetting argument and test the collisions by simply sampling the medial circle and computing the footpoints on the offset surface.

Tilt function optimization. The tilt function is represented as a cubic B-spline, and we optimize the coefficients, which leads to a linear constraint in the approximation term. One could eventually consider optimizing also the knot vector, however, this would lead to non-linear formulation, which would consequently lead to a slower optimization. We showed on the cavity example, which contained many "bumps" in the configuration space, that one has enough degrees of freedom to well approximate the bottom boundary using sufficiently many uniformly distributed knots.

Configuration spaces of more complicated topology. In all our examples, the configuration space contained, at each time instant t_i, only one interval $[\psi_i^0, \psi_i^1]$ that bounded $\psi(t_i)$ and represented the non-colliding region, which is in accordance with the CNC praxis. One could, however, have a more complicated configuration space that could contain, e.g., isolated closed subsets ("islands"), see Fig. 19, and would need

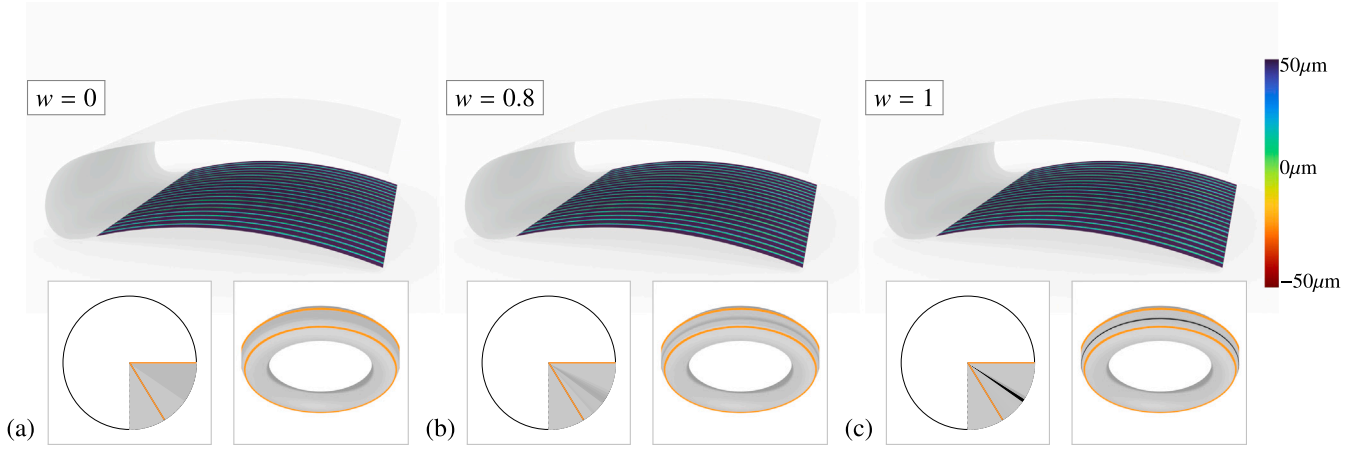


Fig. 15. Machining error vs. tool wear. The bottom side of the cavity geometry, c.f. Fig. 14, is approximated by $n + 1 = 20$ motions of the tool and color-coded by the machining error for various values of w. The bottom images show the tool wear on the circular insert and the whole torus inside the range of admissible tilt angles (orange). Observe a considerable tool wear for $w = 1$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

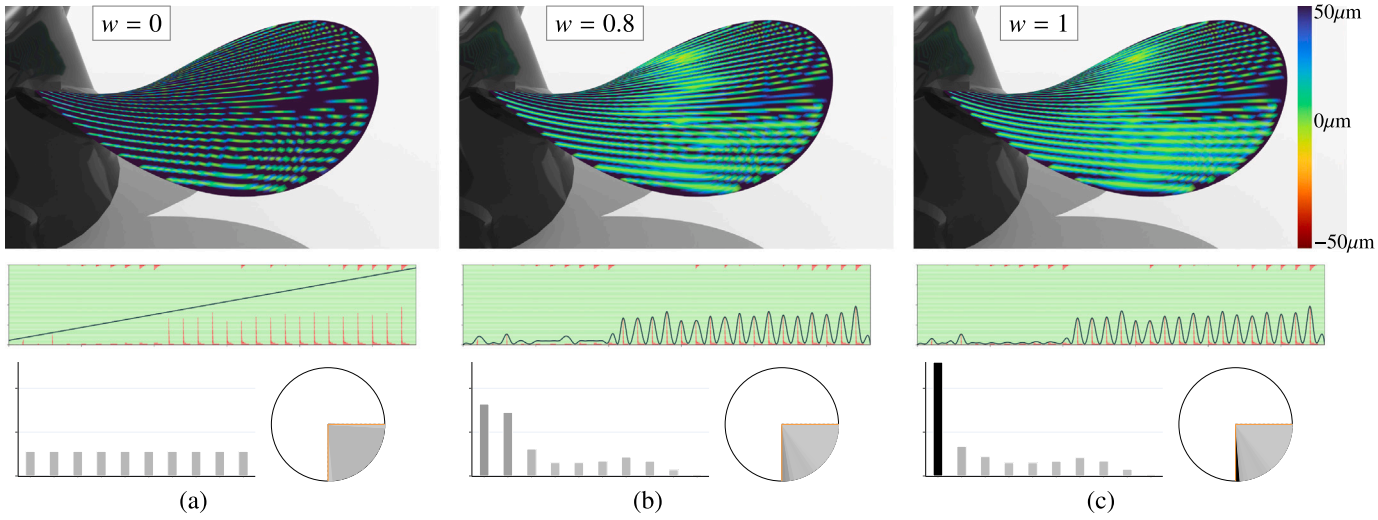


Fig. 16. Propeller. Top: Heatmaps representing the machining error on a blade of the propeller for various weights w are shown. Middle: Configuration spaces and the optimized tilt function Ψ. Bottom: The histograms representing the tool wear and its distribution on the circular insert. (a) While the machining error is mostly above the 50 μm threshold, optimizing Ψ towards accuracy reduces the error considerably (b,c). (c) Optimizing only the machining error ($w = 1$), however, causes considerable tool wear in the bottom part of the tool (black) when compared with balanced accuracy-wear counterpart (b).

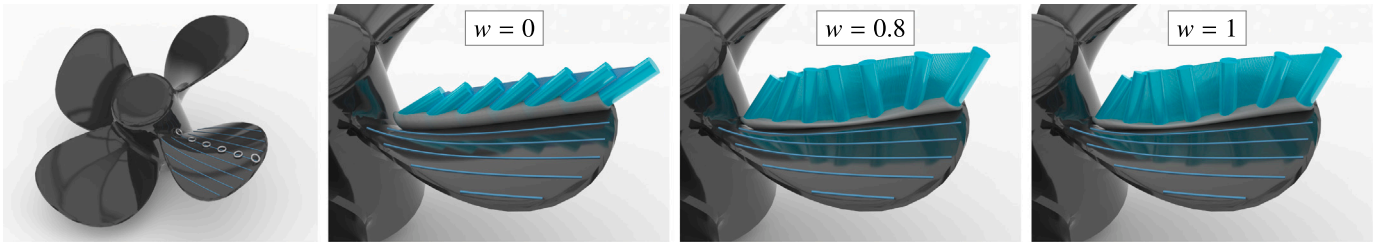


Fig. 17. Optimized motions. On the propeller geometry (left), we show three optimized motions of the toroidal tool along the same contact path. The optimized tilt functions correspond to those shown in Fig. 16 for $w = 0$, $w = 0.8$, and $w = 1$, respectively.

to navigate the tilt function among them. In such a case one could initialize $\Psi(t)$ by the middle values of the lowest interval, if one aims to stay close to the bottom boundary of the configuration space.

Smooth vs. discrete representation of the configuration space. In our framework, we sample the time positions and tilt angles to compute the configuration space in a discrete fashion. One could possibly represent the configuration space, e.g., using the approach [34] to avoid sampling. However, such an approach returns only conservative bounds, and therefore one could loose parts of the configuration space which could offer important non-colliding configurations. Since the most accurate positions occur close to the boundary of the configuration space (second order line contact), we believe that such an approach would not detect highly accurate solutions, which was one of our objectives.

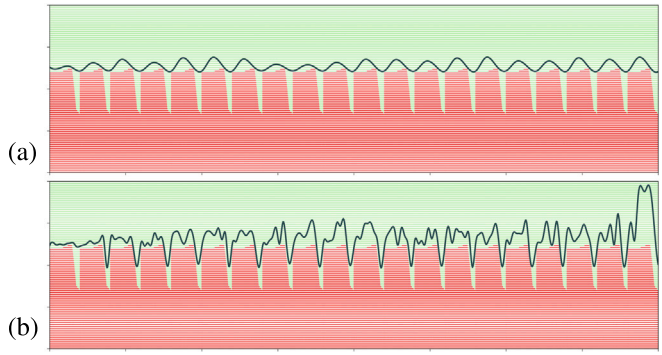


Fig. 18. Experiments with the optimization settings using the cavity model when $w = 0.8$. (a) If 4 control coefficients are used for each M_i, the graph of $\Psi(t)$ does not have the degree of freedom to closely approximate the valleys of the non-colliding regions. (b) By increasing the number of coefficients to 10, but the number of generations in the genetic algorithm is only 250, the result is still not satisfactory. When increasing both of these settings, the number of coefficients to 10 and the number of generations to 500, the optimization results in the spline in Fig. 14.

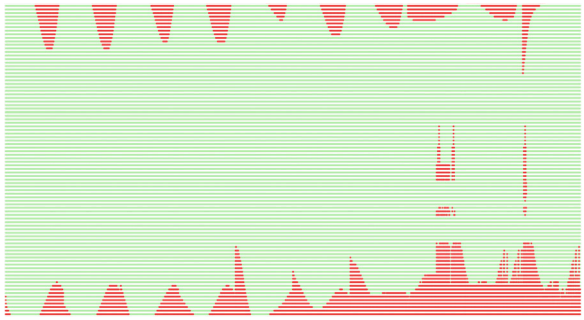


Fig. 19. A synthetic configuration space containing separated "islands" of colliding regions. Constructing $\Psi(t)$ would be more challenging as the initial (linear) guess is not possible.

Optimization of the rotation angle. Similarly to [1], we considered the rotation angle to be perpendicular to the tangent direction of the contact path and optimized only the tilt angle. Even though it would be possible to optimize both the rotation and the tilt angles, such an approach would lead to volumetric (3D) configuration spaces, which would be computationally more expensive, and it is not expected that optimizing the rotation would considerably improve the approximation quality. Clearly, considering rotations does not affect the tool wear term at all.

Computational speedup & alternative optimization. In our framework, while we aimed to reduce the running time through efficient implementation, we did not strive for the absolute best execution times nor optimized the code for maximum speed. Potential improvements include parallelizing the calculation of the intersection points on the graph of $\Psi(t)$. One might also approach the optimization problem with an adaptive population method, where the number of populations is not constant in the different generations but dynamically adapted and modified.

Tilt-wear dependence. In this work, we postulated that the time spent using a certain cutting point on the cutting edge (tilt angle) correlates linearly with the tool wear. In real-life machining, this relation may not need to be linear, that is, the already-worn parts of the cutting edge could wear even faster than those that are worn only a little or not at all yet. Moreover, there are many other factors that influence this relationship in real life machining (e.g., spinning speed vs. feed rate, cutting forces, surface quality from roughing stage, cutter-workpiece engagement, etc.); therefore modeling the wear is difficult [2,31]. We aim to validate the proposed formulation using physical experiments in a future project.

6. Conclusions and future work

We have introduced a new motion-planning algorithm for 5-axis point CNC machining of free-form surfaces using toroidal cutting tools. Given an input free-form surface, a set of contact paths on it, and a toroidal cutter, the proposed algorithm first quickly builds the configuration space, detecting tilt angle intervals that are both locally and globally collision-free. The tilt function that controls the motion of the tool along all the contact paths is designed and optimized towards three major objectives: minimal tool wear and maximum machining accuracy while staying in the collision-free part of the configuration space. Our algorithm has been validated on both synthetic free-form surfaces and industrial benchmarks, showing that one can considerably reduce the tool wear without deteriorating the machining accuracy.

As a future work, we aim for physical validations using metal cutting tools and also to generalize the algorithm for toroidal cutters with non-circular inserts. Such a generalization is a challenging task, especially for fast construction of the configuration space as there one cannot use the offsetting argument and test the collision by simply measuring the point-surface signed distance.

CRediT authorship contribution statement

Kinga Kruppa: Conceptualization, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Juan Zaragoza Chichell:** Software, Validation, Visualization, Writing – review & editing. **Michal Bizzarri:** Conceptualization, Investigation, Methodology, Validation, Visualization, Writing – original draft, Writing – review & editing. **Michael Bartoň:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Basque Government via the AURRERA project (Elkartek KK-2024/00024), by the Spanish Ministry of Science, Innovation and Universities, grant No PID2023-146640NB-I00 funded by MICIU/AEI/10.13039/501100011033 and co-funded by the European Union, and by BCAM "Severo Ochoa" accreditation CEX2021-001142-S.

M. Bartoň was supported by RYC-2017-22649 funded by MICIU/AEI/10.13039/501100011033 and EI ESF "ESF Investing in your future".

M. Bizzarri was supported by the project R&D of Technologies for Advanced Digitization in the Pilsen Metropolitan Area (DigiTech) No.: CZ.02.01.01/00/23_021/0008436 co-financed by the European Union.

Data availability

Data will be made available on request.

References

[1] Bartoň M, Bizzarri M, Rist F, Sliusarenko O, Pottmann H. Geometry and tool motion planning for curvature adapted CNC machining. ACM Trans Graph 2021;40(4):1–16.

[2] Gdula M, Knapčíková L, Husár J, Vandžura R. Modeling and measurement of tool wear during angular positioning of a round cutting insert of a toroidal milling tool for multi-axis milling. Appl Sci 2024;14(22):10405.

[3] Chichell JZ, Rečková A, Bizzarri M, Bartoň M. Collision-free tool motion planning for 5-axis CNC machining with toroidal cutters. Computer-Aided Des 2024;173:103725.

[4] Jensen CG, Red WE, Ernst C. Machining free-form surface cavities using a combination of traditional and non-traditional multi-axis machining methods. Computer-Aided Des Appl 2008;5:241–53.

[5] Zhang Z, Shi Z, Zhong F, Zhang K, Zhang W, Guo J, et al. Continuous tool-path optimization for simultaneous four-axis subtractive manufacturing. Comput Graph Forum 2025;44(1):e15204.

[6] Li SX, Jerard RB. 5-axis machining of sculptured surfaces with a flat-end cutter. Computer-Aided Des 1994;26(3):165–78.

[7] Fan J, Ball A. Flat-end cutter orientation on a quadric in five-axis machining. Computer-Aided Des 2014;53:126–38.

[8] Ezair B, Elber G. Automatic generation of globally assured collision free orientations for 5-axis ball-end tool-paths. Computer-Aided Des 2018;102:171–81.

[9] Dutta N, Zhang T, Fang G, Yigit IE, Wang CC. Vector field-based volume peeling for multi-axis machining. J Comput Inf Sci Eng 2024;24(5).

[10] Campa F, de Lacalle LL, Lamikiz A, Sanchez J. Selection of cutting conditions for a stable milling of flexible parts with bull-nose end mills. J Mater Process Technol 2007;191(1):279–82.

[11] Roth D, Bedi S, Ismail F, Mann S. Surface swept by a toroidal cutter during 5-axis machining. Computer-Aided Des 2001;33(1):57–63.

[12] He D, Li Z, Li Y, Tang K. Quasi-developable and signed multi-strip approximation of a freeform surface mesh for efficient flank milling. Computer-Aided Des 2021;140:103083.

[13] Bo P, Bartoň M, Pottmann H. Automatic fitting of conical envelopes to free-form surfaces for flank CNC machining. Computer-Aided Des 2017;91:84–94.

[14] Engin S, Altintas Y. Mechanics and dynamics of general milling cutters.: Part I: helical end mills. Int J Mach Tools Manuf 2001;41(15):2195–212.

[15] Machchhar J, Plakhotnik D, Elber G. Precise algebraic-based swept volumes for arbitrary free-form shaped tools towards multi-axis CNC machining verification. Computer-Aided Des 2017;90:48–58.

[16] Singh M, Qu J, Duvedi RK, Bedi S, Mann S. A new approach to find gouge free tool positions for a toroidal cutter for Bézier surfaces in five-axis machining. Int J Adv Manuf Techn. 2021;117:3053–62.

[17] Calleja A, Bo P, González H, Bartoň M, López de Lacalle LN. Highly accurate 5-axis flank CNC machining with conical tools. Int J Adv Manuf Technol 2018;1–11.

[18] Rajain K, Gómez Escudero G, Bizzarri M, González Barrio H, Calleja Ochoa A, López de Lacalle LN, et al. High-quality smooth finishing of blade-like geometries via G^1 multi-pass 5-axis flank CNC machining using conical cutting tools. Int J Adv Manuf Technol 2025;1–15.

[19] Harik RF, Gong H, Bernard A. 5-axis flank milling: A state-of-the-art review. Comp-Aided Des 2013;45(3):796–808.

[20] Zhu L, Zheng G, Ding H, Xiong Y. Global optimization of tool path for five-axis flank milling with a conical cutter. Computer-Aided Des 2010;42(10):903–10.

[21] Warkentin A, Ismail F, S. B. Comparison between multi-point and other 5-axis tool positioning strategies. J Mach Tools & Manuf 2000;40:185–208.

[22] Wang XC, Ghosh SK, Li YB, Wu XT. Curvature catering – a new approach in manufacture of sculptured surfaces (part 1. theorem). J Mater Process Technol 1993;38(1–2):159–75.

[23] Wang XC, Ghosh SK, Li YB, Wu XT. Curvature catering – a new approach in manufacture of sculptured surfaces (part 2. methodology). J Mater Process Technol 1993;38(1–2):177–93.

[24] Kim Y, Bartoň M, Elber G, Pottmann H. Precise gouging-free tool orientations for 5-axis CNC machining. Comp-Aid. Des 2015;58:220–9.

[25] Cao L-X, Gong H, Liu J. The offset approach of machining free form surface: Part 2: Toroidal cutter in 5-axis NC machine tools. J Mater Process Technol 2007;184(1–3):6–11.

[26] Jensen CG, Red WE, Pi J. Tool selection for five-axis curvature matched machining. Computer-Aided Des 2002;34(3):251–66.

[27] Yoon JH, Pottmann H, Lee YS. Locally optimal cutting positions for five-axis sculptured surface machining. Comp-Aid. Des 2003;35:69–81.

[28] Zhao P, Liu Z, Li Z, Cao Z. Research on tool axis vector optimization when face milling complex surfaces. Int J Adv Manuf Technol 2023;128(11–12):5081–99.

[29] Twardowski P, Legutko S, Krolczyk GM, Hloch S. Investigation of wear and tool life of coated carbide and cubic boron nitride cutting tools in high speed milling. Adv Mech Eng 2015;7(6):1687814015590216.

[30] Sun Y, Sun J, Li J, Li W, Feng B. Modeling of cutting force under the tool flank wear effect in end milling Ti6Al4V with solid carbide tool. Int J Adv Manuf Technol 2013;69(9–12):2545–53.

[31] Pivkin PM, Grechishnikov VA, Minin IV, Ershov AA, Voronin V, Nadykto AB, et al. A new approach for controlling of curved cutting edges of toroid-shaped end-milling cutter. In: Dimens. Opt. Metrology and Inspection for Pract. Applics. X. 11732, 2021, p. 76–86.

[32] Ozturk E, Tunc LT, Budak E. Investigation of lead and tilt angle effects in 5-axis ball-end milling processes. Int J Mach Tools Manuf 2009;49(14):1053–62.

[33] Blank J, Deb K. pymoo: Multi-objective optimization in Python. IEEE Access 2020;8:89497–509.

[34] Ezair B, Elber G. Using curvature bounds towards collision free 5-axis tool-paths. Graph Model 2019;103:101022.