Chapter 29
A New Method for Bevel Gear Tooth Flank Computation*

Ulf Hutschenreiter†

Abstract

The use of automatic differentiation for an exact computation of geometric properties of bevel gear tooth flanks is discussed. Known models of the kinematics of the machine tools generating bevel gear tooth flanks have been improved. Using that result, we compute curvature properties of the generated envelope surfaces directly. From an analytical point of view, the envelope condition used in differential geometry can be interpreted as a singularity of a vector function in 3-space. By an analytical characterization of this singularity we obtain a two-dimensional parameterization of the generated flank for any machine parameter set. The necessary partial derivatives of the vector function are computed with the help of automatic differentiation. For the first time it is possible to compute the curvature properties and the resulting undercut geometry between tooth and fillet surface of the tooth flanks of any spiral bevel or hypoid gear without any approximation.

Keywords: Spiral bevel gears, hypoid gears, tooth flank computation, undercutting, edge of regression, envelope condition, derivative tensors, higher order differentiation, cusp singularities.

1 Introduction

Modern methods for computing spiral bevel and hypoid gear tooth flanks are based more or less on general models of the kinematics of the gear-generating machine tools. Points of the generated tooth flank can be characterized in two ways. First, besides belonging to the surface generated by the movement of the cutter head, the points have to fulfill the equation of meshing [Litvin1994a] used in the theory of gearing. Second, they have to be points on an envelope of the family of surfaces also representing the movement of the cutter head. Both characterizations result in implicit equations describing the tooth flanks, which can be analyzed by two main approaches.

The first method, mainly represented by [Weck1987a], solves the meshing equations by numerical iteration. As many points as possible are computed on the tooth flank [Baxter1973a], [Guzman1978a], [Schriefer1983a], [Stadtfeld1993a]. This method is applicable for any settings of the machine tool but requires a large effort for computing every point on the tooth flank.

The second method is based on an explicit solution of the law of gearing [Wildgrube1984a] or the envelope condition [Baer1991a], [Dutschk1994a], [Litvin1969a], [Litvin1994a] for special machine tool settings. The effort to compute single points on the

*This work was supported as part of the research project *Optimierung komplexer Modelle* funded by the *Deutsche Forschungsgemeinschaft*, Bonn.

†Technical University Dresden, Mathematics Department, Institute of Geometry, Mommsenstr. 13, 01062 Dresden, Germany, (ulf@math.tu-dresden.de).

tooth flank decreases dramatically. Unfortunately, the method does not work for general machine tool settings.

Using both methods, one obtains two-dimensional grids on the generated tooth flank, where the shape of the grids is influenced mainly by the machine tool parameter settings. These grids are not very useful for further analysis of geometric and kinematic properties of the tooth flanks. Instead, they need to be converted into grids with parameters stemming from the gear wheel geometry. Because the effort for computing single points on such grids is high, different approximation methods have been used, for example, straight lines [Guzman1978a], splines [Schriefer1983a], [Stadtfeld1993a], or Bezier surfaces [Dutschk1994a]. All further investigations (contact properties, transmission errors, curvature properties, tooth strength analysis, etc.) are then based on these approximations.

The method proposed here is also based on a general model of a bevel gear-generating machine tool. By analytical characterization [Griewank1989b] of the solution of the envelope condition used in differential geometry and by automatic differentiation [Griewank1989a], [Griewank1991e], we can compute any point on a given grid on the tooth flank with comparatively small effort and machine accuracy.

Because we do not need an explicit solution of the envelope condition, the new method works for arbitrary machine tool settings. Moreover, it can be used with time-dependent machine tool parameters. Thus, the method is applicable with additional movements such as *modified roll* or *helical motion.*

We demonstrate the possibility of computing first- and second-order partial derivatives of the generated tooth flank. Consequently, we can compute normal directions and curvature properties at any point of the computed tooth flank directly. For the first time, tooth contact analysis does not involve any approximation, and we can predict and compute the occurrence of undercut edges for certain machine settings.

2 Modeling Spiral Bevel and Hypoid Gear-Generating Systems

In this section, we give a short description of a kinematic model of the gear generating process tailored to modern cyclo-palloid systems. Specifications or even simplifications of this model allow investigation of the cutting process for most systems generating spiral bevel gears with tooth flanks of epicycloidic shape (e.g., Klingelnberg) and circular shape (e.g., Gleason Works).

2.1 Kinematics of a Cyclo-Palloid Spiral Bevel Gear-Generating System

Cyclo-palloid bevel gear-generating systems form a continuous generating process (see Figure 1), where all flanks of a gear wheel are cut during one single work cycle. The process consists of three coupled rotational movements. First is the rotation of the cutter head, which supports the blades that cut the teeth flanks with the angular velocity ω_M. Second is the rotation of the cradle that supports the cutter head with an angular velocity ω_W, where $\omega_W \ll \omega_M$. Third is the rotation of the workpiece (the gear wheel itself) with an angular velocity ω_R. The latter can be partitioned into two parts, each of which is coupled with one of the two other rotations. This process can be expressed by the equation

$$\omega_R = \omega_{R(m)} + \omega_{R(w)} = c\omega_M + \omega\omega_W, \tag{1}$$

where $c = z_M/z$, a ratio between the number of blades of each kind on the cutter head z_M and the gear tooth number z. The variable $\omega = 1/\sin\delta$ is a ratio defined by the angle δ of the pitch cone of the gear wheel.

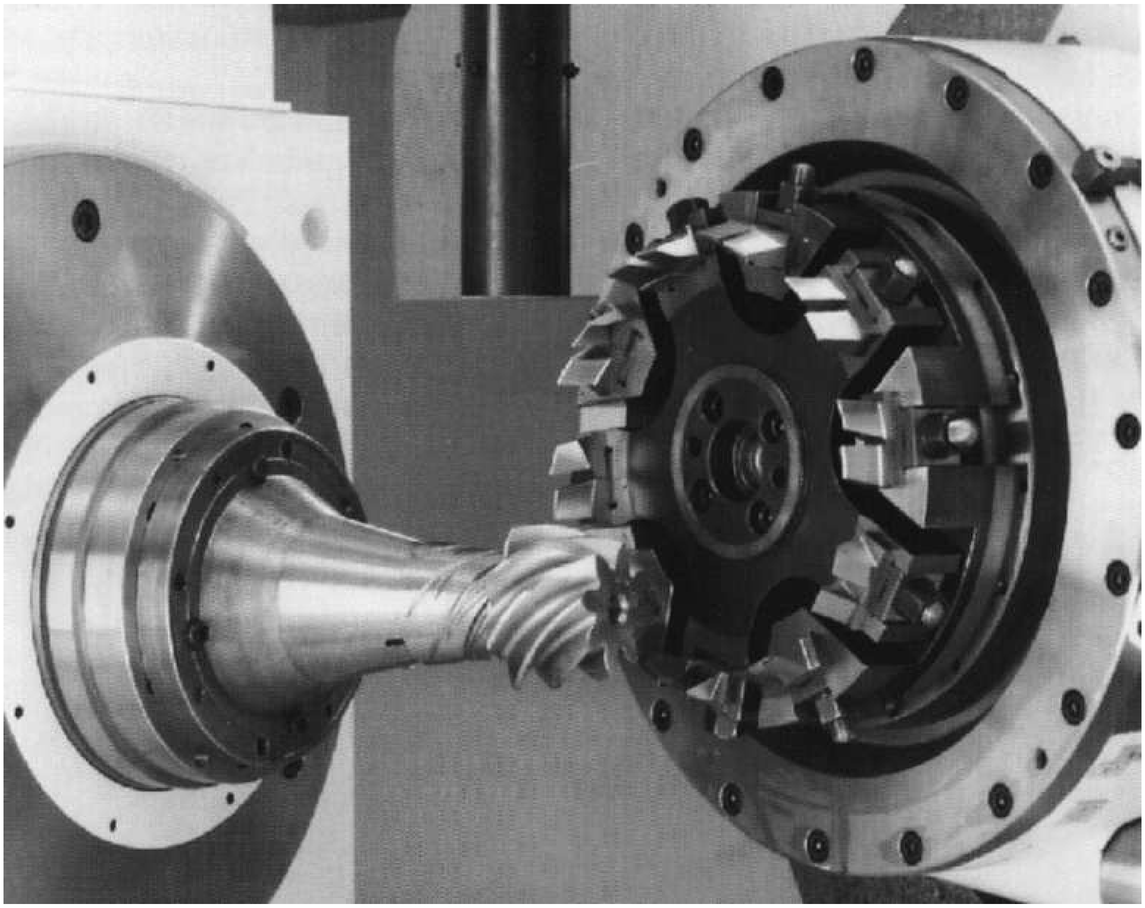

Fig. 1. *Process on a spiral bevel gear generating CNC machine Klingelnberg KNC-40.*

The most important coordinate systems describing the bevel gear-generating process are the blade coordinate system $\mathbf{\Sigma}^{Sk}$, the cutter head coordinate system $\mathbf{\Sigma}^{Mk}$, the cradle coordinate system $\mathbf{\Sigma}^{Wi}$, the gear wheel coordinate system $\mathbf{\Sigma}^{Wr}$, and the epicycloid coordinate system $\mathbf{\Sigma}^{Ep}$ that is used only for modeling [Hutschenreiter1996b]. These are the reference systems of the different curves and surfaces described in the next subsection.

2.2 Determination of the Family of Enveloping Surfaces

First, we describe the displacement of the cutting edge of each blade with respect to the epicycloid coordinate system $\mathbf{\Sigma}^{Ep}$. Using homogeneous coordinates [Litvin1994a], we can describe the cutting edge of the blade as a plane curve in $\mathbf{\Sigma}^{Sk}$ by

$$
{}^{Sk}\widetilde{\boldsymbol{r}}(u) = (1, x(u), 0, z(u))^{\top}\,, \tag{2}
$$

where u is the natural coordinate of the space curve ${}^{Sk}\boldsymbol{r}(u)$. In practice, these edges consist of two (the blade itself and its tip, see Figure 2) or even more segments. The curves are only once continuously differentiable. Considering the machine parameters of the cutter head, we describe this curve with respect to the cutter head coordinate system by

$$
{}^{Mk}\widetilde{\boldsymbol{r}}(u) = \widetilde{\boldsymbol{T}}(\boldsymbol{\varrho})\widetilde{\boldsymbol{D}}_1(\theta_K)\widetilde{\boldsymbol{D}}_3^{\top}(\theta_F)\widetilde{\boldsymbol{D}}_2^{\top}(\theta_M){}^{Sk}\widetilde{\boldsymbol{r}}(u). \tag{3}
$$

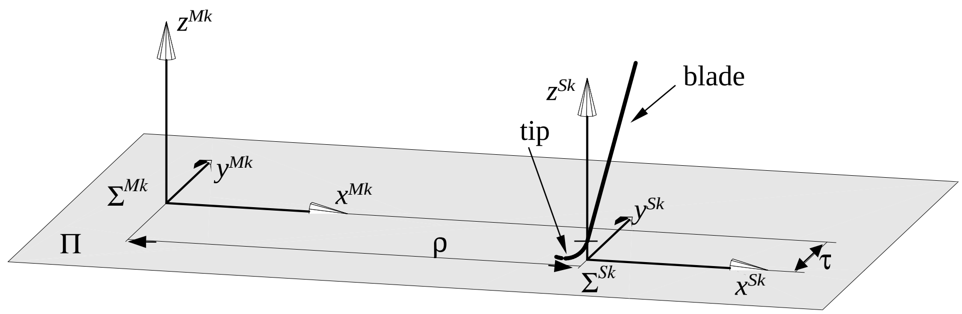

FIG. 2. *The cutting edge (blade and its tip) with respect to the cutter head coordinate system.*

Here $\widetilde{\boldsymbol{D}}_i(\alpha)$ denotes the transformation matrix defined by a rotation around the ith coordinate axis with the angle α, and $\widetilde{\boldsymbol{T}}(\boldsymbol{v})$ represents the transformation matrix defined by a translation vector $\boldsymbol{v}$.

Considering the rotation of the cutter head and several machine parameters that influence the position of the cutter head and its axis with respect to the cradle, and using the parameter $v = \omega_M s$ with some time variable s, we get the following equation:

$$\begin{aligned} {}^{Wi}\tilde{\boldsymbol{r}}(u,v) \quad &= \quad \widetilde{\boldsymbol{D}}_3\left(\tilde{\theta}_W\right)\widetilde{\boldsymbol{T}}(\boldsymbol{x}_{Mw})\widetilde{\boldsymbol{D}}_3^{\top}\left(\tfrac{\pi}{2}\right)\widetilde{\boldsymbol{D}}_2^{\top}(\pi)\widetilde{\boldsymbol{D}}_3(\theta_S)\widetilde{\boldsymbol{D}}_1(\theta_N) \\ &\quad \cdot \widetilde{\boldsymbol{D}}_3^{\top}(\theta_S)\widetilde{\boldsymbol{T}}(\boldsymbol{\sigma})\widetilde{\boldsymbol{D}}_3(v)^{Mk}\tilde{\boldsymbol{r}}(u). \end{aligned} \tag{4}$$

Now we define our epicycloid coordinate system $\boldsymbol{\Sigma}^{Ep}$ that rotates with respect to $\boldsymbol{\Sigma}^{Wi}$ with an angular velocity $\omega_E = \omega_{R(m)}/\omega = c\omega_M/\omega$ that is proportional to $\omega_{R(m)}$. Using the parameter v defined above, we get

$${}^{Ep}\tilde{\boldsymbol{r}}(u,v) = \widetilde{\boldsymbol{D}}_3\left(\tfrac{cv}{\omega}\right){}^{Wi}\tilde{\boldsymbol{r}}(u,v). \tag{5}$$

Summarizing Equations (2) - (5), we have a full description of the displacement of the space curve ${}^{Sk}\boldsymbol{r}(u)$ relative to $\boldsymbol{\Sigma}^{Ep}$:

$$\begin{aligned} {}^{Ep}\tilde{\boldsymbol{r}}(u,v) \quad &= \quad \widetilde{\boldsymbol{D}}_3\left(\tfrac{cv}{\omega}\right)\widetilde{\boldsymbol{D}}_3\left(\tilde{\theta}_W\right)\widetilde{\boldsymbol{T}}(\boldsymbol{x}_{Mw})\widetilde{\boldsymbol{D}}_3^{\top}\left(\tfrac{\pi}{2}\right)\widetilde{\boldsymbol{D}}_2^{\top}(\pi)\widetilde{\boldsymbol{D}}_3(\theta_S)\widetilde{\boldsymbol{D}}_1(\theta_N) \\ &\quad \cdot \widetilde{\boldsymbol{D}}_3^{\top}(\theta_S)\widetilde{\boldsymbol{T}}(\boldsymbol{\sigma})\widetilde{\boldsymbol{D}}_3(v)\widetilde{\boldsymbol{T}}(\boldsymbol{\varrho})\widetilde{\boldsymbol{D}}_1(\theta_K)\widetilde{\boldsymbol{D}}_3^{\top}(\theta_F)\widetilde{\boldsymbol{D}}_2^{\top}(\theta_M)^{Sk}\tilde{\boldsymbol{r}}(u). \end{aligned} \tag{6}$$

As long as the machine parameters are constant, the surface is invariant of the shape with respect to the cradle position t. The name *epicycloid* originates from the points of the cutting edge, which run on epicycloids as long as the axes of cutter head and cradle are parallel.

Although the equation (6) looks like a parameter description of a surface, distinct positions v of the cutter head are mapped to different positions of the gear wheel. We therefore have to look for a description with respect to $\boldsymbol{\Sigma}^{Wr}$. Under the assumption $\omega_W \ll \omega_M$, we can interpret such a description so that a generating surface is swept out by the blades for any fixed position of the gear wheel. The epicycloid coordinate systems $\boldsymbol{\Sigma}^{Ep}$ together with the gear wheel system $\boldsymbol{\Sigma}^{Wr}$ form a coupled rotation around the crossed axes. The ratio between the angular velocities is defined by ω. In accordance with (1), we get the following formula of the *general generating surface* with respect to any cradle position $t = \omega_W s$:

$$^{Wr}\widetilde{\boldsymbol{r}}(u,v)(t) = \widetilde{\boldsymbol{D}}_3\,(cv)\widetilde{\boldsymbol{D}}_3\,(\omega t)\widetilde{\boldsymbol{T}}\widetilde{\boldsymbol{D}}_3^{\top}\,(t)\widetilde{\boldsymbol{D}}_3^{\top}\,(\tfrac{cv}{\omega})\,^{Ep}\widetilde{\boldsymbol{r}}(u,v), \tag{7}$$

with a transformation matrix $\widetilde{\boldsymbol{T}}$ that describes the position of the gear wheel axis relative to the cradle axis:

$$\widetilde{\boldsymbol{T}} = \widetilde{\boldsymbol{D}}_1\,(\delta)\widetilde{\boldsymbol{D}}_3^{\top}\,(\pi)\widetilde{\boldsymbol{D}}_1\,(\tfrac{\pi}{2})\widetilde{\boldsymbol{T}}(\boldsymbol{m}) = \left(\begin{array}{c|ccc} 1 & 0 & 0 & 0 \\ \hline -m & -1 & 0 & 0 \\ 0 & 0 & -\sin\delta & \cos\delta \\ 0 & 0 & \cos\delta & \sin\delta \end{array}\right).$$

In the particular case $t = 0$, we get the *special generating surface*

$$^{Wr}\widetilde{\boldsymbol{r}}(u,v)(0) = \widetilde{\boldsymbol{D}}_3\,(cv)\widetilde{\boldsymbol{T}}\widetilde{\boldsymbol{D}}_3^{\top}\,(\tfrac{cv}{\omega})\,^{Ep}\widetilde{\boldsymbol{r}}(u,v).$$

2.3 Computing the Envelope of a Family of Surfaces of Varying Shape

Equation (7) can be interpreted as a description of a family of surfaces of varying shape. Any surface $^{Wr}\boldsymbol{r}(t)(u,v)$ (disregarding the irrelevant first component of $^{Wr}\widetilde{\boldsymbol{r}}(t)(u,v)$) represents the displacement of the blade relative to the gear wheel at a distinct cradle position t. The sequence of such single cuts during the generating process eventually generates the tooth flank on the gear wheel. Hence, the resulting tooth flank is equivalent to the envelope of the family of these surfaces. From differential geometry, we know that this envelope has to satisfy the envelope condition

$$\left(^{Wr}\boldsymbol{r}_u, {}^{Wr}\boldsymbol{r}_v, {}^{Wr}\boldsymbol{r}_t\right) = 0, \tag{8}$$

where $(.,.,.)$ describes the scalar triple product of three vectors. Traditionally, analytical methods [Baer1972a] have been used to get an explicit equation $\underline{\boldsymbol{r}} = \underline{\boldsymbol{r}}(u,v)$ of the envelope. Cases are preferred where all surfaces of the family have the same shape, and a congruent displacement can be found mapping one surface to another [Litvin1995a].

The new method proposed here requires no any restrictions regarding the machine tool settings or surfaces of the same shape. For any family of surfaces $^{Wr}\boldsymbol{r}(t)(u,v)$ where $^{Wr}\widetilde{\boldsymbol{r}}(t)(u,v)$ satisfies (7), we can represent them as a vector function of 3-space as follows:

$$\begin{aligned} \boldsymbol{F}:\quad & \mathbb{R}^3 \rightarrow \mathbb{R}^3 : (u,v,t)^{\top} \mapsto (x,y,z)^{\top} = \boldsymbol{F}(u,v,t) \\ & \text{with}\quad \boldsymbol{F}(u,v,t) = \boldsymbol{D}_3(cv)\left(\boldsymbol{U}(t)\boldsymbol{r}(u,v) + \boldsymbol{h}(t)\right). \end{aligned} \tag{9}$$

Here $\boldsymbol{D}_3(cv)$ is the rotation matrix part of $\widetilde{\boldsymbol{D}}_3\,(cv)$. $\boldsymbol{U}(t)$ and $\boldsymbol{h}(t)$ are the rotation and vector part of the transformation $\widetilde{\boldsymbol{U}}(t) = \widetilde{\boldsymbol{D}}_3\,(\omega t)\widetilde{\boldsymbol{T}}\widetilde{\boldsymbol{D}}_3^{\top}\,(t)$, and $\boldsymbol{r}(u,v) := {}^{Wi}\boldsymbol{r}(u,v)(t)$ is the vector part of $^{Wi}\widetilde{\boldsymbol{r}}(u,v)(t)$. Moreover, time dependency of any of the machine tool parameters used in the equations (6) or (7) is allowed. Therefore, our method considers additional movements such as *modified roll* or *helical motion.*

In equivalence to (8), we get the analytical form of the envelope condition

$$(\boldsymbol{F}_u, \boldsymbol{F}_v, \boldsymbol{F}_t) = \det\left(\boldsymbol{F}'\right) = 0, \tag{10}$$

which can be interpreted as an implicit definition of a function $t = t(u,v)$. If any value of this function can be computed even if no explicit solution $t(u,v)$ is available, we can get a two-dimensional parameterization of the envelope of the surface family (7) in the form

$$\underline{\boldsymbol{r}}(u,v) := \boldsymbol{F}(u,v,t(u,v)). \tag{11}$$

2.4 Determination of the Curvature Properties

Based on results in [Baer1972a], we can give the formulas for the computation of the curvature properties of an envelope $\underline{\boldsymbol{r}}$. Use of these formulas not only assumes the computability of the function $t = t(u,v)$ but also assumes the computability of the partial derivative of that function. Here *computability* means the availability of a finite algorithm for evaluating a function at all feasible arguments. We solve this task in the following section. For the moment, we assume the availability of these first- and second-order partial derivatives.

Let $\underline{\boldsymbol{r}}$ be the envelope of a family of surfaces. In accordance with (9) and (11), $\underline{\boldsymbol{r}}$ is given by

$$\begin{aligned} \underline{\boldsymbol{r}} &:= \underline{\boldsymbol{r}}(u,v) = \boldsymbol{F}\left(u,v,t(u,v)\right) \\ &= \boldsymbol{D}_3(cv)\left(\boldsymbol{U}\left(t(u,v)\right)\boldsymbol{r}(u,v) + \boldsymbol{h}\left(t(u,v)\right)\right). \end{aligned} \tag{12}$$

Then the quantities

$$E = \underline{\boldsymbol{r}}_u^2; \quad F = \underline{\boldsymbol{r}}_u \underline{\boldsymbol{r}}_v; \quad G = \underline{\boldsymbol{r}}_v^2$$

describe the first order Gaussian fundamental coefficients of surface theory of the envelope $\underline{\boldsymbol{r}}$. Using the definition $W^2 := EG - F^2$, we can describe the second-order fundamental coefficients by

$$L = \frac{1}{W}\left(\underline{\boldsymbol{r}}_u, \underline{\boldsymbol{r}}_v, \underline{\boldsymbol{r}}_{uu}\right); \quad M = \frac{1}{W}\left(\underline{\boldsymbol{r}}_u, \underline{\boldsymbol{r}}_v, \underline{\boldsymbol{r}}_{uv}\right); \quad N = \frac{1}{W}\left(\underline{\boldsymbol{r}}_u, \underline{\boldsymbol{r}}_v, \underline{\boldsymbol{r}}_{vv}\right),$$

where $(.,.,.)$ again is the scalar triple product of three vectors.

Using these fundamental coefficients and other well-known formulas of differential geometry, we can compute (among others) the Gaussian curvature K or the mean curvature H. Thus, both the principal curvature values κ_1 and κ_2 and the principal curvature directions can be obtained. Using Euler's theorem on curvature in surface theory and Meusnier's theorem [Blaschke1924a], we can compute the normal curvature of any spatial curve lying on the envelope at any point of the envelope surface.

To compute the first- and second-order partial derivatives of $\underline{\boldsymbol{r}}$, we use the factorization in (12) and the following shortcuts for the partial derivatives of the single elements of the equation: $\boldsymbol{D}_3 := \boldsymbol{D}_3(cv)$, $\dot{\boldsymbol{D}}_3 := d/dv\,(\boldsymbol{D}_3(cv))$, $\ddot{\boldsymbol{D}}_3 := d^2/dv^2\,(\boldsymbol{D}_3(cv))$; $\boldsymbol{U} := \boldsymbol{U}(t)$, $\dot{\boldsymbol{U}} := d/dt\,(\boldsymbol{U}(t))$, $\ddot{\boldsymbol{U}} := d^2/dt^2\,(\boldsymbol{U}(t))$; $\boldsymbol{h} := \boldsymbol{h}(t)$, $\dot{\boldsymbol{h}} := d/dt\,(\boldsymbol{h}(t))$, $\ddot{\boldsymbol{h}} := d^2/dt^2\,(\boldsymbol{h}(t))$. Using the equations for the partial derivatives of $\boldsymbol{U}$ and $\boldsymbol{h}$ with respect to u and v, namely, $\boldsymbol{U}_u := \dot{\boldsymbol{U}}t_u$, $\boldsymbol{U}_v := \dot{\boldsymbol{U}}t_v$, $\boldsymbol{U}_{uu} := \ddot{\boldsymbol{U}}t_u^2$, $\boldsymbol{U}_{uv} := \ddot{\boldsymbol{U}}t_u t_v$ and $\boldsymbol{U}_{vv} := \ddot{\boldsymbol{U}}t_v^2$ and $\boldsymbol{h}_u := \dot{\boldsymbol{h}}t_u$, $\boldsymbol{h}_v := \dot{\boldsymbol{h}}t_v$, $\boldsymbol{h}_{uu} := \ddot{\boldsymbol{h}}t_u^2$, $\boldsymbol{h}_{uv} := \ddot{\boldsymbol{h}}t_u t_v$ and $\boldsymbol{h}_{vv} := \ddot{\boldsymbol{h}}t_v^2$, respectively, we can describe the partial derivatives of $\underline{\boldsymbol{r}}(u,v)$ by the equations

$$\underline{\boldsymbol{r}}_u = \boldsymbol{D}_3\left(\dot{\boldsymbol{U}}t_u\boldsymbol{r} + \dot{\boldsymbol{h}}t_u + \boldsymbol{U}\boldsymbol{r}_u\right); \tag{13}$$

$$\underline{\boldsymbol{r}}_v = \dot{\boldsymbol{D}}_3\left(\boldsymbol{U}\boldsymbol{r} + \boldsymbol{h}\right) + \boldsymbol{D}_3\left(\dot{\boldsymbol{U}}t_v\boldsymbol{r} + \dot{\boldsymbol{h}}t_v + \boldsymbol{U}\boldsymbol{r}_v\right); \tag{14}$$

$$\underline{\boldsymbol{r}}_{uu} = \boldsymbol{D}_3\left[\ddot{\boldsymbol{U}}t_u^2\boldsymbol{r} + \ddot{\boldsymbol{h}}t_u^2 + \dot{\boldsymbol{U}}\left(t_{uu}\boldsymbol{r} + 2t_u\boldsymbol{r}_u\right) + \dot{\boldsymbol{h}}t_{uu} + \boldsymbol{U}\boldsymbol{r}_{uu}\right]; \tag{15}$$

$$\begin{aligned} \underline{\boldsymbol{r}}_{uv} &= \dot{\boldsymbol{D}}_3\left(\dot{\boldsymbol{U}}t_u\boldsymbol{r} + \dot{\boldsymbol{h}}t_u + \boldsymbol{U}\boldsymbol{r}_u\right) + \boldsymbol{D}_3\left[\ddot{\boldsymbol{U}}t_u t_v\boldsymbol{r}\right. \\ &\quad \left. + \ddot{\boldsymbol{h}}t_u t_v + \dot{\boldsymbol{U}}\left(t_{uv}\boldsymbol{r} + t_u\boldsymbol{r}_v + t_v\boldsymbol{r}_u\right) + \dot{\boldsymbol{h}}t_{uv} + \boldsymbol{U}\boldsymbol{r}_{uv}\right]; \end{aligned} \tag{16}$$

$$\begin{aligned} \underline{\boldsymbol{r}}_{vv} &= \ddot{\boldsymbol{D}}_3\left(\boldsymbol{U}\boldsymbol{r} + \boldsymbol{h}\right) + 2\dot{\boldsymbol{D}}_3\left(\dot{\boldsymbol{U}}t_v\boldsymbol{r} + \dot{\boldsymbol{h}}t_v + \boldsymbol{U}\boldsymbol{r}_v\right) \\ &\quad + \boldsymbol{D}_3\left[\ddot{\boldsymbol{U}}t_v^2\boldsymbol{r} + \ddot{\boldsymbol{h}}t_v^2 + \dot{\boldsymbol{U}}\left(t_{vv}\boldsymbol{r} + 2t_v\boldsymbol{r}_v\right) + \dot{\boldsymbol{h}}t_{vv} + \boldsymbol{U}\boldsymbol{r}_{vv}\right]. \end{aligned} \tag{17}$$

In [Litvin1995a], another method of computation of curvature properties was proposed. The author clearly describes how the computed curvature properties can be used for tooth contact analysis. However, the approach used there works with special parameter settings and assumes that all generating surfaces have the same shape.

3 Computing a Spiral Bevel or Hypoid Gear Tooth Flank

In this section, we give an analytical characterization of the envelope condition and collect useful equations necessary to compute the envelope itself and its partial derivatives. We describe the implementation of our approach using the method of automatic differentiation, and we discuss some of the problems that arise while computing the tooth flanks of spiral bevel or hypoid gears.

3.1 Equivalence between the Envelope Condition and the Solution of a Singularity Problem

In the preceding section, we defined a vector function $\boldsymbol{F}$ (9) of 3-space describing a family of surfaces

$$\begin{aligned} \boldsymbol{F}: \quad & \mathbb{R}^3 \to \mathbb{R}^3 : \boldsymbol{w} \mapsto \boldsymbol{x} = \boldsymbol{F}(\boldsymbol{w}) \\ & \text{with} \quad \boldsymbol{x} = (x, y, z)^\top \quad \text{and} \quad \boldsymbol{w} = (u, v, t)^\top . \end{aligned}$$

From an analytical point of view, the envelope condition (8) of the envelope surface $\underline{\boldsymbol{r}}$ of that family is equivalent to the singularity of the Jacobian of $\boldsymbol{F}$ expressed by

$$\det\left(\boldsymbol{F}'\right) = \begin{vmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} & \frac{\partial x}{\partial t} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} & \frac{\partial y}{\partial t} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} & \frac{\partial z}{\partial t} \end{vmatrix} = 0. \tag{18}$$

Now we investigate the singularity problem described by the equation (18). Additionally, we assume that the rank of $\boldsymbol{F}'$ is not lower than 2 over the domain of interest, and so the singularity set defined by (18) is well determined by

$$\operatorname{rank}\left(\boldsymbol{F}'\right) = 2.$$

In our case, the restriction $\operatorname{rank}(\boldsymbol{F}') \geq 2$ is easy to satisfy because the u and v parameter lines of the surfaces ${}^{W_r}\tilde{\boldsymbol{r}}(t)(u, v)$ in (7) that represents the cutting edge and the rotation of the cutter head are nearly orthogonal. No singularity can appear on these surfaces themselves.

Using the approach of [Griewank1989b],

$$\boldsymbol{A}_{(4,4)}(\boldsymbol{w}) := \begin{pmatrix} \boldsymbol{F}'_{(3,3)}(\boldsymbol{w}) & \boldsymbol{a}_{(3,1)} \\ \boldsymbol{b}^\top_{(3,1)} & 0 \end{pmatrix}.$$

We can always find two locally constant vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ that guarantee that the matrix $\boldsymbol{A}$ is nonsingular. Then the inverse matrix $\boldsymbol{A}^{-1}$ exists, and we can define two vectors $\boldsymbol{c}$ and $\boldsymbol{d}$ and a scalar g by

$$\boldsymbol{A}^{-1}_{(4,4)}(\boldsymbol{w}) =: \begin{pmatrix} B_{(3,3)}(\boldsymbol{w}) & \boldsymbol{d}_{(3,1)}(\boldsymbol{w}) \\ \boldsymbol{c}^\top_{(3,1)}(\boldsymbol{w}) & -g(\boldsymbol{w}) \end{pmatrix}.$$

Using the identities $\boldsymbol{A}(\boldsymbol{w})\boldsymbol{A}^{-1}(\boldsymbol{w}) = I_{(4,4)}$ and $\boldsymbol{A}^{-1}(\boldsymbol{w})\boldsymbol{A}(\boldsymbol{w}) = I_{(4,4)}$, we get eight equations, four of which are of particular interest:

$$\boldsymbol{F}'(\boldsymbol{w})\boldsymbol{d}(\boldsymbol{w}) - g(\boldsymbol{w})\boldsymbol{a} = (0\ 0\ 0)^\top \iff \boldsymbol{F}'(\boldsymbol{w})\boldsymbol{d}(\boldsymbol{w}) = g(\boldsymbol{w})\boldsymbol{a}; \tag{19}$$

$$\boldsymbol{b}^\top \boldsymbol{d}(\boldsymbol{w}) + 0 = 1 \iff \boldsymbol{b}^\top \boldsymbol{d}(\boldsymbol{w}) = 1; \tag{20}$$

$$\boldsymbol{c}^\top(\boldsymbol{w})\boldsymbol{F}'(\boldsymbol{w}) - g(\boldsymbol{w})\boldsymbol{b}^\top = (0\ 0\ 0) \iff \boldsymbol{c}^\top(\boldsymbol{w})\boldsymbol{F}'(\boldsymbol{w}) = g(\boldsymbol{w})\boldsymbol{b}^\top; \tag{21}$$

$$\boldsymbol{c}^\top(\boldsymbol{w})\boldsymbol{a} + 0 = 1 \iff \boldsymbol{c}^\top(\boldsymbol{w})\boldsymbol{a} = 1. \tag{22}$$

To compute the vectors $\boldsymbol{c}$ and $\boldsymbol{d}$ and the scalar g, we need not compute the inverse matrix $\boldsymbol{A}^{-1}$ but rather solve the following systems of linear equations:

$$\boldsymbol{A}(\boldsymbol{w}) \begin{pmatrix} \boldsymbol{d}_{(3,1)}(\boldsymbol{w}) \\ \\ -g(\boldsymbol{w}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}; \tag{23}$$

$$\boldsymbol{A}^\top(\boldsymbol{w}) \begin{pmatrix} \boldsymbol{c}_{(3,1)}(\boldsymbol{w}) \\ \\ -g(\boldsymbol{w}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{24}$$

Because $\boldsymbol{A}$ is nonsingular, the solutions of these systems exist. Moreover, the computation is cheap, since we can use one factorization for both systems.

From Cramer's rule, we know that $-g(\boldsymbol{w}) = \det(\boldsymbol{F}')/\det(\boldsymbol{A})$. Because of the assumed regularity of $\boldsymbol{A}$, it is easy to obtain the crucial equivalence

$$\det\left(\boldsymbol{F}'(\boldsymbol{w})\right) = 0 \iff g(\boldsymbol{w}) = 0. \tag{25}$$

The importance of this relation lies in the fact that now the equation $g(\boldsymbol{w}) = 0$ is equivalent to (8) or (18), respectively. Thus

$$g(\boldsymbol{w}) = g(u, v, t) = g(u, v, t(u, v)) = 0 \tag{26}$$

can be interpreted as the implicit definition of the same function $t(u, v)$ that was originally defined by the envelope condition (8) or (10), respectively.

Applying the results in [Griewank1989b] to our special case, we obtain a collection of equations that is useful for computing the properties of the envelope $\underline{\boldsymbol{r}}$ characterized by (26). The scalar vector function $g(\boldsymbol{w})$ and their partial derivatives can be expressed by

$$\begin{aligned} g(\boldsymbol{w}) &= \boldsymbol{c}^\top(\boldsymbol{w})\boldsymbol{F}'(\boldsymbol{w})\boldsymbol{d}(\boldsymbol{w}); \\ g'(\boldsymbol{w}) &= (g_u, g_v, g_t)^\top = \boldsymbol{c}^\top(\boldsymbol{w})\boldsymbol{F}''(\boldsymbol{w})\boldsymbol{d}(\boldsymbol{w}); \\ g''(\boldsymbol{w}) &= \begin{pmatrix} g_{uu} & g_{vu} & g_{tu} \\ g_{uv} & g_{vv} & g_{tv} \\ g_{ut} & g_{vt} & g_{tt} \end{pmatrix} \\ &= \boldsymbol{c}^\top(\boldsymbol{w})\boldsymbol{F}'''(\boldsymbol{w})\boldsymbol{d}(\boldsymbol{w}) - \left(\boldsymbol{c}^\top(\boldsymbol{w})\right)'\boldsymbol{F}'(\boldsymbol{w})\boldsymbol{d}'(\boldsymbol{w}) - \left(\left(\boldsymbol{c}^\top(\boldsymbol{w})\right)'\boldsymbol{F}'(\boldsymbol{w})\boldsymbol{d}'(\boldsymbol{w})\right)^\top. \end{aligned} \tag{27}$$

The partial derivatives of the vectors $\boldsymbol{c}$ and $\boldsymbol{d}$ occurring in (27) can be computed by using the same factorization as for the solution of (23). Differentiating (23) by u, we obtain the following linear system:

$$\boldsymbol{A}(\boldsymbol{w})\,\frac{\partial}{\partial u}\begin{pmatrix} \boldsymbol{d}(\boldsymbol{w}) \\ -g(\boldsymbol{w}) \end{pmatrix} = \begin{pmatrix} -\left[\frac{\partial}{\partial u}\boldsymbol{F}'\right]\boldsymbol{d}(\boldsymbol{w}) \\ 0 \end{pmatrix}.$$

In analogy to this, the systems of the other partial derivatives of $\boldsymbol{d}$ and by differentiation of (24) those of $\boldsymbol{c}$ can easily be obtained.

Now we have assembled all the equations necessary to compute the partial derivatives of the implicit function $t(u, v)$, referred to in the preceding subsection in the equations (13) through (17). Under the assumption $g_t \neq 0$, we can apply the implicit function theorem to (26) and finally get

$$\begin{aligned}
t_u &= -\frac{g_u}{g_t}; \\
t_v &= -\frac{g_v}{g_t}; \\
t_{uu} &= -\frac{1}{g_t^3}\left(g_{uu}g_t^2 - 2g_u g_{ut} g_t + g_u^2 g_{tt}\right); \\
t_{uv} &= -\frac{1}{g_t^3}\left(g_{uv}g_t^2 - g_{ut}g_v g_t - g_u g_{vt} g_t + g_u g_v g_{tt}\right); \\
t_{vv} &= -\frac{1}{g_t^3}\left(g_{vv}g_t^2 - 2g_v g_{vt} g_t + g_v^2 g_{tt}\right).
\end{aligned}$$

3.2 Realization with ADOL–C

The model presented in Section 2 allows the straightforward implementation of the function $\boldsymbol{F}$ describing the displacement of the blade relative to the gear wheel during the generating process. The complexity of this function consists only in the consideration of many different machine parameters that must be incorporated correctly. The structure of the function is simple: it is merely a sequence of translations and rotations without any iterations. Even then, it might be hard to compute the partial derivatives up to third order symbolically. Therefore, we prefer to use the method of automatic differentiation [Griewank1989a], [Griewank1991e] to solve this problem.

Because we also need higher-order partial derivatives of $\boldsymbol{F}$, we decided to use the automatic differentiation package ADOL–C [Griewank1996b]. ADOL–C provides the following two advantages. First, one can theoretically obtain directional derivatives of arbitrarily high order. Second, one can use the source code of the given function with only minor modifications. The system records the execution trace, referred to as the *tape* of the original evaluation process of the implemented function. By subsequent forward or reverse derivative evaluations using the drivers provided by the system and accessing the written tape(s), we computed derivatives relatively quickly and with machine accuracy. The effort of computing the derivatives grows only quadratically with respect to the order of the derivatives. Moreover, during the derivative evaluation of the tapes, no additional computational error arising from the evaluation method is incurred.

To satisfy the special needs of our project and of some other users, we implemented a new driver [Griewank1995a] that allows the computation of the full third-derivative tensor of a vector function in forward mode. Normally one would compute some directional derivatives of the form $\boldsymbol{c}^\top \boldsymbol{F}'''\boldsymbol{dd}$ by using both forward and reverse modes, but considering the dimension of our problem function in 3-space, it is better to compute the full tensor

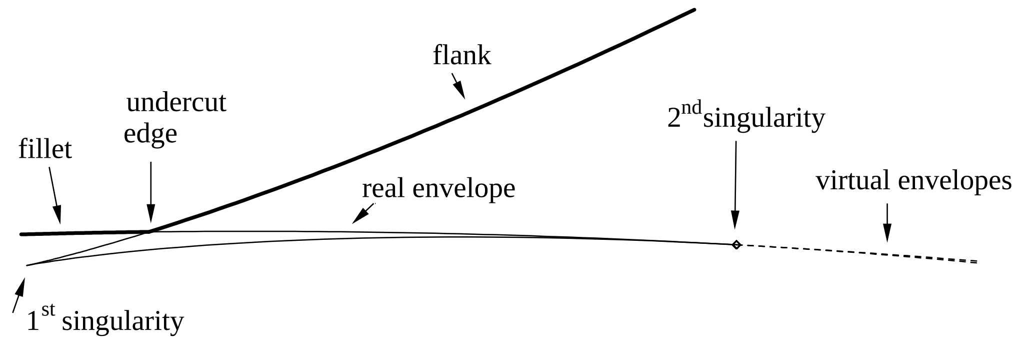

FIG. 3. *Part of the tooth profile with singularities and undercut.*

during a single forward sweep. To compute the 60 distinct components of $\boldsymbol{F}$, $\boldsymbol{F}'$, $\boldsymbol{F}''$, and $\boldsymbol{F}'''$ takes about 50 to 60 times the evaluation time of the original C code for $\boldsymbol{F}$. We get a time ratio of 4 to 5 between the computation of g'' and g.

Based on the computation of $\boldsymbol{F}'$, $\boldsymbol{F}''$, and $\boldsymbol{F}'''$ executed with the help of ADOL–C, it is possible to implement various numerical methods [Schwetlick1991a] with high accuracy. These methods include some variants of Newton methods to compute the points of the tooth flank by solving (26) for given u and v, and eventually to compute such points on the flanks for given grids in spherical coordinates. Thus, by using the methods proposed here, it is possible for the first time to compute normal directions and curvature properties at arbitrary points of the generated tooth flanks directly and for general machine settings. When we have solved (26) for distinct values of u and v, all steps to compute g' and g'' are straightforward.

Figure 4 shows a plot of one tooth of a six-toothed pinion computed with the actual implementation of our method. The grid used in the figure originates from the geometry of the gear wheel only. The tooth profiles running from the tooth root on the convex flank via the tooth tip to the tooth root on the concave flank lie on spheres with the center lying on the pinion axis. This center itself is defined as the foot of the perpendicular between the crossed axes of the two gear wheels, which eventually run in the gear box. Some of the problems occurring during the computation of such profiles will be discussed in the next subsection.

3.3 Singularities at the Envelope and the Undercut Problem

Problems during the computation of points on a tooth flank arise mainly from two related causes. First, the cutting edge on the blade often consists of more than one segment (see Figure 2) with a parameterization only once continuously differentiable. For example, a spherical blade consists of two spherical segments: one for the blade itself with a very small curvature, and one for the tip of the blade with a very small radius and large curvature. Thus, it is necessary to distinguish between the two segments of the blade (2) during the evaluation of $\boldsymbol{F}$ and its partial derivatives.

The package ADOL–C provides the possibility of handling such conditional branches by tagging them while recording the tape and by writing both cases on the tape. In our case, it seemed better to write an extra tape for each segment of the blade. The tapes are not very long, so the memory requirement does not play an important role. Also, we can monitor the branching inside our algorithms but outside the ADOL–C drivers, and so we have a convenient and effective way to evaluate different tapes with the same values of u, v, and t. This enables the computation of points on the envelope generated by some virtual extensions of the single segments of the blade outside their original domains of u, which is useful for the search for the undercut edge described below.

The second problem arises from the occurrence of some singularities on the computed envelopes, which in practice results in the occurrence of some undercut that partly destroys the generated tooth flank. When we used spherical blades in our experiments, as in Figure 2, we found (in agreement with other authors, e.g., [Dutschk1994a]) that whenever undercut occurred, a pair of singularities could be observed. One singularity was on that part of the envelope generated by the blade segment itself, and the other on the part of the envelope generated by the point separating the two curve segments. Practical experience shows that an intersection line, generated by the two segments of the blade, must exist between the two parts of the whole envelope surface. This line can actually be obtained as the so-called undercut edge on real tooth flanks.

Figure 3 illustrates this situation in a profile of the envelope in the environment of the occurring singularities. Both segments were virtually extended over their separating point, resulting in virtual envelopes marked in the figure as dotted lines. The intersection of the real envelope generated by the blade together with the sphere is marked by an ordinary line. The resulting tooth profile, which consists of flank and fillet that contain one point of nondifferentiability, is marked by a thick line. This point of nondifferentiability marks the undercut edge occurring under the conditions of that case. If any singularities occur only at those parts of the envelope generated by the virtual extensions of the blade segments, no singularity occurs on that curve on the envelope that was generated by the separating point, and no undercut on the tooth flank is observed.

To predict the occurrence of undercut and to compute the undercut edge, we must determine the singularity occurring on the envelope of the blade segment together with its virtual extension. Then we can compute the inverse of the one-dimensional singularity inside the original two-dimensional singularity set. Observing the u-values of the inverse used in the parameterization of the blade, we can decide whether the one-dimensional singularity is part of the envelope generated by the real blade and whether any undercut occurs. If the occurrence of undercut is observed and the position of the singularity is computed, the cut line between the two surfaces can be computed. Hence, the position of the undercut edge is available.

We do not know a necessary and sufficient condition to characterize the occurrence of singularities on general surfaces or even on envelope surfaces. Currently, we use the following approach for a necessary condition, with some success:

$$\boldsymbol{d}^{\top}(\boldsymbol{w})g'(\boldsymbol{w}) = 0. \tag{28}$$

This approach is motivated by the fact that if Equation (28) is satisfied, the vector $\boldsymbol{d}$ of the domain of arguments that satisfies the equation (19) (and thus is a zero vector of $\boldsymbol{F}'$ whenever g vanishes) is orthogonal to the gradient g' of the singularity set [Schuster1996a]. That means $\boldsymbol{d}$ attaches only the singularity set, and thus generates another new singularity on the image of the singularity set, which occurs on the envelope originally studied for singularities. The results obtained with this approach are in accordance with the necessary undercut condition proposed in [Dutschk1994a]. The method for determining undercut, as proposed in [Dutschk1994a], is not very satisfactory, especially for an effective implementation, but an exact proof of the correctness of the new approach (28) still is

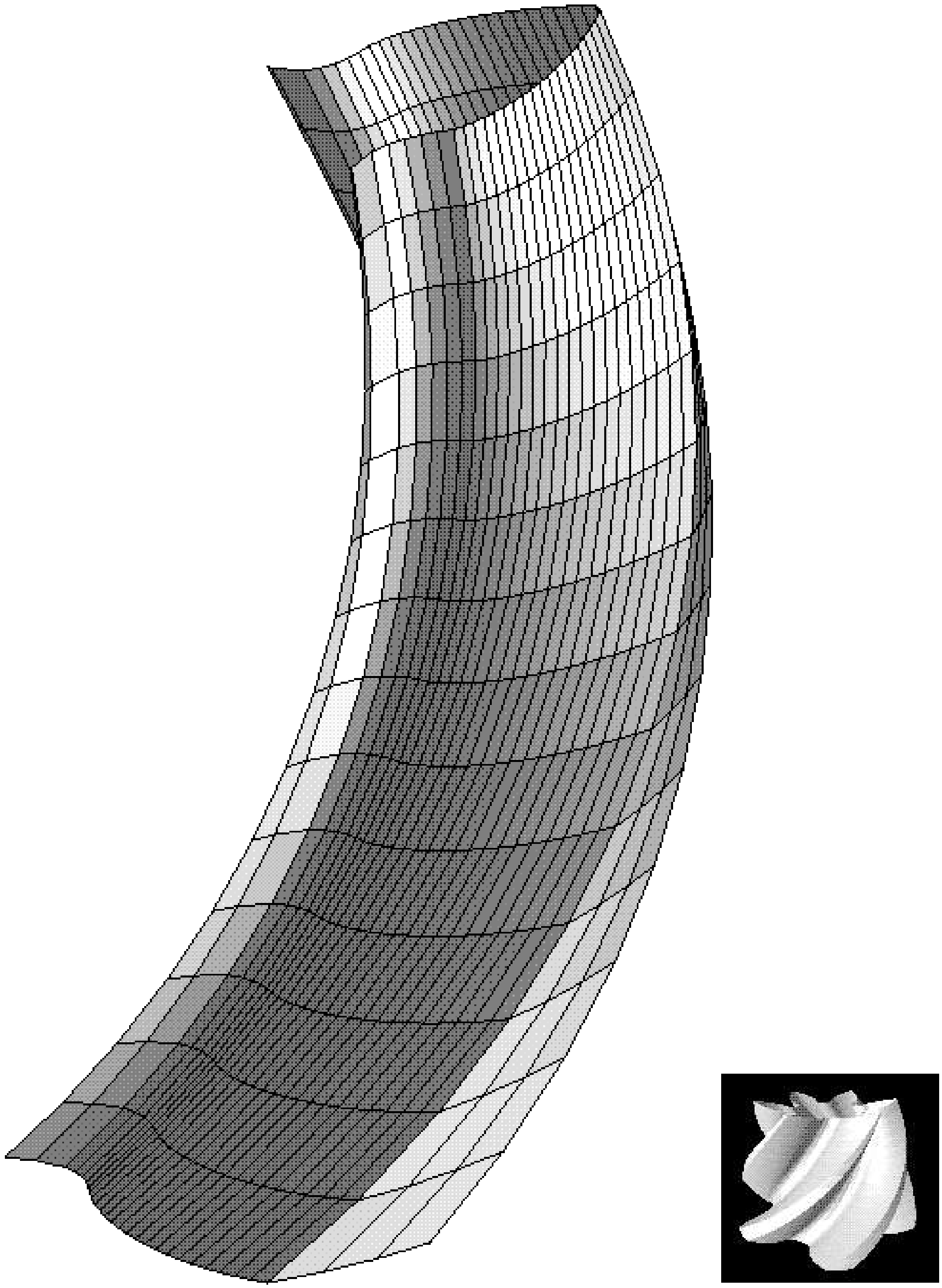

FIG. 4. *Example of a computed tooth with 2 undercut edges – whole pinion on the right.*

lacking.

Nevertheless, the approach proved useful for our computational experiments, and the condition was sufficient in all cases studied with the new implementation. All images of the solutions of the equation (28) have proven to be singularities on the envelope surfaces. Based on the results, we can distinguish between that part of the envelope determining the usable tooth flank and that part determining the fillet at the tooth root. This calculation is independent of the undercut's occurrence. By sweeping over the full face width of the tooth, exact and complete tooth profiles can be computed. In Figure 4, we can see at the upper part of the tooth (the tooth toe) that an undercut edge occurs at both flanks. The angle between the fillet and the flank is nearly 170°. Thus, it is hard to present and observe. Running to the lower part (the tooth heel) the undercut edges end in smooth ranges of the flanks, a result that agrees with numerical calculations.

4 Outlook

Based on our results, the exact contact geometry of any mating pair of toothed bevel gear can be computed for any position of gear and pinion. At a known point of contact that can be computed with high accuracy, the contact line (if it exists) and the curvature of the tooth flanks can be determined analytically.

Therefore, the path of contact and the tooth-bearing area can be computed over the full length of action of the tooth flank surfaces. Thus, we get the transmission function not only for the meshing cycle of one pair of toothed gears but also for multiple teeth pair contacts. Using both known and new evaluation functionals for the position of the path of contact, the tooth-bearing area, the transmission error, the contact, and the tooth root stress that appears between the teeth under load, we hope to develop a goal function usable for an optimization of the tooth flank geometry and kinematic with the machine tool parameter settings and the parameter of the gear box itself.

Using the method of automatic differentiation as a tool for optimizing functions of high complexity, we also plan to implement a new kind of optimization strategy for the spiral bevel and hypoid gears.

Acknowledgments

I thank Gert Bär and Andreas Griewank for many helpful discussions and Olaf Vogel for his assistance in implementing the algorithms we developed.

References

[Baer1972a] G. Bär, *Zur Verzahnungtheorie der Schneckengetriebe*, PhD thesis, TU Dresden, 1972.

[Baer1991a] G. Bär and B. Liebschner, *Fitting flanks and contact properties of hypoid gears*, in 8-th World Congress on the Theory of Machines and Mechanisms, Prague, 1991, pp. 1095–1098.

[Baxter1973a] M. L. Baxter, *Second-order surface generation*, Journal of Industrial Mathematics Society, (1973).

[Blaschke1924a] W. Blaschke, *Differentialgeometrie*, Springer-Verlag, Berlin, 1924.

[Dutschk1994a] R. Dutschk, *Geometrische Probleme bei Herstellung und Eingriff bogenverzahnter Kegelräder*, PhD thesis, TU Dresden, 1994.

[Griewank1989a] A. Griewank, *On automatic differentiation*, in Mathematical Programming: Recent Developments and Applications, M. Iri and K. Tanabe, eds., Kluwer Academic Publishers, Dordrecht, 1989, pp. 83–108.

[Griewank1991e] A. Griewank and G. F. Corliss, eds., *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, SIAM, Philadelphia, Penn., 1991.

[Griewank1996b] A. GRIEWANK, D. JUEDES, AND J. UTKE, *ADOL–C, a package for the automatic differentiation of algorithms written in C/C++*, ACM Trans. On Math. Software, (1996), p. to appear.

[Griewank1989b] A. GRIEWANK AND G. W. REDDIEN, *Computation of cusp singularities for operator equations and their discretizations*, Journal of Computational and Applied Mathematics, (1989), pp. 133–153.

[Griewank1995a] A. GRIEWANK AND J. UTKE, *Evaluating higher derivative tensors by forward propagation of univariate Taylor series*, Technical Report, TU Dresden, Institut für Wissenschaftliches Rechnen, Dresden, 1995.

[Guzman1978a] O. GUZMAN, *Kegelradgetriebe, Berechnung und Messung der Zahnflankengeometrie - Bestimmung der Eingriffsverhältnisse*, PhD thesis, RWTH Aachen, 1978.

[Hutschenreiter1996b] U. W. HUTSCHENREITER, *Zahnflankenberechnung für bogenverzahnte kegelräder mittels automatischen differenzierens*, Technical Report, TU Dresden, Institut für Geometrie, 1996.

[Litvin1969a] F. L. LITVIN, *Die Beziehungen zwischen den Krümmungen der Zahnoberflächen bei räumlichen Verzahnungen*, Zeitschrift für Angewandte Mathematik und Mechanik, (1969), pp. 685–690.

[Litvin1994a] ———, *Gear Geometry and Applied Theory*, Prentice Hall, Englewood Cliffs, N. J., 1994.

[Litvin1995a] ———, *Computerized determination of curvature relations and contact ellipse for conjugate surfaces*, Computer Methods in Applied Mechanics and Engineering, (1995), pp. 151–170.

[Schriefer1983a] H. SCHRIEFER, *Verzahnungsgeometrie und Laufverhalten bogenverzahnter Kegelradgetriebe*, PhD thesis, RWTH Aachen, 1983.

[Schuster1996a] K. SCHUSTER, *Berechnung von Randfaltungen glatter Vektorfunktionen mit Anwendung auf Zahnradflanken*, diplomarbeitt, Institut für Wissenschaftliches Rechnen, TU Dresden, 1996.

[Schwetlick1991a] H. SCHWETLICK AND H. KRETZSCHMAR, *Numerische Verfahren für Naturwissenschaftler und Ingenieure*, Fachbuchverlag Leipzig, 1991.

[Stadtfeld1993a] H. J. STADTFELD, *Handbook of Bevel and Hypoid Gears*, Rochester Institute of Technology, 1993.

[Weck1987a] M. WECK ET AL., *Das lauf- und beanspruchungsverhalten bogenverzahnter kegelradgetriebe*, in Arbeitstagung Zahnrad- und Getriebuntersuchung, WZL TH Aachen, 1987.

[Wildgrube1984a] E. WILDGRUBE, *Geometrische Grundlagen und Rechnergestützte Auswertung von Dreikoordinaten-Messungen an Zyklo-Palloid-Kegelrädern*, PhD thesis, MLU Halle-Wittenberg, 1984.