



HAL
open science

Optimization of 5-axis high-speed machining using a surface based approach

Sylvain Lavernhe, Christophe Tournier, Claire Lartigue

► **To cite this version:**

Sylvain Lavernhe, Christophe Tournier, Claire Lartigue. Optimization of 5-axis high-speed machining using a surface based approach. *Computer-Aided Design*, 2008, 40 (10-11), pp.1015-1023. 10.1016/j.cad.2008.08.006 . hal-00373701

HAL Id: hal-00373701

<https://hal.science/hal-00373701>

Submitted on 7 Apr 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Optimization of 5-axis high-speed machining using a surface based approach

Sylvain Lavernhe¹, Christophe Tournier¹, Claire Lartigue^{1,2}*

¹ ENS de Cachan

² IUT de Cachan – Université Paris Sud 11

Laboratoire Universitaire de Recherche en Production Automatisée

ENS de Cachan – Université Paris Sud 11

61 avenue du Président Wilson

94235 Cachan cedex – France

name@lurpa.ens-cachan.fr

Abstract: This paper deals with optimization of 5-axis trajectories in the context of high-speed machining. The objective is to generate tool paths suited to high speed follow-up during machining in order to respect cutting conditions while ensuring the geometrical conformity of the machined part. For this purpose, the optimization of the tool axis orientations is performed using a surface model for the tool path which allows integrating kinematical limits of the machine tool as well as classical geometrical constraints. The illustration of the optimization through an example highlights the gain in machining time, thereby demonstrating the feasibility of such an approach.

Keywords: Multi-axis machining, inverse time, tool path optimization, kinematical performance, surface model

* Corresponding author

1. INTRODUCTION

Within the fields of automotive and aeronautics, 5-axis High Speed Machining (5-HSM) becomes nowadays a competitive process for the elaboration of sculptured surfaces. The objective is to obtain a part which respects the geometrical specifications with regard to given productivity criteria.

5-HSM process consists of various activities defining a digital chain (Fig. 1): definition of a CAD model, generation of the tool trajectory from the CAD model (CAM activity), transformation of the data (post-processing activity), driving and monitoring of the process, etc.

Fig. 1. 5-axis digital process.

Many technological difficulties can be highlighted at each stage of the digital chain. CAM activity consists in calculating the trajectory of the tool tip from the CAD model. The result must be a collision free trajectory with optimized tool/surface positioning in order to guarantee the conformity of the part with respect to the required quality [1]. The calculated tool path is a series of C_L points (Cutter Location points) and corresponding tool orientations, plus various machining parameters defining the CL file. As this format is not recognized by the Numerical Control unit which only interprets "G-codes" (standard ISO 6983), a post-processing stage is necessary to convert the calculated tool path into an adapted file for the Numerical Controller (NC). This file, called the NC file, contains the set of tool postures (tool positions and tool axis orientations) and the corresponding feedrates. In 5-axis machining, there is no direct correspondence between the part coordinate system (PCS) and the joint space of the machine tool. As the tool path is calculated in the PCS, tool postures are expressed in the joint space via the Inverse Kinematical Transformation (IKT) which transforms the

tool path into direct axis commands. The IKT is most generally carried out during the post-processing stage. The NC unit thus performs the tool path interpolation and the trajectory follow-up. However, the actual axis behavior is altered by physical limitations such as kinematical capacities, cycle times of the NC, as well as by numerical problems linked to the IKT (singularities, multiple joint solutions, etc.) [2][3][4]. Hence, from the CAM stage to actual machining, numerous parameters influence performance in 5-axis HSM affecting machining time as well as surface quality [5][6].

Usually, the tool path optimization is performed by optimizing the two angles that define the tool axis orientation. Methods have evolved from finding the best tool positioning so that gouging and collisions are avoided based on concepts of differential geometry such as local curvature properties [1][7][8][9][10], to methods that also include axis tool orientation smoothing [10][11][12][13][14]. To avoid abrupt orientation changes and large cutting errors, Ho et al [10][11] have developed a procedure which couples the tool orientation smoothing method (TOS method) to the cutting improvement method. The TOS method relies on a quaternion interpolation algorithm widely used in robotics. In their approach, Jun et al [12] defined a machining configuration space (C-space), which corresponds to the parameter space of the axis orientations defined by the two angles (α , β). For each C_C point (Cutter Contact point), a feasible region is built in the C-space bounded by geometric and machining constraints. To minimize the cusp heights while avoiding gouging and collision, authors showed that the optimal orientation lies on the boundary of the feasible region. The smoothing is thus carried out in the C-space by determining the shortest C-distance between the previous optimal tool orientation and the next candidate. Another way to optimize the tool-path is to consider the geometry and architecture of the machine tool [15][16][17][18]. Considering the interpolation process of the NC unit, kinematical

errors and local behavior near singularity points can be taken into account during the CAM stage or the post-processing operation. By this way, the trajectory in the joint space of the machine is controlled.

These methods of tool path optimization rely on a geometrical approach, essentially based on the CAD model and geometrical parameters (surface curvatures, tool geometry and sometimes machine-tool geometry and architecture). In the context of High Speed Machining, actual velocities seldom match the programmed ones. Therefore, when the geometrical conformity of the part is reached, the objective becomes to control trajectory follow-up during machining, which means actual local feedrate and trajectory traveling time. In this direction, Kim et al. [19] introduced the concept of time-optimal tool paths. They proposed a method that machines an entire surface as quickly as possible while respecting both geometrical specifications and kinematical limits of the machine tool. The “greedy approach” they developed consists in finding the directions of maximal material removal, directions to which a vector field is fitted. Tool paths are thus modeled as streamlines of this vector field. However, the method only considers the motors speed limits. Farouki et al [4] proposed an interesting approach which takes into account limiting constraints such as the maximal torque and the maximal power of the machine tool in the calculation of the tool path feedrate. An illustration is presented within the context of 3-axis machining. Acceleration and power constraints expressed in the part frame directly correspond to axis constraints. Therefore, inertia and cutting forces can be taken into account. From the previous works on feedrate optimization [20][21]. Sencer et al [22] proposed a method which determines for a given trajectory the maximal feedrate that can be locally programmed and reached during machining while respecting maximum velocities, accelerations and jerks of drives. The velocity profile is expressed as a B-Spline curve.

In their study, López de Lacalle et al [23] propose to use the estimation of deflection forces as a criterion for the best choice of tool paths. It is then possible to select tool orientations which produce low deflection forces respecting geometrical requirements.

Few authors include specific constraints linked to HSM in tool trajectory calculation and optimization. The best use of 5-axis HSM requires determining the most adapted tool trajectories to ensure geometrical specifications while minimizing machining time. This leads to an optimization problem for which all the phenomena presented above must be taken into account in tool path calculation. In previous works, we have proposed to represent the tool path using a surface model, the Machining Surface (MS) [7][24]. The MS is a surface including all the information necessary for the driving of the tool, so that the envelope surface of the tool movement sweeping the MS gives the designed surface. In 5-axis milling, the MS consists of two surfaces. The first one ensures the respect of the geometrical specifications by the exact positioning of the tool on the designed surface. The second surface permits to manage the tool axis orientations taking into account different kinds of constraints. In this context, we address the issue of the optimization of tool axis orientation integrating kinematical constraints. Indeed, this paper aims at finding optimized axis orientations so that actual feedrate is maximized during trajectory follow-up. The proposed approach is related to 5-axis finishing of sculptured surfaces using ball-end tool or filleted-end tool. For this purpose, an analysis of the limiting factors which affect trajectory follow-up is necessary, particularly within the context of HSM. Using the inverse time formalism, these limiting factors can be expressed as constraints for the optimization problem. Therefore, the paper is organized as follows: section 2 presents specific limits linked to 5-axis HSM altering actual feedrates; section 3 deals with the Machining Surface model and the formulation of the tool axis optimization problem; section 4 is dedicated to the illustration of the efficiency

of our method through an example.

2. SPECIFIC LIMITS IN 5-AXIS HSM

Basically, the most influencing factors affecting 5-axis HSM performance are the cutting conditions, the part set-up, the machining strategy and the fitting of the set {machine tool, NC unit}. As far as kinematics performance is concerned, both the machine tool and the NC unit play a major role, in particular in the trajectory follow-up. Hence, the study proposed next explores more particularly the influence of the set {machine tool, NC unit} on the trajectory follow-up within the context of 5-axis HSM. The geometry of the machine tool is assumed to be perfect.

In 5-axis machining, the trajectory follow-up requires a coordinated movement of all the 5 axes of the machine according to the axis orders which are calculated in real-time by the NC. As stated above, the actual feedrate seldom matches the programmed ones. Indeed, the kinematical behavior of the set {machine tool, NC unit} is limited by the characteristics of the machine itself, the characteristics of the NC unit and by the HSM specific functions of the NC. The study proposed in the paper focuses on a RRTTT type of machine architectures. Obviously, other types of machine tool configurations could be similarly studied.

As the tool path is defined by a set of tool positions (X_{pr}, Y_{pr}, Z_{pr}) and tool axis orientations (i, j, k) , corresponding axis configurations $(P^{axis1}, P^{axis2}, P^{axis3} \dots)$ are calculated via IKT.

$$\left(P^{axis1}, P^{axis2}, P^{axis3}, P^{axis4}, P^{axis5} \right) = IKT \left(X_{pr}, Y_{pr}, Z_{pr}, i, j, k \right) \quad (1)$$

In addition, axis velocities are computed from axis configurations taking the programmed feedrate, V_{prog}^f into account. Fig. 2 shows a segment of the trajectory defined by two successive tool postures. If the length of the segment is L_{12} , the tool is

supposed to be moving during ΔT_{12} from one configuration to the second one with a constant feedrate:

Fig. 2. Trajectory segment in PCS.

$$\Delta T_{12} = \frac{L_{12}}{Vf_{prog}} \quad (2)$$

Hence, the velocity V_{f12}^i of each axis i to cover the segment $P_1^i P_2^i$ in the MCS is given by:

$$V_{f12}^i = \frac{P_2^i - P_1^i}{\Delta T_{12}} = \frac{P_2^i - P_1^i}{L_{12}} \cdot Vf_{prog} \quad (3)$$

At this stage, various factors act as limits and lead to velocity decreasing.

- Axis kinematical limits

As axes are made of mechanical components, they have physical limits. In order to preserve the mechanical components, the NC reduces the maximal values that can be reached through numerical limits. Each axis i thus possesses limits of its kinematical characteristics: maximal velocity, V_{max}^i , maximal acceleration A_{max}^i , and maximal jerk, J_{max}^i . Therefore, during the follow-up of the theoretical trajectory, axes are managed so that these numerical limits are respected. This acts as a limiting factor for axis coordination in the MCS.

- Limits linked to NC cycle time

Between two tool configurations, the NC unit needs at least one interpolation cycle time to calculate axis orders. For small length segments and high-programmed feedrates, the NC reduced the programmed feedrate in order to satisfy this cycle time [25] according to the following equation:

$$V_{max}^{NC} = \frac{L_{12}}{T_{cycle\ time}} \quad (4)$$

The actual feedrate is thus locally lower than the programmed one due to the interpolation cycle time. Fig. 3 shows results highlighting velocity reduction for different small length segments. Indeed, let us consider a trajectory defined as succession of segments along the X-axis: a large segment P_1P_2 (100mm), a small one P_2P_3 (variable length from 3 to 0.5mm), a large one P_3P_4 (100 – P_3P_4 mm). To travel this trajectory, the programmed feedrate is set to 10m/min. If the cycle time $T_{cycle\ time}$ is equal to 12ms, only segments the length of which is greater than 3mm can be traveled at the programmed feedrate. As illustrated in Fig. 3, the registration of the X-axis velocity brings out a velocity reduction when traveling a 2mm-length segment.

Fig. 3. Influence of small length segments on actual feedrate during follow-up.

- Limits linked to specific functions of the NC unit

Various limits have here to be taken into account within the context of HSM. A recurrent difficulty in HSM is the management of discontinuities in the joint space of the machine appearing at block transitions. Tangency discontinuities are the most critical ones. Passing exactly through these discontinuities with a non-null feedrate would require infinite acceleration on each axis which is physically not possible. Rounding tolerances are thus introduced to improve the follow-up, while controlling the geometrical deviation to the trajectory. As a result, the velocity must be adapted which may cause slow-downs.

The combination of multi axes in the context of HSM imposes to use CNC functions the choice of which results of a compromise between productivity and quality. For instance, when using a NC unit Siemens 840D, it is required to use the Soft mode for which the acceleration motion profile is trapezoidal [25]. Although it is slower than the classical trapezoidal velocity motion profile, such a mode permits to preserve mechanical components while confining errors to the trajectory.

In the same way, in order to control axis behavior, error management is performed in the joint space (MCS) that means for each axis.

Finally, for each segment trajectory, the velocity is limited by the minimum value of the constraints. This yields to:

$$\begin{cases} 0 \leq v_{12}^i \leq \min(V_{f12}^i, V_{\max12}^{NC}, V_{\max}^i) \\ -A_{\max}^i \leq a_{12}^i \leq A_{\max} \\ -J_{\max}^i \leq j_{12}^i \leq J_{\max}^i \end{cases} \quad (5)$$

As axis capacities are different, the follow-up of the trajectory is limited by the less powerful axis. Moreover, the nature of rotation and translation movements is different which makes difficult a direct comparison of axis capacities. To overcome this difficulty, we have proposed in previous works to express axis kinematical capacities using the inverse time method [5][6]. With such formalism, it becomes possible to find the kinematical limits and to express them as limiting constraints in the optimization problem as we will see in section 3.3.

3. OPTIMIZATION USING A SURFACE BASED APPROACH

3.1. Location in the digital chain

The optimization method we propose is located at the interface between the CAM stage and the NC processing stage (Fig. 4). Indeed, the method requires the definition of an initial trajectory calculated by the CAM software, independently of the NC unit. This initial solution is the support of the optimization process, which relies on the following information concerning the machine tool:

- Kinematical model of the machine tool (IKT)
- Kinematical limits of each axis and of the NC unit
- Part setup onto the machine tool table

Obviously, parameters such as the tool geometry, the machining tolerance, the feedrate and the scallop height allowed are already set in the initial CAM stage.

Fig. 4. Modification of the digital chain.

Two different ways or points of view can be adopted to implement the optimization. The first one simply consists in extending input data at the CAM stage. A simulation stage including axis capacities and NC limits can thus be carried out, and the trajectory can afterwards be modified to avoid collisions, or to answer kinematical criteria. In this case, the optimization can be executed in a post-processing stage to dedicate the calculus to a specific machine.

Considering the second point of view, a part of the calculation stage is transferred within the NC unit itself. This can be enabled by selecting an optimization option. The main drawback is that more computational capacities are required. Furthermore, trajectory simulation can only be carried out on the machine.

This optimization process especially suits the STEP-NC project [26][27] to promote the exchange of information between CAD / CAM / NC systems and provide a better integration and interoperability of data.

3.2. The surface model for tool path computation

The 5-axis HSM optimization relies on a surface description of the tool path: the trajectory of a particular point of the tool is expressed as a surface or as a set of surfaces [7][24]. The surface model, also called the “Machining Surface” (MS), ensures the respect of functional requirements and design intents while integrating machining constraints. Concerning the general case of 5-axis machining in point milling using a filleted-end tool, the calculation of the tool trajectory requires the definition of two parameterized surfaces (Fig. 5): the guiding surface S_G and the orientation surface S_O .

The guiding surface ensures tool positioning whatever the machining strategy, whereas the axis tool orientation is independently managed by the orientation surface.

Let r and R be respectively the corner radius and the radius of the tool (Fig. 5). If C_C is the contact point between the tool and the surface to be machined, K is defined as the offset point of C_C by a distance value equal to r . The triplet $(K;C_L;\mathbf{n})$, where \mathbf{n} is the contact normal, allows the positioning of the tool. The guiding surface is thus the locus of the K point, whereas the orientation surface is the locus of the C_L point.

Fig. 5. Definition of the Machining Surface (MS) in 5 axes.

With such modeling, the tool path is described as a continuous surface which contains more information than the classical model made up of a set of ordered points. The MS not only ensures the continuity of the tool paths but it also uncouples functional requirements and dynamical requirements. The calculation of the trajectory using the MS can be divided into the following stages:

- Calculation of the guiding surface (S_G) from the CAD model (S_D):

S_G is simply obtained by offsetting the surface to be machined $S_D(u, v)$ by a distance equal to the corner radius r where $\mathbf{n}(u, v)$ is the normal to the surface:

$$S_G(u, v) = S_D(u, v) + r \cdot \mathbf{n}(u, v) \quad (6)$$

- Calculation of the orientation surface (S_O) according to the machining strategy and the tool orientation management:

S_O is calculated as the generalized offset of the surface:

$$S_O(u, v) = S_D(u, v) + r \cdot \mathbf{n}(u, v) + (R - r) \cdot \mathbf{v}(u, v) \quad (7)$$

- Calculation of the tool postures according to the CAM parameters.

As the guiding surface is independent of the machining strategy, the trajectory of the tool tip is defined as sets of curves $\{S_G(u(s), v(s)), S_O(u(s), v(s))\}$ in the parametric

space of the guiding surface. Therefore, tool paths exactly belong to the guiding surface. When the tool moves along a curve $S_G(u(s), v(s))$, the contact point is exactly positioned on the surface to be machined. Hence, the guiding surface guarantees the conformity of the part as regards the geometrical specifications. The orientation surface defines the tool axis orientation according to the machining strategy. It becomes thus possible to optimize the position of S_O so that constraints can be taken into account. In particular, constraints linked to 5-axis HSM can be integrated into the optimization of the tool trajectory.

3.3. Optimization of the tool orientation

Within the context of finish machining, the optimization of the tool axis orientation is expressed in a generic way for a filleted-end tool as follows (the case of a ball-end tool corresponds to $R=r$):

“Considering the trajectory defined in the parametric space, $(u(s), v(s))$, find the best orientation surface $S_O(u(s), v(s))$ (i.e. the best tool orientations, $\theta_n(s)$ and $\theta_t(s)$) that provides minimum machining time T_m , while respecting geometrical and kinematical constraints.”

Minimizing machining time while respecting the programmed feedrate V_{fprog} leads to minimize the difference between the actual feedrate $v_f(s)$ and the programmed one all trajectory long:

$$\min(T_m) \approx \min \left(\int_0^L \frac{1}{\|V_{fprog} - v_f(s)\|} \cdot ds \right) \quad (8)$$

Machining time minimization involves the instantaneous feedrate, the length of the trajectory and real-time parameters associated to the trajectory processing. It is clear that performance optimization is strongly linked to the machine tool architecture and the

NC unit capacities. At this stage, we have to express $v_f(s)$ in function of the trajectory $(u(s), v(s))$. Choosing the piloted point C_L , describing the orientation surface, this yields to:

$$\begin{cases} v_f(s) = \left\| \frac{d}{dt} C_c(u, v) \right\| = \left\| \frac{d}{dt} S_G(u, v) - r \cdot \mathbf{n}(u, v) \right\| \\ S_O(s) = S_G(u, v) - R \cdot \mathbf{v}(u, v, \theta_n, \theta_t) \end{cases} \quad (9)$$

where the unknowns are $s(t)$, $\theta_n(s)$ and $\theta_t(s)$. The crux here is the relationship between time and position over the trajectory: $s(t)$. As this relationship is physically carried out in real-time during trajectory processing, it is difficult to be expressed in a mathematical form. It is also important to notice that during trajectory processing, axis movements obey the law of movements imposed by jerk piloting mode:

$$p^i(t) = p_0^i(t) + v^i(t) \cdot t + \frac{1}{2} \cdot a^i(t) \cdot t^2 + \frac{1}{6} \cdot j^i(t) \cdot t^3 \quad (10)$$

where kinematical parameters are constraint as stated in eq.(5).

If we focus now on the geometrical constraints, we have to consider the following ones:

- No collision, no gouging
- Cusp height limit

The first constraint imposes orientation parameters to be included within an admissible space (similar to the C-space [12]). The second one implies the respect of the maximum cusp height allowed, h_{max} , in order to ensure part conformity.

In its general form, the problem of optimizing S_O is very complex. The main difficulty is to establish the relationship between the tool position and the time parameter integrating constraints linked to the trajectory processing (maximal kinematical performance, cycle time, look ahead functions...) which makes it difficult to express such a relationship whether explicitly or implicitly.

Hence, we propose to simplify the problem by only considering one pass at a time. The main idea is to make the best use of the axes, according to their kinematical characteristics.

3.4. Optimization of the tool orientation: simplified problem

At this stage, we suppose that axis positions P^i are calculated according eq.(1) for each tool posture $(X_{pr}, Y_{pr}, Z_{pr}, i, j, k)$.

During trajectory processing, position orders are coordinated to ensure the respect of several constraints (eq.(5)). As axes are different, they are solicited differently; hence, trajectory follow-up is limited by the less dynamic axis with regards to the solicitations. As movements are of different nature, axes can not be directly compared. For this purpose, we propose to express kinematical characteristics ($V \sim s^{-1}$, $A \sim s^{-2}$, $J \sim s^{-3}$) using the inverse time formalism which simply traduces the coordination between axes.

Let us consider the movement of the axis i from the position P_1^i to the position P_2^i . The axis displacement from one configuration to the other one is:

$$\Delta P_{12}^i = P_2^i - P_1^i \quad (11)$$

By assuming that the interpolation between these two positions is linear in the joint space, the current position is given by:

$$P_{12}^i = \Delta P_{12}^i \cdot \alpha^i \quad \alpha^i \in [0,1] \quad (12)$$

where α^i is the fraction of total displacement between P_1^i and P_2^i , which corresponds to the expression of the current position in the inverse time formalism:

$$\hat{p}_{12}^i = \frac{P_{12}^i}{\Delta P_{12}^i} = \alpha^i \quad (13)$$

The coordination of all the axes into the joint space implies that the fraction of displacement is considered equal for each axis:

$$\hat{p}_{12}^{axis 1} = \hat{p}_{12}^{axis 2} = \dots = \hat{p}_{12} \quad (14)$$

We can express in such a way other kinematical parameters:

$$\begin{cases} \hat{v}_{12} = \frac{v_{12}^i}{\Delta P_{12}^i} \\ \hat{a}_{12} = \frac{a_{12}^i}{\Delta P_{12}^i} \\ \hat{j}_{12} = \frac{j_{12}^i}{\Delta P_{12}^i} \end{cases} \quad (15)$$

Equation (10) becomes:

$$\hat{p}(t) = \hat{p}_0(t) + \hat{v}(t) \cdot t + \frac{1}{2} \cdot \hat{a}(t) \cdot t^2 + \frac{1}{6} \cdot \hat{j}(t) \cdot t^3 \quad (16)$$

In which the kinematical parameters are constraint as stated in equation (5), which is now expressed as follows:

$$\begin{cases} 0 \leq \hat{v}(t) \leq \min(\hat{V}_{\max}, \hat{V}_{prog}, \hat{V}_{NC}) \\ -\hat{A}_{\max} \leq \hat{a}(t) \leq \hat{A}_{\max} \\ -\hat{J}_{\max} \leq \hat{j}(t) \leq \hat{J}_{\max} \end{cases} \quad (17)$$

with:

$$\begin{cases} \hat{V}_{\max} = \min_{i=1..N} \left(\frac{V_{\max}^i}{\Delta P^i} \right); \hat{A}_{\max} = \min_{i=1..N} \left(\frac{A_{\max}^i}{\Delta P^i} \right); \hat{J}_{\max} = \min_{i=1..N} \left(\frac{J_{\max}^i}{\Delta P^i} \right) \\ \hat{V}_{NC} = \frac{1}{T_{cycle\ time}} \\ \hat{V}_{prog} = \frac{V_{prog}}{L_{i12}} \end{cases} \quad (18)$$

In order to minimize machining time, the proposed approach consists in finding the set of tool orientations so that the values of the three constraints \hat{V}_{\max} , \hat{A}_{\max} and \hat{J}_{\max} are reached. As maximizing velocity might not be similar to maximizing acceleration or jerk, we have to determine the most limiting kinematical characteristic. Hence, the optimization process consists of three main stages (Fig. 6).

Fig. 6. Optimization process structure.

First, a global optimization step is realized considering the constraint \hat{J}_{\max} .

Supposing that the jerk can take one of the following three values:

$$\hat{j}(t) = \begin{cases} -\hat{J}_{\max}^i \\ 0 \\ \hat{J}_{\max}^i \end{cases} \quad (19)$$

axis accelerations and velocities can be reconstructed according to a predictive model [5]. Then, new stages of optimization can be locally performed considering \hat{A}_{\max} and next \hat{V}_{\max} . Indeed, if acceleration saturations are detected, the orientation is locally modified so that the acceleration \hat{A}_{\max} is reached (step 2). The last step is similar for velocity: once velocity saturations are detected, tool orientations are modified along the trajectory portion which is concerned so that the velocity is increased. The process is iterative as the local modifications of velocity and acceleration may provide jerk modifications.

With this model, assuming that the constraints \hat{V}_{prog} and \hat{V}_{NC} are not reached in eq.(17), to each step of calculus corresponds one limitation: \hat{J}_{\max} , \hat{A}_{\max} or \hat{V}_{\max} .

Let consider \hat{Q}_{\max} a kinematical characteristic which can be velocity, acceleration or jerk expressed in its inverse time form. It can be projected onto the tool-path at the C_L point:

$$\begin{cases} Q_{\max} = \hat{Q}_{\max} \cdot \Delta C_L \\ \hat{Q}_{\max} = \min_{i=1..N} \left(\frac{Q_{\max}^i}{\Delta P^i} \right) \end{cases} \quad (20)$$

As the inverse time formalism is used, the optimization problem can be written in a similar way whatever the kinematical characteristic.

Therefore, the objective function to be maximized is:

$$F_Q = \int_0^L Q_{\max}(s) \cdot ds \quad (21)$$

If $(u(s), v(s))$ defines the trajectory in the parametric space, the new formulation of the optimization problem is summarized in eq.(22). The aim is to find optimal evolution laws of the yaw and tilt angles $(\theta_n(s), \theta_t(s))$ all along the trajectory. As tool paths are considered one by one, geometrical constraints (no gouging, maximal cusp height) are expressed as limitations on both the yaw and tilt angles.

$$\left\{ \begin{array}{l} \text{Finding} \quad (\theta_n(s), \theta_t(s)) \quad (\text{or the } \mathbf{S}_O(u(s), v(s))) \\ \text{so that} \quad : \quad \max(F_Q) \\ \text{subject to} \\ \quad \theta_{t \min} \leq \theta_t(s) \leq \theta_{t \max} \\ \quad \theta_{n \min} \leq \theta_n(s) \leq \theta_{n \max} \end{array} \right. \quad (22)$$

The function F_Q must provide the maximum speeds achievable along the trajectory according to the orientation laws of the tool axis. Fig. 7 illustrates the construction process of the function F_Q .

The first step is the calculation of the tool trajectories from the Machining Surface (eq.(6) and eq.(7)). The tilt and yaw angles are chosen constant in this first stage. They define the initial solution of the optimization problem. Considering the surface, these values can initially be chosen so that the machined width strip is maximized or to respect the maximal cusp height h_{\max} . For each tool positioning defining a single pass calculated in the parametric space of the guiding surface, coordinates of the tool tip and vector cosines $(X_{pr}, Y_{pr}, Z_{pr}, i, j, k)$ are built in the PCS. The second stage concerns the evaluation of the kinematical constraints. For this purpose, the joint space trajectory is constructed based on the IKT and the geometrical sampling model (eq.(1)). Then, constraints are determined on the joint trajectory (eq.(18)). In function of the stage which is considered in the optimization process, the associated characteristic Q , are summed for the whole single pass (eq.(20) and eq.(21)).

Fig. 7. Construction process for the cost function.

This non-linear optimization problem is solved using a sequential quadratic programming (SQP) method. To ensure that the computed solution is close to the global one, the problem is solved several times with different initial solutions $(\theta_n(s), \theta_t(s))$ by sampling the domain $[\theta_{n \min}(s), \theta_{n \max}(s)] \times [\theta_{t \min}(s), \theta_{t \max}(s)]$.

To illustrate our purpose, the proposed approach is applied to the machining of a hyperbolic paraboloid.

4. APPLICATION

The approach is here conducted on a pass, considering that the analysis can be repeated throughout the whole machining tool path. In this example, we have directly modeled the guiding surface as a hyperbolic paraboloid as a Bezier patch (Fig. 8).

The patch is machined using parallel planes guiding strategy so that the tool follows the rules of the surface. Hence, tool paths are lines in the PCS. Initially the tilt angle θ_t and the yaw angle θ_n are set respectively to 1° and 0° . The tool radii are $R=5\text{mm}$ and $r=1.5\text{mm}$.

Experiments are carried out using the milling centre of our laboratory, a MIKRON UCP710 with a RRTTT kinematics equipped with a NC unit Siemens 840D [5][6].

The programmed feedrate is set to 5m/min . The IKT is performed in real time by the NC unit. The part set-up is chosen so that the frame of the part system (X_{pr}, Y_{pr}, Z_{pr}) corresponds to the frame of the machine tool (X_m, Y_m, Z_m) . Note that for this surface, all the 5 axes of the machine tool are solicited.

Fig. 8. Machining of the hyperbolic paraboloid.

We study the longest tool path, in the middle of the part. Fig. 9(a)(b)(c) show the

resulting maximal reachable feedrate, acceleration and jerk all path long given axis kinematical constraints before. These values correspond to the maximum values of the considered parameters which are reached given the less powerful axis for each block of the path. We observe that each of the three parameters reaches a minimum in the middle of the tool path, especially velocity and acceleration that are near zero.

Fig. 9. Resulting maximum velocity (a), acceleration (b), jerk (c) along the tool path.

Measured velocities on the 5 axes during machining are reported in Fig. 10. Results highlight the difficulties encountered by the C-axis during machining with this initial tool axis orientation. This axis is much more solicited compared to other axes. Indeed, in the middle of the tool path, we are close to the singular point of the machine tool kinematics.

Fig. 10. Measurements of axis velocities.

Hence, the actual relative feedrate tool/surface does not match at all the programmed one, especially in the middle of the path; cutting conditions are not respected and machining time is increased (Fig. 11).

Fig. 11. Initial and optimized relative feedrates.

Without loss of generality, we propose an illustration of the optimization approach based on the optimization of the velocity only. Therefore, the objective function according to eq. (21) can be written as follows:

$$F_Q = \int_0^L V_{\max}(s) \cdot ds \quad (23)$$

Let us now express the geometrical constraints for $(\theta_i(s), \theta_n(s))$. A preliminary analysis allows the determination of the admissible space of the angles so that gouging is avoided. As the shape of the surface is simple, the chosen values are constant all the

trajectory long:

$$\begin{cases} 0^\circ \leq \theta_t \leq 45^\circ \\ -90^\circ \leq \theta_n \leq 90^\circ \end{cases} \quad (24)$$

In the case of more complicated parts, a study based on the local curvature of the surface and on the possible tool accessibilities should be conducted to define more precisely the gouging-free admissible space. Solutions for $(\theta_t(s), \theta_n(s))$ are proposed in a polynomial form. The choice of the polynomial orders depends on the complexity of the surface. To let some degrees of freedom and avoid large movements of the tool axis orientation that can damage cusp height limit, the polynomial orders are set to the maximal order of the surface plus one.

Considering the studied surface, this yields to:

$$\begin{cases} \text{Finding} & \begin{cases} \theta_n(s) = a_0 + a_1 \cdot s + a_2 \cdot s^2 + a_3 \cdot s^3 \\ \theta_t(s) = b_0 + b_1 \cdot s + b_2 \cdot s^2 + b_3 \cdot s^3 \end{cases} \\ \text{so that} & \text{maximize}(F_Q) \\ \text{subject to} & \begin{cases} 0^\circ \leq \theta_t \leq 45^\circ \\ -90^\circ \leq \theta_n \leq 90^\circ \end{cases} \end{cases} \quad (25)$$

The resolution leads to the evolution of $(\theta_t(s), \theta_n(s))$ proposed in Fig. 12. The value of the yaw angle is quasi constant around -50° whereas the tilt angle varies from 40° to 45° .

Fig. 12. Optimized tilt and yaw angle.

Once the velocity is optimized for all axes, kinematical performances are clearly improved. This remark stands for the velocity, which is the objective function, but also for acceleration and jerk (Fig. 9). Indeed, axis displacements are completely different, rotary axes are less solicited (Fig. 10). The effective velocity along the tool path is quite equal to the programmed one (Fig. 11), which guarantees the respect of cutting conditions for the main part of the trajectory. The remaining difference between actual

optimized feedrate, 4.5m/min, and the programmed one, 5m/min, is due to short length segments and the constraint from NC unit cycle time for the interpolation (eq.(4)). But in this example, such short length segments are necessary to respect the chordal deviation. Finally, machining time of the optimized trajectory is three times shorter.

Nevertheless, considering the entire surface machining, the resulting tilt and yaw angles penalize the cusp height. Indeed, based on equation provided in [7], the cusp height is no more respected. If we consider the approximation of the effective profile of the filleted-end tool as a circle [28] we can roughly compute the necessary distance between guiding planes to maintain the same cusp height. In the worst case, for which θ_i equal 45° , the distance between guiding planes should be approximately ten times smaller, which significantly increases machining time. Then, in order to maintain both acceptable machining time and surface quality, it should be interesting to add the equivalent tool radius as a constraint in the optimization problem or to let degrees of freedom for the workpiece setup [29].

5. CONCLUSION

Within the context of multi-axis machining of complex surfaces, we have proposed to optimize tool trajectories and their follow-up during machining.

The optimization problem relies on the Machining Surface model for tool path which is of great interest in the generation of optimal trajectories. In 5-axis the Machining Surface consists of a set of two surfaces: the guiding surface and the orientation surface. The first one ensures the part geometrical conformity whereas the second one manages the tool axis orientations. As the surface model allows the uncoupling of geometrical and kinematical constraints, tool axis orientations can be optimized by optimizing the orientation surface. In the paper, we have focused on finding the best orientations so

that kinematical performances of the axes are optimized. For this purpose, kinematical constraints related to the set {machine tool, NC unit} are expressed as limiting constraints. The explicit formalization of the optimization problem being very complex, the proposed approach is simplified. The problem can be formulated as finding the set of tool orientations so that the values of the maximal kinematical parameters are reached. This optimization is implemented through an example validating its feasibility on a single pass.

- [1] Rao A, Sarma R. On local gouging in five-axis sculptured surface machining using flat-end tools. *Computer Aided Design* 2000;32:409-420.
- [2] Altintas Y. *Manufacturing Automation, Metal Cutting Mechanics, Machine Tool Vibrations and CNC Design*, Cambridge University Press, ISBN 0-521-65973-6, 2000.
- [3] Dugas A, Lee JJ, Hascoët JY. High Speed Milling–Solid simulation and machine limits, *Integrated Design and Manufacturing in Mechanical Engineering*, Kluwer Academic Publishers 2002;287-294.
- [4] Farouki RT, Tsai YF, Wilson CS. Physical constraints on feedrates and feed accelerations along curved tool paths. *Computer Aided Geometric Design* 2000;17(4):337-359.
- [5] Lavernhe S, Tournier C, Lartigue C. Kinematical performance prediction in multi-axis machining for process planning optimization. *International Journal of Advanced Manufacturing Technology* 2008;37(5-6):534-544.
- [6] Lavernhe S, Tournier C, Lartigue C. Model for performance prediction in multi-axis machining. *CIRP 2nd International Conference High Performance Cutting*, the University of British Columbia, Vancouver, Canada, 12-13 June 2006.
- [7] Duc E, Lartigue C, Tournier C, Bourdet P. A new concept for the design and the manufacturing of free-form surfaces: the machining surface. *Annals of the CIRP* 1999;48(1):103-106.
- [8] Lee YS. Admissible tool orientation control of gouging avoidance for 5-axis complex surface machining. *Computer-Aided Design* 1997;29(7):507-521.
- [9] Yoon JH, Pottmann H, Lee YS. Locally optimal cutting positions for 5-axis sculptured surface machining. *Computer-Aided Design* 2003;35(1):69-81.

- [10] Chiou CJ, Lee YS. A machining potential field approach to tool path generation for multi-axis sculptured surface machining. *Computer-Aided Design* 2002;34(5):357-371.
- [11] Ho MC, Hwang YR, Hu CH. Five-axis tool orientation smoothing using quaternion interpolation algorithm. *International Journal of Machine Tools and Manufacture* 2003;43(12):1259-1267.
- [12] Jun CS, Cha K, Lee YS. Optimizing tool orientations for 5-axis machining by configuration-space search method. *Computer-Aided Design* 2003;35(6):549-566.
- [13] Fleisig RV, Spence AD. Constant feed and reduced angular acceleration interpolation algorithm for multi-axis machining. *Computer-Aided Design* 2001;33(1):1-15.
- [14] Wang N, Tang K. Automatic generation of gouge-free and angular-velocity-compliant five-axis toolpath. *Computer-Aided Design* 2007;39(10):841-852.
- [15] Bohez ELJ. Compensating for systematic errors in 5-axis NC machining. *Computer-Aided Design* 2002;34(5):391-403.
- [16] Munlin M, Makhanov SS, Bohez ELJ. Optimization of rotations of a five-axis milling machine near stationary points. *Computer-Aided Design* 2004;36(12):1117-1128.
- [17] Sørby K. Inverse kinematics of five-axis machines near singular configurations. *International Journal of Machine Tools and Manufacture* 2007;47(2):299-306.
- [18] Makhanov S. Optimization and correction of the tool path of the five-axis milling machine. Part1. Spatial optimization. *Mathematics and Computers in Simulation* 2007;75:210–230.

- [19] Kim T, Sarma SE. Toolpath generation along directions of maximum kinematic performance; a first cut at machine-optimal paths. *Computer-Aided Design* 2002;34(6):453-468.
- [20] Erkorkmaz K, Altintas Y. High Speed CNC system design – Part I: Jerk limited trajectory generation and quintic spline interpolation. *International Journal of Machine Tools & Manufacture* 2001;41(9):1323-1345.
- [21] Altintas Y, Erkorkmaz K. Feedrate optimization for spline interpolation in high speed machine tools. *CIRP Annals* 2003;52:297-302.
- [22] Sencer B, Altintas Y, Croft E. Feed optimization for five-axis CNC machine tools with drive constraints. *International Journal of Machine Tools and Manufacture* 2008;48(7-8):733-745.
- [23] López de Lacalle LN, Lamikiz A, Sánchez JA, Salgado MA. Toolpath selection based on the minimum deflection cutting forces in the programming of complex surfaces milling. *International Journal of Machine Tools and Manufacture* 2007(2):388-400.
- [24] Tournier C, Duc E, Lartigue C, Contri A. The concept of the machining surface in 5-axis milling of free-form surfaces. *Integrated Design and Manufacturing In Mechanical Engineering*, Kluwer Academic Publishers; 2002, p. 279-286.
- [25] Siemens. Description of Functions - Sinumerik 840D/840Di/810D - Basic Machine 2004. www.automation.siemens.com/doconweb/.
- [26] Mueller P. STEP-NC new data interface for NC programming, *STEP-NC News Letter* 1, 2 & 3; 2000.
- [27] ISO 14649, draft international standard (ISO TC184/SC1 and ISO TC184/SC4)
- [28] Lo CC. Efficient Cutter-path planning for five-axis surface machining with flat-end cutter. *Computer-Aided Design* 1999;31(9):557-566.

- [29] Anotaipaiboon W, Makhanov SS, Bohez ELJ. Optimal setup for 5-axis machining. *International Journal of Machine Tools and Manufacture* 2006;46(9):964-977.

Figure 1

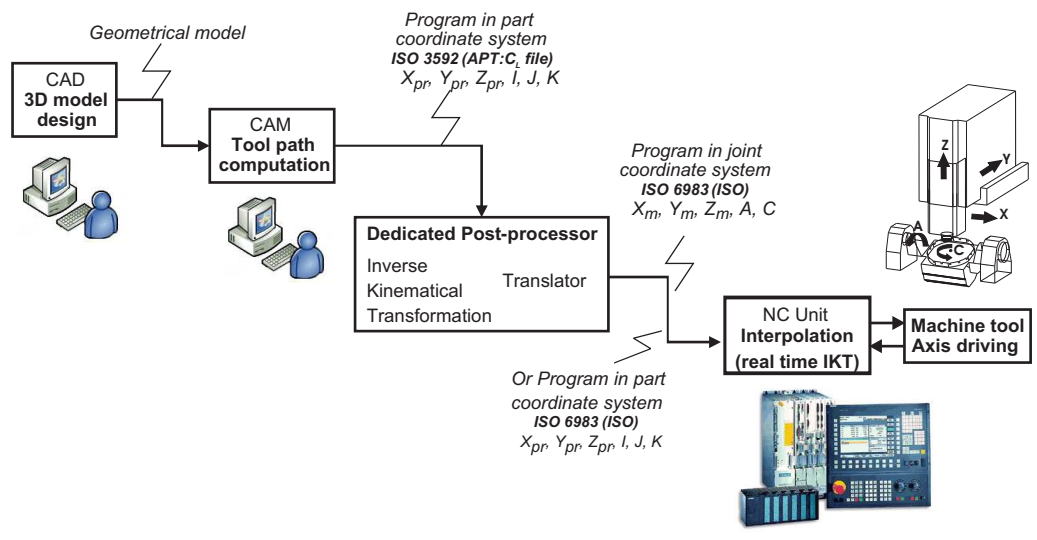


Figure2

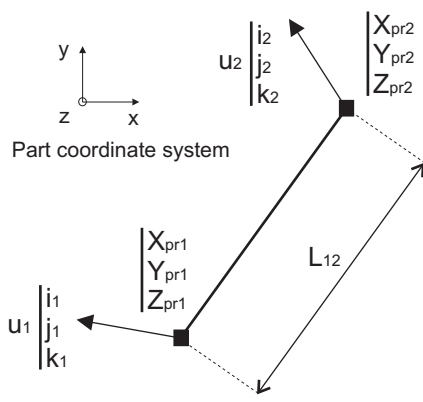
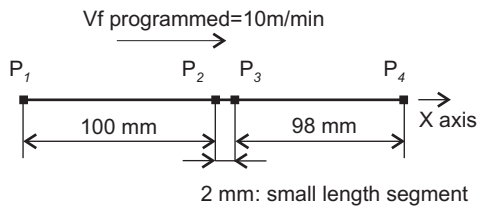


Figure3



segment length (mm)	3	2	1	0.1
measured local feedrate (m/min)	10	8.98	4.5	0.45

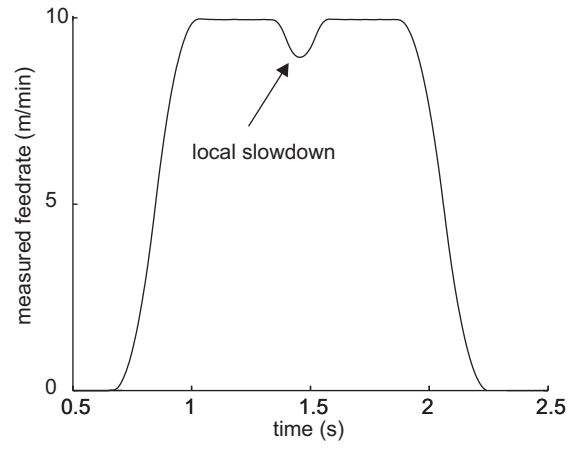


Figure4

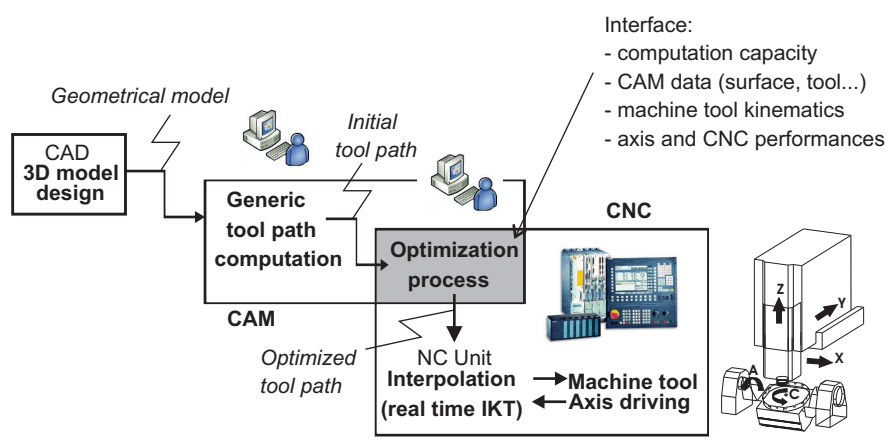


Figure 5

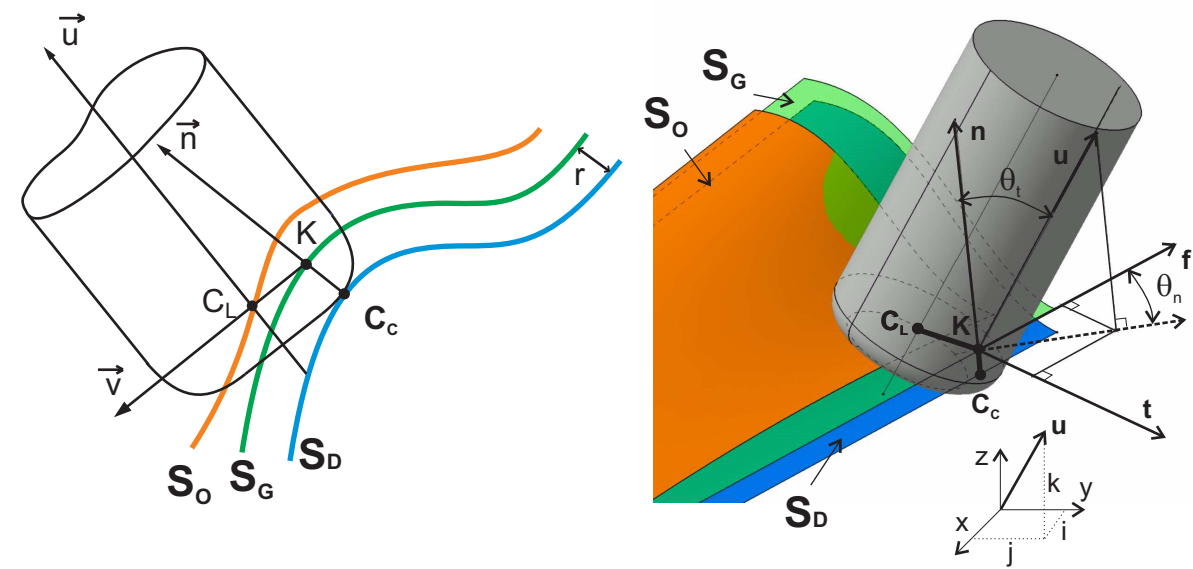


Figure 6

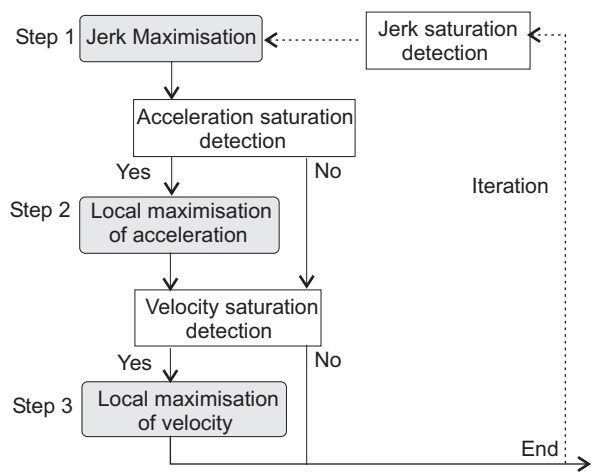


Figure 7

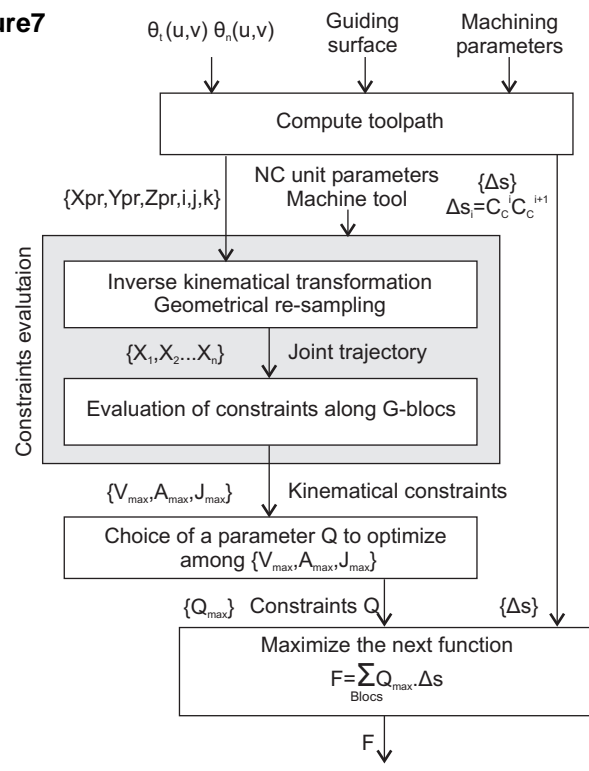


Figure 8

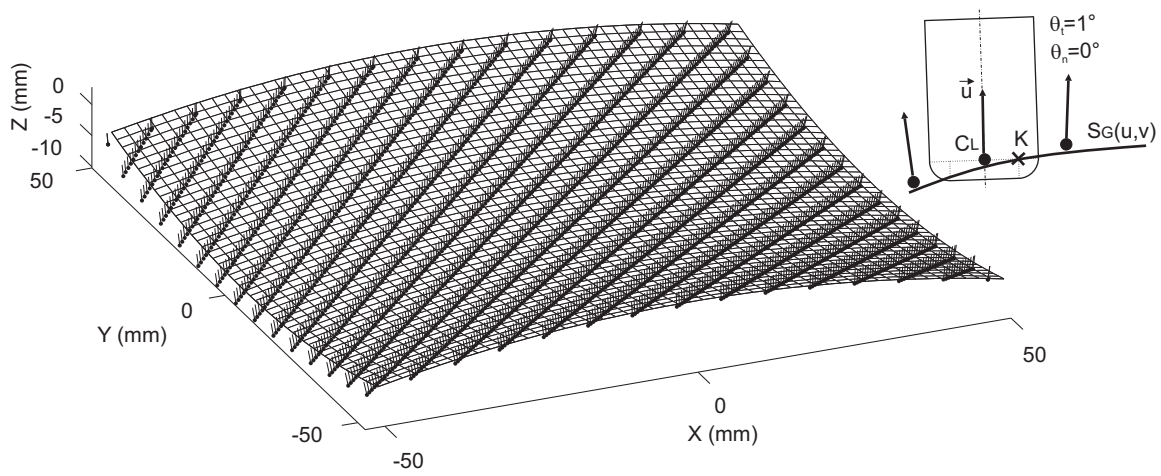
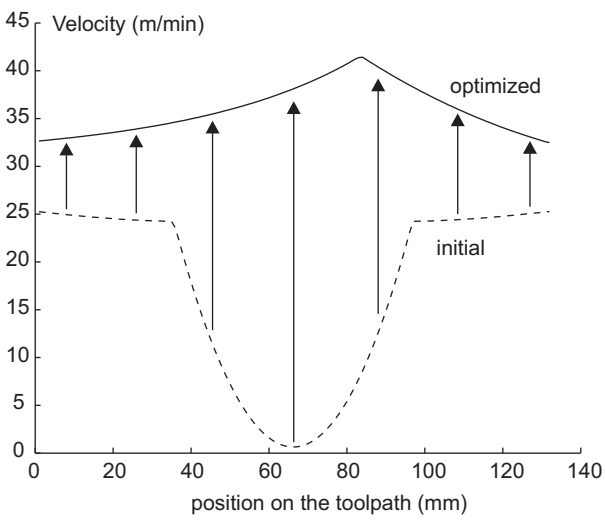
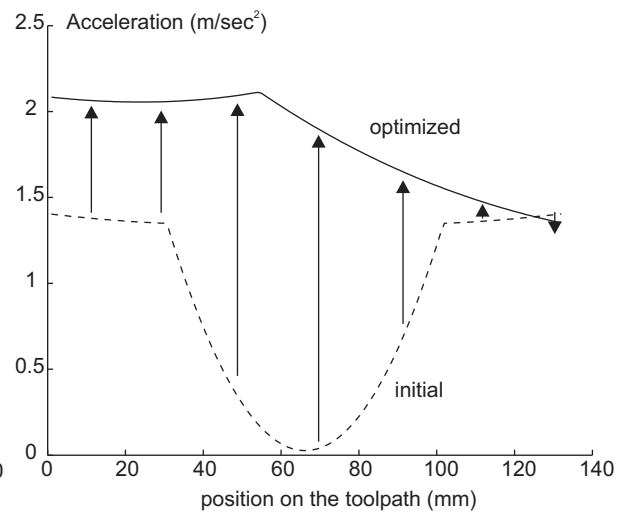


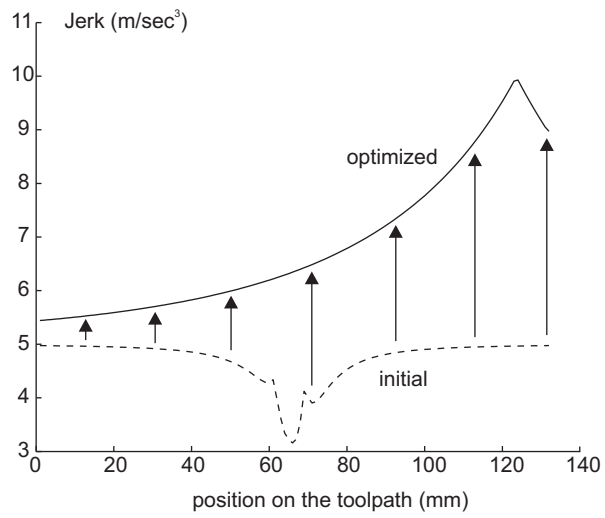
Figure9



(a)



(b)



(c)

Figure10

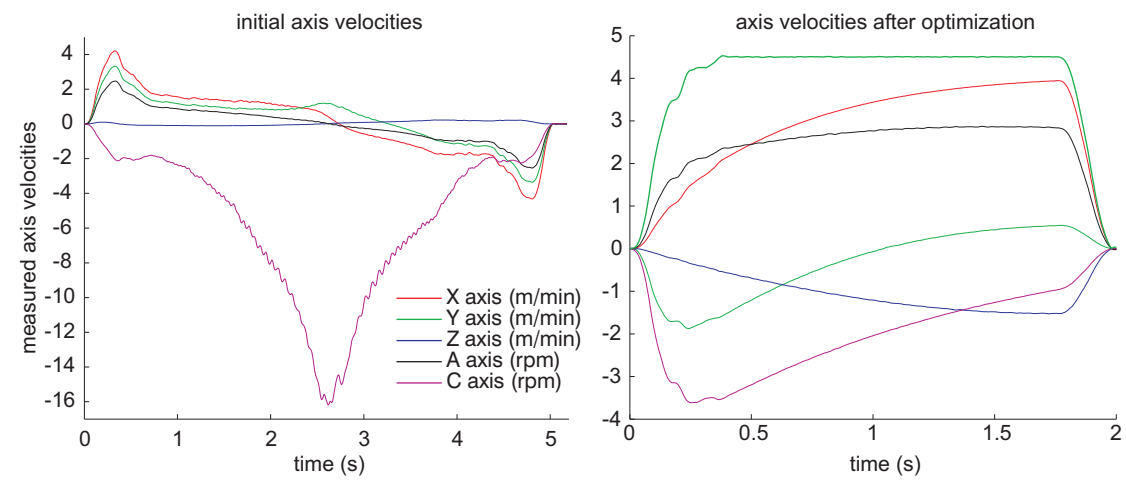


Figure11

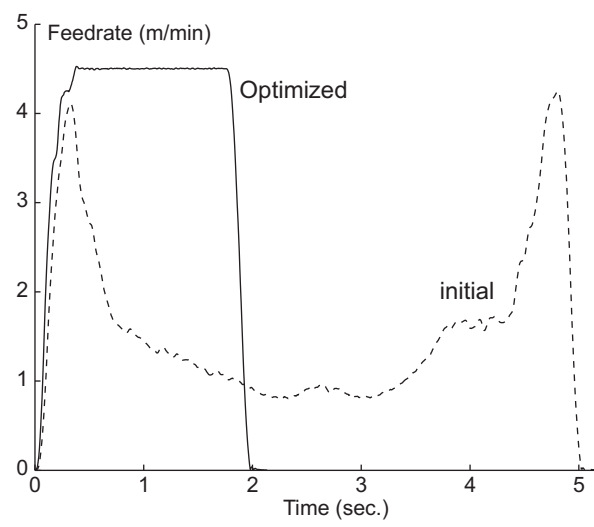


Figure12

