



**HAL**  
open science

## Contour error pre-compensation for five-axis high speed machining: offline gain adjustment approach

Tan-Quang Duong, Pedro Rodriguez-Ayerbe, Sylvain Lavernhe, Christophe Tournier, Didier Dumur

### ► To cite this version:

Tan-Quang Duong, Pedro Rodriguez-Ayerbe, Sylvain Lavernhe, Christophe Tournier, Didier Dumur. Contour error pre-compensation for five-axis high speed machining: offline gain adjustment approach. *International Journal of Advanced Manufacturing Technology*, 2018, 100, pp.3113-3125. 10.1007/s00170-018-2859-z . hal-01908400

**HAL Id: hal-01908400**

**<https://hal.science/hal-01908400v1>**

Submitted on 30 Oct 2018

**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.

---

# Contour error pre-compensation for 5-axis high speed machining: off-line gain adjustment approach

Tan-Quang Duong · Pedro Rodriguez-Ayerbe · Sylvain Lavernhe ·  
Christophe Tournier · Didier Dumur

**Abstract** This paper presents an Off-line Gain Adjustment (OGA) approach to reduce contour error in 5-axis high speed machining. The proposed contour error formulation is based on the estimation of tool contact points and the OGA is inspired from the idea of Model Predictive Control (MPC). The control gains used in the position loop of servo drives are optimally adjusted off-line to reduce the contour error for the considered trajectory. The obtained gain profiles are computed preserving axis kinematic limitations, stability criterion of servo drives and the motor current constraints. The OGA is developed thanks to a validated machine simulator. The simulation results prove that the OGA reduces significantly the contour error in 5-axis high speed machining.

**Keywords** Contour error · CNC · Off-line gain adjustment · Receding horizon · Model predictive control

## 1 Introduction

5-axis high speed machining is replacing the traditional 3-axis machining in sculpturing complex surfaces, such as impellers, molds, or aeronautic parts, etc. To enhance the performance of the process, computer processor, actuator and feedrate planning techniques have been much improved.

---

This work, N° 2014-812D Projet OMEGA, is supported by DIGITEO foundation, France.

Duong T-Q<sup>1</sup>, Rodriguez-Ayerbe P<sup>2</sup>, Lavernhe S<sup>3</sup>, Tournier C<sup>4</sup>, Dumur D<sup>5</sup>

<sup>1</sup>E-mail: Tanquang.Duong@centralesupelec.fr

<sup>2</sup>E-mail: Pedro.Rodriguez@centralesupelec.fr

<sup>3</sup>E-mail: Sylvain.Lavernhe@ens-paris-saclay.fr

<sup>4</sup>E-mail: Christophe.Tournier@ens-paris-saclay.fr

<sup>5</sup>E-mail: Didier.Dumur@centralesupelec.fr

<sup>1,2,5</sup>L2S, CentraleSupélec-CNRS-Univ. Paris-Sud, Université Paris-Saclay, 91190 Gif-sur-Yvette, France

<sup>1,3,4</sup>LURPA, ENS Paris-Saclay, Univ. Paris-Sud, Université Paris-Saclay, 94235 Cachan, France

However, its control structure and strategy have undergone less improvement. Each axis drive is controlled by Computer Numerical Control (CNC) through a classical cascaded structure, including position, velocity and current loops ranging from outer to inner loops respectively. While the two inner loops are controlled by the proportional - integral controllers, the external position loop is controlled by the proportional controller, combined with the feed forward action for velocity and current loops [1].

The general motivation is that if the control action performs really well, the finishing part will be very close to the desired one. Consequently, the hand finishing and polishing steps after machining can be eliminated or at least will require less efforts. Hence, the cutting quality, productivity, and economical benefit can obviously be achieved. To reach the desired accuracy and quality of the machined surfaces, all digital stages or physical phenomena have to be mastered. Digital chain covers CAD models for parts, 5 axis tool path strategies and definitions [2], post-processor and pre-processing issues [3], feedrate planning for real time interpolation [4]. Then, axis dynamics, actual behaviour of the machine tool geometry and cutting have to be considered previously through various models (machine geometry deviations, tool deflection, surface topography, etc.) [5]. The connection between the numerical part and the physical behaviour is made by the axis control ; it is therefore a critical point in the process.

This study mainly focuses on the improvement of the axis control stage, particularly the position controller of axis drive in the NC machining process. Considering that machining free-form surfaces is primarily concerned, there are many reasons saying that with the above classical axis control, it is difficult to obtain high contouring accuracy. On the one hand, in high speed machining, the high cutting speed leads to high feedrate. By interpolating such a feedrate, the required axis velocity, acceleration and jerk are also increased. Thus, by following the free-form profile the machine tool

axes may have high kinematic characteristics. On the other hand, the classical axis control structure has a limited bandwidth. Furthermore, machine tool contains uncertainty characteristics and disturbances, such as variation in drive parameters due to the varying workpiece mass and lubrication condition, coulomb friction and cutting force [6,7]. These detrimental facts increase tracking error, and generate large contour error, which is classically defined as the orthogonal distance between the machined and desired profiles.

To reduce contour error, authors proposed advanced controllers to minimize axis tracking errors, known as *tracking control*, indirectly yielding the contour error reduction. Altintas et al. [6] built a sliding mode controller to reduce the tracking error of each axis in the presence of external disturbances such as friction and cutting force, indirectly decreasing the contour error. Erkorkmaz et al. [7] used a pole placement controller with disturbance cancellation in the feedback loop, in order to address the detrimental effects of friction, cutting forces, and drive parameter variations. Moreover, Dumur et al. proposed an axis drive predictive control approach to deal with free-form machining in the context of high speed machining [8]. The benefit of this philosophy is that the idea is simple, as it is based on the axis tracking error reduction to reduce the contour error. Moreover, the tracking error is easily obtained during the control process thanks to the measuring devices. However, it should be noted that the decrease of tracking error, while it is still larger than zero, may be less effective in reducing the contour error or can even increase the contour error in some cases of multi-axis machining [9].

Another philosophy for contour error reduction is *contouring control*, that is to directly decrease contour error in machining process. This means that all axis tracking errors and position responses are firstly collected and used to estimate the contour error and then the control signal of each axis is modified appropriately by one of advanced contouring control laws or controllers to reduce the contour error [10,11]. A contouring control manner usually refers to two sub-groups: contour error compensation in on-line process and contour error pre-compensation in off-line process. In this way, clearly, there are two main factors affecting the contouring performance: (1) the precision of contour error estimation and (2) contour error compensation technique.

For the on-line contour error compensation, the contour error formulas need to be consistent with the computation in real-time process. Koren et al. [10] proposed that contour error is proportional to the tool tracking error in bi-axial machining. Moreover, Chiu et al. [11], in dealing with a feedback controller design, has proposed that some projections of the actual position tracking error in a moving task coordinate frame can be used to approximate contour error. In 5-axis machining, the influence of the actual tool orientation on contour error must be taken into considera-

tion. Altintas et al. [12] developed two analytical models, for both tool tip contour error and tool orientation contour error, based on differential path geometry and the kinematics of the machine. Yang et al. [13] proposed that the contouring error components contributed by all axes drives are estimated through interpolated position commands and the generalized Jacobian function. Once the contour error is estimated on-line, the advanced on-line contouring controllers can be developed. Koren et al. [10] proposed a Variable Cross Couple Control (CCC) for compensating the contour error on-line. Cheng et al. [14] proposed an integrated motion control scheme combined with a fuzzy logic-based feedrate regulator, which adjusts the value of the desired feedrate, to reduce the contour error. Meanwhile, Khalick et al. [15] added the real time estimated contour error into the optimization problem of MPC, in yielding the contour error reduction. The benefit of the on-line contouring control is that it can evaluate and control the real behavior of the machine and the axis drive dynamic can be excited intelligently to compensate for the contour error. The drawback is, however, the computing load in real time, especially in 5-axis machining. Moreover, the advanced on-line contouring controllers require modification of the classical control structure of the commercial CNC. That may be inconvenient and costly for machine tool manufacturers.

In contrast, the off-line contour error pre-compensation technique is related to the contour error prediction thanks to dynamic axis model. Then calibration efforts in the machine simulator are performed to pre-compensate the predicted contour error, before this error may occur in machining process. The idea of contour error prediction is based on the availability of the programmed tool path and the simulated one obtained from the dynamic model of the machine axes [16,17]. Afterwards, the position or velocity setpoints are usually modified to achieve the purpose. Khoshdarregi et al. [18] proposed generating the shaping position commands that suppress the residual vibration and compensate the resulting contour error. Zhang et al. [19] modified the trajectory commands in the FFW blocks, in pre-compensating the contour error and respecting all axis kinematic constraints. Yang et al. [16] used the idea of MPC to adjust the position setpoints, preserving the axis velocity and acceleration constraints. Furthermore, the contouring accuracy in corner machining application is also improved based on this philosophy in [17]. The advantage of this method is that the computational load for contour error prediction and control law is not a constraint anymore and the modified positions is easily implemented by users in CNC. Furthermore, the classical axis control structure is still kept unchanged in this case. However, the performance of this method much depends on the precision of the dynamic axis model.

The original idea of this study is to seek an optimal set of variable control gains, based on a validated axis dynamic

model in off-line process, that could appropriately excite the axis drive dynamic and/or adaptively adjust the controller behaviors to reduce the contour error during 5-axis machining process. It means that instead of using fixed control gains in the position controller of CNC, variable gains will be computed off-line and used in on-line process while preserving the classical cascaded control structure with feed forward actions.

In the literature, various methods are proposed to tune PID control gains in servo system and CNC. Jaen-Cuellar et al. [20] proposed a PID tuning approach by coupling the gain phase margin method with the genetic algorithm, in improving the response in servo system. Le Flohic et al. [21] developed a model based tuning approach for PID control gain of CNC, considering the dynamic behavior of the machine tool structure. However, to the best of the author's knowledge, almost all the existing gain tuning approaches for CNC application consider tracking rather than contouring performances. Moreover, they only serve for finding one fixed PID control gain set, used for all of the different machining parts. These methods cannot obtain the high contouring accuracy in machining free-form surfaces. It is observed by CNC users that axis drive tuning can be different in function of the surface to machine. Thus, the goal of this study is to optimize the axis control gains depending on the geometry of the tool path, with the objective of contour error reduction.

In the proposed approach, control gains are generated off-line by solving an optimization problem, being subject to constraints, including axis kinematic limitations, stability criterion of servo drives and motor current limits. The off-line execution is based on a non-linear axis model, in which the disturbance coming from friction model is considered. Another specificity is the use of "receding horizon" approach of Model Predictive Control (MPC) to solve the optimization problem for the whole trajectory. This proposed approach, so-called Off-line Gain Adjustment (OGA), can be used to reduce the contour error for multi-axis high speed machining, either 3-axis or 5-axis machining. It is proposed that the obtained optimal variable gains could be used in the future commercial CNC. The only modification on the CNC is the extended functions, that are the gain update at each sampling time and the required memory to store the gain values for the given trajectory. Thus, it could be easily implemented by the CNC manufacturers. The general idea of OGA is illustrated in Fig. 1.

The OGA is developed thanks to the non-linear axis model, that is proposed by Susanu et al. [22], enhanced by Prévost et al. [23] and validated by Beudaert et al. [24]. Therefore, the chosen model, acting as a machine simulator, has a high confidence in simulating the machining behavior. Preliminary results for 2-axis machining are presented in [25,26]. In the present paper, OGA is implemented for the global

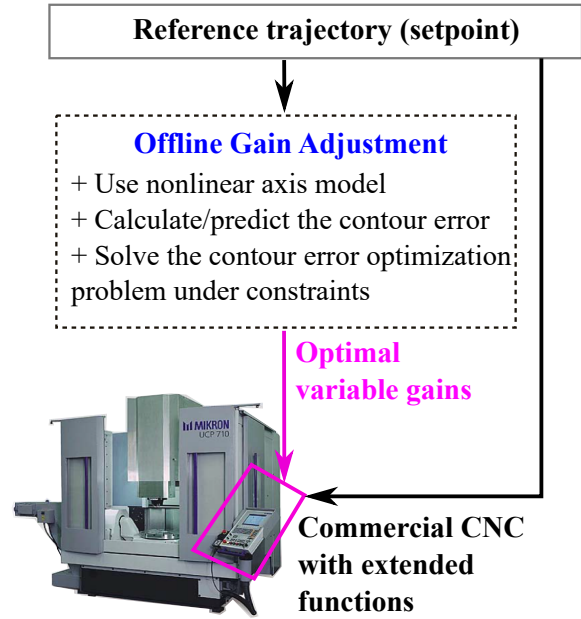


Fig. 1 General idea of OGA approach

case, along with a new contour error estimation approach in multi-axis machining. To the best of the author's knowledge, the usual estimated contour error approaches designate the contour error as deviations between the actual tool location point, tool orientation and the desired ones, to approximate the effective cutting errors, known as under cut and/or over cut. Another paradigm is proposed here, in which the contour error derived from the estimated tool contact point is used, as it is more accurate to approximate the effective cutting errors, instead of using the tool location point.

The rest of the paper is organized as follows: Section 2 is dedicated to the OGA principle. Then, the OGA optimization under constraints and its solution are presented in Section 3. The machine simulator based results are discussed in Section 4. Finally, the conclusions and perspectives are given in Section 5.

## 2 OGA principle

### 2.1 Control structure

The control structure with specificity of OGA is illustrated in Fig. 2, and explained as follows.  $r$  and  $s$  superscripts represent the reference and simulated values in view of the axis model simulation, while  $m$  and  $w$  subscripts denote for the articular (machine axis) and workpiece spaces respectively. For simplicity, the velocity and current loops, motor and friction model have been hidden. The study intention is to modify the proportional and feed forward gains in the position loop, denoted as  $K_a^P$  and  $K_a^F$  respectively, where  $a$  represents for both the three linear axes  $\{X_m, Y_m, Z_m\}$  and the

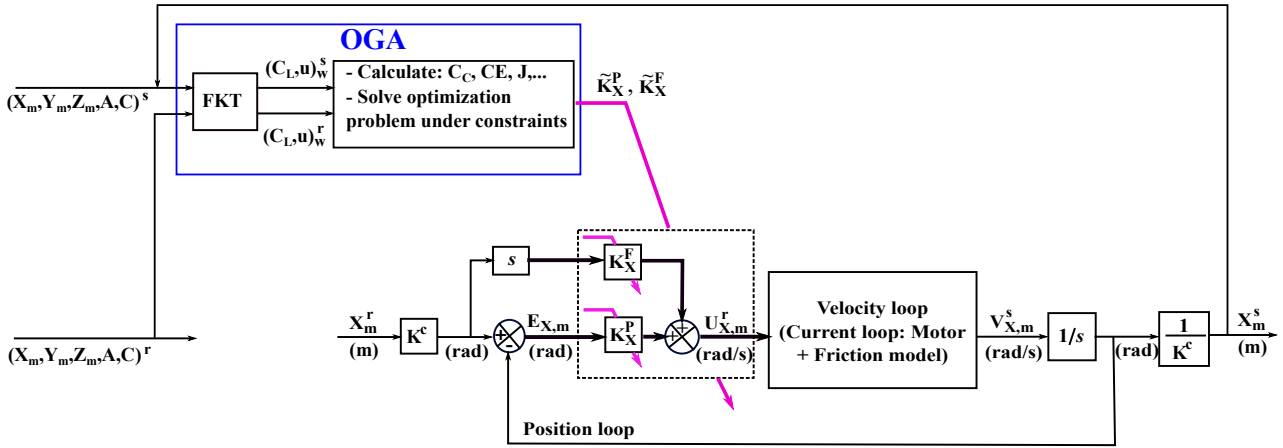


Fig. 2 Proposed control structure with the specificity of OGA, e.g. the case of X axis is shown

two rotary axes  $\{A, C\}$  of a 5-axis machining center. The torque feed forward action of the current loop is not considered in this work in order to highlight the effects of the velocity feed forward control action and its modification based on the OGA. The blocks  $s$  and  $1/s$  are the Laplace domain transfer functions for the derivative and integrator operations respectively.  $K^c$  is a conversion factor from meter unit to radian unit. The unit of  $K_a^P$  is [1/second], while  $K_a^F$  is a constant without unit.

The inputs of OGA are the axis position setpoints  $(X_m, Y_m, Z_m, A, C)^r$  and responses simulated by the non-linear axis model  $(X_m, Y_m, Z_m, A, C)^s$ . In OGA, firstly the Forward Kinematic Transformation (FKT) problem is solved to obtain the reference and simulated tool behaviors, respectively denoted as  $(\mathbf{C}_L^r, \mathbf{u}^r)$  and  $(\mathbf{C}_L^s, \mathbf{u}^s)$ , in which  $\mathbf{C}_L(x, y, z)$  and  $\mathbf{u}(i, j, k)$  are in turn the tool location point and tool axis orientation. The tool contact points and the contour error are then estimated for the entire tool path. Finally, solving the contour error optimization problem under constraints allows to obtain the optimal variable gains  $(\tilde{K}_X^P$  and  $\tilde{K}_X^F)$  at the output of OGA. The gains are afterwards used to update the control gains in the position loop of axis drive.

## 2.2 Contour error estimation

As shown in Fig. 3.a, CAM firstly generates a programmed tool path, characterized by a set of  $(\mathbf{C}_L^r, \mathbf{u}^r)$ . A series of  $\mathbf{C}_L^r$  represents the reference  $\mathbf{C}_L$  path. If the tool follows exactly such a programmed tool path, it will sculpture the workpiece to obtain the desired machining profile, that approximates the nominal surface with the allowable geometrical errors, e.g. machining tolerance. This desired machining profile can be illustrated as the reference  $\mathbf{C}_C$  path, represented by a set of reference tool contact points  $\mathbf{C}_C^r$ . At each contact point,  $\mathbf{f}$ ,  $\mathbf{n}$  and  $\mathbf{t} = \mathbf{f} \times \mathbf{n}$  represent the unit feed, surface normal and transverse vectors, respectively.

The above programmed tool path is interpreted into NC code, that is then fed into the CNC. Through the simulations and calculations proposed in Fig. 2, the corresponding simulated tool positioning  $(\mathbf{C}_L^s, \mathbf{u}^s)$ , is obtained. Similarly, other notations can be defined, including the simulated tool contact point  $\mathbf{C}_C^s$ , the simulated  $\mathbf{C}_L$  and  $\mathbf{C}_C$  paths, as illustrated in Fig. 3.b.

In fact,  $(\mathbf{C}_L^s, \mathbf{u}^s)$  is lagged behind  $(\mathbf{C}_L^r, \mathbf{u}^r)$ . This is due to the limited bandwidth of the axis drives and the non-linear characteristics, such as friction or cutting forces. Such a lagged distance is the tool tracking error,  $e_t$ , defined as:

$$e_t = \|\mathbf{C}_L^r - \mathbf{C}_L^s\| \quad (1)$$

In 5-axis machining, there are usually three kinds of errors referring to the precision of tool path displacement:

- the tool tip contour error is represented by the orthogonal distance ( $distance_{\perp}$ ) between  $\mathbf{C}_L^s$  and the reference  $\mathbf{C}_L$  path, denoted as  $\varepsilon_L$  in Fig. 3.b and defined as:

$$\varepsilon_L = distance_{\perp}(\mathbf{C}_L^s, \text{Reference } \mathbf{C}_L \text{ path}) \quad (2)$$

- the tool orientation contour error is represented by the deviation of the tool orientation, denoted as  $\varepsilon_u$  in Fig. 3.c and given by:

$$\varepsilon_u = angle(\mathbf{u}^s, \mathbf{u}^r) \quad (3)$$

- the effective cutting error, is defined by the orthogonal distance between  $\mathbf{C}_C^s$  and the reference  $\mathbf{C}_C$  path, denoted as  $\varepsilon_C$  in Fig. 3.b and 3.c, provided as:

$$\varepsilon_C = distance_{\perp}(\mathbf{C}_C^s, \text{Reference } \mathbf{C}_C \text{ path}) \quad (4)$$

Note that  $\varepsilon_C$  is the most important error in the above three errors, as it refers directly to the quality of surface finishing, e.g. over or under cuts.

In the literature, authors [12, 13] defined contour error models as  $\varepsilon_L$  and  $\varepsilon_u$ , assuming that when they exist,  $\varepsilon_C$  exists as well, as illustrated in Fig. 3.b and 3.c. The advantage

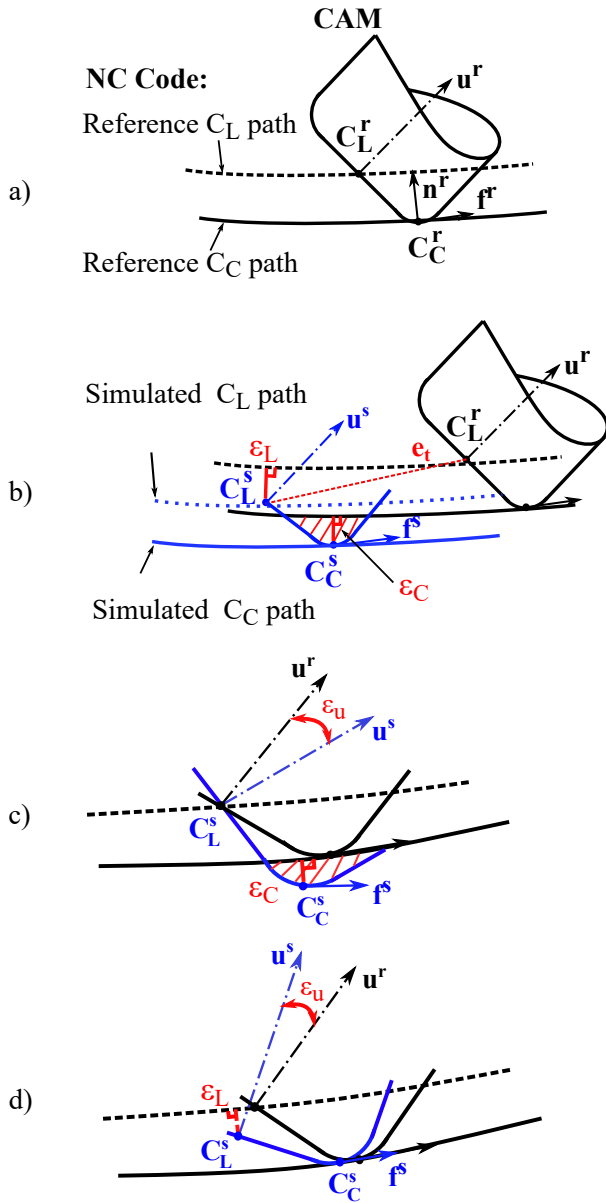


Fig. 3 Contour error definition

of this is that  $\varepsilon_L$  and  $\varepsilon_u$  can be determined thanks to the availability of  $(\mathbf{C}_L, \mathbf{u})$ . In contrast,  $\varepsilon_C$  is hardly accessible due to the lack of  $\mathbf{C}_C$  information at the control stage.

Reducing to zero simultaneously  $\varepsilon_L$  and  $\varepsilon_u$  requires a lot of effort to minimize  $\varepsilon_C$ . However, as seen in Fig. 3.d, even if both  $\varepsilon_L$  and  $\varepsilon_u$  are non-null, the effective cutting error  $\varepsilon_C$  may be null. The advantage of such a way to manage the actual tool positioning is to relax constraints and enlarge the possibilities of the axes and the control to reach the required accuracy on the machined surface.

Therefore, one of the motivations of this study is to improve the contouring performance by controlling directly the position of tool contact  $\mathbf{C}_C$  and defining the contour error as  $\varepsilon_C$ .

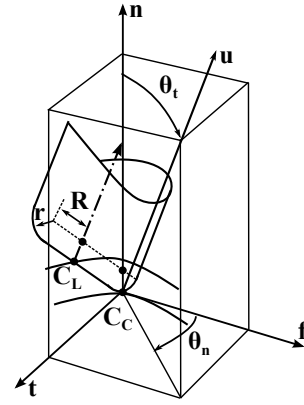


Fig. 4 Tool positioning parameters

To do this, it is necessary to estimate the tool contact point  $\mathbf{C}_C$ .

Taking the case of toric tool in Fig. 4 as an example,  $\mathbf{C}_C$  is obtained as [27]:

$$\mathbf{C}_C = \mathbf{C}_L - r(\mathbf{n} - \mathbf{u}) - R \frac{\mathbf{u} \wedge \mathbf{n}}{\|\mathbf{u} \wedge \mathbf{n}\|} \wedge \mathbf{u} \quad (5)$$

where  $R$ ,  $r$  are the large and small radius of the toric tool. From Eq. (5), the estimation of  $\mathbf{n}$  is the key step to determine  $\mathbf{C}_C$ . This paper considers the case in which  $\mathbf{n}$  is calculated through the estimation of  $\mathbf{f}$ .

In CAM process, tilt angle  $\theta_t$  and yaw angle  $\theta_n$  represent the angles  $\angle(\mathbf{u}, \mathbf{n})$  and  $\angle(\mathbf{u} - (\mathbf{u} \cdot \mathbf{n})\mathbf{n}, \mathbf{f})$  respectively, as illustrated in Fig. 4. In the case  $\theta_t \neq 0$  and  $\theta_n = 0$ , the unique relation between  $\mathbf{n}$ ,  $\mathbf{u}$  and  $\mathbf{f}$  is given by:

$$\mathbf{n} = \frac{\mathbf{f} \wedge \mathbf{u}}{\|\mathbf{f} \wedge \mathbf{u}\|} \wedge \mathbf{f}$$

To estimate  $\mathbf{f}$ , the following assumptions are made:

- the tool diameter is much smaller than the trajectory curvature;
- if present, the change of  $\theta_t$  angle is quite low along the path, e.g.  $\dot{\theta}_t$  is as small as several micro-degrees/second.

As a result, the unit feed vector  $\mathbf{f}$  at  $\mathbf{C}_C$  is approximated to the unit tangential vector as a function of tool displacement at  $\mathbf{C}_L$ . The latter is calculated by the first derivative of  $\mathbf{C}_L$  as a function of the tool displacement  $s$ . Finally,  $\mathbf{f}$  is estimated by:

$$\mathbf{f} = \frac{d\mathbf{C}_L}{ds} \quad (6)$$

where the tool displacement at the  $k$  instant, denoted as  $s^k$ , is defined as:

$$s^k = \sum_{i=1}^k \|\mathbf{C}_L^{r,i} - \mathbf{C}_L^{r,i-1}\|$$

Once  $\mathbf{f}$  is estimated by Eq. (6),  $\mathbf{C}_C$  and  $\varepsilon_C$  are then calculated by Eq. (5) and Eq. (4), respectively.

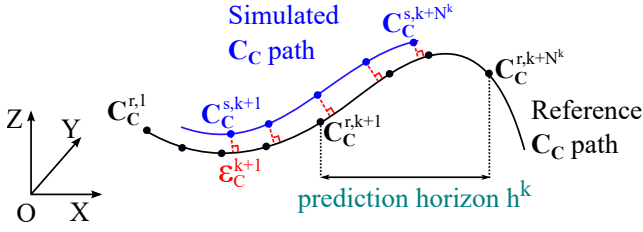


Fig. 5 Definition of contour error within prediction horizon  $h^k$

### 2.3 Gain modification influence on the contour error

Basically, when OGA adjusts the proportional gain of the position controller  $K_a^P$ , it leads to adjust the dynamic of axis drives [28]. While if OGA adjusts the feed forward gain of the position controller  $K_a^F$ , it modifies the impact of feed forward action on the velocity loop [29]. As a result, the axis tracking errors are changed. Then, the amplitude and direction of the tool tracking error,  $e_t$  in Eq. (1), and the resulting tangential feedrate vector  $f$  are consequently modified. Therefore, it can be said that the gain adjustments in OGA can finally modify the tangential feedrate vector so that the actual tool contact point can approach the reference  $C_C$  path, in order to remove or reduce the contour error  $\epsilon_C$ .

## 3 OGA optimization

The OGA optimization, as illustrated in Fig. 5, consists of three main steps:

- Define a prediction horizon  $h^k$ , covering the future tool contact points from  $k+1$  to  $k+N^k$  instant;
- Predict contour error  $\epsilon_C$  over the horizon  $h^k$  based on the simulation model; and
- Tune  $K^P$  and  $K^F$  which minimizes the contour error  $\epsilon_C$  over the horizon, respecting constraints, via receding horizon way.

The mathematical presentation of the above steps is given below.

### 3.1 Optimization problem

Assume that from the beginning to the instant  $k$ , the control gain values, the reference and simulated values of the axes motions and the tool behaviors are known. The objective is to find out the optimal future values of the control gains from the instant  $k+1$  to the instant  $k+N^k$  so that the future contour errors are minimized, over the horizon  $h^k$ .

A matrix containing all reference position setpoints over the horizon  $h^k$ ,  $\mathbb{S}^r$ , is defined as follows:

$$\mathbb{S}^r = \begin{bmatrix} X_m^{r,k+1} & Y_m^{r,k+1} & Z_m^{r,k+1} & A^{r,k+1} & C^{r,k+1} \\ X_m^{r,k+2} & Y_m^{r,k+2} & Z_m^{r,k+2} & A^{r,k+2} & C^{r,k+2} \\ \dots & \dots & \dots & \dots & \dots \\ X_m^{r,k+N^k} & Y_m^{r,k+N^k} & Z_m^{r,k+N^k} & A^{r,k+N^k} & C^{r,k+N^k} \end{bmatrix} \quad (7)$$

Assume that within such a horizon, the values of  $K_a^P$  and  $K_a^F$  for all axes are given by the matrix  $\mathbb{K}^P$  and  $\mathbb{K}^F$  respectively, defined as:

$$\left\{ \begin{array}{l} \mathbb{K}^P = \begin{bmatrix} K_X^{P,k+1} & K_Y^{P,k+1} & \dots & K_A^{P,k+1} & K_C^{P,k+1} \\ K_X^{P,k+2} & K_Y^{P,k+2} & \dots & K_A^{P,k+2} & K_C^{P,k+2} \\ \dots & \dots & \dots & \dots & \dots \\ K_X^{P,k+N^k} & K_Y^{P,k+N^k} & \dots & K_A^{P,k+N^k} & K_C^{P,k+N^k} \end{bmatrix} \\ \mathbb{K}^F = \begin{bmatrix} K_X^{F,k+1} & K_Y^{F,k+1} & \dots & K_A^{F,k+1} & K_C^{F,k+1} \\ K_X^{F,k+2} & K_Y^{F,k+2} & \dots & K_A^{F,k+2} & K_C^{F,k+2} \\ \dots & \dots & \dots & \dots & \dots \\ K_X^{F,k+N^k} & K_Y^{F,k+N^k} & \dots & K_A^{F,k+N^k} & K_C^{F,k+N^k} \end{bmatrix} \end{array} \right. \quad (8)$$

The variation amounts of the above gains of the axis  $a$ , in the considered horizon, are denoted as  $\delta K_a^P$  and  $\delta K_a^F$ , and defined in Eq. (9). Note that  $K_a^{P,k}$  and  $K_a^{F,k}$  are known at the  $k$  instant.

$$\left\{ \begin{array}{l} \delta K_a^P = \begin{bmatrix} K_a^{P,k+1} - K_a^{P,k} \\ K_a^{P,k+2} - K_a^{P,k+1} \\ \dots \\ K_a^{P,k+N^k} - K_a^{P,k+N^k-1} \end{bmatrix} \\ \delta K_a^F = \begin{bmatrix} K_a^{F,k+1} - K_a^{F,k} \\ K_a^{F,k+2} - K_a^{F,k+1} \\ \dots \\ K_a^{F,k+N^k} - K_a^{F,k+N^k-1} \end{bmatrix} \end{array} \right. \quad (9)$$

To simulate the tool movement from the  $k+1$  to  $k+N^k$  instant over the horizon  $h^k$ , each row of  $\mathbb{S}^r$  in Eq. (7),  $\mathbb{K}^P$  and  $\mathbb{K}^F$  in Eq. (8) are simultaneously sent to the machine simulator. It generates the simulated axis positions  $\mathbb{S}^s$ :

$$\mathbb{S}^s = \begin{bmatrix} X_m^{s,k+1} & Y_m^{s,k+1} & Z_m^{s,k+1} & A^{s,k+1} & C^{s,k+1} \\ X_m^{s,k+2} & Y_m^{s,k+2} & Z_m^{s,k+2} & A^{s,k+2} & C^{s,k+2} \\ \dots & \dots & \dots & \dots & \dots \\ X_m^{s,k+N^k} & Y_m^{s,k+N^k} & Z_m^{s,k+N^k} & A^{s,k+N^k} & C^{s,k+N^k} \end{bmatrix} \quad (10)$$

From Eq. (10), solving the FKT problem allows to obtain a vector representing the predicted tool location and orientation  $\mathbb{P}^s$ :

$$\mathbb{P}^s = \left[ P_w^{s,k+1}, P_w^{s,k+2}, \dots, P_w^{s,k+N^k} \right] \quad (11)$$

where  $P_w^{s,k+1}$  represents for  $(\mathbf{C}_L^{s,k+1}, \mathbf{u}^{s,k+1})$ .



According to the assumptions in Section 2.2 and Eq. (11), a vector containing the predicted tool contact points,  $\mathbb{C}_C^s$ , is obtained as follows:

$$\mathbb{C}_C^s = \left[ \mathbf{C}_C^{s,k+1}, \mathbf{C}_C^{s,k+2}, \dots, \mathbf{C}_C^{s,k+N^k} \right] \quad (12)$$

The contour error is then calculated by the orthogonal distance from each element of  $\mathbb{C}_C^s$  to the reference  $\mathbf{C}_C$  path. To do this, a vector containing a set of the reference tool contact points, denoted as  $\mathbb{C}_C^r$ , is employed. As it can be noticed, all of the simulated tool contact points from  $\mathbf{C}_C^{s,k+1}$  to  $\mathbf{C}_C^{s,k+N^k}$  are lagged behind their reference ones, from  $\mathbf{C}_C^{r,k+1}$  to  $\mathbf{C}_C^{r,k+N^k}$ , due to the inherent tool tracking error. Therefore, the size of  $\mathbb{C}_C^r$  must be greater than that of  $\mathbb{C}_C^s$  to ensure that the contour error can always be calculated. Consequently,  $\mathbb{C}_C^r$  is proposed as follows:

$$\mathbb{C}_C^r = \left[ \mathbf{C}_C^{r,k-N^b}, \mathbf{C}_C^{r,k-N^b+1}, \dots, \mathbf{C}_C^{r,k+N^k} \right] \quad (13)$$

in which  $0 \leq N^b \leq k-1$ . If  $N^b = k-1$ , the first element of vector  $\mathbb{C}_C^r$  is always  $\mathbf{C}_C^{r,1}$ , the contour error can always be calculated, however it induces a larger computation burden. This is because the size of  $\mathbb{C}_C^r$  will be increased when  $k$  increases. To overcome such an inconvenience, depending on the dynamic response of the axis drive,  $N^b$  can be chosen as a specific value to reduce the size of  $\mathbb{C}_C^r$  and so reduce the computation time.

From Eq. (4), Eq. (12) and Eq. (13), the predicted contour error values over the horizon  $h^k$  are calculated and represented by  $\mathbb{E}_C$  as follows:

$$\mathbb{E}_C = \left[ \varepsilon_C^{k+1}, \varepsilon_C^{k+2}, \dots, \varepsilon_C^{k+N^k} \right] \quad (14)$$

From Eq. (14), the cost function,  $J$ , is proposed as:

$$J = \mathbb{E}_C (\mathbb{E}_C)^T \quad (15)$$

From Eq. (15), the optimization problem of OGA is formulated by:

$$[\tilde{\mathbb{K}}^P, \tilde{\mathbb{K}}^F] = \text{argmin}(J) \quad (16)$$

where  $\tilde{\mathbb{K}}^P$  and  $\tilde{\mathbb{K}}^F$  are the optimal values of  $\mathbb{K}^P$  and  $\mathbb{K}^F$  in Eq. (8).

The optimization problem in Eq. (16) is subject to constraints, that are discussed below.

### 3.2 Constraints

In order to assure machine kinematic limitations, stability and feasibility, the optimization of OGA in Eq. (16) is subject to the three following constraints.

*Stability criterion.* To satisfy the stability criterion of the servo drives, it is proposed that the variable values of proportional gain in the position controller, generated by OGA, respect the limitations of phase margin and gain margin of the position open loop of the axis drive [30], as given below:

$$\begin{cases} \text{Gain margin} \geq 15 \text{ (dB)} \\ \text{Phase margin} \geq 70 \text{ (Deg)} \end{cases} \quad (17)$$

in order to guarantee sufficient damping of the position trajectories, thus reducing contour errors.

*Axis kinematic limitations.* The second constraint refers to the axis kinematic limitations of each machining center. They are the axis velocity, acceleration and jerk limitations, denoted by  $V_a^{\max}$ ,  $A_a^{\max}$ , and  $J_a^{\max}$  respectively:

$$\begin{cases} -V_a^{\max} \leq V_a^{s,i} \leq V_a^{\max} \\ -A_a^{\max} \leq A_a^{s,i} \leq A_a^{\max} \\ -J_a^{\max} \leq J_a^{s,i} \leq J_a^{\max} \end{cases} \quad \text{with } k+1 \leq i \leq k+N^k \quad (18)$$

where  $V_a^{s,i}$  for example are the predicted velocity values of the axis  $a$  from the  $k+1$  to  $k+N^k$  instant.

*Motor current limitations.* To maintain a good behavior of the machine the motor current  $I_a^i$  and its increment  $\Delta I_a^i = I_a^i - I_a^{i-1}$  are constrained during the prediction horizon:

$$\begin{cases} I_a^k \leq I_a^{\max} \\ |\Delta I_a^k| \leq \Delta I_a^{\max} \end{cases} \quad \text{with } k+1 \leq i \leq k+N^k \quad (19)$$

This constraint considers the sampling time of the current loop, smaller than the sampling time of the position loop, to avoid current high changes during a position loop sampling time. The constraint is verified in the machine simulator results. During the optimization procedure, for each evaluated gain the constraint is tested, and the gain is discarded if the constraint is not verified.

### 3.3 Receding horizon based optimization solving technique

To solve the OGA optimization in the whole trajectory, the receding horizon is used as in the MPC to find the optimal gains for the whole horizon  $h^k$  at the  $k$  instant but only the first optimal gain of the horizon is kept for the instant  $k+1$ , as illustrated in Fig. 6.a and 6.b. Then the horizon is receded to the instant  $k+1$  and the optimization is repeated, as shown in Fig. 6.c. The method of receding horizon allows to apply the OGA optimization for the whole trajectory. The use of MPC comes by the fact that non-linear simulators of the machine are available. This control technique considers



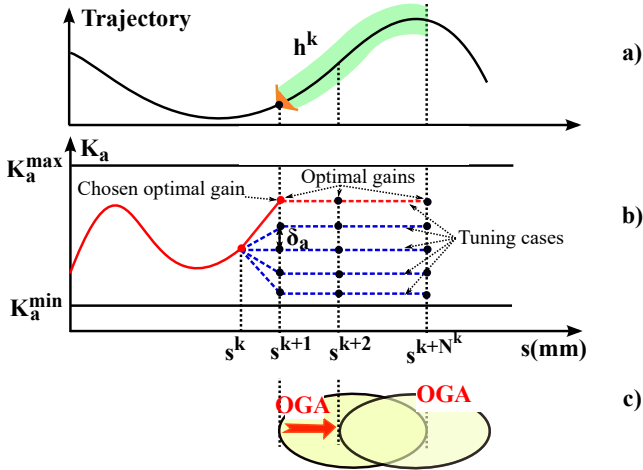


Fig. 6 Solving the OGA optimization: a) Horizon length; b) Tuning the constant gains; c) Receding horizon with overlap

a model to predict the behaviour of the process. This prediction is used to obtain the input of the process by minimization of a criterion, such as the error between the desired trajectory and the predicted one [31,32]. This strategy is used here thanks to a validated machine simulator. The control gains are in this case the considered inputs modifying the behaviour of the machine, and the nonlinear simulator is used as the model of the process.

In Fig. 6 b, during the prediction horizon  $h^k$ , the gain can take different values for each sampling time. In order to simplify the optimization problem and as receding horizon is used, it is proposed that the gain is kept constant during the prediction horizon. Moreover, if OGA is performed within an unlimited gain range, the computation burden turns to be too high. To tackle this, OGA should be performed within one limited range for each control gain, given by:

$$K_a^{\min} \leq K_a^k \leq K_a^{\max} \quad (20)$$

It is proposed that this admissible gain range is defined in advance in the way that the classical axis control can use any fixed gain values in this range without violating any constraints in Eq. (17), Eq. (18), and Eq. (19).

Assume that  $\delta K_a$ , representing for either  $\delta K_a^P$  or  $\delta K_a^F$  in Eq. (9), is the gain modifications within the horizon  $h^k$ , and is obtained by:

$$\delta K_a = \begin{bmatrix} K_a^{k+1} - K_a^k \\ K_a^{k+2} - K_a^{k+1} \\ \dots \\ K_a^{k+N^k} - K_a^{k+N^k-1} \end{bmatrix} = \begin{bmatrix} \delta_a \\ 0 \\ \dots \\ 0 \end{bmatrix} \quad (21)$$

The meaning of Eq. (21) is that setting  $\delta_a$  as positive, zero and negative values allows to obtain the tuning cases illustrated in Fig. 6 b. Solving the OGA optimization problem with this method consists in tuning  $\delta_a$  in Eq. (21) within the admissible gain ranges in Eq. (20). Then, by checking all of

the possibilities of  $K^P$  and  $K^F$  in Eq. (8) by means of the machine simulator, the optimal variable gains minimizing Eq. (15) are found, fulfilling all of the constraints.

#### 4 Machine simulator based OGA results

To validate the OGA approach, virtual machine based simulations are performed. Firstly, Fig. 7.a shows the desired part and its 5-axis milling simulation.

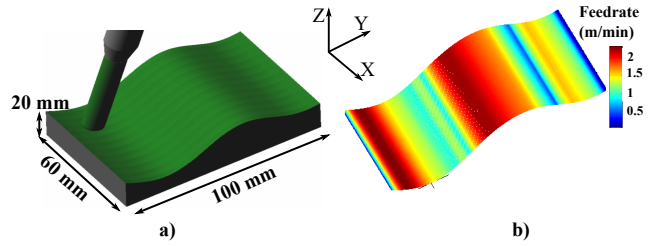


Fig. 7 a) Desired part b) Desired feedrate cartography

The programmed tool path is generated by a CAM software, with the configuration given in Table 1.

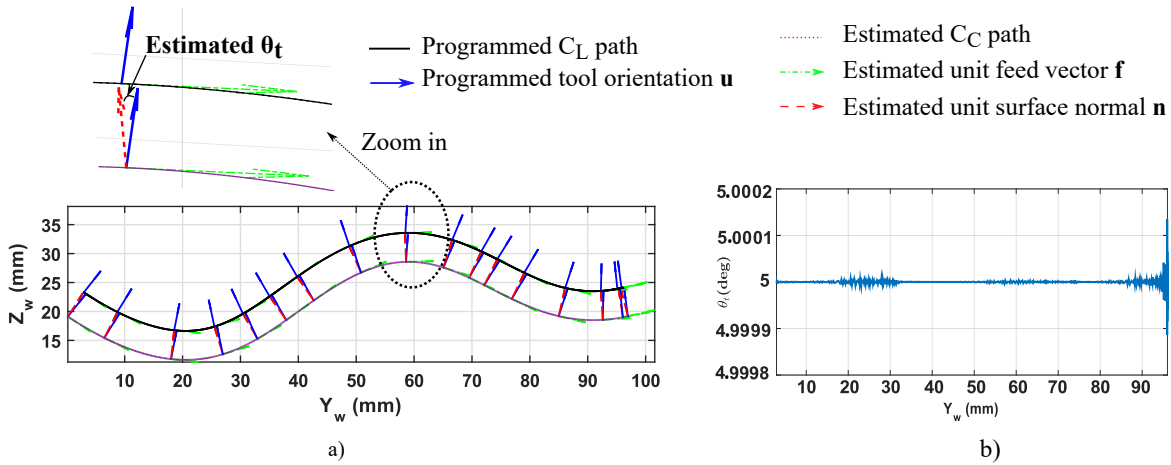
Table 1 Configuration in CAM process

Parameters	Value	Unit
Yaw angle	0	degree
Tilt angle	5	degree
Hemispheric tool: radius	5	mm
Machining tolerance	$1 \times 10^{-4}$	mm
Machining strategy	parallel plane (YZ)	

Fig. 8.a shows the programmed tool path, the estimated vectors  $\mathbf{f}$ ,  $\mathbf{n}$  and the estimated  $\mathbf{C}_C$  path. In Fig. 8.b, the estimated  $\theta_t$  values accurately approximate the programmed value, with an error of less than  $1.10^{-4}$  degrees. Therefore, the approximation of vector  $\mathbf{n}$  has a high confidence, leading to the high precision of the estimated  $\mathbf{C}_C$  and  $\mathbf{e}_C$ .

The feedrate planning, shown in Fig. 7.b, is executed by VPOp algorithm [33], with 6 ms Bspline interpolation and the programmed feedrate of 3 m/min. Without loss of generality in the 5-axis machining and for the simplicity purpose, this example requires two linear axes  $Y, Z$  and one rotary axis  $A$ , whose kinematic performances of the 5-axis machine tool used are given in Table 2.

Considering the constraints in Section 3.2, the admissible range for  $K^P$  and  $K^F$  defined in Eq. (20), as well as the best fixed gains for contour following,  $K^{P*}$  and  $K^{F*}$ , are obtained in Table 3. These best fixed gains have been obtained by testing all combinations of  $K^P$  and  $K^F$  in the admissible range with a grid step of 0.1 m/min/mm or rad/min/mrad for



**Fig. 8** a) Illustrations of the programmed tool path, the estimated  $C_C$  path and the estimated vectors; b) Estimated  $\theta_t$

$K^P$  and 0.01 for  $K^F$ . Other configurations of OGA parameters are pre-defined as follows:

- Horizon length:  $N^k = 25$  (sampling instants)
- Choosing  $\delta_0 = 0.001$  and tuning  $\delta_a$  from Eq. (21) within 5 cases  $\{-2\delta_0, -\delta_0, 0, \delta_0, 2\delta_0\}$
- From (19),  $\Delta I_a^{max}$  is chosen to be equal to  $k_0 = 2\%$  of the nominal motor current values [34]

Table 4 highlights the results of OGA in the use cases, mentioned below:

- REF: Reference case with classical control using the best fixed gains
- Case 1: OGA for tuning only  $K^P$
- Case 2: OGA for tuning only  $K^F$
- Case 3: OGA for tuning both  $K^P$  and  $K^F$

Contouring accuracy is evaluated through the mean of contour error. Overall, the OGA has reduced the contour error much more than the classical control using the best fixed

gains. It can be seen that the best performances of OGA in Case 1, Case 2 and Case 3 can respectively reduce around 70%, 95% and 98% the contour error of the reference case REF. Case 2 takes the same computation time as Case 1, but the former reduces around 83% the contour error of the latter. Meanwhile, the contouring accuracy of Case 3 is improved around 66% compared with Case 2, but the former takes 7 times longer to finish the computation than the latter. In fact, the simulation can be speeded up by:

- reducing the number of tuning cases of  $\delta_a$  from Eq. (21) or reducing the horizon length  $N_k$ ,
- increasing the receding step size to several samples, or up to half of the horizon length.

This speed reduction would reduce the compensation effect. These parameters can be used to adapt the proposed approach in function of the considered trajectory, the available calculation capacity and the desired compensation effect.

For simplicity, OGA (Case 3) is shown in detail, in which Fig. 9 and Fig. 10 show the resulting variable gains and their effects on axis kinematic and motor current responses respectively. Axis velocities and accelerations are not much modified. Jerk variations are higher than in the reference case, but still respect the corresponding limits. The achieved variable gains of Y and A axes contribute more in the OGA

**Table 2** Axis kinematic constraints of Mikron UCP 710 machine

	Y	Z	A
$V^{max}(m/min - rpm)$	30	30	15
$A^{max}(m/s^2 - rad/s^2)$	3	2.1	$0.83 \times 2\pi$
$J^{max}(m/s^3 - rad/s^3)$	5	50	$5 \times 2\pi$

**Table 3** Admissible gain ranges for  $K^P$ ,  $K^F$ ; and the best fixed gain values  $K^{P*}$ ,  $K^{F*}$

	Unit	Y	Z	A
$K^P$	m/min/mm rad/min/mrad	[0.9 : 3.4]	[0.9 : 2.7]	[0.9 : 2.7]
$K^F$	-	[0 : 1.08]	[0 : 1.95]	[0 : 1.52]
$K^{P*}$	m/min/mm rad/min/mrad	1.2	1.8	1
$K^{F*}$	-	0.9	0.8	0.9

**Table 4** Comparison between OGA and the reference case

	$\bar{\varepsilon} (\mu m)$	Improvement	Computing time
REF	14.4	-	-
Case 1	4.2	70.83 % REF	$\simeq 6.5$ h
Case 2	0.68	95.28 % REF 83.81 % Case 1	$\simeq 6.5$ h
Case 3	0.23	98.40 % REF 66.18 % Case 2 67.14 % Case 3	$\simeq 45.5$ h $(\simeq 1.89$ days)

(Case 3) than that of Z axis. No major change is observed for current evolutions in all of the three axes.

Fig. 11 highlights that the adjustment of the control gains has changed the feedrate, tool tracking error and contour error. While the contour error is much reduced by OGA, the resulting tool tracking error of OGA is sometimes increased as compared with REF. This confirms the fact that a zero contour error can be obtained even if the tool tracking error still exists. The zoomed views prove that OGA makes the tool displacement very accurate over the peak of the convex curve zone, which is one of the critical regions in machining the free-form trajectory.

Fig. 12.a shows that OGA causes much less contour error marks on the surface finishing than the reference case. Fig. 12.b indicates that the contour error of the reference case REF mainly ranges from 0.5 to 50  $\mu\text{m}$ . While, OGA (Case 3) have a centered normal distribution of contour error ranging from -0.5 to 0.5  $\mu\text{m}$ .

Furthermore, the comparison of 3D contour error presentation of OGA between Case 1, Case 2 and Case 3 is shown in Fig. 13. From Case 1 to Case 3, the contour error marks on the simulated machining surface of the part are remarkably reduced.

## 5 Conclusions and perspectives

A new approach OGA has been proposed in this paper. Its objective is to improve the contouring accuracy in machining free-form part in the context of multi-axis high speed machining, based on the means of an off-line process. Its main strategy is to exploit all possibilities of the classical control structure of axis drive, proposing the elementary base of a method that could be considered for an implementation in a commercial CNC. Inspiring from the motivations, the main contributions of this paper are summarized below.

Simulation results based on a validated machining simulator proved that the OGA reduces significantly the contour error in all cases of study, as compared with the classical control structure using the best fixed gains for contour following. Furthermore, it can be concluded that if the computation burden is less critical or overcome within the off-line execution phase, tuning both  $K^P$  and  $K^F$  is the best choice for the contouring accuracy. If not only a high contouring accuracy but also a short computation time are needed, then is preferably to tune only  $K^F$ . The main contribution of this study is the formulation of an off-line gain adjustment method for improving the contouring accuracy of a general 5-axis machining case, being subject to the axis kinematic constraints, the stability criterion of servo drive, and the limits of motor current; and naturally considering the geometrical effects of a free-form machining profile. Another contribution is related to the contour error formulation in 5-axis point

milling with a toric tool, based on  $\varepsilon_C$ , using the proposed estimation approach for the tool contact points  $C_C$ .

In perspective, the OGA will consider more non-linear characteristics and disturbances effects, e.g. machine flexibilities, into the optimization problem under constraints of OGA. In addition, other techniques of optimization, e.g. heuristic method or genetic algorithm, etc, should be examined. The objective is to reduce the computation burden in solving the OGA optimization problem and to produce more optimal control gains for the contour error reduction. Analysis of obtained gain variation profiles is expected to eliminate or reduce the optimization stage. Moreover, optimization results of similar trajectories would be used to deduce gain modification profiles in function of trajectories. Last but not least, the experimental tests of OGA are expected to be performed in the framework of an OPEN CNC.

## References

- Altintas Y (2000) Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Calibrations, and CNC Design. Cambridge University Press, Cambridge.
- Hu P, Chen L, Tang K (2017) Efficiency-optimal iso-planar tool path generation for five-axis finishing machining of freeform surfaces. Computer-Aided Design, 83:33-50.
- Huang J, Du X, Zhu LM (2018) Real-time local smoothing for five-axis linear tool-path considering smoothing error constraints. International Journal of Machine Tools and Manufacture, 124:67-79
- Erkorkmaz K, Chen QG C, Zhao MY, Beudaert X, Gao, XS (2017). Linear programming and windowing based feedrate optimization for spline toolpaths. CIRP Annals, 66(1):393-396
- Altintas Y, Kersting P, Biermann D, Budak E, Denkena B, Lazoglu I (2014). Virtual process systems for part machining operations. CIRP Annals, 63(2):585-605.
- Altintas Y, Erkorkmaz K, Zhu WH (2000) Sliding Mode Controller Design for High Speed Feed Drives. CIRP Annals - Manufacturing Technology, 49(1):265-270.
- Erkorkmaz K, Altintas Y (2001) High speed CNC system design. Part III: high speed tracking and contouring control of feed drives. International Journal of Machine Tools and Manufacture, 41(11):1637-1658.
- Dumur D, Susanu M, Aubourg M (2008) Complex form machining with axis drive predictive control. CIRP Annals - Manufacturing Technology, 57(1):399-402.
- Tang L, Landers RG (2013) Multiaxis Contour Control - The State of the Art. IEEE Transactions on Control Systems Technology, 21(6):1997-2010.
- Koren Y, Lo CC (1991) Variable-Gain Cross-Coupling Controller for Contouring. CIRP Annals - Manufacturing Technology, 40(1):371-374.
- Chiu GC, Tomizuka M (2011) Contouring control of machine tool feed drive systems: a task coordinate frame approach. IEEE Transactions on Control System Technology, 9(1):130-139.
- Altintas Y, Sencer B (2010) High speed contouring control strategy for five-axis machine tools. CIRP Annals - Manufacturing Technology, 59(1):417-420.
- Yang J, Altintas Y (2015) A generalized on-line estimation and control of five-axis contouring errors of CNC machine tools. International Journal of Machine Tools and Manufacture, 88:9-23.

14. Cheng MY, Su KH (2009) Contouring accuracy improvement using a tangential contouring controller with a fuzzy logic-based feedrate regulator. *The International Journal of Advanced Manufacturing Technology*, 41(1):75-85.
15. El Khalick MA, Uchiyama N (2011) Discrete-time model predictive contouring control for biaxial feed drive systems and experimental verification. *Mechatronics*, 21(6):918-926.
16. Yang S, Ghasemi AH, Lu X, Okwudire CE (2015) Precompensation of servo contour errors using a model predictive control framework. *International Journal of Machine Tools and Manufacture*, 98:50-60.
17. Erkorkmaz K, Yeung CH, Altintas Y (2006) Virtual CNC system. Part II. High speed contouring application. *International Journal of Machine Tools and Manufacture*, 46(10):1124-1138.
18. Khoshdarregi MR, Tappe S, Altintas Y (2014) Integrated Five-Axis Trajectory Shaping and Contour Error Compensation for High-Speed CNC Machine Tools. *IEEE/ASME Transactions on Mechatronics*, 19(6):1859-1871.
19. Zhang K, Yuen A, Altintas Y (2013) Pre-compensation of contour errors in five-axis CNC machine tools. *International Journal of Machine Tools and Manufacture*, 74:1-11.
20. Jaen-Cuellar AY, Romero-Troncoso R de J, Morales-Velazquez L, and Osornio-Rios RA (2013) Pid-controller tuning optimization with genetic algorithms in servo systems. *International Journal of Advanced Robotic Systems*, 10(324):1-14.
21. Le Flohic J, Paccot F, Bouton N, Chanal H (2018) Model-based method for feed drive tuning applied to serial machine tool. *The International Journal of Advanced Manufacturing Technology*, 95(1-4):735-745.
22. Susanu M, Dumur D (2006) Hierarchical Predictive Control within an Open Architecture Virtual Machine Tool. *CIRP Annals - Manufacturing Technology*, 55(1):389-392.
23. Prévost D, Lavernhe S, Lartigue C, Dumur D (2011) Feed drive modelling for the simulation of tool path tracking in multi-axis High Speed Machining. *International Journal of Mechatronics and Manufacturing Systems*, 4(3-4):266-284.
24. Beudaert X, Lavernhe S, Tournier C (2014) Direct trajectory interpolation on the surface using an open CNC. *The International Journal of Advanced Manufacturing Technology*, 75(1):535-546.
25. Duong TQ, Rodriguez-Ayerbe P, Lavernhe S, Tournier C, Dumur D (2016) Offline gain adjustment with constraints for contour error reduction in high speed milling. 2016 IEEE International Conference on Advanced Intelligent Mechatronics, 201-206.
26. Duong TQ, Rodriguez-Ayerbe P, Lavernhe S, Tournier C, Dumur D (2016) Receding Horizon Based Offline Gain Adjustment for Contour Error Reduction in High Speed Milling. *Procedia CIRP- 10th CIRP Conference on Intelligent Computation in Manufacturing Engineering*, 62:227-232.
27. Tournier C, Duc E (2005) Iso-scallop tool path generation in 5-axis milling. *The International Journal of Advanced Manufacturing Technology*, 25(9):867-875.
28. Pritschow G (1996) On the Influence of the Velocity Gain Factor on the Path Deviation. *CIRP Annals - Manufacturing Technology*, 45(1):367-371.
29. Tung ED, Tomizuka M (1993) Feedforward Tracking Controller Design Based on the Identification of Low Frequency Dynamics. *ASME Journal of Dynamic Systems, Measurement, and Control*, 115(3):348-356.
30. Younkin GW (2003) *Industrial servo control systems: Fundamentals and applications*. Marcel Dekker, Inc.
31. Camacho EF, Bordons Alba C, *Model predictive control*. Springer Science & Business Media, (2013)
32. Maciejowski JM, *Predictive control with constraints*. Pearson education (2002)
33. Beudaert X, Lavernhe S, Tournier C (2012) Feedrate interpolation with axis jerk constraints on 5-axis NURBS and G1 tool path. *International Journal of Machine Tools and Manufacture*, 57:73-82.
34. Siemens (2005), *Siemens Configuration Manual Simodrive 611 / Masterdrives MC, 1FT6 Synchronous Motors, Manual*, pp 144-145

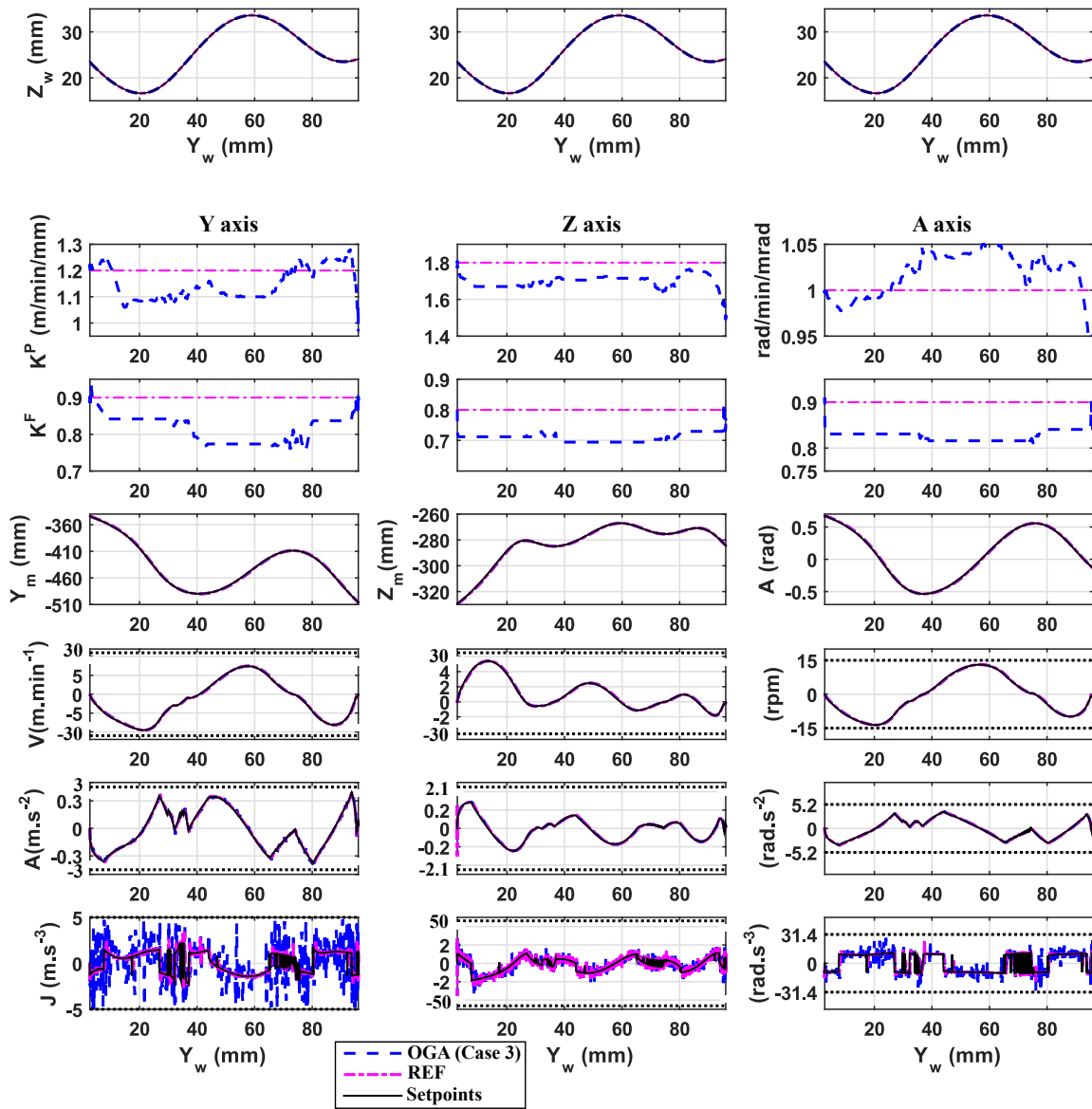


Fig. 9 Axis kinematic responses of OGA (Case 3)

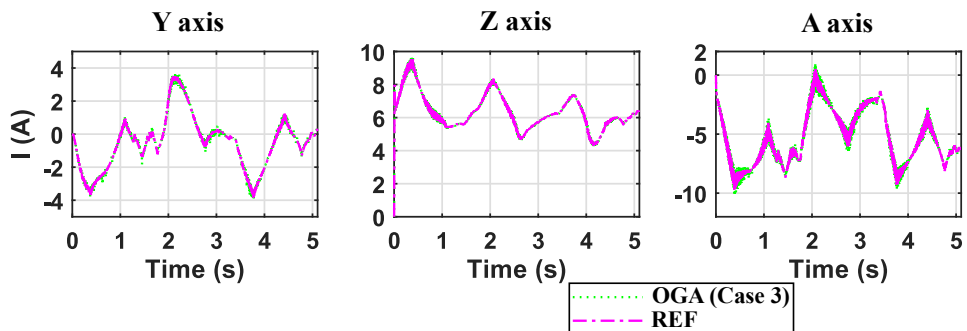


Fig. 10 Motor current verification of OGA (Case 3)

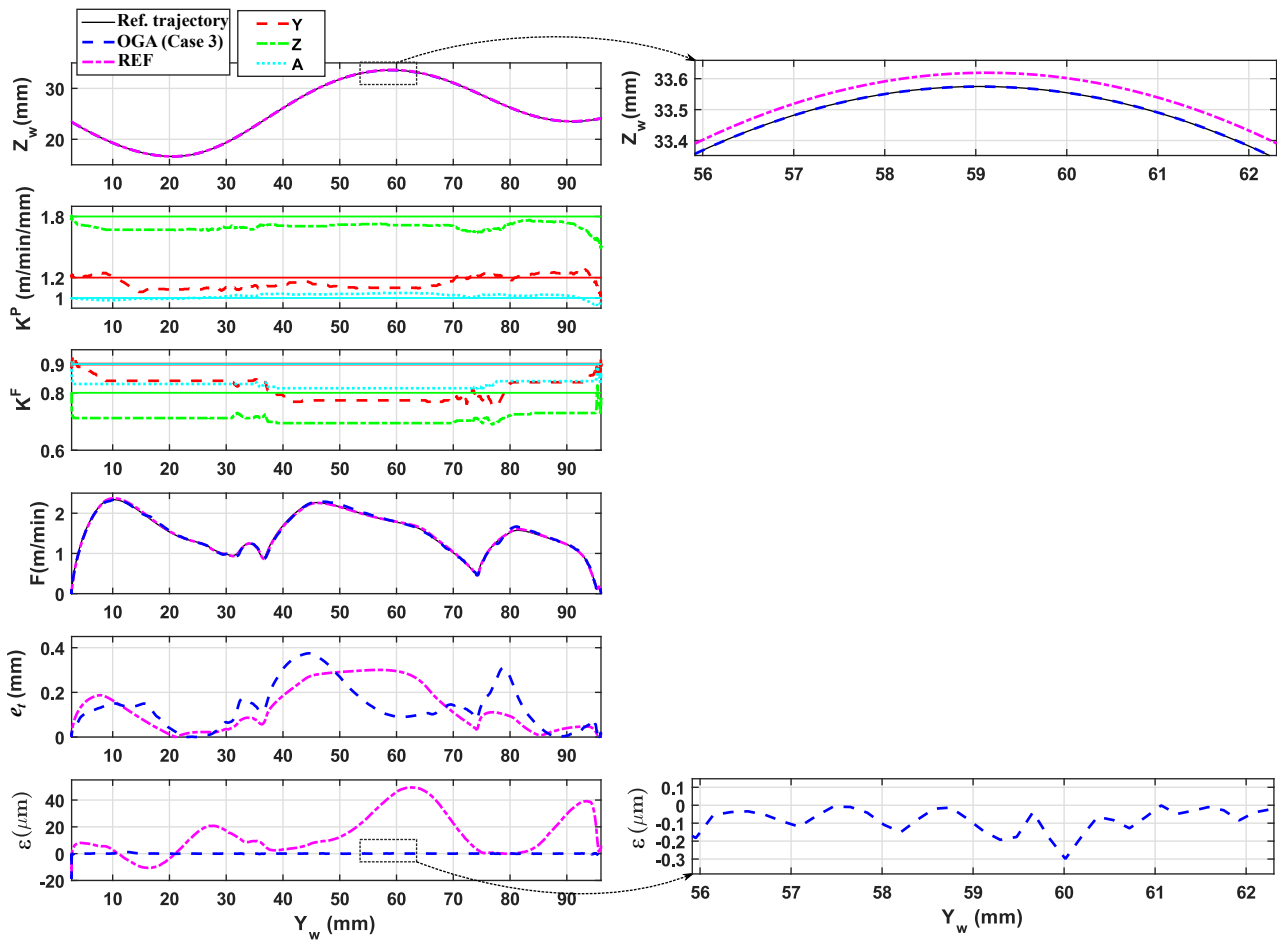


Fig. 11 Impact of variable gains of OGA (Case 3) on the resulting feedrate, tool tracking error and contour error

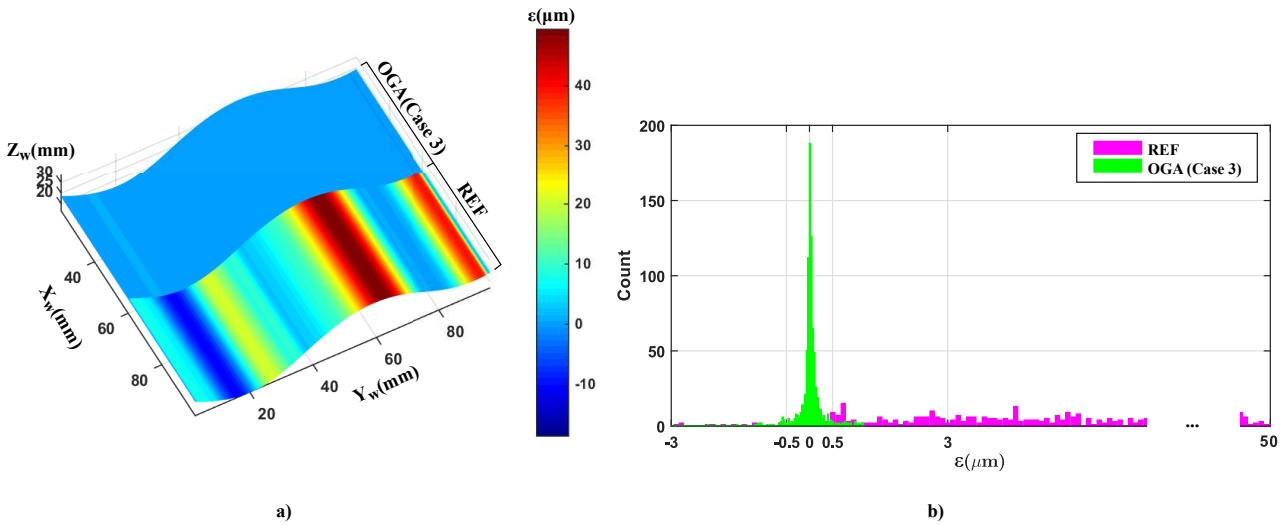
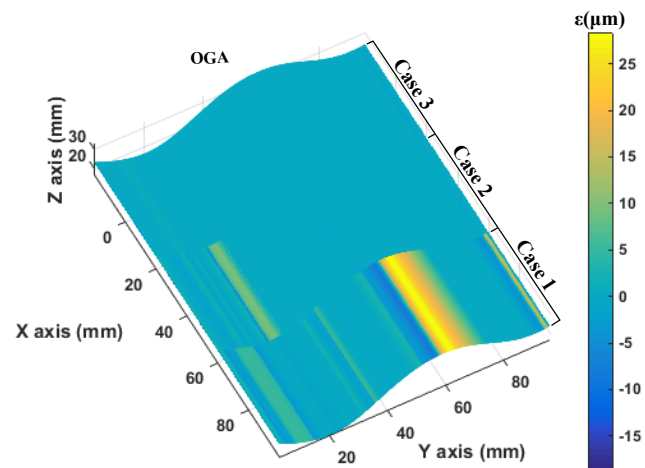


Fig. 12 a) Contour error in 3D representation on the part; b) Contour error histogram



**Fig. 13** Comparing 3D contour error presentation of OGA between Case 1, Case 2 and Case 3