



### **Science Arts & Métiers (SAM)**

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>  
Handle ID: <http://hdl.handle.net/10985/8155>

#### **To cite this version :**

Mohamed DAMAK, Adel OLABI, Richard BEAREE, Olivier GIBARU - Feedrate planning for machining with industrial six-axis robots - Control Engineering Practice - Vol. 18, n°5, p.471-482 - 2010

Any correspondence concerning this service should be sent to the repository

Administrator : [scienceouverte@ensam.eu](mailto:scienceouverte@ensam.eu)



## **Feedrate planning for machining with industrial six-axis robots**

**Adel Olabi**

Metrology and Applied Mathematics Laboratory (L2MA)  
Arts et Metiers ParisTech  
8, Boulevard Louis XIV, 59046 Lille Cedex (France)

**Richard Béarée\***

Laboratory of Electrical Engineering and Power Electronics (L2EP)  
Arts et Metiers ParisTech  
8, Boulevard Louis XIV, 59046 Lille Cedex (France)  
Fax: +33 3 20 62 27 50 ; Email: [richard.bearee@ensam.eu](mailto:richard.bearee@ensam.eu)

**Olivier Gibaru**

Metrology and Applied Mathematics Laboratory (L2MA)  
Arts et Metiers ParisTech  
8, Boulevard Louis XIV, 59046 Lille Cedex (France)

**Mohamed Damak**

Dynalog-France  
1. Bd de Valmy, 59650 Villeneuve d'Ascq (France)

\*Corresponding author

*Abstract*

Nowadays, the adaptation of industrial robots to carry out high-speed machining operations is strongly required by the manufacturing industry. This new technology machining process demands the improvement of the overall performances of robots to achieve an accuracy level close to that realized by machine-tools. This paper presents a method of trajectory planning adapted for continuous machining by robot. The methodology used is based on a parametric interpolation of the geometry in the operational space. FIR filters properties are exploited to generate the tool feedrate with limited jerk. This planning method is validated experimentally on an industrial robot.

**Key words**

Trajectory planning; Industrial Robots; Jerk; Vibration; Machining.

## 1. INTRODUCTION

Over the last four decades industrial robots were used to realize many industrial tasks like material handling, welding, cutting, and spray painting. Nowadays they are widely used in many fields of industry, like automobile industry and aircraft industry. Compared to machine tools, industrial robots are cheaper and more flexible with more important work space. This is why industrials are enthusiastic to replace machine tools by robots. These industrial robots can carry out machining applications like, prototyping, cleaning and pre-machining of cast parts as well as end-machining of middle tolerance parts. This kind of applications requires high accuracy in positioning and path tracking. Unfortunately industrial robots were designed to realize repeatable tasks. So they are repeaters but not that accurate. The robot repeatability ranges typically from 0.03 to 0.1 mm, and the accuracy is often measured to be within several millimeters (Damak *et al.* 2004). Due to their serial structure, articulated robot has lower stiffness than classical machine tools. The stiffness of an industrial robot is usually less than 1 N/ $\mu\text{m}$ , while the stiffness of machine tools is often greater than 50 N/ $\mu\text{m}$  (Pan *et al.* 2006). This poor accuracy and stiffness are caused by many factors, such as geometric parameter errors: manufacturing tolerances, wear of parts and components replacement, as well as non-geometric factors, such as flexibility of links and gear trains, gear backlashes, encoder resolution errors, and thermal effects (Elatta *et al.* 2004) (Shiakolas *et al.* 2002) (Khalil & Dombre, 2004).

Many fields of investigation are proposed to increase the accuracy of industrial robots like; robots calibration, process development and control system. Robot calibration improves the accuracy of positioning by reducing the deviation between the commanded pose and the real one. The complete procedure of robot calibration basically consists of four stages: modeling, measurement, identification, and compensation (Meng & Zhuang, 2007). A kinematic model

of a robot is the mathematical description of its geometry and motion. To construct this model Denavit-Hartenberg convention is usually used. In kinematic calibration, geometric defaults are modeled and compensated. In this method robot joints are assumed to be perfectly rigid (Elatta *et al.* 2004). On the other hand, in non-kinematic calibration, flexibility of robot joints and the other non geometric defaults are taken into account (Ziaei *et al.* 2009) (Ostring *et al.* 2003). In (Abele *et al.* 2007), authors have worked on modeling the Cartesian compliance of an industrial robot according to its joints compliance to analyze the system's stiffness. Other works were interested in the machining process itself, like in (Pan *et al.* 2006) where the authors show the effect of the conditions of the machining process on its stability. Regarding the control field, a large number of works have been done on trajectory planning, feedback control, system compensation and feedforward control (Lambrecht *et al.* 2005) (Hakvoort *et al.* 2008) (Goto *et al.* 2007) (Huey *et al.* 2008). Trajectory planning is one of the important control aspects. It is a fundamental problem in robotics. A well-planned trajectory guaranties a good path tracking and excites less the mechanical structure of the robot and the servo control system, so vibrations can be avoided. For the machining applications these vibrations damage the quality of the machined surfaces.

Trajectory planning can be defined as: determining a temporal motion law along a given geometric path, with respecting certain kinematic and dynamic limits. Therefore, from a geometric path, the planner generates the temporal references of position, speed and acceleration for each joint. For industrial robots, the end effector trajectory can be planned in both joint space and Cartesian space (operational space). Classically, motion planning in joint space is more used. This approach has many advantages like: Both joints actuators and dynamic constraints are in the same level (joint level), so the verification of the respect of these constraints is easier for the control system which, in robotics, acts on the joints actuators

rather than the end effector. Trajectory planning in the joint space allows avoiding the problems arising with kinematic singularities and manipulator redundancy (Gasparetto & Zanotto, 2008). The main disadvantage of planning the trajectory in the joint space is that the performed motion by the robot end effector is not easily foreseeable. This is due to the nonlinearities introduced when transforming the trajectories of the joints into the end-effector trajectory through direct kinematic model. This strategy is suitable for classical tasks like, pick and place, where the movement of the end effector is free between the two extremes positions (Chettibi *et al.* 2004); on the other hand, for machining applications, controlling the feedrate of the cutting tool is indispensable. Planning the trajectories in Cartesian space allows to impose the desired motion law, thus, to control the cutting tool movement. This approach is classically used to plan cutting tools trajectories for the machines tools (Erkorkmaz & Altintas, 2001).

In this paper, a strategy of trajectory planning in robot operational space is introduced. This strategy is adapted to plan trajectories of end-effector of industrial robot intended to realize machining processes. In this method, a smooth motion law is generated by means of a parametric speed interpolator. This interpolator makes advantage of the properties of finite-impulse response (FIR) filters (Kong & Yang, 2005) to give a smooth pattern to the feedrate profile (jerk limited or others). To illustrate the efficiency of this method, trajectories resulting from this strategy are tested on a machining industrial robot, depicted in Figure.1.



Fig. 1. Machining robot with high-speed spindle.

## 2. CARTESIAN SPACE MOTION PLANNING STRATEGY

As mentioned before, this paper is interested in machining applications by industrial robots. So, unlike other applications it is concerned with planning of continuous-path motions instead of point-to-point motions. In this section, a method of trajectory generation for planning the motion of the cutting tool along a prescribed path is presented. This planning is realized in Cartesian space. The procedure used to generate motion commands is as follows. Firstly, the motion of the cutting tool on a parametric curve is planned by using a smooth feedrate profile (with different jerk patterns). Secondly, the parametric interpolator generates the position of the cutting tool (end effector) at each sampling time. Thirdly, these sampled Cartesian positions are converted into joint coordinate commands by using the inverse kinematics model. Fourthly, the joint kinematics constraints, expressed by means of upper bounds on speed, acceleration and jerk are checked and if necessary the feedrate is adapted. Finally, the joint space trajectories are used as references for the joints servos. Figure 2 illustrates the flowchart of the motion planning strategy detailed in this section.

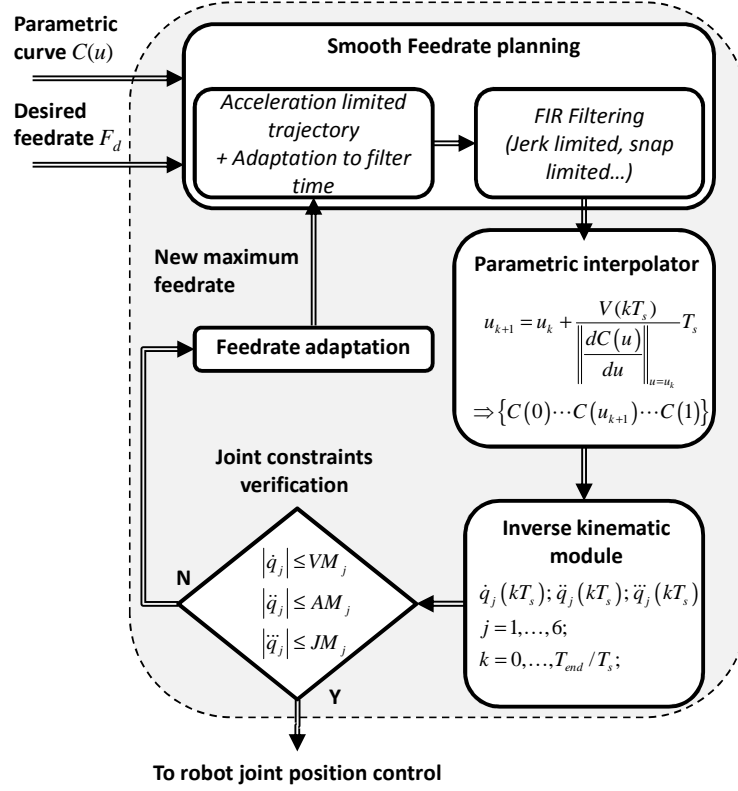


Fig. 2. Cartesian space motion planning strategy.

## 2.1 End Effector feedrate Planning Algorithm

The generation of end-effector smooth motion is divided into two steps. In the first step, simple trapezoidal speed profiles are generated. These profiles are filtered in the second step by a Finite Impulse Response filter.

Motion planning is usually divided into: acceleration stage, constant speed stage (the desired feedrate, denoted  $F_d$ , if reachable) and deceleration stage. Considering classical trapezoidal velocity profile and noting  $F_k$  the feedrate at time  $t = kT_s$ , the feedrate evolution during the acceleration stage is given by

$$F_k = F_0 + A_M \cdot k \cdot T_s \quad (1)$$

where  $T_s$  is the sampling time,  $k$  is the sampling number,  $F_0$  is the start feedrate and  $A_M$  is the kinematic constraint on the maximum end-effector acceleration. If the desired or the maximum feedrate is reached, the system enters into constant feedrate zone. The start time of the deceleration stage can be easily calculated according to the curve length. But, for this



simple speed profile, it is much more convenient to use the concept of speed horizon. The speed horizon  $F_h$  is based on the Work-Kinetic Energy theorem and can be expressed as

$$F_h(L_r) = \sqrt{F_f^2 + 2 \cdot A_M \cdot L_r} \quad (2)$$

Where  $F_f$  is the final velocity and  $L_r$  is the remaining distance to be travelled. If the speed horizon becomes lower than the next speed calculated by (1) or the constant desired feedrate  $F_d$ , the system enters into deceleration stage. One notes that the computation of  $L_r$  requires the curve arc length, noted  $L$ , which, except for specific cases, does not admit analytic reduction, therefore, the remaining distance has to be approximated. At the end of the deceleration stage, the required feedrate could be not synchronized with the distance to be traveled. To overcome this problem, the solution proposed in this algorithm is to adjust the feedrate such that the travelling time is an integer multiple of the sampling time.

The trapezoidal velocity profile with piecewise constant acceleration have discontinuities which industrial systems, and especially robots, cannot follow, whatever the performances of the actuators. These discontinuities excite the mechanical structure in transitory stages and are responsible for a great part of the damage of the dynamic behavior (Erkorkmaz & Altintas, 2001). To overcome this default, different types of feedrate profiles can be planned. Modern CNC systems used a least S-shaped speed profile with piecewise constant jerk value. In the present work, speed convolution technique with linear FIR filter, presented in (Kong & Yang, 2005) or (Chang, 2005), is used to smooth the trapezoidal velocity profile calculated previously, taking account of Jerk limitations. Such methodology offers the advantage of easiness of implementation in Open CNC systems. Therefore, the jerk limited profile does not have to be calculated analytically, which is a computer time consuming task. The simple trapezoidal velocity profile can be reconstructed based on the velocities bounds  $F_0$ ,  $F_d$  and  $F_f$  calculated for each block by the existing CNC lookahead function. One notes that others elegant approaches can be used directly to generate a jerk limited trajectory in polynomial

form adapted for real time implementation (Boryga *et al.* 2009) (Osornio-Rios *et al.* 2007) (Zheng *et al.* 2009).

The linear FIR speed filter (Figure 3) can be expressed as

$$F'_k = \left( \sum_{i=1}^m c_i \right)^{-1} \sum_{i=1}^m c_i F_{k-i+1} \quad (3)$$

where  $c_i$  denotes the  $m$  filter weight coefficients and  $F'_k$  is the filtered feedrate reference. One can note that the weight coefficients used here are symmetrical ( $c_i = c_{m-i+1}$ ). In order to respect both the ending velocity  $F_f$  and the distance  $L$  to be traveled on the block, the trapezoidal velocity profile has to be adapted to the filter time constant. Firstly, to respect the ending velocity a constant velocity stage of value  $F_f$  with a time duration equal to the filter time delay  $mT_s$  is added at the end of the block. Secondly, to compensate for the additional motion, noted  $\Delta L$ , induced by this constant velocity stage, the remaining distance to be travelled has to be initially set to  $L - \Delta L$ . Considering the symmetry property of the weight coefficient,  $\Delta L$  can be expressed as  $\Delta L = F_f \frac{(m+1)}{2} T_s$ . Figure 4 shows the flowchart of the feedrate planning algorithm including this time filter compensation.

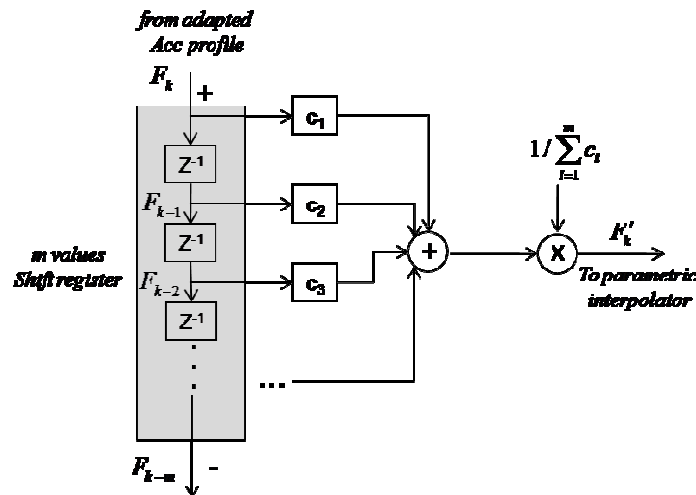


Fig. 3. Linear FIR speed filter.

Starting from the adapted trapezoidal velocity profile, a jerk limited trajectory is simply obtained by setting all weight coefficients equal to one. The constant jerk time resulting from this moving average filter is given by the filter time. Then, the maximum jerk value will be

$$J_M = A_M / (mT_s).$$

Others type of smooth profiles can easily be produced. For example, the squared-sine acceleration profile corresponds to the following coefficients

$$c_1 = \sin^2\left(\frac{\pi}{2m}\right); \quad c_i = \sin^2\left(\frac{i\pi}{2m}\right) - \sin^2\left(\frac{(i-1)\pi}{2m}\right) \quad i = 2, \dots, m$$

and the resulting maximum jerk value will be  $J_M = \pi A_M / (2mT_s)$ . Linear FIR filters can also be cascaded to obtain for example snap limited trajectory (snap is the jerk time derivative). In this case, noting  $m$  and  $n$  are the first and second filter lengths respectively and assuming  $2n \leq$

$m$ , the maximum jerk and snap value can be calculated with  $J_M = A_M / ((m-n)T_s)$  and

$$S_M = A_M / (n(m-n)T_s^2).$$

Figure 5 shows an example of different velocity profiles resulting from this strategy with the same filter time. Considering now a constraint on the maximum jerk value, the constant jerk limited profile leads obviously to the minimum time movement.

It will be seen in Section 3 that this maximum jerk value for a constant Jerk limited trajectory could have a predictable effect on the vibrations of the system. One can note that for others profiles and especially for snap limited trajectory, it becomes very difficult to give a physical meaning to the constraints on higher order time derivatives. For all these reasons, the jerk limited trajectory is used in the following for the validation of the algorithm.

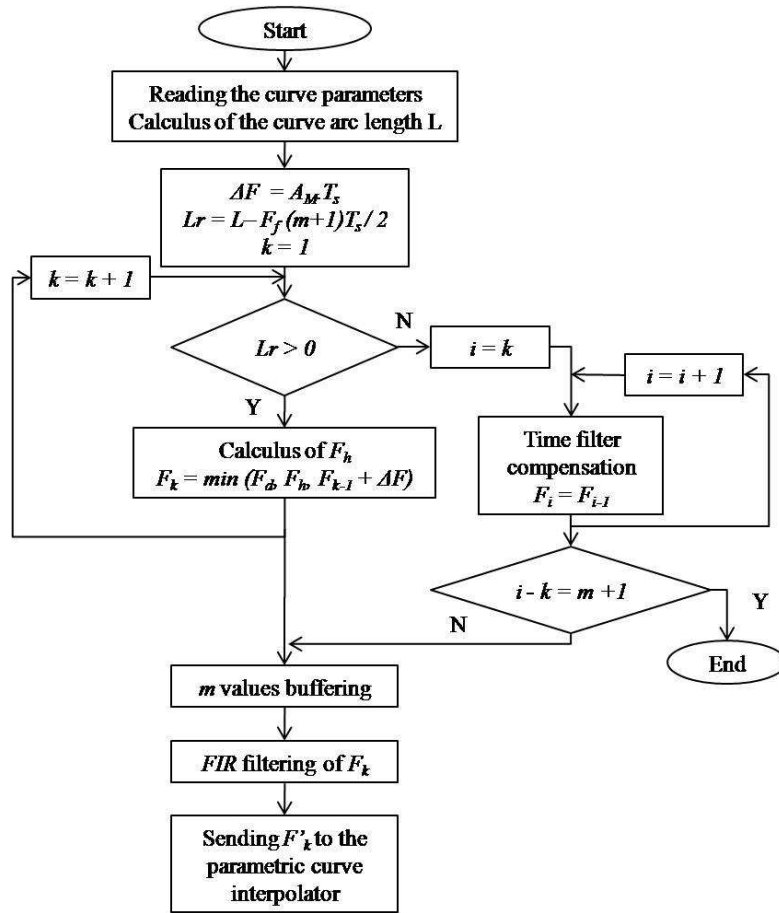


Fig. 4. Flowchart of the feedrate planning method.

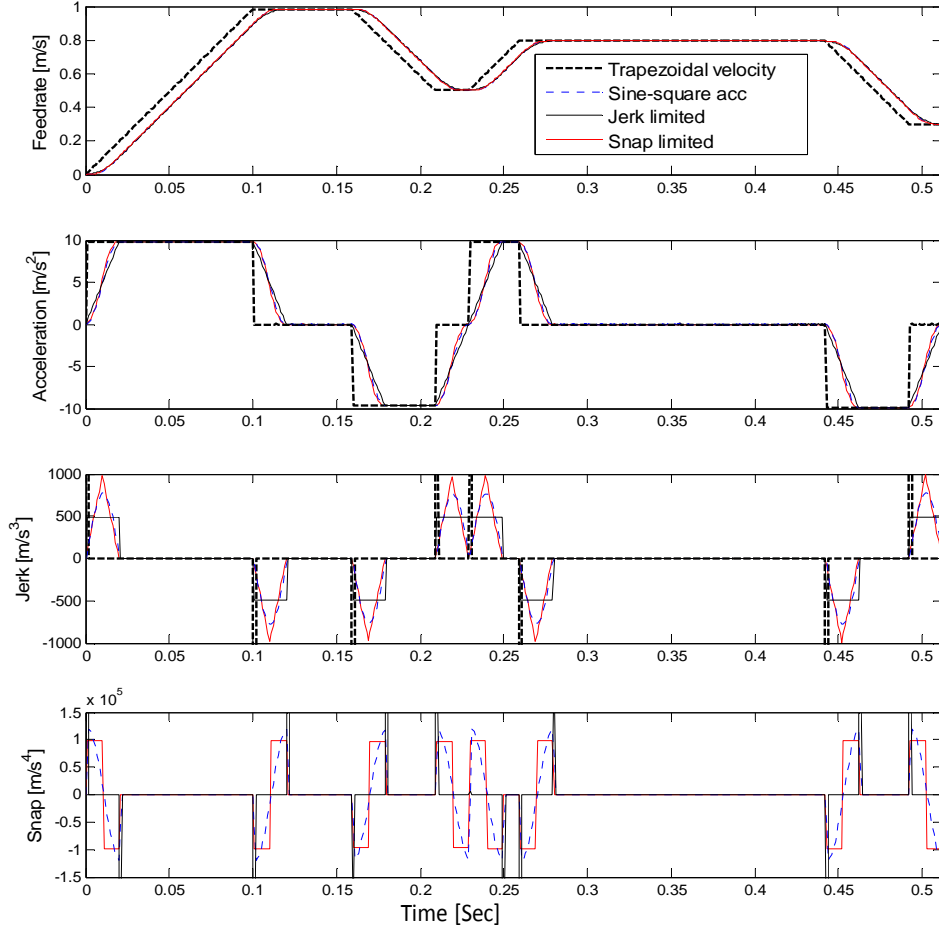


Fig. 5. Feedrate profiles and their time derivatives resulting from the feedrate planning algorithm (example with two connected blocks with different desired feedrate values and extremum conditions).

## 2.2 Parametric curve interpolator

In this paper tool-paths or Tool Center Point paths are presumed to be designed in CAD systems with parametric forms such as NURBS or L1 Splines (Auquier *et al.* 2007).

Considering such a parametric curve  $C(u)$ , the successive parameter  $u$  corresponding to the controller sampling period  $T_s$  is evaluated from the calculated feedrate  $F(t)$  using numerical integration

$$F(t) = \left\| \frac{dC(u(t))}{dt} \right\| \quad (4)$$

with  $\|\cdot\|$  denoting the Euclidean norm in Cartesian 3D space. As the function  $u(t)$  is a strictly monotonic increasing function, then the parametric velocity can be expressed as

$$\frac{du}{dt} = \frac{F(t)}{\left\| \frac{dC(u)}{du} \right\|} \quad (5)$$

Since the parametric speed cannot be expressed in closed form for common parametric curves, Taylor's series approximation is classically used. For computing efficiency the first order Taylor's approximation is used. Noting  $u_k$  the parameter at time  $t = kT_s$ , the next numerical parameter is given by

$$u_{k+1} = u_k + \frac{F_k}{\left\| \frac{dC(u)}{du} \right\|_{u=u_k}} T_s \quad (6)$$

One notes that the approximation of the parametric velocity can induce feedrate fluctuations that become sensitive for very short segments lengths. In such case, adding the second order term is a solution, which has proven to be efficient in (Mohan *et al.* 2008). Then, the Cartesian position references that lie on the original curve can be generated in real-time by replacing the parameter value given by (7) in the parametric curve expression. In order to achieve multi-joints control, robot joints references must be derived by means of the inverse kinematics model.

### 2.3 Inverse kinematic model

The Inverse Geometric Model of the studied robot is derived analytically by using classical Denavit-Hartenberg convention. The main difficulty of the inverse geometric model is that for a desired end effector configuration (position and orientation); there are eight solutions in joint space. In this work only one solution is taken into account. This solution corresponds to configuration of machining on a table placed in front of the robot (see figure.6).

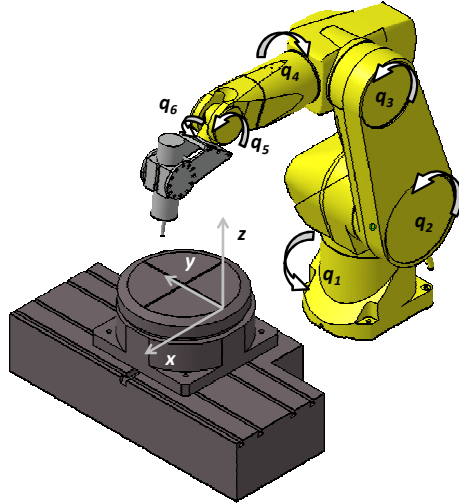


Fig. 6. Machining configuration.

All manipulators have singularities at the boundary of their workspace, and most have loci of singularities inside their workspace (Craig, 1989). For the machining robot used in this study, workspace boundary singularities are avoided by limiting its task space and positioning it away from the workspace boundaries. During machining process, the last three axes are mainly concerned with the work space interior singularities problem. This singularity can be caused by the two axes 4 and 6 lining up ( $q_5=0$ ). This can be avoided by limiting the rotation of the joint 5. In other cases, specific machining strategies adapted to the work-piece have to be developed and can be incorporated into CAM software. The optimization of the robot configuration and the definition of the switching paths between two configurations are out of the scope of this paper.

#### 2.4 Joint Constraints and Feedrate Optimization

The last stage of the motion planning in Cartesian space consists in checking that the joint kinematic constraints of axis- $j$  expressed by means of upper bounds on velocity  $V_{Mj}$ , acceleration  $A_{Mj}$  and jerk  $J_{Mj}$  are not violated

$$|\dot{q}_j(t)| \leq V_{Mj}; \quad |\ddot{q}_j(t)| \leq A_{Mj}; \quad |\dddot{q}_j(t)| \leq J_{Mj}; \quad j = 1, \dots, 6 \quad (7)$$

In the developed methodology, the desired feed value  $F_d$  is the only optimization parameter (as classically done in the machine-tool field). When the desired feedrate of a segment or curve is incompatible with (8), a bisection search method finds efficiently a kinematically feasible feedrate. The range of the search space is bounded by zero at the bottom, and the desired feedrate at the top. The search algorithm bisects the feed search space iteratively until a feasible solution is found within a specified tolerance. This method was used for computational efficiency reason, but such optimization problem can be solved with fast sequential quadratic programming or heuristic methods. Finally, the joint position corresponding to the kinematically compatible feed can be transferred at each sampling time to the joint position control.

### 3. INFLUENCE OF THE MAXIMUM JERK VALUE ON THE VIBRATION

Compared to acceleration limited profile, jerk limited profile reduces endpoint vibration and in some cases can totally suppress residual vibration. As demonstrated in (Barre *et al.* 2005), the maximum jerk value for an axis can be chosen to suppress vibration induced by the dominating vibratory mode of this axis. A simple explanation is based on the complex poles cancellation principle. The jerk limited law is a multiswitch bang-bang law, which can be represented in continuous domain as a sum of time delayed step function

$$P(s) = \frac{J_M}{s^4} \cdot F(s); \quad F(s) = \sum_{i=1}^n A_i \cdot e^{-sT_i} \quad (9)$$

with  $P(s)$  the axis position reference,  $J_M$  the maximum jerk value for this axis,  $n$  is the number of commutation ( $n = 4, 6$  or  $8$  according to dynamic limitations).  $F(s)$  is a time delayed filter, the coefficients  $A_i$  take their values in the ensemble  $\{1, 2, -2, -1\}$  and  $T_i$  are the switching



time. The cancellation of the conjugated complex poles associated to a dominating vibratory mode is obtained by the presence of conjugated complex zeros at the same location in the delayed filter

$$\sum_{i=1}^n A_i \cdot e^{-sT_i} \Big|_{s=-\zeta_n \omega_n \pm j\omega_n \sqrt{1-\zeta_n^2}} = 0. \quad (10)$$

Assuming a lightly damped mode ( $\zeta_n = 0$ ), a trivial solution to (10) consists in choosing the time duration between two commutations (jerk time) equal to a multiple of the natural period of the vibrational mode. In other words, the minimum time of the moving average filter used in the feedrate planning can be chosen equal to the dominating natural period. Figure 7 (a) shows the residual vibration of the end-effector resulting from a simple rotation of the first joint around Z axis of world frame. In this configuration, the flexibility of the first joint induced a flexural motion of the end-effector associated to a modal frequency near 6 Hz. According to the previous remarks, the jerk time of 160 ms (6,3Hz) leads to a motion without residual vibration. One notes that the vibration level during the movement is significantly decreased too. Indeed, if jerk time is finely tuned, vibration can only occur during constant jerk stages. This result corresponds to specific case, and it isn't valid for general applications where the robot realizes continuous paths. Because, in this method there is only one degree of freedom to adjust the curvilinear jerk, while, when the end-effector tracks a path on the operational space the six axes of the robot contribute to the movement. This setting can therefore lead to an improvement in vibration behavior if a single axis dominates the response of the end-effector. On the other hand, the eigenfrequencies of the robot depend on its configuration. However, assuming a reduced workspace these variations can reasonably be neglected.

Experimental modal analysis was conducted on the studied robot. It demonstrates a significant influence of the first, second, and third links of the robot, producing lightly damped poles in

the neighbourhood of 10 Hz and 34 Hz. In this study, the workspace is included in a cube of 500 mm and the modal frequency variations are below 8 %. Figure 7 (b) shows the result of a movement along the Y axis. In this particular case, the axis n°1 contributes mainly to the displacement of the end- effector. The frequency associated to the flexibility of the joint 1 is close to 10 Hz, which gives a period of optimum curvilinear jerk about 100 ms. One notes that such vibration reduction cannot be obtained for movements in X or Z direction, because of the vibratory coupling between each robot axis. Then, as a compromise the filtering time will be chosen equal to the first natural period (i.e 100 ms).

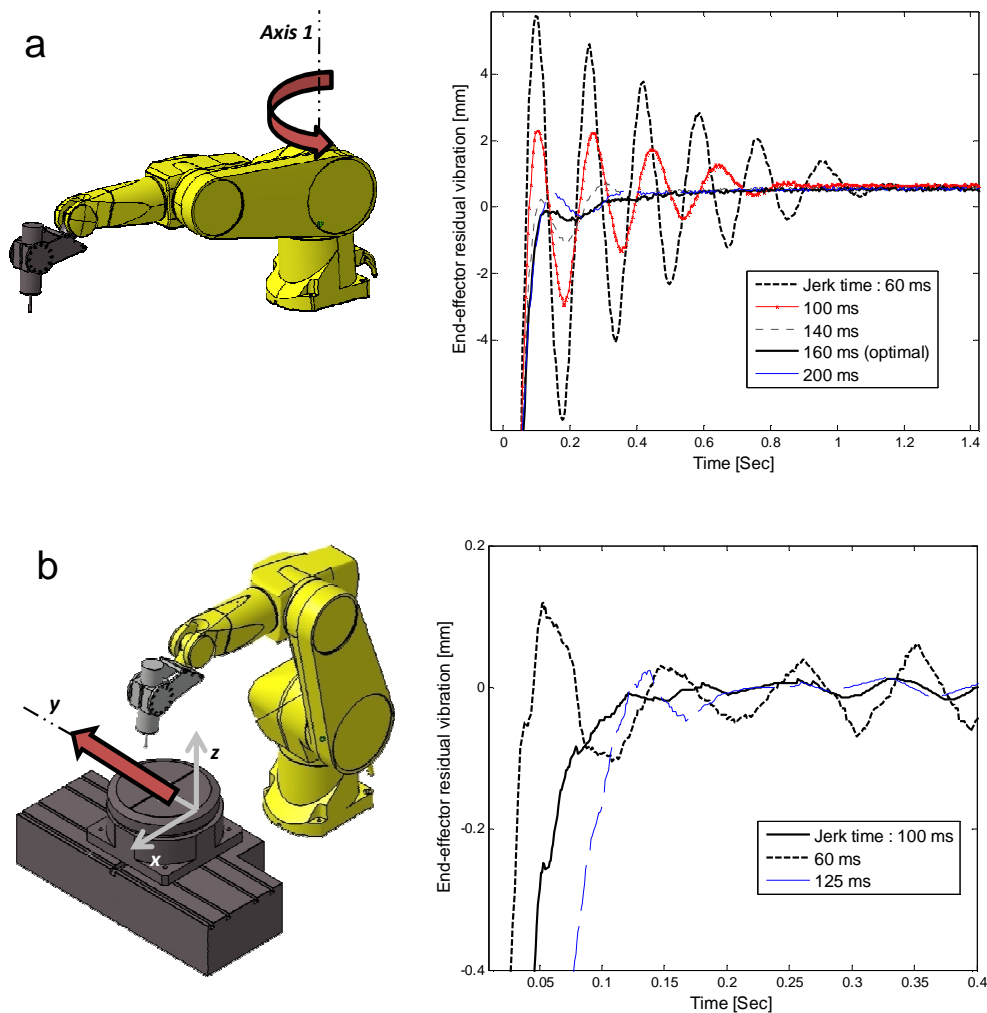


Fig. 7. Jerk time influence on residual vibration: (a) joint motion around axis 1, (b) Cartesian motion along Y axis.

## 4. EXPERIMENTAL RESULTS

### 4.1. Reference path

In this section, the effectiveness of the method will be demonstrated by testing it on a real industrial robot. The proposed feedrate planning algorithm assumes that the reference path has a continuous curvature variation. If it is not the case, the system goes to full stop at the curvature discontinuities. To avoid these problem the reference path has to be interpolated incorporating this constraint with the specified contour error before planning the trajectory. These aspects are not developed in this paper. The theoretical geometric path used for demonstrations is a logarithmic spiral in the XY plane of the robot world frame (see figure.8). This simple curve was chosen for its geometric properties, especially for the monotonous curvature variation and the exact analytic calculation of its arc length. In addition, this path can be considered similar to paths used for pocketing processes.

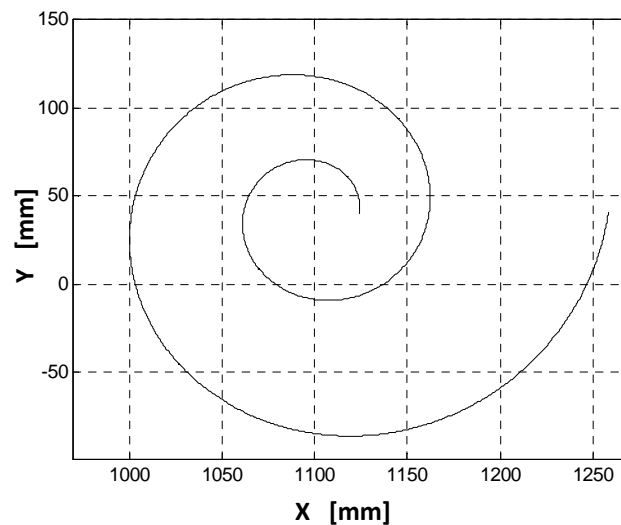


Fig. 8. Logarithmic Spiral, the Theoretical Path.

### 4.2. Trajectory elaboration

The robot used in this work is a six-axis vertical articulated industrial robot (RX170B from Stäubli Robotics) shown in Figure.1 and 6. A high-speed motor spindle was directly installed on the sixth axis, without any modifications of the robot mechanical structure. This robot is intended to realize machining processes like milling, deburring, drilling...etc. The robot has

some opening in its controller. It is particularly possible to impose joints trajectories for the six axes. The proposed trajectory planning is done offline because of its computational complexity. Nevertheless, since the trajectory is already known a look-ahead function can be generated and integrated into the control system for on-line implementation. The resulting joint trajectories are sampled to  $T_s = 4\text{ms}$ . Then, joint references are read sequentially by the robot controller. The motion parameters and joint constraints, such as feedrate, acceleration, and jerk are summarized in Table 1.

Table 1. Experimental parameters

Logarithmic Spiral $R = a \times e^{b \cdot \theta}$	$a=24, b=0.15, \theta = 4\pi$
Maximum Feedrate [m/sec]	1
Maximum Acceleration [m/sec <sup>2</sup> ]	1
Filtering time (jerk time) [sec]	0.08
Joints speed limits [deg/sec]	65,65,65,90,100,150
Joints Acceleration limits [deg/sec <sup>2</sup> ]	140,110,150,450,300,700
Joints Jerk limits [deg/sec <sup>3</sup> ]	2500

A smooth motion law for the cutting tool along the desired path (figure.7) is generated by the method developed in this paper. The feedrate is intended to be constant, as usually wanted for machining applications (figure.9). The corresponding joint trajectories of axes 1 and 2 are presented in figure.10.

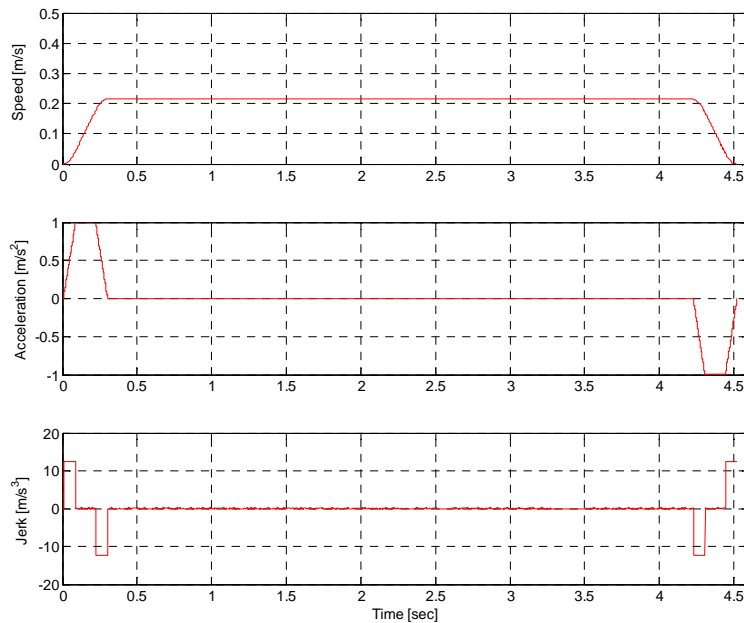


Fig. 9. Tool Motion Law with constant feedrate.

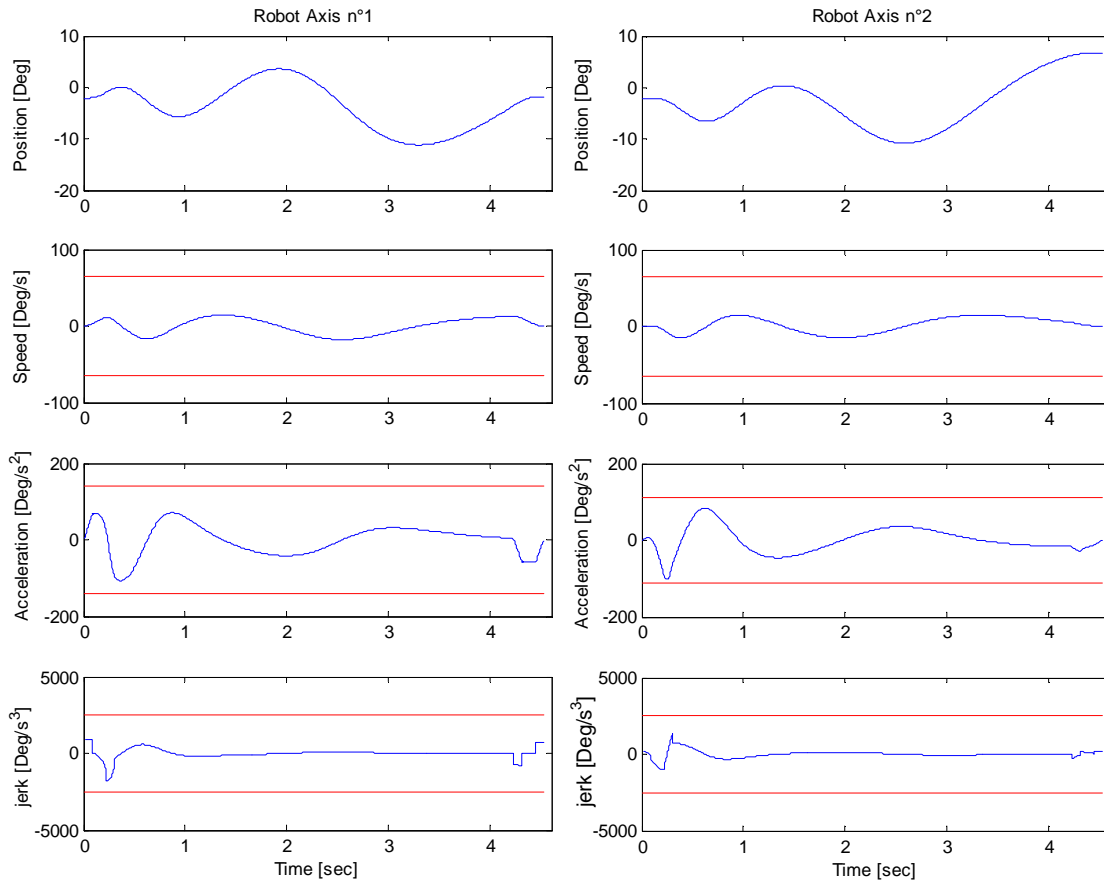


Fig. 10. Robot axis 1, 2 reference trajectories

It has to be noted that the maximum feedrate has not been reached. This is due to the saturation of axis 2, because of an important curvature variation at the beginning of the trajectory. This curvature variation becomes less important by getting far from the start point. So the tool feedrate is minimized to overcome this overrun of axis 2 speed at the beginning of the trajectory. In the first algorithm, constraints verification is done globally, so the minimized feedrate will be maintained over the entire trajectory. Maintaining a constant feedrate along the path can be coherent when trying to ensure a fine quality of the surface. But, in most cases, the optimized feedrate is less than the desired one, it is therefore essential to modulate it. A simple solution consists of segmenting the curve. The verification of constraints is performed locally on each segment. The segmentation points can be

automatically calculated during the geometric interpolation of the path considering for example a maximum curvature variation criterion.

The path used in this study was segmented into four and eight curve segments. Figure 11 shows the variation of the curvature along the trajectory. Figure 12 illustrates tool motion laws on the segmented trajectories. The feedrate evidently increases when the tool passes from one segment to another, since the curvature decreases. Table 2 presents the traveling time over the trajectory with and without segmentation. It can be noticed that by segmenting the geometric path in four curves the total traveling time is decreased of 26.8%. On the other hand, segmenting the path in eight curves doesn't have a great effect on traveling time because of the saturation of axis 2 on the test curve. The corresponding planned trajectories of axis 2 are presented in Figure 13.

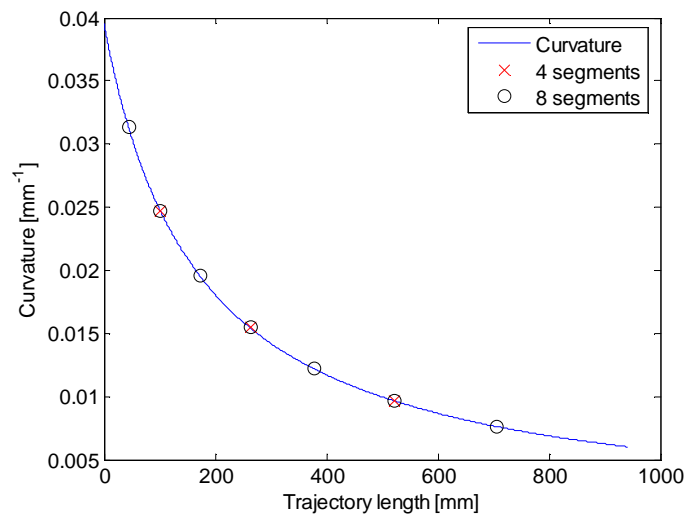


Fig. 11. Curvature variation and cutting points

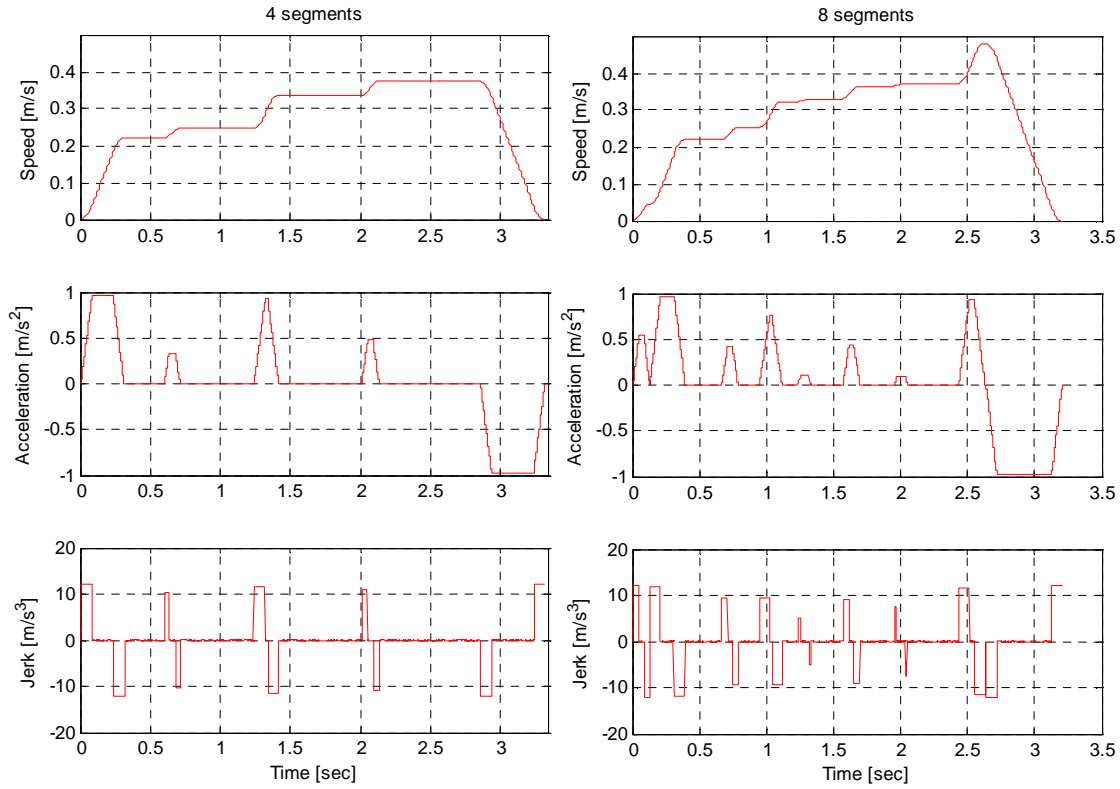


Fig. 12. Tool Motion Laws on segmented trajectories

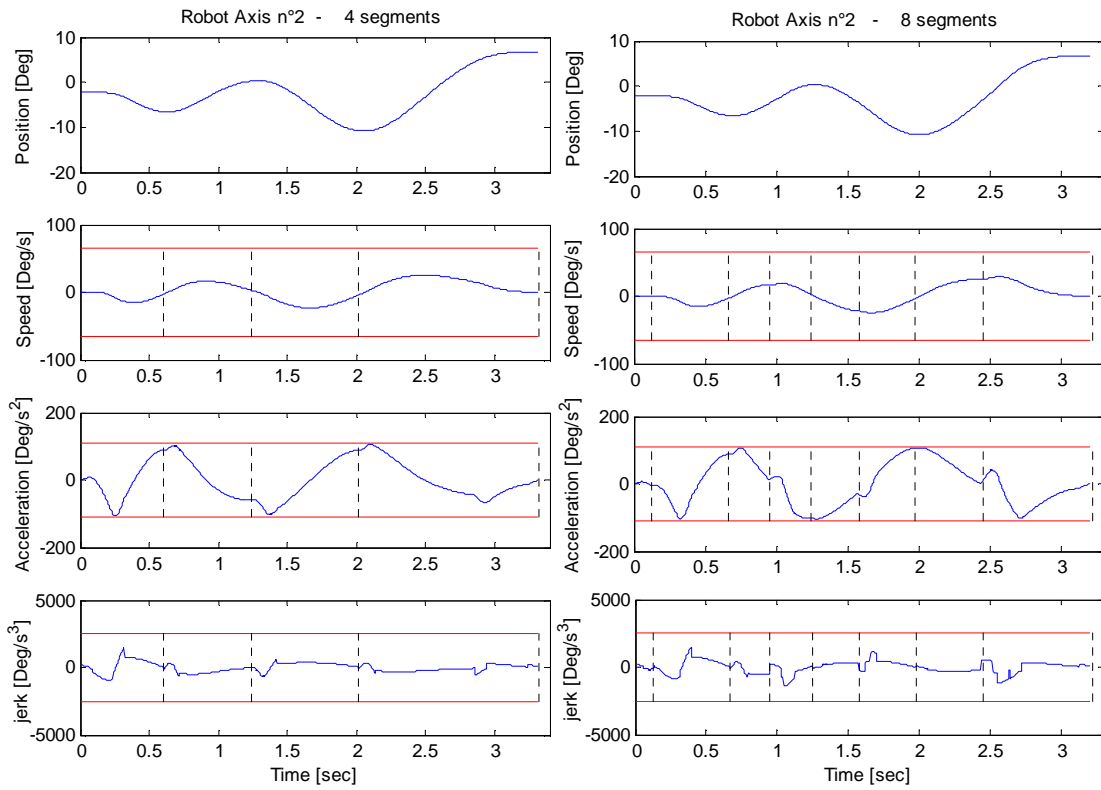


Fig. 13. Axis 2 trajectories

Table 2. Traveling time and max. federate according to segments number.

<i>Segments N°</i>	<i>Traveling Time[sec]</i>	<i>Max Feedrate[mm/sec]</i>
1	4.52	210
4	3.31 (- 26.8 %)	370 (+76.2 %)
8	3.2 (-29.2 %)	480 (+128 %)

#### 4.3. Results

The three trajectories were tested on the machining robot. Figure 14 shows an example of the machined spiral. However, in order to clearly validate the control of the feedrate with the proposed trajectory planning methodology, the trajectories of the robot end-effector have been measured without machining by means of a 3D measurement system, CompuGauge<sup>TM</sup>. It is a simple measuring instrument consisting of two triangulation beams and software for data acquisition (Dyналog<sup>TM</sup>). The sampling rate of data acquisition is 1 kHz (1000 samples per second).



Fig. 14. Machining operations with the robot (cutting depth of 3 mm in resin material).

Figure 15 shows the measured end-effector feedrates. The dashed black curves represent the references of the three feedrates for the same spiral with one, four, and eight segments. These experiments show that using the proposed feedrate planning result in a predictable control of the end-effector feedrate, which can contribute to the improvement of continuous machining operations with industrial robots.



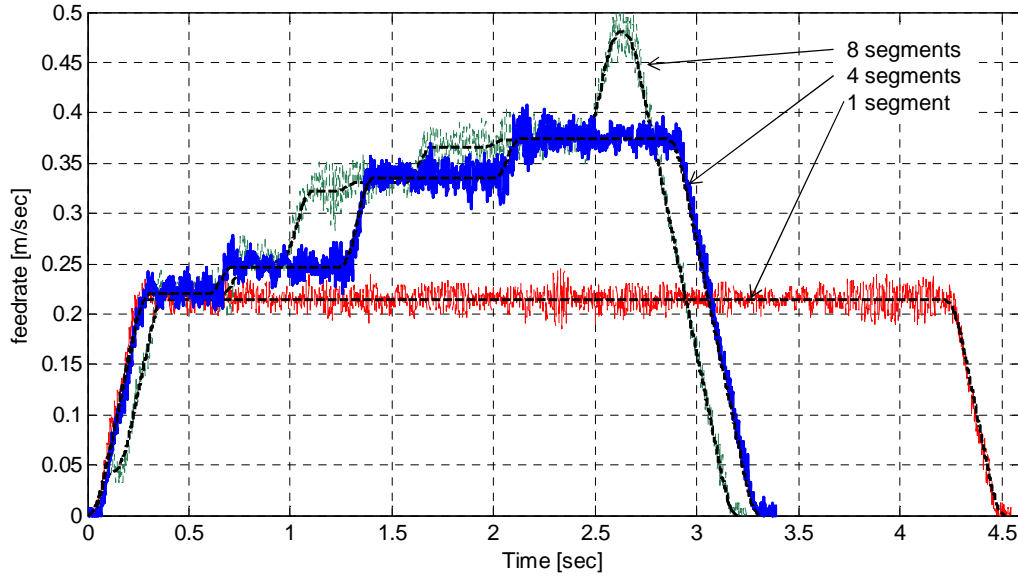


Fig. 15. Measured end-effector feedrates.

Figure 16 illustrates these trajectories realized by the robot end effector. Figure 17 shows the resulting contour errors along the three trajectories and the maximum and Root Mean Square contour errors are summarized in Table 3. The contour error is defined as  $\Delta R = |R_{mes} - R|$  where  $R$  is the theoretical radius defined in Table 1 and  $R_{mes}$  is the radius of the measured point. Experimental results show that the first trajectory ensures a better path tracking, due to the fact that the feedrate is constant most of time. On the other hand, the two others induced higher joint accelerations, which excite more the vibration modes of the mechanical structure, so the path tracking is damaged. For the four segments curve, the travelling time is decreased by 27%, but the RMS contour error is increased by more than 35%.

Table 3. Contour error

<i>Segments</i> <i>N°</i>	<i>RMS contour</i> <i>error[mm]</i>	<i>Max contour</i> <i>error[mm]</i>
1	0.2817	0.8531
4	0.3822 (+ 35.7%)	0.9394
8	0.4038 (+43.3%)	1.053

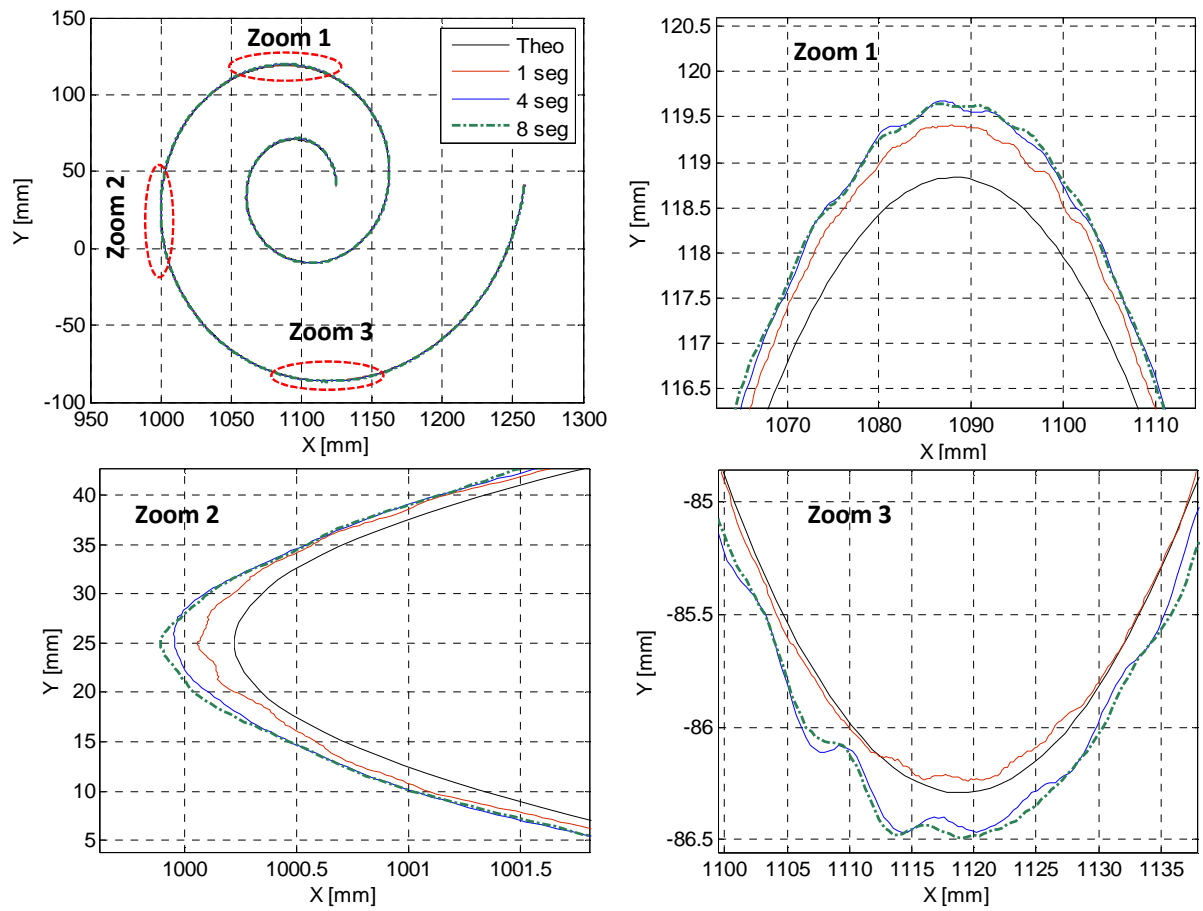


Fig. 16. Measured Paths of the end-effector.

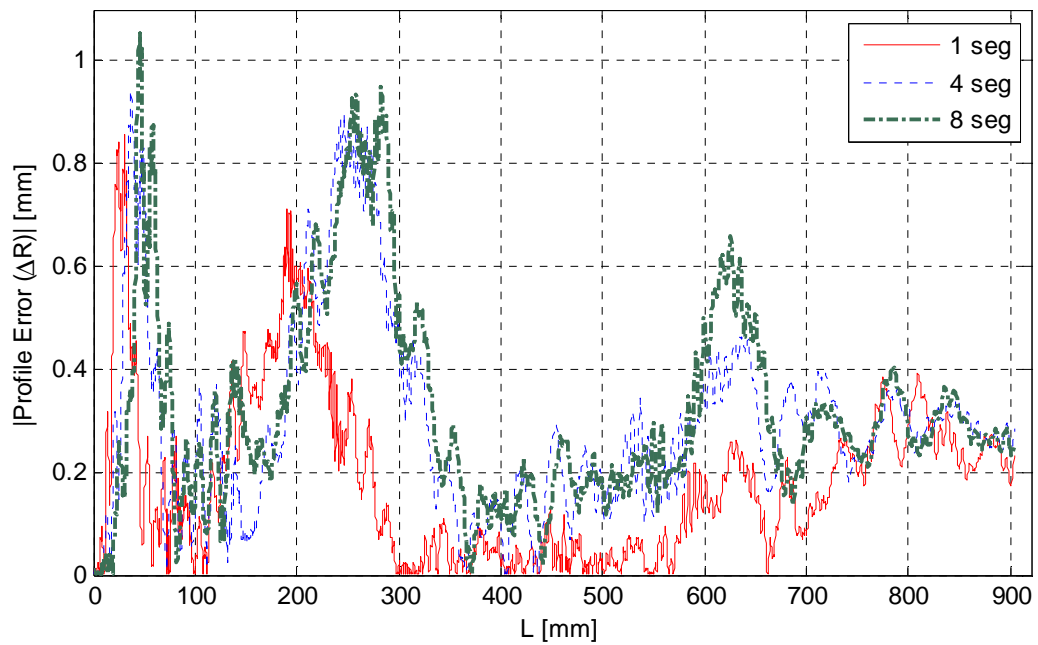


Fig. 17. Contour errors.

## 5. CONCLUSIONS

In this paper, a feedrate planning method adapted for continuous machining with industrial robot has been proposed. Starting from a parametric representation of the tool paths, this method generates a smooth jerk limited law of motion for the tool, respecting the robot joints constraints. Different trajectories along a logarithmic spiral has been planned and tested in a six-axis industrial robot. It was shown that the feedrate planning strategy is an effective solution for controlling the tool motion for a robot. This provides a first step toward the improvement of machining with an industrial robot. Experimental results underline the presence of path tracking errors as well. Reducing these errors and improving the positioning accuracy of the robot end-effector are the objectives of future works.

## ACKNOWLEDGMENT

The authors want to thank Stäubli for providing the necessary information of the controller, Dynalog for its contribution to the experimental validations and X. Helle for its material contributions.

## REFERENCES

- Abele, E. Weigold, M. & Rothenbücher, S. (2007). Modeling and identification of an industrial robot for machining applications. *Annals of the CIRP* 56(1), 387-390.
- Auquiart, P. Gibaru, O. Nyiri, E. (2007). C1 and C2-continuous polynomial parametric Lp splines ( $p \geq 1$ ), *Computer Aided Geometric Design*, 24, 373-394.
- Barre, P-J. Bearee ,R., Borne ,P. & Dumetz, E. (2005). Influence of a jerk controlled movement law on the vibratory behaviour of high-dynamics systems. *Journal of Intelligent and Robotic Systems*. 42, 275-293.
- Boryga, M., Grabos, A. (2009). Planning of manipulator motion trajectory with higher-degree polynomials use. *Mechanism and Machine Theory*. 44(7), 1400-1419.
- Chang, Y-F. (2005). Design and implementation of a linear jerk filter for a computerized numerical controller. *Control Engineering Practice*. 13, 567–576.
- Chettibi, T. Lehtihet, H.E. Haddad, M. Hanchi, S.(2004). Minimum cost trajectory planning for industrial robots .*European Journal of Mechanics A/Solids*. 23, 703–715.
- Craig, J. (1989). *Introduction to robotics - Mechanics and control* 2nd edition . Addison-Wesley Publishing Company.
- Damak, M. Grosbois, J. De Smet, P. (2004). Vision robot based absolute accuracy measurement-calibration and uncertainty evaluation . *THE 35th INTERNATIONAL SYMPOSIUM ON ROBOTICS*. Paris-Nord Villepinte .
- Elatta, A.Y. Pei Gen, Li. Liang Zhi, Fan. Daoyuan, Yu & Fei, Luo. (2004). An overview of robot calibration. *Information Technology Journal*, 3(1) , 74-78.
- Erkorkmaz, K. & Altintas, Y. (2001). High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation. *International Journal of Machine Tools & Manufacture*. 41,1323–1345.

Gasparetto, A. & Zanotto, V. (2008). A technique for time-jerk optimal planning of robot trajectories, *Robotics and Integrated Manufacturing*. *Robotics and Computer-Integrated Manufacturing* 24(3), 415-426.

Goto, S. Usui, T. Kyura, N. & Nakamura, M.( 2007). Forcefree control with independent compensation for industrial articulated robot arms. *Control Engineering Practice*. 15, 627–638.

Hakvoort, W.B.J. Aarts, R.G.K.M, van Dijk, J. & Jonker, J.B.(2008) Lifted system iterative learning control applied to an industrial robot .*Control Engineering Practice*.16, 377–391.

Huey, J. Sorensen, K. & Singhose W.(2008). Useful applications of closed-loop signal shaping controllers. *Control Engineering Practice*. 16, 836–846.

Khalil, W. & Dombre, E.(2004). *Modeling, Identification and Control of Robots*. Butterworth.

Kong, T; & C. H. Yang, D. (2005) A Hybrid Parametric Interpolator for Complete Motion Profile Generation. *Computer-Aided Design & Applications*. 2, 105-114.

Lambrechts, P. Boerlage, M. & Steinbuch, M. (2005). Trajectory planning and feedforward design for electromechanical motion systems . *Control Engineering Practice*. 13(2), 145–157.

Meng, Y. & Zhuang, H.(2007). Autonomous robot calibration using vision technology. *Robotics and Computer-Integrated Manufacturing*. 23(4), 436–446.

Mohan, S. Kweon, S-H. Lee, D-M. Yang. S-H. (2008). Parametric NURBS Curve Interpolators: A review. *International journal of precision engineering and manufacturing*, 9(2), 84-92.

Osornio-Rios, R. A., Romero-Troncoso, R. J., Herrera-Ruiz, G., Castaneda-Miranda, R. (2007). Computationally efficient parametric analysis of discrete-time polynomial based acceleration-deceleration profile generation for industrial robotics and CNC machinery. *Mechatronics*. 17, 511-523.

Ostring, M. Gunnarsson, S. & Norrlof, M. (2003). Closed-loop identification of an industrial robot containing Flexibilities. *Control Engineering Practice*. 11, 291–300.

Pan, Z. Zhang, H, Zhu, Z. & Wang, J.(2006). Chatter analysis of robotic machining process. *Journal of Marerials Processing Technology*.173, 301-309.

Shiakolas, P. Conrad, K.L. Yih, T.C.(2002). On the accuracy, repeatability, And degree of Influence of kinematics parameters for industrial robots. *International Journal Of Modelling and Simulation*. 22(4), 245-254.

Zheng, C., Su, Y., Müller, P.C. (2009). Simple online smooth trajectory generations for industrial systems, *Mechatronics*, 19(4), 571-576.

Ziaei, K. Liya, Ni. & W.L.Wang, D. (2009). QFT-based design of force and contact transition controllers for a flexible link manipulator. *Control Engineering Practice*. 17, 329–344.

Figure legends

Fig. 1. Machining robot with high-speed spindle.

Fig. 2. Cartesian space motion planning strategy.

Fig. 3. Linear FIR speed filter.

Fig. 4. Flowchart of the feedrate planning method.

Fig. 5. Feedrate profiles and their time derivatives resulting from the feedrate planning algorithm (example with two connecting blocks or curves with different desired feedrate values and extremum conditions).

Fig. 6. Machining Configuration.

Fig. 7. Jerk time influence on residual vibration: (a) joint motion around axis 1, (b) Cartesian motion along Y axis.

Fig. 8. Logarithmic Spiral, the Theoretical Path.

Fig. 9. Tool Motion Law with constant feedrate

Fig. 10. Axis 1, 2 reference trajectories

Fig. 11. Curvature variation and cutting points

Fig. 12. Tool Motion Laws on segmented trajectories

Fig. 13. Axis 2 trajectories

Fig. 14. Machining operations with the robot (cutting depth of 3 mm in resin material).

Fig. 15. Measured end-effector feedrates.

Fig. 16. Measured Paths of the end-effector.

Fig. 17. Contour errors.