Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 133 (2025) 412–417

www.elsevier.com/locate/procedia

20th CIRP Conference on Modeling of Machining Operations

Feed rate optimisation scheme in robotic machining operations for dynamic error compensation

Valentin Dambly[*,a], Bryan Olivier[b], Édouard Rivière-Lorphèvre[a], François Ducobu[a], Olivier Verlinden[b]

[a] *UMONS Research Institute for Materials Science and Engineering, Machine Design Production and Engineering Lab, Place du parc 20, Mons, Belgium*
[b] *UMONS Research Institute for Materials Science and Engineering, Theoretical Mechanics, Dynamics and Vibrations Lab, Place du parc 20, Mons, Belgium*

* Corresponding author. *E-mail address:* valentin.dambly@umons.ac.be

Abstract

The manufacturing sector demands shift towards parts with more complex geometries with the need for flexibility in production, driving interest to robotic machining. This advancing technology brings advantages like affordability, versatility, and ease of implementation, making it well-suited for agile production environments. Nevertheless, robotic machining struggles with accuracy issues due to the inherent flexibility of robots, which results in deviations and vibrations. The positioning error along a robotic machining trajectory is composed of two contributions: the steady state error and the transient. Initially generated from CAM software, the trajectory is considered as a path with a speed profile. It is then discretised in elementary sections, modelled with Hermite splines and connected by nodes. To address, offline, the lack of accuracy, an updated trajectory is computed by iteratively replacing these nodes space based on the error estimated from the dynamics simulation, strongly reducing the steady state error. However, in transient sections, the error reduction is not sufficient.

This research focusses on the impact of the feedrate modification in transient areas, typically the entrance and exit of tool in the workpiece. Specific speed profiles are defined for these sections by applying linear segments with parabolic blends expressed in terms of the curvilinear abscissa. An optimisation scheme is proposed to update their feedrate considering the node repositioning necessary to compensate tool-tip deviation. The investigation of the feedrate update is based on the results from virtual machining simulator including the robot dynamical model, responsible for steady-state and transient errors respectively and a cutter-workpiece engagement module.

Keywords: Robotic Machining; Offline Trajectory Correction; Feedrate Update; Robot Modelling

1. Introduction

In the current industrial landscape, robots are extensively used for tasks such as manipulation, assembly, welding and painting. The growing demands for manufacturing, including reducing mechanical assembly in favour of components with advanced geometries and materials, have driven the industry toward more flexible production methods. One way to meet these requirements is to perform machining operations with industrial robots since dexterity is one of their main features [1]. However, robotic operations are subject to accuracy problems, which has led the scientific community to focus on modelling flexibility and compensating for the deviations it entails. Based on either quasi-static or dynamic simulations, operational path is updated [2].

Feedrate scheduling consists in adjusting the feedrate dynamically along the tool path to optimise machining performance and ensure quality [3]. Typically, feedrate scheduling methods are used to optimise the material removal rate while guaranteeing a healthy operation, including a good surface condition of the workpiece and acceptable cutting forces, i.e. those that do not risk causing instability and premature tool wear [4]. Also, it is typical to update the feedrate for elaborated geometries, where complex tool motions are necessary [5–7]. In the field of robotic machining, the feedrate adaptation is used as online action to help an offline updated trajectory [2]. Moreover, in traditional or robotic machining, the feedrate update can be used to avoid unstable conditions as well, where it is known that a reduction of feedrate tends to stabilise the process.

10.1016/j.procir.2025.02.071

In this work, modifying the feedrate is considered not only for the sake of its stability aspects but mostly for its influence on improving the trajectory compensation, infamously necessary in robotic operations. Indeed, after the computation of compensated trajectory, there are remaining positioning errors in specific sections of the paths. Typically, it occurs when the robot is subjected to severe changes of loads (when the tool enters and leaves the workpiece) or cutting forces directions.

This research paper is a continuation of the trajectory compensation procedure for robotic machining [8]. The first section introduces the robotic machining technology and challenges in the industrial landscape. The second section outlines first the modelling approach for robotic milling operations, then proceeds to the description of the trajectory formalism to finally present the feedrate optimisation scheme. The third section shows the application of the feedrate on a documented case and explores the sensitivity of the remaining error to the shape of the trajectory sections whose feedrate have been modified. Finally, the conclusion about the efficiency of such approach is presented, commenting about the determination of updated trajectory sections.

Nomenclature

a_e Radial depth of cut [mm]
a_p Axial depth of cut [mm]
h_k Uncut chip thickness for tooth k [mm]
$\mathbf{H}_{i,j}$ Hermite Trajectory composed of Homogeneous transformation matrix nodes
J_k Jerk ($\frac{d^3x}{dt^3}$)
$K_{l,c}$ Cutting force coefficients with $l = t, r, a$ being the tangential, the radial and the axial directions ,respectively [MPa]
$K_{l,e}$ Edge force coefficients with $l = t, r, a$ being the tangential, the radial and the axial directions, respectively [N/mm]
$\mathbf{M}_q$ The system inertia matrix
$\mathbf{h}_q$ Gathering of Coriolis, gyroscopic and centrifugal effects of a multibody system
N Rotation speed [RPM]
s Curvilinear variable [m]
f_z Feed per tooth [mm/th]
$f_{d,i}$ Feedrate of the path section i [mm/s]
τ External torques applied on the systems bodies

2. Modelling

This section is dedicated to the description of the robot model, the cutting force module and the simulation of the machining process. The trajectory update methodology is presented, from path modification up to the investigations of feedrate update. The simulation scheme is represented in Figure 1 with the insertion of the offline trajectory update scheme.

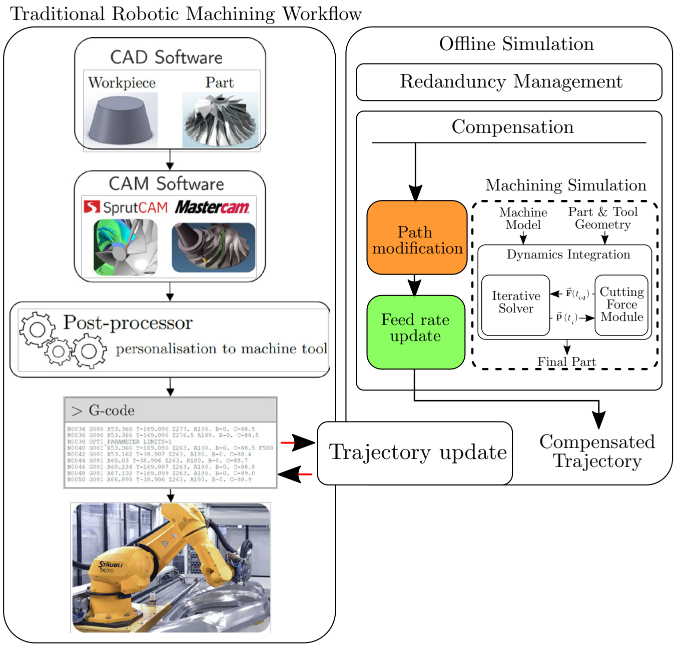

Fig. 1. Robotic machining workflow and the proposed offline trajectory update based on dynamic simulations.

2.1. Robot modelling

As aforementioned, the offline update of the trajectory is based on dynamic simulation of the machining operation, hence requiring a dynamic model of the robot as well as a cutting module suited for such simulations. In UMONS robotic machining cell, the industrial robot dedicated to milling is a Stäubli TX200. The dynamics of the system is expressed through its equations of motion. Developing the principle of virtual power for the multibody structure, the generalised equations for a system composed of rigid bodies are

$$\mathbf{M_q}(\vec{\mathbf{q}})\ddot{\vec{\mathbf{q}}}(t) + \mathbf{h_q}(\vec{\mathbf{q}}, \dot{\vec{\mathbf{q}}}) = \tau(t) \tag{1}$$

where $\mathbf{M_q}(\vec{\mathbf{q}})$, $\mathbf{h_q}(\vec{\mathbf{q}}, \dot{\vec{\mathbf{q}}})$ and $\tau(t)$ are defined in minimal coordinates. $\mathbf{M_q}(\vec{\mathbf{q}})$ is the mass matrix, $\mathbf{h_q}(\vec{\mathbf{q}}, \dot{\vec{\mathbf{q}}})$ is the vector gathering the Coriolis, gyroscopic and centrifugal forces and $\tau(t)$ the contribution of the forces applied on the system [9]. Vector $\vec{\mathbf{q}}$ regroups the actuated degrees-of-freedom $\vec{\mathbf{q}}_a$ and the unactuated degrees of freedom $\vec{\mathbf{q}}_u$.

As discussed in the literature [10–14], the main areas introducing flexibility in the robot are the joints, representing up to 80% of the overall flexibility of the robot and the long links (typically arm and forearm), whose flexible behaviour is responsible for the remaining 20% [15]. In this study, the articular (or joint) flexibility is modelled with the tri-axial flexibility approach [16, 17]. It consists in inserting a virtual joint (VJM [18]) at each articulation in the form of a tri-directional set of torsional spring-dampers in orthogonal directions. The articular flexibility is then modelled with the addition of 18 unactuated degrees of freedom.

The links flexible behaviour is modelled with the corotational approach [19]. The corotational formulation relies on describing the motion of a flexible body using a simplified representation based solely on the positions of some nodes, selected at key positions. Typical key nodes are the interface ones, connecting the body to the rest of the mechanical system. Additional internal nodes can be introduced to improve the accuracy of the model. The deformation of the flexible body is then determined from the displacements of these nodes with respect to the reference frame (adding 6 unactuated degrees of freedom per flexible body considered). In the model of the TX200, the arm and the forearm are then modelled as equivalent beams whose nodes are located at these bodies joints. In summary, the flexible behaviour of the robot is represented with these 30 unactuated degrees of freedom (18 for the six joints and 12 for the two flexible bodies). The values of the spring-dampers have been identified by fitting the simulated frequency response functions from a measured set collected via experimental modal analysis with roving hammer and a two-stage optimisation algorithm [10].

This flexible multibody model is coupled with a machining simulator to assess the milling forces at each time step. The latter are computed following the Altintas mechanistic model, or linear force model [20]. This model consists in discretising the cutting tool along its axis into elements and compute the force contribution of each element through the following expression

$$dF_{l,i} = \mathbf{K}_{l,c} \cdot h_i \cdot dz + \mathbf{K}_{l,e} \cdot dS, \qquad l = t, r, a \tag{2}$$

with, for tooth i, dz the height of the elementary tool slice, dS the elementary cutting edge length and $K_{l,c}$, $K_{l,e}$ the shear force and the edge force coefficients respectively, for the indices t, r, a being the tangential, radial and axial directions respectively. Depending on the tool/material couple, these coefficients must be identified experimentally [21]. The values used for the following simulations detailed in section 3 (Table 1) have been obtained from an inverse identification method [22]. The key parameter in Equation 2 is the uncut chip thickness h_i, as it represents the amount of matter removed at each time step of the simulation. It is computed by considering the interference between the elementary volume swept by the cutting edges in motion (discretised) and a tri-dimensional network of dexels. The tri-dexel simulator presented in previous work is used to evaluate the evolution of the chip thickness, and hence the cutting forces [23].

The global milling force is computed as the sum of each of these elementary contributions as follows:

$$\vec{\mathbf{F}}_m = \sum_{i=1}^{n_s} \begin{Bmatrix} dF_x \\ dF_y \\ dF_z \end{Bmatrix}_{m,i} = \mathbf{R}_{m,i} \cdot \begin{Bmatrix} dF_t \\ dF_r \\ dF_a \end{Bmatrix}_i , \tag{3}$$

with

$$\mathbf{R}_{m,i} = \begin{bmatrix} -\cos\theta & -\sin\theta\sin\kappa & -\sin\theta\cos\kappa \\ \sin\theta & -\cos\theta\sin\kappa & -\cos\theta\cos\kappa \\ 0 & -\cos\kappa & -\sin\kappa \end{bmatrix}_i \tag{4}$$

and θ the edge rotation angle, κ the axial immersion angle and $dF_{t,r,a}$ respectively the tangential, radial and axial elementary force contributions in the local tooth frame indicated by the subscript i. Once computed, these milling forces $\vec{\mathbf{F}}_m$ are applied at the tool centre point (TCP) body of the multibody model, corresponding to the term $\tau_m = \mathbf{J}_i^T\{\mathbf{F}_m\}$ for $i = n_{body,tcp}$, part of $\tau(t)$ from Eq. 1. The machining operation is then simulated by integrating the equations of motion with the α-generalised scheme using a strong explicit coupling with the cutting force module, as described in [23].

As presented in the literature, in robotic machining operations, even when stable, the tool is subjected to deviations [1, 12]. These deviations can be compensated offline, by editing the trajectory sent to the robot according to the deviations estimated from the dynamic simulations. To achieve such compensation, the trajectory is updated by iteratively simulating the machining operation, driving the error to its minimum [24].

A trajectory can be considered as a path on which a speed profile is applied. For the compensation of the deviations, only the shape of the path is updated considering the positioning error. Editing the path allows to strongly reduce the error (up to 90% in Figure 3). However, the speed profile can also be updated. So far, the compensation is mainly carried out by modifying the TCP frame location, but, as shown with the simulation results from Figure 3, in some areas the error cannot be more decreased by solely changing the path.

2.2. Feedrate update

Once the compensation algorithm has computed a modified trajectory, the error profile is analysed. If there are still areas with an error over the maximum acceptable deviation, they are identified as ”troubled areas” and require further modifications, other than just the shape of the path. The goal of this section is then to investigate the option of reducing the error in troubled areas by locally modifying the speed profile therein.

In the literature, it is common to use linear segment with parabolic blending for trajectory (*lspb*) [25]. The point of the blending is to smooth transitions, i.e. in space to avoid sharp angles or time, for the speed profile. From an execution point of view, path blending has been introduced in operations to allow the effector to maintain its velocity as long as possible along a segment-composed path. As aforementioned, it is convenient to consider a trapezoidal velocity for the trajectory. A simple velocity trapeze leads to a step acceleration, which is not recommended in dynamic simulation. It is however more suitable and practical to look for a continuity class C^1 on acceleration. With a proper interpolation formulation of the trajectory, the path and the speed profile are decoupled, hence editing locally the speed profile is not modifying the shape of the path [8].

The typical expression of the trajectory corresponds to the integration in time of a jerk parameter J_k with k the section number. For the sake of illustration, the position, velocity, acceleration and jerk are gathered in Equation 5.

$$\begin{cases} s = \frac{1}{6}J_k.\Delta t^3 + \frac{1}{2}\ddot{s}_{k-1}.\Delta t^2 + \dot{s}_{k-1}.\Delta t + s_{k-1} \\ \dot{s} = \frac{1}{2}J_k.\Delta t^2 + \ddot{s}_{k-1}.\Delta t + \dot{s}_{k-1} \\ \ddot{s} = J_k.\Delta t + \ddot{s}_{k-1} \\ \dddot{s} = J_k \end{cases} \tag{5}$$

with $\Delta t = t - t_k$, for t_k the start of the section. The *lspb* is defined along the path, following the curve, hence the use of the curvilinear variable s in its formulation which gives the evolution of s, $\frac{ds}{dt}$ and $\frac{d^2s}{dt^2}$.

It must be specified that two error indices are used throughout the work on the compensation algorithm: the cord error and the accumulated error. The cord and accumulated errors are considered as:

$$e_{cord} = \max\left(\|\vec{\mathbf{e}}_{milling,t}\|\right), \text{ for } t \in [t_{m,0}, t_{m,f}] \tag{6}$$

$$e_{acc} = \sum_{k=1}^{n_{eval}} \|\vec{\mathbf{e}}_{milling,k}\|^2 \tag{7}$$

with n_{eval} the number of evaluations, $t_{m,0}$ and $t_{m,f}$ the moment the tool starts and stops cutting, respectively. The cutting error $\vec{\mathbf{e}}_{milling}$ is the difference between the expected TCP position and the actual position. In the case of linear trajectories, only the position is considered [24].

The feedrate update algorithm is presented in Figure 2 and described hereafter. The first step is to identify the problematic path sections. To detect them, the resulting error is filtered using a moving average. Then the sections of path where this error is above a threshold are saved to have their speed modified. The threshold value is set at 50 µm, which is the repeatability value of the Stäubli TX200. As one might expect, these areas are mainly located where there is a change of load or cutting forces direction. Indeed, it has been shown in the literature that the steady-state error can be compensated mostly by modifying the shape of the path but it is not the case for transient phases.

Once the sections are identified, a local *lspb* is applied on the top of the operation speed profile. The value of the speed level of each *lspb* are gathered and considered as the design variables of an optimisation problem, aiming to reduce the overall positioning error (Eq. 7). It must be noted that modifying the feedrate directly changes the cutting forces from the relation $f_d = f_z.z.N$, where the feed per tooth f_z is equivalent to the maximum chip thickness, being an image of the forces. The direct consequence of a different cutting force magnitude is the modification of the deviations at the tool-tip. This implies that, in these sections, the updated path, obtained in the "path modification" step of the offline workflow (Figure 1) is not valid anymore. Hence, in these sections, another compensation of the path positions is necessary at each iteration of the feedrate optimisation algorithm. The compensation of the path is carried with the modified mirroring approach [24].

The optimisation problem is then expressed as $\min_{f_{d,k}} e_{acc}$ such that $f_{d,min} < f_{d,k} < f_{d,max}$, where the error (Eq. 7) is the one obtained after the compensation loop. The optimisation method used is a Nelder-Mead non-linear simplex, since, from a physical point of view, the search is made around the initial feedrate so a deterministic method is suited for this kind of problems.

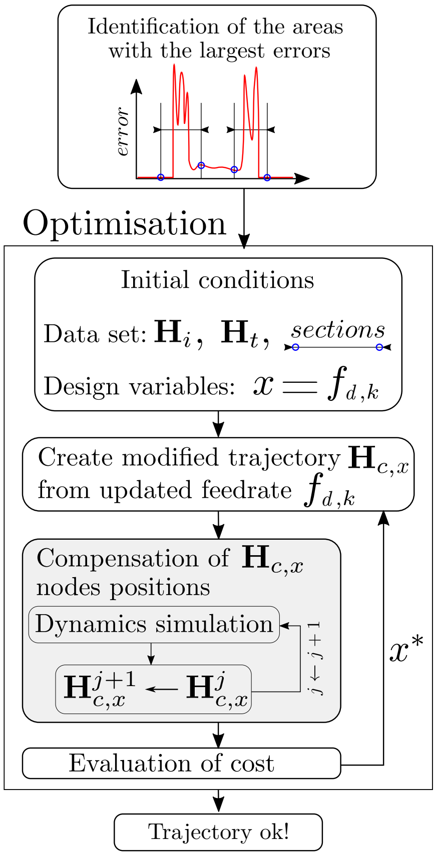

Fig. 2. Feedrate optimisation scheme.

3. Results

The feedrate algorithm has been tested for a shoulder milling operation of aluminium Al6082 with half immersion of the tool. The operation has been selected to emphasis the deviations of the tool normally to its feed direction. It is represented in Figure 3. The operational parameters of the simulations are given in Table 1. An assumption has been made in the series of simulation on the cutting coefficients. This assumption is acceptable if the feedrate is varying around its value when identifying the coefficients [26]. In this case, they are not significantly varying in the range of admissible feedrate. The optimisation bounds the space of design variables for the feedrates values to avoid changing of machining mode.

In Figure 3, the error resulting from the updated path (corresponding the result of the "Path modification" step from Figure 1) is represented as well. As it can be seen, the error is strongly reduced however the entrance and exit of the tool lead to rather high cord error (120 µm).

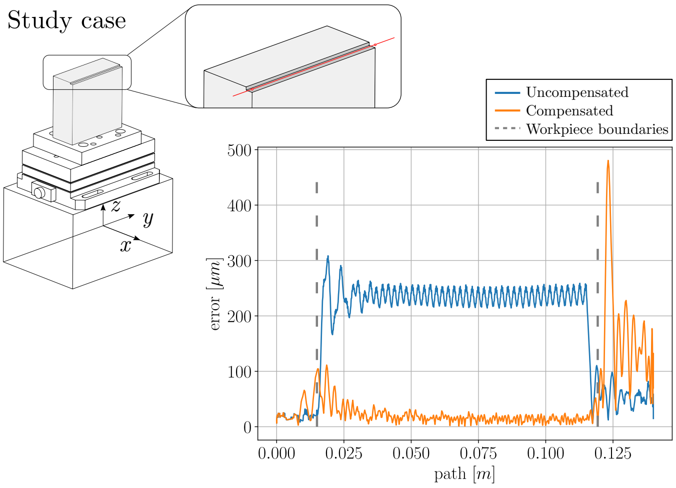

Fig. 3. Simulation case with initial error (Uncompensated) and remaining error after position compensation (Compensated)

Table 1. Simulation parameters for the trajectory compensation

Operation type		Shoulder Milling
Workpiece Material		Al6082
	Operation parameters	
Tool Type		Flat-end mill
Tool Material		Carbide
Diameter	[mm]	10
N° of edges (flutes)		2
Pitch	[°]	170-190
Helix angle	[°]	30
Rotation Speed	[rpm]	11250
Axial / Radial depth of cut	[mm]	2 / 5
	Simulation parameters	
Number of slices (tool)		16
Time steps dt	$[s]$	1e-4
$K_{t/r/a,c}$	$[MPa]$	733.5 / 346.5 / 127.9

The moving average window size defines the path sections where the feedrate changes will be operated. Several window sizes have been tested from 50 µs to 300 µs. The difference is that a tighter window gives smaller sections, closely centred around the error areas and wider ones allow more amplitude around the areas. The issue with smaller sections is that as a change of velocity is asked to the robot, it naturally acts as a perturbation on the system, and small sections tends to add excitation and make the feedrate update under perform. The improvements are gathered in Table 2.

Table 2. Indices of performances for the feedrate updated trajectory with respect to the initial trajectory and the path compensated trajectory.

	Cord error	Accumulated error
Gain w.r.t. initial trajectory	69.25 %	99.15%
Gain w.r.t. compensated trajectory	20.24%	40.24%

It can be seen that the maximum deviation (cord error) is located at the entrance of the tool. However, with the feedrate change, the impact of deviation propagates less far into the part. The Figure 4 presents the final speed profile and the comparison of the tool positioning error for the initial, the path compensated and the feedrate optimised trajectories. For the sake of comparison these results are presented along the curvilinear coordinates.

An interesting index to extract from the error signal is the distribution of the machining error in magnitude. It quantifies the proportion of the tool path whose positioning error is below a threshold. The evolution of the proportion of the error for the initial, path compensated and feedrate optimised trajectories is given in Figure 5. It can be seen that, with the path compensated trajectory, more than 90% of the error is below 50 µm. The feedrate updated one managed to reduce errors sufficiently to have 96% of the error below the 50 µm threshold (magnitude chosen according to the Stäubli TX200 repeatability value).

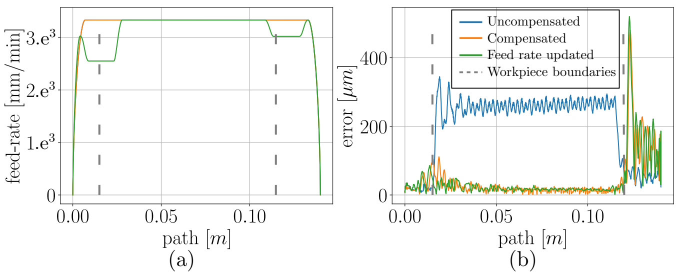

Fig. 4. (a) Updated speed profile, superimposed with the original speed profile. (b) Comparison of the errors along the tool path.

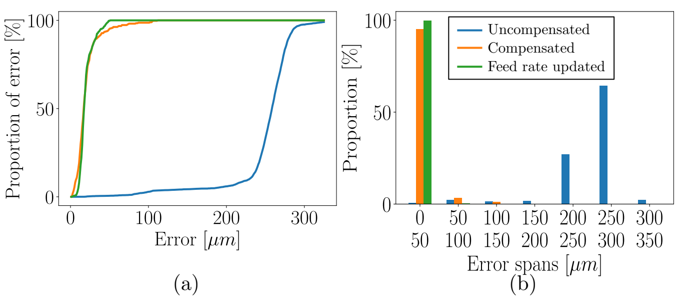

Fig. 5. (a) Evolution of the proportion of the tool positioning error. (b) Distribution of the positioning error in 50 µm groups.

4. Conclusion

Within the field of offline trajectory compensation in robotic machining, the impact of feedrate modification in the view of improving the compensation has been numerically investigated. It has been shown that the feedrate update in the sections with deviations, where a trajectory compensation method by only path update cannot decrease sufficiently the errors, can contribute to the decrease of the cord error and the overall machining error. The feedrate update managed to reduce the cord error and decrease the impact of the tool penetration in the part. These simulation results confirm the interest of having a two-levels trajectory compensation algorithm with first, the path update and secondly the feedrate update. It open the way for further investigations and experimental campaigns.

Considering the prospects of this work, two main direction are considered are given below:

- The assumption made about the cutting coefficients will be discarded by considering them varying.
- Similarly to the compensation algorithm [24], extensive experimental campaign is planned to challenge the trajectories obtained from the optimisation and confirm the expected improvements.

Acknowledgements

This work was supported by the European Union and the Walloon regional government funding this research (HybridAM research project).

References

[1] Alexander Verl, A. Valente, S. Melkote, C. Brecher, Erdem Ozturk, and Taner Tunc. Robots in machining. *CIRP Annals*, pages 799–822, 06 2019.

[2] M.F. Zaeh, F. Schnoes, B. Obst, and D. Hartmann. Combined offline simulation and online adaptation approach for the accuracy improvement of milling robots. *CIRP Annals*, 69, 05 2020.

[3] Y. Altintas and K. Erkorkmaz. Feedrate optimization for spline interpolation in high speed machine tools. *CIRP Annals*, 52(1):297–302, 2003.

[4] Yusuf Altintas. *Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations, and CNC Design*. Cambridge University Press, 2 edition, 2012.

[5] Xavier Beudaert, Pierre-Yves Pechard, and Christophe Tournier. 5-axis tool path smoothing based on drive constraints. *International Journal of Machine Tools and Manufacture*, 51(12):958–965, 2011.

[6] Xiaoyong Huang, Fei Zhao, Tao Tao, and Xuesong Mei. A novel local smoothing method for five-axis machining with time-synchronization feedrate scheduling. *IEEE Access*, 8:89185–89204, 2020.

[7] Ashan Eranga, Supun Dissanayaka, and Chaowalit Hamontree. Feed rate optimization for five axis milling with an iso-scallop tool path. In *2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, pages 1–6, July 2022.

[8] Dambly Valentin. *Robotic machining: Tool trajectories adaptation to correct the dynamic and static inaccuracies*. PhD thesis, University of Mons, 09 2023.

[9] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. Springer, 2009.

[10] Hoai Nam Huynh, Hamed Assadi, Valentin Dambly, Édouard Rivière-Lorphèvre, and Olivier Verlinden. Direct method for updating flexible multibody systems applied to a milling robot. *Robotics and Computer-Integrated Manufacturing*, 68, 04 2021.

[11] Monica Katherine Gonzalez, Nikolas Alexander Theissen, Asier Barrios, and Andreas Archenti. Online compliance error compensation system for industrial manipulators in contact applications. *Robotics and Computer-Integrated Manufacturing*, 76:102305, 2022.

[12] Seong Hyeon Kim, Eunseok Nam, Tae Ha, Soon-Hong Hwang, Jae Lee, Soo-hyun Park, and Byung-Kwon Min. Robotic machining: A review of recent progress. *International Journal of Precision Engineering and Manufacturing*, 20, 08 2019.

[13] Marcel Cordes and Wolfgang Hintze. Offline simulation of path deviation due to joint compliance and hysteresis for robot machining. *The International Journal of Advanced Manufacturing Technology*, 90, 04 2017.

[14] Alexandr Klimchik, Yier Wu, Stéphane Caro, Benoît Furet, and Anatol Pashkevich. Accuracy improvement of robot-based milling using an enhanced manipulator model. *Mechanisms and Machine Science*, 22, 09 2014.

[15] Alexandr Klimchik, Yier Wu, Claire Dumas, Stéphane Caro, Benoît Furet, and Anatol Pashkevich. Identification of geometrical and elastostatic parameters of heavy industrial robots. *Proceedings - IEEE International Conference on Robotics and Automation*, 11 2013.

[16] C. Reinl, Martin Friedmann, J. Bauer, M. Pischan, Eberhard Abele, and Oskar Von Stryk. Model-based off-line compensation of path deviation for industrial robots in milling applications. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, pages 367–372, 07 2011.

[17] Matthias Neubauer, Hubert Gattringer, Andreas Mueller, Armin Steinhauser, and Wolfgang Höbarth. A two-stage calibration method for industrial robots with joint and drive flexibilities. *Mechanical Sciences*, 6:191–201, 09 2015.

[18] Eberhard Abele, Stefan Rothenbücher, and Matthias Weigold. Cartesian compliance model for industrial robots using virtual joints. *Production Engineering*, 2:339–343, 09 2008.

[19] O. Verlinden, H.N. Huynh, G. Kouroussis and E. Rivière-Lorphèvre. Modelling flexible bodies with minimal coordinates by means of the corotational formulation. *Multibody System Dynamics*, 42(4):495–514, 2018.

[20] S Engin and Y Altintas. Mechanics and dynamics of general milling cutters Part I: Helical end mills. *International Journal of Machine Tools and Manufacture*, 41:2195–2212, 12 2001.

[21] Yusuf Altintas and Serafettin Engin. Generalized modeling of mechanics and dynamics of milling cutters. *CIRP Annals-manufacturing Technology*, 50:25–30, 12 2001.

[22] Edouard Riviere-Lorphèvre, Enrico Filippi, and Pierre Dehombreux. Inverse method for cutting forces parameters evaluation. *Engineering MECHANICS*, 14:354–357, 08 2007.

[23] Valentin Dambly, Edouard Rivière-Lorphèvre, François Ducobu, and Olivier Verlinden. Tri-dexel-based cutter-workpiece engagement: computation and validation for virtual machining operations. *The International Journal of Advanced Manufacturing Technology*, 2023.

[24] Valentin Dambly, Bryan Olivier, Édouard Rivière-Lorphèvre, François Ducobu, and Olivier Verlinden. Iterative offline trajectory correction based on dynamic model for compensating robot-dependent errors in robotic machining. *Robotics and Computer-Integrated Manufacturing*, 2025. accepted for publication.

[25] Yusuf Altintas, P. Kersting, Dirk Biermann, Erhan Budak, Berend Denkena, and Ismail Lazoglu. Virtual process systems for part machining operations. *CIRP Annals - Manufacturing Technology*, 63, 12 2014.

[26] M.Y. Tsai, S.Y. Chang, J.P. Hung, and C.C. Wang. Investigation of milling cutting forces and cutting coefficient for aluminum 6060-t6. *Computers & Electrical Engineering*, 51:320–330, 2016.