World Academy of Science, Engineering and Technology
International Journal of Industrial and Manufacturing Engineering
Vol:10, No:9, 2016

Milling Simulations with a 3-DOF Flexible Planar Robot

Hoai Nam Huynh, Edouard Rivière-Lorphèvre, Olivier Verlinden

Abstract—Manufacturing technologies are becoming continuously more diversified over the years. The increasing use of robots for various applications such as assembling, painting, welding has also affected the field of machining. Machining robots can deal with larger workspaces than conventional machine-tools at a lower cost and thus represent a very promising alternative for machining applications. Furthermore, their inherent structure ensures them a great flexibility of motion to reach any location on the workpiece with the desired orientation. Nevertheless, machining robots suffer from a lack of stiffness at their joints restricting their use to applications involving low cutting forces especially finishing operations. Vibratory instabilities may also happen while machining and deteriorate the precision leading to scrap parts. Some researchers are therefore concerned with the identification of optimal parameters in robotic machining. This paper continues the development of a virtual robotic machining simulator in order to find optimized cutting parameters in terms of depth of cut or feed per tooth for example. The simulation environment combines an in-house milling routine (`DyStaMill`) achieving the computation of cutting forces and material removal with an in-house multibody library (`EasyDyn`) which is used to build a dynamic model of a 3-DOF planar robot with flexible links. The position of the robot end-effector submitted to milling forces is controlled through an inverse kinematics scheme while controlling the position of its joints separately. Each joint is actuated through a servomotor for which the transfer function has been computed in order to tune the corresponding controller. The output results feature the evolution of the cutting forces when the robot structure is deformable or not and the tracking errors of the end-effector. Illustrations of the resulting machined surfaces are also presented. The *consideration of the links flexibility* has highlighted an increase of the cutting forces magnitude. This proof of concept will aim to enrich the database of results in robotic machining for potential improvements in production.

Keywords—Control, machining, multibody, robotic, simulation.

I. Introduction

ROBOTIC machining is a growing technology in the field of mechanical manufacturing. This technique consists in machining materials with an industrial robot used to position and orientate the spindle handling the cutting tool. For the moment, robotic machining is dedicated to applications involving low cutting forces such as finishing operations, a low precision such as trimming operations or the shaping of soft materials. The main fields of cutting applications using industrial robots are deburring, pre-machining, cutting, trimming, cleaning, sanding, prototyping, degating of cast metal parts and finishing operations [1]. Soft materials like foam, wood or plastics can be machined with an equivalent accuracy compared with a conventional machine-tool. However, it does not prevent the use of robotic machining to shape material like CFRP (Carbon Fiber Reinforced Polymer), aluminium, stone and steel for end-machining operations of moderate precision parts (Fig. 1).

Fig. 1 Applications of robotic machining

Using industrial robots for machining operations represents a very promising alternative to conventional machine-tools as they are less expensive for larger workspaces. However, they are less accurate but it is accepted that the cost-saving would be around 30% compared with an equivalent machine-tool with a comparable workspace [2]. The productivity gain is therefore more interesting when larger parts are machined with industrial robots: it is the case in aerospace and in sailing for which large composite parts could be finalised by machining robots for finishing operations such as trimming still achieved manually. The automation of those tasks would drive to a reduction of cost and scrap parts [3]. Whereas industrial robots can reach every location on the workpiece with a desired orientation thanks to their agility, their lack of stiffness causes the appearance of vibrations which lead to inaccuracy on the produced parts. This problem constitutes the main obstacle that prevents the adoption of robots for classical machining processes. The stiffness of a serial robot is typically less than 1 N/μm which means fifty times lower than the rigidity of a machine-tool [4]. The greater flexibility of robots also

Hoai Nam Huynh is with the Faculty of Engineering UMONS, Theoretical Mechanics, Dynamics and Vibration Unit, Mons, 7000 Belgium (phone: +32 (0)65 37 42 16; fax: +32 (0)65 37 41 83; e-mail: HoaiNam.Huynh@umons.ac.be).

Edouard Riviere-Lorphevre is with Faculty of Engineering UMONS, Machine Design and Production Engineering Unit, Mons, 7000 Belgium (phone: +32 (0)65 37 45 47; fax: +32 (0)65 37 45 45; e-mail: Edouard.Riviere@umons.ac.be).

Olivier Verlinden is with the Faculty of Engineering UMONS, Theoretical Mechanics, Dynamics and Vibration Unit, Mons, 7000 Belgium (phone: +32 (0)65 37 41 84; fax: +32 (0)65 37 41 83; e-mail: Olivier.Verlinden@umons.ac.be).

lowers their natural frequencies, up to 100 times lower than for machine-tools, and may generate milling instabilities as the chatter phenomenon [5]. Chatter vibrations sometimes occur when the tool machines a previous milled surface yielding to wavy surfaces [6].

Recent research in robotic machining focuses on the prediction of vibratory instabilities by analysing the interaction with the material [7], the compensation of trajectory errors in robotic machining using 3D-piezo mechanism by adapting the position of the spindle [8] or by modifying the tool path offline [9], the optimization of the placement of the workpiece with respect to the robot [2] or the identification of joints stiffness of six-revolute industrial serial robots [10].

A project devoted to the identification of optimal cutting parameters in robotic machining has recently started at the Faculty of Engineering of Mons (Belgium) combining our skills and experiences in machining and in multibody systems dynamics. Since their identification is still empirical, the combination of multibody systems coupled with milling is particularly welcome to simulate the dynamic behaviour of robotic machining processes and improve their performance. The first step of the project aims to build a robotic machining virtual simulator in order to perform simulations. This model should be validated with machining experiments afterwards for eventually allowing a better understanding of the milling phenomena as well as possible new optimization methods. Optimal cutting parameters will therefore be obtained quickly, which would allow a potential cost reduction in the industries. So far, a coupling between an in-house multibody library called `EasyDyn` [11], [12] and an in-house milling routine called `DyStaMill` (*DY*namics *STA*bility of *MILL*ing operations) has been performed to reproduce classical results in milling such as stability lobes (axial depth of cut as a function of the spindle speed) and thus validate their association [13]; an extension of the model to a 3-DOF planar robot composed of rigid links was also presented at that time. Nonetheless, as stated by a paper of S. Mousavi [14], an accurate prediction of robotic machining stability is only possible by considering the joints flexibility and the robot links flexibility as well.

This paper will therefore pursue the modelling of a 3-DOF planar robot with **flexible links** submitted to cutting forces. The first section sets the simulation environment by presenting the main characteristics of the multibody library `EasyDyn` and the milling routine `DyStaMill` as well as their coupling. A planar robot model with flexible links completed with the compliance of its servomotors of each joint is then built within the simulation environment. A position controller guides the robot end-effector through an algebraic inverse kinematics scheme. Tracking errors at the end-effector with and without being submitted to cutting forces are presented as well as a comparison of the evolution of the cutting forces according to whether the links are rigid or flexible. Finally, the machined surfaces are illustrated.

II. Simulation Environment

The simulation environment gathers in one `C++` single program a multibody library and a milling routine. This section presents the different components of the simulation environment as well as the processing of their interactions.

A. *`EasyDyn`: The Multibody Library*

The framework `EasyDyn` is an in-house `C++` multibody library which was initially developed for teaching purposes at the Theoretical mechanics, Dynamics and Vibration unit of the Faculty of Engineering of Mons (Belgium) [11], [12]. It sometimes contributes to research projects as it is the case here and is available for free on the Internet.

`EasyDyn` allows its users simulating problems represented by second-order differential equations and, more particularly, the dynamic behaviour of a mechanical multibody system. The approach currently implemented is based on the minimal coordinates for the choice of the configuration parameters q expressing the motion of the bodies. The kinematics of all bodies is provided by the user through homogeneous transformation matrices $\mathbf{T}$ function of the chosen configuration parameters as far as there is a "one to one" relationship between them and the configuration of the system, which means that the configuration parameters have to be independent. The number of configuration parameters is consequently equivalent to the number of Degrees Of Freedom (DOF). The expressions of position and orientation of each body i coming from their transformation homogeneous matrix $\mathbf{T}_{0,i}$ are then derived symbolically in order to get the expressions of their velocities and accelerations ($\mathbf{T}_{0,i}$ gives the position and orientation of the frame attached to the body i with respect to the global reference frame 0) (Fig. 2). The kinematics of each body i is accordingly a function of q and its time-derivatives $\dot{q}$ and $\ddot{q}$. Aside from the kinematics, the user may also apply forces on each body.

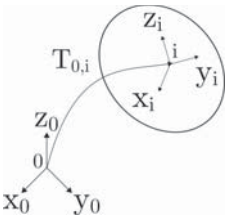

Fig. 2 Frame situation of body i with respect to the global reference frame

The n_{cp} equations of motion are then built according to the given kinematics involving the n_{cp} configuration parameters and the applied forces on the n_b bodies. Their form derives from the application of the d'Alembert's principle:

$$\sum_{i=1}^{n_B}[d^{i,j}\cdot(R_i - m_i a_i) + \theta^{i,j}\cdot(M_{G_i} - \Phi_{G_i}\dot{\omega}_i - \omega_i \times \Phi_{G_i}\omega_i)] = 0 \quad j = 1, n_{cp} \tag{1}$$

where: m_i = the mass of body i; Φ_{G_i} = central inertia tensor of body i; R_i = resultant force of all applied forces on body i; M_{Gi} = resultant moment at the center of mass G_i of all applied forces on body i; v_i = velocity of the center of mass of body i; a_i = acceleration of the center of mass of body i; ω_i = rotational velocity of body i; $d^{i,j}$ = partial contribution

of $\dot{q}_j$ in the translational velocity of body i: $v_i = \sum_{j=1}^{n_{cp}} d^{i,j} \cdot \dot{q}_j$; $\theta^{i,j}$ = partial contribution of $\dot{q}_j$ in the rotational velocity of body i: $\omega_i = \sum_{j=1}^{n_{cp}} \theta^{i,j} \cdot \dot{q}_j$.

The consideration of the bodies flexibility is taken into account by implementation of the so-called co-rotational approach proposed by A. Cardona [15]. Whereas one coordinate system attached to a rigid body allows describing its motion, it seems natural that the motion of a flexible body would be defined by the one its nodes. The co-rotational approach implies the definition of an additional frame called the *co-rotational frame* * which co-rotates with the flexible body and represents the mean motion of its nodes. This particular frame is floating and is not necessarily fixed to one of the nodes. For instance, the motion of a flexible beam is characterized by the one of the two coordinates systems corresponding to its end nodes and its co-rotational frame through their corresponding homogeneous transformation matrices $\mathbf{T}_{0,i}$ (Fig. 3).

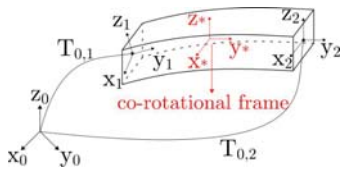

Fig. 3 Flexible beam using the co-rotational approach

To recover the contributions of the flexible bodies in the equations of motion, the co-rotational approach proposes to express the kinetic energy of a N nodes flexible body as:

$$T = \frac{1}{2} \sum_i \sum_j \begin{Bmatrix} v_i^* \\ \omega_i^* \end{Bmatrix}^T \mathbf{M}_{ij} \begin{Bmatrix} v_j^* \\ \omega_j^* \end{Bmatrix} \tag{2}$$

where: T = kinetic energy of a N nodes flexible body; v_i^* = velocity of node i projected in the co-rotational frame *; ω_i^* = rotational velocity of node i projected in the co-rotational frame *; M_{ij} = partition of the FEM mass matrix of the body corresponding to nodes i (lines) and j (columns).

The main interest of the co-rotational approach lies in the direct use of the mass and stiffness matrices of the FEM model of the flexible body. Finally, the equations of motion are obtained by applying the Hamilton principle or the Euler-Lagrange theorem on the expression of the kinetic energy (2). Naturally, the elastic forces are computed from the stiffness properties of the flexible element (see [16] for more details).

B. `DyStaMill`: The Milling Routine

`DyStaMill` standing for *DY*namics *STA*bility of *MILL*ing is an in-house routine written in `C++` for the stability analysis of milling operations. The routine is based on a macroscopic model of the machining process considering cutting forces acting on the system as force elements. Dynamic simulation of milling operations involves the modelling of the workpiece surface and the dynamic system as well as the prediction of the cutting forces [17], [18].

The modelling of the surface is inspired by the 'eraser of matter' model proposed by G. Peigné [19]. The workpiece profile is approximated by linked segments and updated throughout the machining process. On the other hand, the tool is discretized into slices along its revolving axis (Fig. 4) [20]. It is assumed that the displacement of the tool lies in a plane perpendicular to its axis thus generating $2\frac{1}{2}$D shapes. From this material removal model, the routine allows simulating milling operations such as slotting, contouring or pocket hollowing. The modelling of workpiece surface must be accurate enough in order to predict milling instabilities. This is especially true when chatter phenomena occur as it is a regenerative process arising when the tool removes material from a previously machined surface [6].

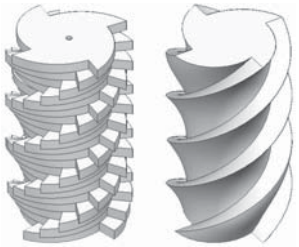

Fig. 4 Axial tool discretization

Dynamic characteristics of the system are obtained from a modal identification of the frequency response at the tooltip by modal analysis. The complete dynamics of the system is then modelled as the superposition of n_{cp} second order differential equations in a modal basis as:

$$m \cdot \ddot{q}_i + c \cdot \dot{q}_i + k \cdot q_i = f_i \qquad i = 1, n_{cp} \tag{3}$$

where: m = modal mass; c = damping coefficient; k = stiffness coefficient; f = applied force.

Fig. 5 illustrates the equivalent mass-spring-damper system corresponding to one of the modes contributing to the tool frequency response.

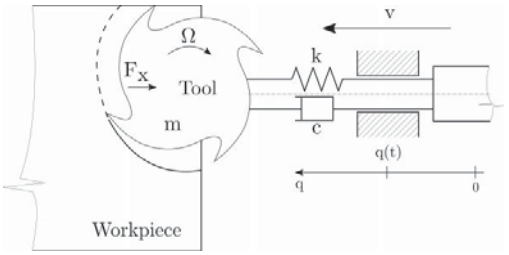

Fig. 5 Single-axis vibrating mill model where: Ω = spindle speed; F_x = cutting forces along x-axis

The numerical integration of the system dynamics provides the time evolution of the tool position from which is based the computation of the cutting forces. The cutting forces are determined using analytical laws as long as a tool/material couple is provided through cutting coefficients K. The model proposed by *Y. Altintas* was adopted here for the computation

of the elementary cutting forces dF_t, dF_r and dF_a along the *t*angential, *r*adial and *a*xial directions of each disc discretizing the tool:

$$\left.\begin{aligned} dF_t &= K_{t,e}\, dS + K_{t,c}\, h\, db \\ dF_r &= K_{r,e}\, dS + K_{r,c}\, h\, db \\ dF_a &= K_{a,e}\, dS + K_{a,c}\, h\, db \end{aligned}\right\} \quad (4)$$

where: $K_{\_,e}$ = edge cutting coefficients; $K_{\_,c}$ = shear force coefficients; h = uncut chip thickness normal to the cutting edge; dS = infinitesimal length of a helical cutting edge segment; db = projected length of an infinitesimal cutting flute in the direction along the cutting velocity (See [20] for details). These local cutting forces are then integrated over the n_z teeth and over the n_e discs to recover the global cutting forces projected into the global reference frame:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \sum_{e=1}^{n_e}\sum_{j=1}^{n_z}[\mathbf{B}] \begin{bmatrix} dF_t \\ dF_r \\ dF_a \end{bmatrix} \quad (5)$$

where: $[\mathbf{B}]$ = transformation matrix to project the local cutting forces into the global reference frame; F_x, F_y, F_z = cutting forces projected into the global reference frame.

C. Coupling of the Routines

For the purpose of this research project, the `DyStaMill` routine was recently integrated within the multibody framework `EasyDyn` for being able to simulate complete robotic machining operations. The framework `EasyDyn` was naturally chosen for hosting the milling routine due to its high scalability when particular features must be added such as the computation of the cutting forces, the update of the machined surface, the joint compliance and actuators of the robot as well as its controller.

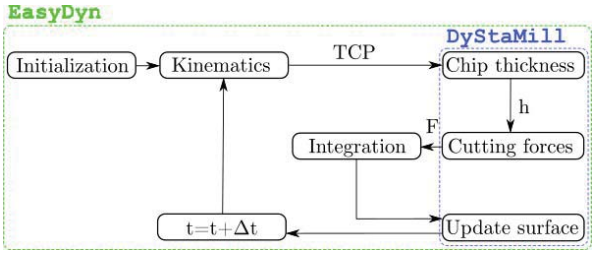

Fig. 6 Solving of the coupled dynamic system

In order to validate the coupling, the first step consisted in replicating results previously obtained for machine-tool simulations with the milling routine working standalone. Several examples were successfully verified with the coupled system ranging from stable and unstable simulations to stability lobes diagrams. The mechanical system was eventually extended to a 3-DOF planar robot composed of rigid links thanks to the potential of the multibody framework. At that time, only stable cutting conditions were tested as the kinematics of the robot end-effector was imposed [13].

Fig. 6 presents the interactions between the milling routine and the multibody framework. Once the data for the multibody system (the number of bodies, the number of degrees of freedom, the kinematics, ...) and the milling simulation (the tool dynamics and its geometry, the material, ...) have been collected, the solving of the coupled system can begin: the tool position (TCP: *T*ool *C*enter *P*oint) is first sent to `DyStaMill` in order to determine the chip thickness h which is then used for the computation of the cutting forces F. An integration step of the coupled dynamic system can finally take place using one of the multibody library functions before updating the workpiece surface for the next time step. Note that the chosen time step must lie above 30 steps per spindle revolution to make sure that the modelling of the workpiece surface is sufficiently updated to predict milling instabilities.

III. Robot Modelling

Having coupled the milling with the multibody aspects, the enhancement of the robot model was the next preoccupation. As compared with the previous robot model, the current one is now composed of flexible links and its joints are driven by the torques coming from servomotors. Therefore, the end-effector motion is this time dynamically actuated.

A. Inspiring Robot and Model

The robot model is inspired by a Fanuc machining robot from the M-20*i*A^{TM} series shown in Fig. 7.

Fig. 7 Fanuc robot (M-20*i*A^{TM} series)

More precisely, a planar version comprising 3 degrees of freedom was chosen as a case of study. The simplified model of the robot consists of 3 links and 3 revolute joints to position and orientate the end-effector. The corresponding kinematic chain thus includes 3 configuration parameters defining the joints motion whose rotation axes are parallel to the z-axis. The cutting forces are applied on the end-effector along the x- and y-axes as shown in Fig. 8. Note that in the implemented model, the cutting forces are directly exerted on the end node of the last link such that there is no offset along the z-axis.

Flexibility of the links has been taken into account by applying the aforementioned co-rotational approach to equivalent flexible beams. As the CAD[1] models of the Fanuc robot were provided by the manufacturer, dimensions and mass

[1]CAD: Computer-Aided Design

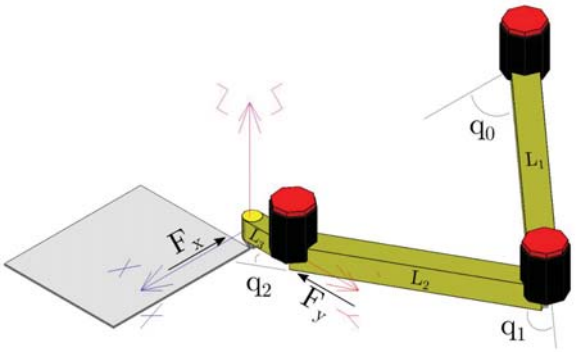

Fig. 8 Fanuc robot model

properties of the three links were reasonably identified for a simplified model. Links characteristics are given in Appendix A.

For the sake of simplicity, the co-rotational frame of each link is located at the first node of each flexible beam. Moreover, in addition of the joints motion, each flexible beam introduces 6 more configuration parameters in order to represent its deformation. Therefore, the motion of the end effector node may be expressed from the product of the elementary homogeneous transformation matrices corresponding to the succession of joints/links along the kinematic chain as:

$$\begin{aligned}\mathbf{T}_{0,6} =&\mathbf{T}_{rotz}(\mathbf{q}_0)\cdot\mathbf{T}_{disp}(L_1+q_3,q_4,q_5)\cdot\mathbf{T}_{rotx}(q_6)\\ &\cdot\mathbf{T}_{roty}(q_7)\cdot\mathbf{T}_{rotz}(q_8)\\ &\cdot\mathbf{T}_{rotz}(\mathbf{q}_1)\cdot\mathbf{T}_{disp}(L_2+q_9,q_{10},q_{11})\cdot\mathbf{T}_{rotx}(q_{12})\\ &\cdot\mathbf{T}_{roty}(q_{13})\cdot\mathbf{T}_{rotz}(q_{14})\\ &\cdot\mathbf{T}_{rotz}(\mathbf{q}_2)\cdot\mathbf{T}_{disp}(L_3+q_{15},q_{16},q_{17})\cdot\mathbf{T}_{rotx}(q_{18})\\ &\cdot\mathbf{T}_{roty}(q_{19})\cdot\mathbf{T}_{rotz}(q_{20})\end{aligned} \tag{6}$$

The expression of the kinematics for the other nodes is similar. The planar robot model thus comprises a total of 21 configuration parameters: 3 of which are dedicated to the joints motion (q_0, q_1 and q_2) while the remaining 18 ones (q_3 to q_{20}) represent the deformations of the links.

B. Motors Modelling

Robots typically rely on servomotors as actuators to position accurately their moving links. The model adopted here is based on the control of a DC motor for each joint. This simple model only needs two equations to represent the dynamics of the motor: the torque equation and the equation of the terminal voltage. The latter which can be seen as the motor command is written as:

$$U = L\cdot\frac{di}{dt} + R_a\cdot i + K_b\cdot\dot{\theta} \tag{7}$$

where: U = motor command provided by the joint controller; L = motor inductance; i = motor current; R_a = armature resistance; K_b = motor speed constant; θ = motor joints position (either q_0, q_1 or q_2).

Note thati due to its small inductance, the first term of (7) is neglected. Using (7), the output torque applied on each joint is expressed as:

$$\tau = K_m\cdot i = K_m\cdot\frac{U - K_b\cdot\dot{\theta}}{R_a} \tag{8}$$

where: τ = motor torque; K_m = motor torque constant.

Servomotor data were found in a datasheet provided by the robot manufacturer. These data are given in Appendix B.

The multibody modelling of the servomotor has been achieved by considering two rigid bodies: one for the case and the other one for the output shaft (Fig. 9). Rigid bodies properties are also given in Appendix B.

Fig. 9 Servomotor (A06B series)

C. Robot Controllers

A two-level controller is required to correctly position and orientate the robot end-effector throughout its motion (Fig. 10). The first level is in fact the inverse kinematic algorithm which converts the desired end-effector position (x_{e-e_d}, y_{e-e_d}) and orientation (θ_{e-e_d}) from the task space into joint desired positions (q_{d_0}, q_{d_1} and q_{d_2}). An algebraic solution of the inverse kinematic algorithm is implemented within the simulation environment [21]. Then, each desired joint position q_{d_i} is compared to the actual position of the corresponding joint q_i leading to an error ϵ_i. The second control level deals with each joint position error using a discrete PID controller.

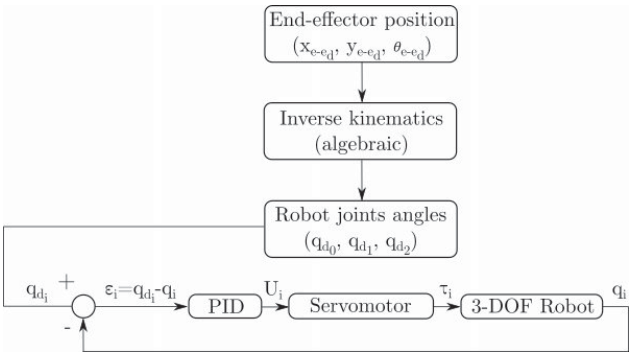

Fig. 10 Inverse kinematics and joints controllers

Each discrete PID controller was tuned separately regarding the load seen by each joint along the robot kinematic chain. In order to set the gains of each PID controller, the transfer functions in position were computed for each joint. The location of the joint along the kinematic chain had mainly effects on their time constant. Afterwards, an iterative process took place to tune each controller separately relying on a frequency domain approach [22]. Once all gains had been

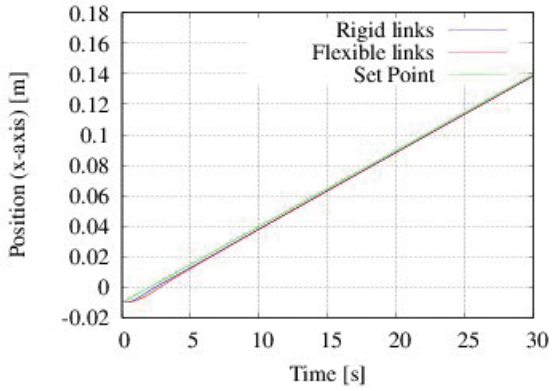

Fig. 11 Tracking along x-axis without machining

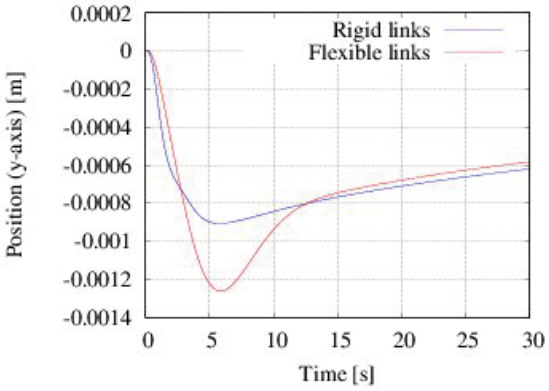

Fig. 12 Tracking along y-axis without machining

set with this method, the proportional gains were increased to speed up the dynamic of all the joints. Note also that a low-pass filter is applied on the derivative term of each PID controller for noise reduction. The final gains values and the joints transfer functions are given in Appendix C.
Finally, PID controllers output a motor command U_i for each joint servomotor.

D. Set-Point Tracking without Machining

Before running robotic machining simulations, the two-level-controller was tested without any material removal to assess its performances. The end-effector was imposed to follow a straight line trajectory along the x-axis at a speed of 0.005 [m/s]. Since the robot model now includes flexible links, the results are compared with its previous version including rigid links. From a simulation point of view, the controller loop was intended to start 0.05 [s] after the beginning of the test so that the system would be at rest. The sampling frequency was 100 [Hz].

The graphs hereafter mainly focus on the differences of the results obtained with a rigid links robot and a flexible links robot. Fig. 11 shows the end-effector tracking along the x-axis in both cases. At the beginning, it seems that the end-effector of the flexible links robot is slower but catches up its delay after some seconds. Afterwards, the tracking is the same for both robots.

Fig. 12 depicts the y-axis tracking errors at both end-effectors. Beside the fact that the end-effectors should stay along the x-axis, a slight vertical deviation along the y-axis occurs. Larger errors are made more visible for the flexible links robot. The y-axis errors ultimately stabilize around the same values for both robots.

IV. Results and Discussion

Robotic machining simulations were then achieved to examine the effect of the cutting forces on the system. Machining parameters have been fixed to focus on the results obtained with the rigid links robot and the flexible links robot. Shear force coefficients defining the tool/material couple are inspired from an article of T. Insperger [23] in which a 7075-T6 aluminium alloy is machined. On the other hand, the axial depth of cut was fixed at 0.05 [mm] and the spindle speed was settled at 8000 [RPM] for a feed of v_f=0.05 [mm/tooth]. For each test, a number of 700 spindle rotations, each divided in 30 steps, were simulated to machine the material with a one-tooth mill of 10 [mm] diameter. A summary of the machining parameters can be found in Appendix D.

As before, the set point tracking of each simulation starts 0.05 [s] after the beginning of the test and the initial location of the end-effector is set at (-0.01,0). Concerning the location of the workpiece, its left upper corner lies at (0,0) as seen in Fig. 8. The robot controller can therefore stabilize by the time its end-effector arrives at the workpiece.

A. Unstable Machining Simulation

Even with a small axial depth of cut of 0.05 [mm], the machining simulations turned out to be unstable for both robots. Cutting forces rose suddenly after a few seconds of simulation (Fig. 13). As the flexible links robot was a bit slower, instabilities were detected after those generated by the rigid links robot. If the first unstable peaks are compared, one can observe that the cutting forces generated by the flexible links robot are somewhat greater.

A look at the tracking of the end-effectors along the x-axis clearly shows a slowdown of their speed as they progress into the material. At the end of the simulation, they even seem to be stuck in the material (Fig. 14).

The end-effectors tracking along the y-axis was much worse as the tool center points deviate by 0.01 [m] at the end of the simulation instead of staying close to y=0 [m].

All those observations lead to the intermediate conclusion that the generated instabilities were related to the controller itself and not to the machining process. Indeed, as mentioned in numerous textbooks related to the control of manipulators [21], [24], a mere position control might not suffice to deal with any contact between the end-effector and the working environment. An hybrid position and force controller would be more appropriate since it would compensate for the cutting forces along the y-axis while following the

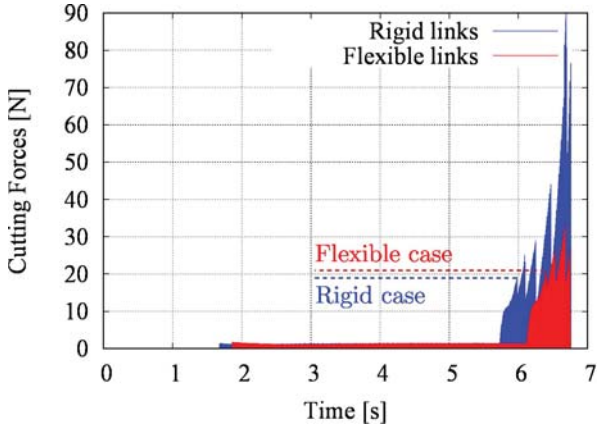

Fig. 13 Cutting forces

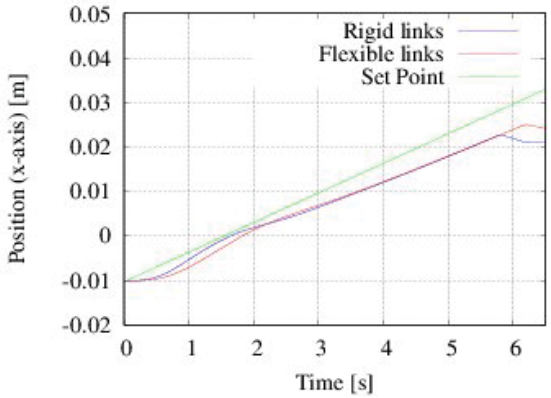

Fig. 14 Tracking along x-axis while machining

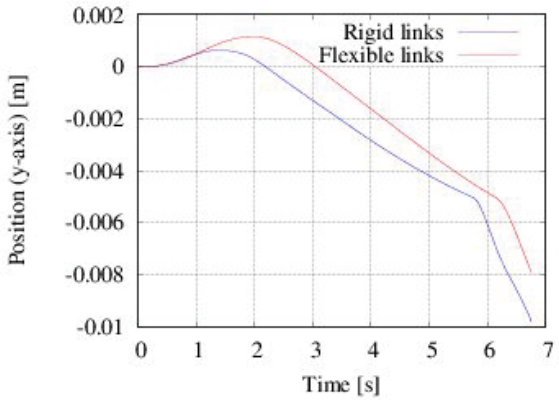

Fig. 15 Tracking along y-axis while machining

set point along the x-axis. Both the force and the position cannot be controlled along the same axis at the same time [25].

B. Stable Machining Simulation

For this machining experiment, the cutting forces along the y-axis seemed to be the major source of disturbance since the end-effectors deviate under their effects. Then, to compensate those cutting forces and somehow reproduce the effects of force controller, a spring whose stiffness coefficient is K_r=9E8 [N/m] and damping coefficient is C_r=100 [Ns/m] was inserted between both end-effectors and the ground. This y-axis spring is maintained vertically throughout the machining process (Fig. 16).

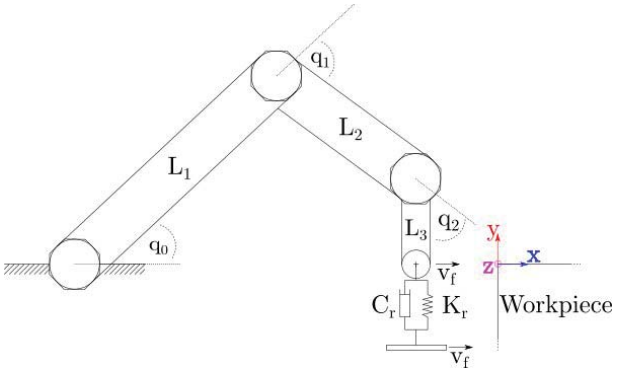

Fig. 16 Robot with spring at end-effector along y-axis

End-effectors set point tracking along the x-axis is improved without the presence of the y-axis cutting forces. The end-effector of the rigid links robot is still a bit faster but then, both end-effectors follow the same trajectories (Fig. 17).

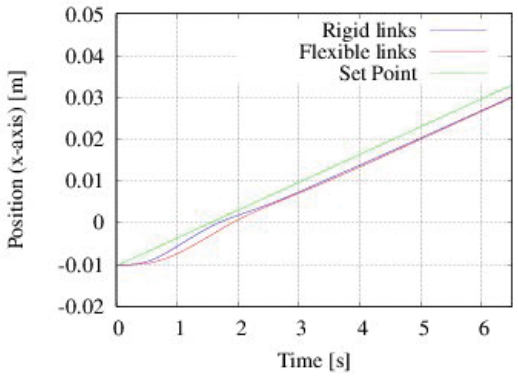

Fig. 17 Tracking along x-axis while milling (+spring)

The end-effectors displacements along the y-axis are accordingly driven by the spring dynamics. They now remain near y=0 [m] with an error around 10e-9 [m] which means that the end-effectors move essentially along the x-axis (Fig. 18).

Concerning the obtained cutting forces, it is now more visible that taking into account the links flexibility increases their amplitude for the tested operation. The cutting force peak shown in Fig. 19 and due to the first contact with the workpiece is also higher for the flexible links robot. Overall, an increase of about 15 % of the cutting forces amplitude can be observed for this machining experiment.

C. Output Surface Quality

The milling routine is also able to display the machined surfaces. The obtained surfaces shown in Fig. 20 comes from the simulations performed with the rigid links robot.

As expected, the flexible links robot have deteriorated the surface finish a little more due to vibratory effects (Fig. 21).

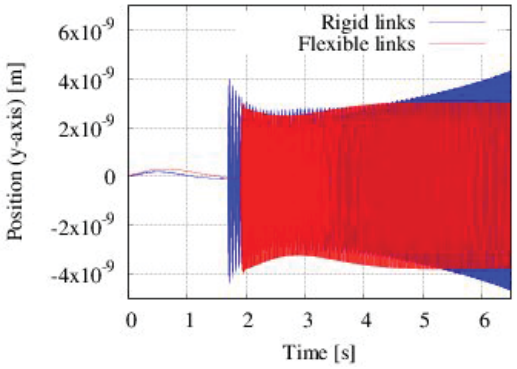

Fig. 18 Tracking along y-axis while milling (+spring)

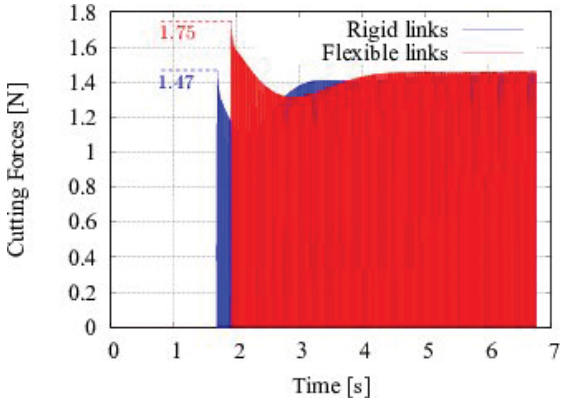

Fig. 19 Cutting forces (+spring)

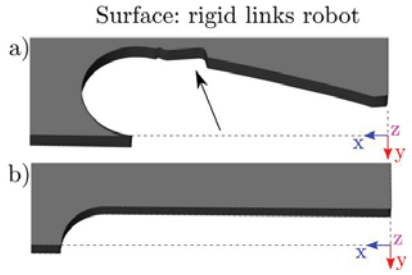

Fig. 20 Machined surface: robot with rigid links - a) without spring (unstable case) - b) with spring (stable case)

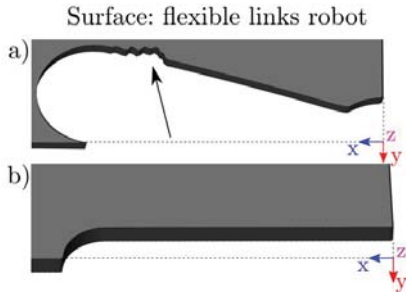

Fig. 21 Machined surface: robot with flexible links - a) without spring (unstable case) - b) with spring (stable case)

V. Summary

This paper presented the extension of a 3-DOF planar robot model by taking into account its links flexibility to achieve robotic machining operations. Indeed, it is said in the literature that the flexibility of the robot links has a great influence on the stability of robotic machining. For this purpose, the so-called co-rotational approach was used to model flexible bodies in a simulation environment combining the multibody and the milling aspects. Then, the flexible links robot was controlled through an inverse kinematic scheme. Its end-effector was imposed to follow a straight line to achieve half-immersion milling operations.

This proof of concept showed that cutting forces obtained with the flexible links robot were about 15 % greater than those retrieved from its homologous rigid links robot for the tested operation. As expected, the quality of the surface finish is also degraded due to vibratory effects coming from the flexibility of the links. On the other hand, it was also observed that a single position control of the robot end-effector was not sufficient in order to deal with the material interactions. Future prospects will therefore focus on the implementation of a hybrid position and force controller to compensate the cutting forces disturbances and eventually simulate stability lobes diagrams commonly found in machining.

Appendix A

Links characteristics if a steel square solid-section is assumed:

	L [m]	Mass [kg]	I_{yy} [m^4]	I_{zz} [m^4]
$Link_1$	1	117	2.8125E-5	2.8125E-5
$Link_2$	0.68	119.34	4.21875E-5	4.21875E-5
$Link_3$	0.15	16.23	1.8066E-5	1.8066E-5

Classical steel properties are assumed:

Density [kg/m^3]	Young's Modulus [Pa]	Poisson's Ratio
7800	2.1E11	0.3

Appendix B

Electrical properties for the torque constant K_m (speed constant K_b) and the armature resistance R_a:

$$\begin{aligned} K_m &= 0.77\ [\tfrac{N \cdot m}{A}] \\ K_b &= 0.77\ [V/rad/s] \\ R_a &= 0.17\ \Omega \end{aligned}$$

Servomotor characteristics:

	Mass [kg]	I_{xx} [kg·m^2]	I_{yy} [kg·m^2]	I_{zz} [kg·m^2]
Case	17	0.0995	0.0995	0.0359
Shaft	1	0.003	0.0062	0.0062

Appendix C

Identified transfer functions under their Laplace form:

$$G_{q_0}(s) = \frac{1.3}{s(59.06s + 1)} \tag{9}$$

$$G_{q_1}(s) = \frac{1.3}{s(10.27s+1)} \tag{10}$$

$$G_{q_2}(s) = \frac{1.3}{s(0.04s+1)} \tag{11}$$

Controller feedback gains used in simulations:

PID	K_p $[\frac{N\cdot m}{rad}]$	K_i $[\frac{N\cdot m}{rad\cdot s}]$	K_p $[\frac{N\cdot m\cdot s}{rad}]$	Cut freq. [Hz]
q_0	19.62	0.79	36.57	4.39
q_1	5.45	0.13	17.37	2.56
q_2	2.18	1.29	0.28	64.52

APPENDIX D

Summary of the machining parameters:

Material	=	7075-T6 aluminium alloy
$[K_{t,c}, K_{r,c}K_{a,c}]$	=	[550,200,0] [MPa]
Mill diameter	=	10 [mm]
Number of edge	=	1 [tooth]
Helix angle	=	0 [degree]
Feed	=	0.05 [mm/tooth]
Spindle speed Ω	=	8000 [RPM]
Axial depth of cut	=	0.05 [mm]
Direction of rotation	=	up-milling
Cutting conditions	=	half-immersion

ACKNOWLEDGMENT

The authors would like to thank the Belgian Fund for Scientific Research (FRIA-FNRS) for the financial support.

REFERENCES

[1] A. Abele, M. Weigold, S. Rothenbücher, "Model and identification of an industrial robot for machining applications," *Annals of CIRP*, vol. 56-1, pp. 387–390, 2007.
[2] S. Caro, C. Dumas, S. Garnier, B. Furet, "Workpiece placement optimization for machining operations with a kuka kr270-2 robot," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2921–2926, May, 2013.
[3] C. Dumas, A. Boudelier, S. Caro, S. Garnier, M. Ritou, B. Furet, "Développement d'une cellule robotisée de détourage des composites," *Mécanique et industries*, vol. 12, pp. 487–494, 2011.
[4] H. Zhang, J. Wang, G. Zhang, Z. Gan, Z. Pan, H. Cui, Z. Zhu, "Machining with flexible manipulator: Toward improving robotic machining performance," *Proc. IEEE-ASME International Conference on Advanced Intelligent Mechatronics*, pp. 1127–1132, USA, July, 2005.
[5] Z. Pan, H. Zhang, Z. Zhu, J. Wang, "Chatter analysis of robotic machining process," *Journal of Material Processing Technology*, vol. 173, pp. 301–309, 2006.
[6] J. Tlusty, M. Polacek, "The stability of the machine tool against self-excited vibration in machining," *ASME International Research in Production Engineering*, pp. 465–474, 1963.
[7] S. G. Mousavi, V. Gagnol, B.C. Bouzgarou, P. Ray, "Dynamic behaviour model of a machining robot," *ECCOMAS Multibody Dynamics*, pp. 771–779, July, 2013.
[8] U. Schneider, M. Drust, A. Puzik, A. Verl, "Compensation of errors in robot machining with a parallel 3d-piezo compensation mechanism," *Procedia CIRP*, vol. 7, pp. 305–310, 2013.
[9] N.R. Slavkovic, D.S. Milutinovic, M.M. Glavonjic, "A method for off-line compensation of cutting force-induced errors in robotic machining by tool path modification," *Int J Adv Manuf Technol, Springer*, vol. 70, pp. 2083–2096, 2014.
[10] C. Dumas, S. Caro, S. Garnier, B. Furet, "Joint stiffness identification of six-revolute industrial serial robots," *Robotics and Computer-Integrated Manufacturing*, vol. 27, pp. 881–888, 2011.
[11] O. Verlinden, G. Kouroussis, C. Conti, "EasyDyn: a framework based on free symbolic and numerical tool for teaching multibody systems," in *Multibody Dynamics 2005, ECCOMAS Thematic Conference*, Madrid, Spain, 21-24 June 2005.
[12] O. Verlinden, L. Ben Fékih and G. Kouroussis, "Symbolic generation of the kinematics of multibody systems in EasyDyn: From MuPAD to Xcas/Giac," *Theoretical and Applied Mechanics Letters*, vol. 3, no. 1, pp. 013012, doi: 10.1063/2.13013012, 2013.
[13] H.N. Huynh, E. Rivière, O. Verlinden, "Integration of machining simulation within a multibody framework: application to milling," *IMSD: The 4th Joint International Conference on Multibody System Dynamics*, Canada, June, 2016.
[14] S. Mousavi, V. Gagnol, B. C. Bouzgarrou, P. Ray, "Dynamic model and stability prediction in robotic machining," *Int J Adv Manuf Technol (2016), Springer*, pp. 1–13, June 2016.
[15] A. Cardona, "Superelements modelling in flexible multibody dynamics," *Multibody System Dynamics*, vol. 4, pp. 245–266, 2000.
[16] O. Verlinden, H.N. Huynh, E. Rivière, "Modelling of flexible bodies with minimal coordinates by means of the co-rotational formulation," *The 4th Joint International Conference on Multibody System Dynamics*, Canada, June, 2016.
[17] E. Rivière, E. Filippi, P. Dehombreux, "Forces, vibrations and roughness prediction in milling using dynamic simulation," *Proceedings*, (Fifth International Conference on High Speed Machining (HSM 2006)), Mars, Metz, France 2006.
[18] E. Rivière, E. Filippi, P. Dehombreux, "Chatter prediction using dynamic simulation," *International Review of Mechanical Engineering (I.RE.M.E.)*, vol. 1, pp. 78–86, 2007.
[19] G. Peigné, H. Paris, D. Brissaud, "A model of milled surface generation for time domain simulation of high-speed cutting," *Proceedings of the Institution of Mechanical Engineers*, vol. 217, pp. 919–930, 2003.
[20] S. Engin, Y. Altintas, "Mechanics and dynamics of general milling cutters. Part I: helical end mills," *International Journal of Machine Tools and Manufacture*, vol. 41, pp. 2195–2212, 2001.
[21] John J. Craig, *Introduction to Robotic: Mechanics and Control*. Pearson, Prentice Hall, 2005.
[22] C. Renotte, A. VandeWouwer, M. Remy, "A simple frequency domain approach to the tuning of pid control: design of an interactive software tool," *Journal A. Benelux Quaterly Journal on Automatic Control*, vol. 42-3, pp. 23–27, 2001.
[23] T. Insperger, B.P. Mann, G. Stépán, P.V. Bayly, "Stability of up-milling and down-milling, part I: alternative analytical methods," *International Journal of Machine Tools and Manufacture*, vol. 43, pp. 25–34, 2003.
[24] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer, 2010.
[25] M.H. Raibert, J.J. Craig, "Hybrid position/force control of manipulators," *Journal of Dynamic Systems, Measurement and Control*, vol. 103-2, pp. 126–133, 1981.

Hoai Nam Huynh was born in Hô-Chi-Minh-Ville (Viêt Nam) in 1992. He is a researcher at the 'Faculté Polytechnique' (Faculty of Engineering) of Mons in the Department of Theoretical Mechanics, Dynamics and Vibration. He is funded by the Belgian Fund for Scientific Research to achieve a PhD thesis about 'Robotic Machining'. He received an engineer degree in mechanics-mechatronics in 2015. His research interests are in the area of robotic machining simulations for cutting parameters optimization.

Édouard Rivière-Lorphèvre was born in Ath (Belgium) in 1980. He is an associate professor at the 'Faculté Polytechnique' (Faculty of Engineering) of Mons in the Department of Machine Design and Production Engineering. His teaching activities include several topics such as geometry and graphical communication, metrology, topography, manufacturing processes and mechanical design. He received the engineer degree in mechanics in 2003 and his PhD concerning the study and the simulation of machining processes in machining in 2007. His research interests are in the area of simulation of machining techniques and detection of instabilities in machining.

Olivier Verlinden is a full professor at the 'Faculté Polytechnique' (Faculty of Engineering) of Mons in the Department of Theoretical Mechanics, Dynamics and Vibration. His teaching activities include the fields of the motion of mechanical systems (kinematics, statics and dynamics of mechanical systems), computer simulation of multibody system dynamics, sound and vibration, and mechatronics (active mechanical systems, actuators, sensors, microcontrollers). He received the engineer degree in mechanics in 1988 and his PhD concerning the dynamic simulation of flexible multibody systems in 1994. He is involved in research activities in the fields of vibrations (vibrations induced by railway traffic, vibration design of electronic boards), computer simulation of multibody systems and mechatronics.