An intelligent and modular adaptive control scheme for automating the milling process

Luis Rubio^{*}, Andrew Longstaff, Simon Fletcher and Alan Myers

Centre for Precision Technologies, University of Huddersfield, UK *L.R.Rodriguez@hud.ac.uk

ABSTRACT

This research work describes an intelligent and modular architecture for controlling the milling process. For this purpose, it is taking into account the admissible input cutting parameters given by the stability lobes and restrictions of the system, calculating the quasi-optimal cutting parameters. The quasi-optimal cutting parameters are obtained considering a cost function with multi-objective purpose. Furthermore, parallel multi-estimation adaptive control architecture is proposed in order to allow adaptation of the system when the cutting parameters are changed, for example, due to production requirements. It incorporates an intelligent supervisory system to address the problem of choosing the most adequate control signal at each required time. The fundamental idea of the control system is to work automatically, with a simple interface with the operator, based around the admissible cutting parameter space given by the well-known stability lobes.

KEYWORDS: Milling, Adaptive Control, Optimization

1. INTRODUCTION

Dynamic complexity of milling processes combined with their exigent performance requirements requires sophisticated and complex control systems. The selection of adequate cutting parameters for multi-objective optimization in milling processes has been the subject of extensive research in manufacturing literature [1-3], where computer aided programming planning, decision support systems and bio-inspired systems have been used to cope with the problem of multi-objective optimization. Moreover, the control of milling forces has been applied successfully in a broad range of milling applications [4-6].

This paper presents an intelligent and modular architecture for controlling the milling process. It is intelligent because finds out parameters of the system. Moreover, it has the ability to learn from previous experience. Its modularity is based on the idea that the control represents one external module which can be implemented in the system. It is based on models of the milling process. The dynamic equation leads to the time-domain and the well-known stability plots and the linearization around the equilibrium points is represented by transfer functions.

Then, taking into account the admissible input cutting parameters given by the stability lobes and restrictions of the system, the quasi-optimal cutting parameters are calculated. The quasi-optimal cutting parameters are obtained while taking into account a cost function with a multi-objective purpose. In this way, the quasi-optimal cutting parameters can be found automatically based on the multi-criteria of maximising material remove rate and tool life while minimising the surface roughness and maintaining a robust, stable working point. This first toolbox of the modular architecture is designed so the operator can interact with it in a simple and efficient manner, leading to schedule programmed cutting parameters in production.

Furthermore, parallel multi-estimation adaptive control architecture is proposed in order to allow adaptation of the system when the cutting parameters are changed, for example, due to variability in production requirements. It deals with non-linearity, cutting parameter and material dependent milling process factors. Each estimator of the parallel scheme incorporates a recursive least square estimator and produces a control law at the same sampling instant.

The parameters of each transfer function, such as the dynamics and the parameters of the cutting process can vary at each working point and in the transitions between working points. To address this problem, an intelligent supervisory system is presented which selects adequate trajectories in the state space according to design requirements. Furthermore, it is able to learn trajectories and compare them based on a cost function from previous states to generate a knowledge based. This supervisory system closes the control loop. A simulation example is provided to explain how the system acts giving an intuitive understanding of the problem.

2. SYSTEM DESCRIPTION

Milling processes are well characterized as mechanical systems that are particularly sensitive to acquiring vibrations. In this section, the milling process is modelled as a second order differential equation, which is excited by forces whose inherent terms describe excitation of the modal parameters of the system. This fact results in the conversion of resultant energy into vibrations of the system. Those vibrations are generated under certain cutting conditions depending on the process being carried out, clamping of the workpiece, tool and workpiece materials, etc.

2.1. Self-excited vibrations

The standard milling system can be described as a second order differential equation excited by the cutting forces,

$$M \cdot r(t) + B \cdot r(t) + C \cdot r(t) = F(t)$$
⁽¹⁾

where $r(t) = \{x(t), y(t)\}^T$ are the relative displacements between the tool and the workpiece in the X - Y plane, $F(t) = \{F_x(t), F_y(t)\}^T$, and M, B and C are the modal

mass, damping and stiffness matrices respectively, all of them represented in two dimensions. The milling cutting force is represented by a tangential force proportional to the instantaneous chip thickness, and a radial force which is expressed in terms of the tangential force [6],

$$F_t(t) = K_t \cdot a_{dc} \cdot t_c(t) \text{ and } F_r(t) = K_r \cdot F_t(t)$$
(2)

where K_t and K_r , the tangential and radial specific cutting constants which are dependent on the tool material for any specific geometry, a_{dc} , the axial depth of cut and, $t_c(t)$, the chip thickness, obtaining the cutting forces in Cartesian coordinates. The most critical variable in the equation of motion, the chip thickness, $t_c(t)$, consists of a static part and a dynamic one. The static part is proportional to the feed rate and it is attributed to the rigid

body motion of the cutter. The dynamic part models two subsequent passes of the tool through the same part of the work-piece. The phase shift between two consecutive passes of a tool tooth on the work-piece is widely modelled and represented [6] by,

$$t_{c}(t) = f_{r} \cdot \sin \phi_{j} + \left[x(t-\tau) - x(t) \right] \cdot \sin \phi_{j} + \left[y(t-\tau) - y(t) \right] \cdot \cos \phi_{j}$$
(3)

where f_r is the feedrate, ϕ_j the immersion angle and τ is a delayed term defined as $\tau = \frac{60}{N.S_c}$, N_t is the number of teeth and S_s the spindle speed in rpm. Figure 1 pictures

 $N_t S_s$ this mathematical representation in a drawing.



Figure 1: Cross-sectional view of a milling tool [4].

2.2. Machine tool transfer function

The transfer function of the system, in chatter and resonant free zones, can be separated as a series decomposition of the transfer function that relates the actual feed delivered by the drive motor and the resultant force due to the deflection of the tool and, the transfer function that represents the computerized numerical control (CNC). Then, a continuous transfer function which relates both signals, measured resultant force and the actual feed delivered by the drive motor can be shown as a first order dynamic [4],

$$G_p(s) = \frac{F_p(s)}{f_a(s)} = \frac{K_c a_{dc} r(\phi_{st}, \phi_{ex}, N_t)}{N_t S_s} \frac{1}{\tau_c s + 1}$$
(4)

where $K_c(N/mm^2)$ is the resultant cutting pressure constant, $a_{dc}(mm)$ is the axial depth of cut, $r(\phi_{st}, \phi_{ex}, N_t)$ is a non-dimensional immersion function, which is dependent on the immersion angle and the number of teeth in cut, N_t is the number of teeth in the milling cutter, $S_s(rpm)$ the spindle speed and $\tau_c = \frac{1}{N_t S_s}$. At the same time, the relationship between the machine tool controls, the CNC and, the motor drive system can be approximated as a first

order system within the range of working frequencies [4]. This transfer function relates the actual, f_a , and the command, f_c , feed velocities,

$$G_s(s) = \frac{f_a(s)}{f_c(s)} = \frac{1}{\tau_s s + 1}$$
(5)

where τ_s represents an average time constant.

The combined transfer function of the system is given by,

$$G_c(s) = \frac{F_p(s)}{f_c(s)} = \frac{K_p}{(\tau_c s + 1)(\tau_s s + 1)}$$
(6)

with $K_p \left(\frac{kN \cdot s}{mm} \right) = \frac{K_c a_{dc} r_{dc}}{N_t S_s}$

3. COST FUNCTION FOR SELECTION OF OPTIMAL CUTTING PARAMETERS

A novel cost function has been conceived to allow an inference engine to carry out the selection of suitable cutting parameters. The tool cost model for a single milling process can be calculated using the following equation,

$$J\left(TOL, MRR, SURF; R, c_{i(i=1,\dots,3)}\right) = c_1 \cdot NF_1 \cdot TOL + c_2 \cdot NF_2 \cdot MRR + \frac{c_3 \cdot NF_3}{SURF}$$
(7)

The cost function has three terms. Each term is composed of a weighting factor (c_i) , a normalisation factor (NF_i) and the function that defines the process efficiency. These functions are: the life of the tool, TOL; the material remove rate, MRR; and the desired surface finish, SURF. The tool cost function is designed to be directly proportional to the life of the tool and material remove rate and inversely proportional to surface roughness. Hence, optimal solutions will maximise TOL and MRR while minimising SURF. These parameters play an important role when selecting cutting parameters since they are usually used as benchmark indices in industries to measure the performance of the system.

The authors describe in [7] the definition of the parameters and the algorithm that automatically manages the selection of adequate cutting parameters. Nine parameters that compose the cost function and two algorithmic methodologies to manage the weighting factors of the cost function are required, depending on production requirements. The paper also discusses restrictions which limit the milling process and some expert skills are included in order to solve the optimization process, giving a complete procedure to obtain quasioptimal cutting parameters. These definitions are applied in the example application described in section 4 in this paper.

The cost function calculates cutting parameters depending on process requirements. The below explained model reference adaptive control scheme is able to keep the forces below a prescribed upper bound in spite of sudden changes on cutting parameters, spindle speed and depth of cuts, pre-programmed by the optimization algorithm.

4. MODEL REFERENCE ADAPTIVE CONTROL SCHEME

The objective of this section is to provide a control scheme which improves the transitory behaviour of the system and the transitions in between optimal working points

through an intelligent plan. For this purpose, the trajectory between optimal cutting parameters in the state space parameter is selected.

4.1. State space trajectories generation

described as:

Consider
$$q_a = (S_{s,a}, a_{dc,a}, f_{c,a})$$
 and $q_b = (S_{s,b}, a_{dc,b}, f_{c,b})$ the points which optimise J

(equation 7) and it is required transition between. Consider the transfer functions $G_a = \frac{Na}{D}$

and $G_b = \frac{N_b}{D_b}$ which describe the behaviour of the system in those points. The transfer function which describes the system at equilibrium points between q_a and q_b can be

$$G_{\alpha}(s) = \frac{N_{\alpha}(s)}{D_{\alpha}(s)} = \frac{\alpha N_{a}(s) + (1-\alpha)N_{b}(s)}{\alpha D_{a}(s) + (1-\alpha)D_{b}(s)}$$
(8)

for any $\alpha \in [0,1]$ which is dependent on the equilibrium point. This parameterization allows moving around the state space; just adjusting the parameter α , provides the system with the capacity to improve transitory states and enhance transitions in between optimal working points travelling through different states in the state space.

The transfer function around each equilibrium point can be defined by an α -value in equation 8. The trajectory of the system passes through the different operational points when the system changes from one to other optimal cutting parameters due to process requirements. Then, the linear model with the best approximation to the behaviour of the system at each instant will vary during the switching process. This motivates the idea of considering different linear models of the system, each one associated with one different possible working point in between two optimal cutting parameters of the system to be switched, i.e., with a different value of α in equation 8. Furthermore, the objective of the discrete controller is to follow the prefixed dynamics of a reference model. For this reason, it is considered as a discretization for each linear model to design the controller. In this sense, it is defined a set of possible values of α as $S_{\alpha} = \left\{ \alpha^{(1)}, \alpha^{(2)}, ..., \alpha^{(n_{\alpha})} \right\}$, with $\alpha^{(i)} \in [0,1]$ for $1 \le i \le n_{\alpha}$. Then, the multi-model scheme is composed from different discretization for each working point. The set of discrete transfer

is composed from different discretization for each working point. The set of discrete transf functions can be written as.

$$H_{\alpha^{(i)}}(z) = H^{(i)}(z) = Z\left[h(s) \cdot G_{\alpha^{(i)}}(s)\right] = \frac{B_{\alpha}(z)}{A_{\alpha}(z)} = \frac{k_{\alpha} \cdot B_{\alpha}(z)}{A_{\alpha}(z)}$$
(9)

for $1 \le i \le n_{\alpha}$. The different models are used to parameterize the set of controllers which generate the possible control signals. Each discrete control signal is reconstructed through a zero order hold (ZOH). Then, the effect of the continuous system over the milling system is simulated, obtaining a set of outputs defined as $F_p^{(i)}(t)$ for $1 \le i \le n_{\alpha}$. Those signals are compared with the reference continuous output, $F_{pm}(t)$, to obtain a following error, $e^{(i)}(t)$, associated with each discrete linear model. Figure 2 shows the control scheme for the full set of α -values. In this figure, there are n_{α} blocks, each one consists of basic controllers explained below.



Figure 2: Proposed control scheme.

4.2. Performance index

The supervisory controller evaluates a performance index based on the following error associated to each possible discrete time model. The used index has the following form:

$$J_{k}^{(i,j)} = \gamma \sum_{\substack{l=k-M\\ l=k-M}}^{k} \lambda^{k-l} \int_{(l-1)T_{s}}^{lT_{s}} \left(F_{pm}(\tau) - F_{p}^{(i,j)}(\tau) \right) d\tau + (1-\gamma) \sum_{\substack{l=k+1\\ l=k+1}}^{k+M} \lambda^{l-k} \int_{(l-1)T_{s}}^{lT_{s}} \left(F_{pm}(\tau) - F_{p}^{(i,j)}(\tau) \right) d\tau$$
(10)

for $1 \le i \le n_{\alpha}$, where $\lambda \in (0,1]$ and M > 0 are real parameters of the design. λ is a "forgetting factor" that gives more importance to the last sampling instants. $F_{pm}(t)$ denotes the reference continuous output to be followed and $F_{p}^{(i)}(t)$ is the output obtained from simulating the effect of the controller associated with the non-linear milling system.

4.3. On-line updating of the α gain

When the supervision system chooses switching from the active controller, which is associated to the gain of the linear model α_{old} , to another controller associated to the gain of the linear model α_{new} , this switching does not carry out instantaneously. The transition from one α -value to another is implemented over a predetermined number of samples. At the same time, the controller is selected associated to the gain of the linear model α_{k}^{act} , with $\alpha_{k}^{act} \in [\alpha_{old}, \alpha_{new}]$ if $\alpha_{old} < \alpha_{new}$ or $\alpha_{k}^{act} \in [\alpha_{new}, \alpha_{old}]$ if $\alpha_{old} > \alpha_{new}$, at each sampling time

 $t_k = kT_s$ in the switching process. For this reason, α_k^{act} could not be one of the possible values of S_{α} and they can take any value in the interval [0,1].

The following algorithm describes the switching mechanism and, in particular, the method of updating the value of the gain α_k^{act} which determines the value, α , of the active discrete model used to parameterize the applied controller.

$$if k = jN_r, then$$

$$\alpha_{new} = \alpha^{(i_{\min})} \text{ with } i_{\min} = \arg\left(i : J_k^{i} = \min_{1 \le i \le n_\alpha} \left(J_k^{i}\right)\right)$$

$$\alpha_{old} = \alpha_k^{act}$$
end if
$$\alpha_{k+1}^{act} = \begin{cases} \min\left(\alpha_k^{act} + \left(\alpha_{new} - \alpha_{old}\right) / N_\alpha, \alpha_{new}\right) \text{ if } \alpha_{new} \ge \alpha_{old} \\ \max\left(\alpha_k^{act} + \left(\alpha_{new} - \alpha_{old}\right) / N_\alpha, \alpha_{new}\right) \text{ if } \alpha_{new} \le \alpha_{old} \end{cases}$$

where N_{α} is the number of samples over which the scheme updates the value of α_k^{act} from α_{old} to α_{new} .

4.4. Basic controller

Finally, it is necessary to calculate as many control laws as there are linear discrete models. Moreover, since the active value of α_k^{act} in the active discrete model can be different to the included in S_{α} , it is necessary to calculate $n_{\alpha} + 1$ control signals. However, only the control signal associated to α_k^{act} will be used to generate the actual control signal.

In the control scheme, the reference discrete transfer function is calculated as follows:

$$H_m(z) = \frac{B_m(z)A_o(z)}{A_m(z)A_o(z)} = Z[h_o(s) \cdot G_m(s)]$$
(11)

where $A_o(z)$ is cancelled in order to design a causal controller. Now, define $B_m(z) = B'_m(z)k^{(i)}$, where $k^{(i)} = k_{a^{(i)}}$ is the gain in equation 11. Then, the polynomials of the controller are $T(z) = B'_m(z)A_o(z)$, and $R^i(z)$ (monic) and $S^i(z)$ which are unique solutions of the Diophantine equation

$$A^{(i)}(z)R_{1}^{(i)}(z) + k^{(i)}S^{(i)}(z) = A_{m}(z)A_{o}(z)$$
(12)

fulfilling the following conditions on the polynomial grades:

$$gr\left(S^{(i)}\right) = 2n - 1; gr\left(A_{o}\right) = 2n - gr\left(A_{m}\right) - gr\left(B^{(i)}\right) - 1$$
$$gr\left(R^{(i)}\right) = gr\left(A_{o}\right) + gr\left(A_{m}\right) + gr\left(B^{(i)}\right) - n$$

for $1 \le i \le n_{\alpha}$ with $R^{i}(z) = B^{i}(z)R_{1}^{i}(z)$ at each sampling time. Leading to the

following control law,
$$f_{c,k}^{(i)} = \frac{T(q^{-1})}{R_k^{(i)}(q^{-1})} f_{c,k} - \frac{S_k^{(i)}(q^{-1})}{R_k^{(i)}(q^{-1})} F_{p,k}^{(i)}$$

(13) for $1 \le i \le n_{\alpha}$. with $\hat{\theta}_{k+1}^{(l)} = \hat{\theta}_{k}^{(l)} + \frac{P_{k}^{(l)}\phi_{k}^{(l)}e_{k}^{(l)}}{1 + \phi_{k}^{(l)}P_{k}^{(l)}\phi_{k}^{(l)}}; \hat{\theta}_{o}^{(l)} arbitrary$, provided that $tr\left(\hat{A}_{o}, \hat{B}_{o}\right)$ is a co-prime

pair,

$$P_{k+1}^{(l)} = P_{k}^{(l)} - \frac{P_{k}^{(l)} \phi_{k}^{(l)} \phi_{k}^{(l)^{T}} P_{k}^{(l)}}{1 + \phi_{k}^{(l)} P_{k}^{(l)} \phi_{k}^{(l)}}; P_{o}^{(l)} = P_{o}^{(l)^{T}} > 0 \text{ where } e_{k}^{(l)} = F_{p,k} - \hat{F}_{p,k}^{(l)} \text{ is the identification error}$$

for the k^{th} sample $\forall l \in N_{\alpha} = \left\{1, 2, ..., n_{\alpha}\right\}$.

An example of implementing the proposed control scheme is described in the next section.

5. EXAMPLE APLICATION

A practical 3-tooth, 30 mm diameter end mill with the modal characteristics in the X and Y directions corresponding to table 1 is chosen for this example. The work-piece is a rigid aluminium block whose specific cutting energy is $k_t = 600kN \cdot mm^{-1}$ and the proportional factor is taken to be $k_r = 0.07$.

	$\omega_n\left(rad\cdot s^{-1}\right)$	$\xi(\%)$	$k(KNmm^{-1})$
Х	603	3.9	5.590
Y	666	3.5	5.715

Table 1 Modal parameters of the tool.

For the model reference control, the transfer function of equation 6 has a cutting pressure selected to be constant and equal to $1200N \cdot mm^{-2}$ in all range of cutting parameters, the *CNC* time constant, $\tau_m = 0.1ms$ and, $\tau_c = 1/N_t S_s$. The continuous model reference system of the adaptive control is chosen to be a typical continuous second order plant with $\xi = 0.75$ and $\omega_n = 2.5/(4 \cdot T)$, where *T* is the sampling period. Also, it is desirable for the reference force to be maintained at 1200N.

The input space parameter where the system searches for quasi-optimal cutting parameters is given by the stability borderline (first graph of figure 3). This figure indicates that if programming cutting parameters are over the borderline then chatter vibrations will appear and the system will become unstable [4]. However, if programmed cutting parameters are below this borderline the system will work correctly without suffering chatter vibrations. Other mechanical and electrical restrictions when searching for programming adequate cutting parameters are related to spindle power consumption and feed drive limitations. Other safety constraints can be added in order to avoid uncertainty in searching regions and avoiding impractical minima during optimisation.

For example purposes, it is supposed that the following two cutting parameters represent two Pareto optimal fronts, $S_{s1} = 2710$ and $a_{dc1} = 0.3710$, as point 1, and $S_{s1} = 3260$, $a_{dc1} = 0.4127$, as point 2. A more in-depth explanation of how to obtain Pareto optimal cutting parameters is provided by Rubio et al. [7].

Regarding the control scheme, it is taken into account that $\alpha = \{1, 0.7, 0.5, 0.3, 0\}$, being $\alpha_{\min} = 1$ and $\alpha_{\max} = 0$, corresponding with the points 1 and 2, respectively. Figure 3 depicts, from top to bottom, the stability lobes with the situation of the suggested Pareto optimal cutting parameters, the system output (resultant and reference forces) and the programmer feed rate when it is required to change the cutting parameter from point 1 to point 2 with and without applying the controller. The left hand figures show the outputs and control commands when the control scheme is applied. The right side corresponds to the case where the control scheme is not implemented. When the cutting conditions change, a peak appears in the resultant forces. This peak can lead to excessive wear and damage or even breakage of the tool or machine components. Moreover, those peaks can have a detrimental influence on the surface finish of the workpiece. In the proposed control algorithm the cutting forces are maintained constant by adjusting the feed rate according to the presented algorithm.

Then, as shown in figure 3, the Pareto optimal cutting parameters proposed by the selfoptimized system are below the stability borderline in the stable zone. Furthermore, the presented controller is able to move automatically around the allowable cutting space parameter, keeping the forces below a prescribed upper limit bound while programming feasible command federate in spite of changes in cutting parameters.

Finally, the implemented controller can also be tuned in order to reduce the overshoot of the transitory state, which could also lead to damage or breakage of the tool or tool-holder and machine components and leave uneven surface finish.



Figure 3: Situation of the programmed cutting parameters in stability lobes, output force signal and control signal with and without applying the control

6. CONCLUSIONS

This paper proposes a novel control scheme for controlling cutting parameters in milling applications. It is composed of two levels. In the first level, the self-optimised cutting parameters layer comprises life of the tool, material remove rate, surface roughness and the robustness of the system. While the second layer, the parallel multi-estimation controller, provides an environment to control the milling process automatically under changes in cutting conditions. The change of the cutting parameters is scheduled by production requirements. For this purpose, an algorithm methodology is proposed in order to automatically adjust the parameter α choosing the most suitable controller among the set designed for each programmed optimal cutting parameters. The designed controller is able to smooth the transition between discrete control models and so reduce the peaks which appear when sudden changes are made in the cutting parameters.

The fundamental idea of the control system is to work automatically, with a simple interface with the operator, based around the admissible cutting parameter space given by the well-known stability lobes. First, the optimization cost function is used to obtain the cutting parameters according to multiple objectives optimization. Secondly, the adaptive control scheme proposes different control laws working in parallel to address the non-linear and changeable milling process. Finally, the supervisory scheme manages the system so it can work automatically in between optimal working points. Simulation results support the performance of the system.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the UK's Engineering and Physical Science Research Council (EPSRC) funding of the EPSRC Centre for Innovative Manufacturing in Advance Metrology (Grant Ref: EP/I033424/1).

REFERENCES

- F. Cus, and J. Balic, "Optimization of cutting process by genetic algorithm approach," *Robotics and compute-integrated manufacturing*, vol. 19, pp. 113-121, 2003.
- [2] J.V. Abellan, F. Romeros, H.R. Siller, A. Estrud and C. Vila, "Adaptive control optimization of cutting parameters for high quality machining operations based on neural network," *Advance in robotics, Automation and Control*, Itech, Vienna, Austria, 2008.
- [3] U. Zuperl, and F. Cus, "Optimization of cutting conditions during cutting using neural networks," *Robotics and computer integrated manufacturing*, vol. 19, no. 1-2, pp. 189-199, 2003.
- [4] Y. Altintas, Manufacturing automation, Cambridge University Press, 2012.
- [5] U. Zuperl, F. Cus and M. Milfelner, "Fuzzy control strategy for an adaptive force control of forces in milling processes," *Journal of Material Processing Technology*, pp.1472-1478,2005.
- [6] L. Rubio, M. De La Sen, and A. Bilbao-Guillerma, "Intelligent adaptive control of forces in milling processes," Proceedings of the 2007 IEEE Mediterranean Conference on Control, and Automation, pp. 1-6, 2007.
- [7] L.Rubio, M. De La Sen, A.P.Longstaff and S. Fletcher, "Model-based expert system to automatically adapt milling forces in Pareto optimal multi-objective working points", Expert Systems with Applications, 40, 2312-2322, 2013.