Iterative Learning Control for Machining with Industrial Robots

Cano Marchal, Pablo; Sörnmo, Olof; Olofsson, Björn; Robertsson, Anders; Gómez Ortega, Juan; Johansson, Rolf

Link to publication

Total number of authors:
6

Creative Commons License:
Unspecified

Iterative Learning Control for Machining with Industrial Robots [★]

Pablo Cano Marchal [*] **Olof Sörnmo** [**] **Björn Olofsson** [**]
Anders Robertsson [**] **Juan Gómez Ortega** [*] **Rolf Johansson** [**]

[*] *Group of Robotics, Automation and Computer Vision,*
University of Jaén, ES–23071 Jaén, Spain (E-mail: pcano@ujaen.es).
[**] *Department of Automatic Control, LTH, Lund University,*
SE–221 00 Lund, Sweden (E-mail: firstname.lastname@control.lth.se).

Abstract: We consider an iterative learning control (ILC) approach to machining with industrial robots. The robot and the milling process are modeled using system identification methods with a data-driven approach. Two different model-based ILC algorithms are proposed and subsequently experimentally verified in a milling scenario. The difference between the two approaches is the required sensors for acquiring relevant input data for the algorithms. The results from the experiments indicate that the proposed methods have the potential of significantly decreasing the position errors in robotic machining, up to 85% in the considered milling scenario.

1. INTRODUCTION

Milling and other machining processes are common tasks in modern industrial manufacturing. Usually, the milling task is executed using dedicated computer numerical control (CNC) machines. The employment of industrial robots for this kind of operations is of interest, mainly because of the lower investment cost of a robot cell compared to a CNC machine, but also the reconfiguration flexibility of the robot. However, the usage of industrial robots for machining has been limited, as the machining tolerances generally cannot be met—typical values for the accuracy that can be achieved are in the range of 1 μm for CNC machines and approximately two magnitudes higher for conventional industrial robots. A significant contribution to the insufficient accuracy in milling tasks performed with industrial robots is the combination of comparably low stiffness of the joints in a serial-kinematic manipulator and external process forces affecting the end-effector. Typically, the robot motion control is based on distributed feedback from the resolvers in the individual joint motors. Consequently, the desired arm-side or task-space measurements are not available. Since a major part of the induced position deflections in the milling process appear on the arm-side of the robot joints, they cannot be detected by the joint resolvers. Hence, feedforward control based on high-accuracy robot and process models or online feedback from external task-space sensors are required in order to be able to compensate the arm-side position deviations. External position and orientation sensors for task-space measurements with the required accuracy and resolution are typically quite expensive, compared to the cost of the robot. On the other hand, wrist-mounted force sensors are a cheaper alternative of acquiring task-space information of the robot, although relating the measured data with the position of the robot in a machin-
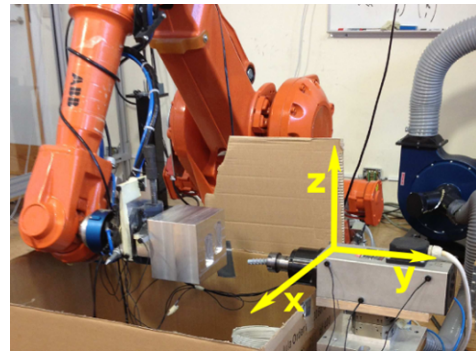

Fig. 1. The experimental platform used for evaluation of ILC in milling scenarios, with the robot holding the workpiece and a fixed spindle with the milling tool.

ing scenario is not straightforward and requires robot-specific model information.

Background   Previous methods for increasing the accuracy of robotic machining are primarily based on contact-force control (Hogan, 1985) and stiffness compensation, *e.g.*, (Wang et al., 2009; Reinl et al., 2011). Iterative learning control (ILC) has been proposed as an offline method for achieving higher accuracy in motion control for robots (Arimoto et al., 1984), and later robots with joint and link flexibilities (Miyazaki et al., 1986; Norrlöf, 2000, 2002; Hakvoort et al., 2007) performing repetitive tasks. The idea is to measure the position deviations during the first execution and then update the control inputs or the specified path and trajectory for subsequent iterations. Under certain assumptions, convergence of this iterative procedure can be theoretically proven. However, all of the mentioned references consider situations where the robot end-effector is moving in free space. In addition, it should be noted that real-time algorithms for ILC have been proposed (Xu et al., 2010).

Problem Formulation   In this paper, the application of ILC for the geometric robot path in a machining task—such as milling, deburring, and grinding—is investigated with the purpose of

increasing the obtained accuracy of the machined parts. The difference to previous applications of ILC in robotics is the contact between the workpiece attached to the robot and the tool on the spindle, which is required for completion of the machining task. This constitutes a major difficulty in applying the iterative scheme, since the force interaction between the tool and the workpiece must be considered when determining the updated geometric path. The motivation for seeking ILC-based solutions to the problem of increasing the position accuracy in machining is the often batch-oriented nature of modern production, which comprises significant repetitiveness in the tasks to be performed. Two different versions of ILC algorithms are developed and subsequently investigated in milling experiments in this paper; the difference between the two being the available sensor data. Obviously, with more relevant sensor data available, higher performance can be achieved. First, ILC is considered based on arm-side measurements in task-space of the robot with an optical 6D tracking system, in order to demonstrate the effectiveness of the strategy for reducing the position deviations. Then, a model-based ILC algorithm is developed, which is only based on the internal joint resolver position data and measurements provided by a wrist-mounted force/torque sensor attached to the robot end-effector. The latter approach is appealing, since it eliminates the need for expensive external tracking systems.

This paper is organized as follows: Section 2 presents the modeling of the robot and the milling process, a subsequent identification procedure for the required models, and the details of the proposed ILC algorithms. The experimental setup and the obtained results from ILC applied to milling in aluminium are presented in Sec. 3. The obtained results are commented upon and the method as such is contrasted to other approaches for increasing the robotic machining accuracy in Sec. 4. The paper is summarized in Sec. 5, where conclusions also are drawn.

2. METHOD

In order to provide a basis for the design of model-based ILC algorithms, the robot and the effects of the milling process on the robot are modeled using system identification methods (Johansson, 1993).

2.1 System Modeling and Identification

As mentioned in Sec. 1, the main difficulty when milling with an industrial robot is the deflections that occur because of process forces that act on the robot end-effector, which in turn result in position errors at the end-effector. In order to model this dynamic compliance relation, it is considered as being composed of two different inherently coupled phenomena; the deflection of the robot when a force is applied to its end-effector, and the process forces arising when milling is performed. The model for the first phenomenon aims to link the applied force with the deflection by the robot—*i.e.*, the relationship which is commonly modeled using Hooke's Law (Ugural and Fenster, 2003) or generalizations thereof—while the second model intends to relate the path traversed by the robot end-effector with the forces that appear during the milling. The block diagram of the complete model of the robotic milling system is presented in Fig. 2. The model variables and subsystems of the block diagram are:

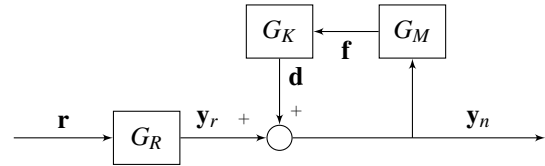- **r**: position reference for each spatial coordinate;

Fig. 2. Block diagram of the milling system model.

- $\mathbf{y}_r$: Cartesian position of the robot computed from joint resolvers;
- $\mathbf{y}_n$: actual Cartesian position of the robot;
- **d**: deflection disturbance caused by the milling process-forces;
- **f**: measured forces on the end-effector along the three Cartesian axes;
- G_R: model of the controlled robot in Cartesian space (including the internal joint-position feedback controllers). More specifically, the transfer function from **r** to $\mathbf{y}_r$ when the robot is moving in free space;
- G_M: model relating $\mathbf{y}_n$ to **f**—*i.e.*, the model describing the process forces in the milling task;
- G_K: compliance model relating the applied force **f** and the deflection **d** of the robot in Cartesian space.

The following relations formalize the modeling approach described in the previous paragraph and illustrated in Fig. 2:

$$\mathbf{y}_n = G_R\,\mathbf{r} + \mathbf{d}, \quad \mathbf{d} = G_K\,\mathbf{f}, \quad \mathbf{f} = G_M\,\mathbf{y}_n. \qquad (1)$$

Robot Identification   A model G_R of the controlled robot motion in Cartesian space is estimated using system identification methods. Since the controlled robot motion is considered only in a limited Cartesian workspace in the milling task [1], linear models can be justified. In particular, a chirp excitation-signal is applied along each direction, and the response in position is measured. Using a time-series modeling approach, third-order discrete-time dynamic models are identified and subsequently used in the ILC algorithm. The models capture the inherent resonant character of the mechanical structure, which is a result of the joint and link flexibilities of the robot.

Deflection Model Identification   As stated earlier in this section, the purpose of the model G_K is to relate the deflections exhibited by the robot with the forces applied on its end-effector. For the identification of this compliance dynamics, the main idea is to compare the behavior of the system when the robot end-effector is moving in free-space, and there are no external forces acting on the robot end-effector, to the case when the milling is performed and process forces are required. Thus, the identification procedure is composed of two experiments, where the same geometric reference path is chosen in both:

- Robot holding a workpiece and moving in free space;
- Robot holding a workpiece and performing milling.

The sensor data signals to be recorded for identification purposes during the experiments are:

- The forces acting on the end-effector (**f**);
- The estimated position of the robot end-effector, computed from the joint-motor resolvers using the forward kinematics of the robot ($\mathbf{y}_r$);

---

[1] Considering that the robot configuration influences the controlled dynamics of the robot end-effector, configuration-dependent linear parameter-varying models or nonlinear models (and thus nonlinear ILC methods) explicitly accounting for this property are required if larger workspaces are to be considered.

- The arm-side position of the robot end-effector measured by an external tracking system $(\mathbf{y}_n)$.

The required sensor data can be collected during the initial uncompensated ILC iteration using the nominal geometric path, thus avoiding the need for separate identification experiments prior to application of the ILC iterations.

The tracking system data acquired during the free-space experiment, with only inertial forces from the workpiece affecting the robot end-effector, are subtracted from the data obtained during the milling experiment. Hence, the remaining quantity is the difference between the actual robot path in the milling and the free-space motion—*i.e.*, the deflection to be modeled—because of the milling. Then, models of order eight were fitted for each Cartesian axis, with the force as input and the deflection on the arm-side as output, using the N4SID subspace algorithm (van Overschee and De Moor, 1994), with the implementation in the System Identification Toolbox (Ljung, 2010) in MATLAB. For simplicity it is assumed that the orientation of the workpiece is constant during the milling. This means that the fixed coordinate system in Fig. 1 is used throughout the paper. Considering the milling geometry for the current setup, see Fig. 1, the majority of the material removal will be performed in the *X-Z*–plane, and it is therefore clear that the significant position deviations will occur along these axes. Hence, without loss of generality, the focus in the presentation here is on these axes. Figures 3 and 4 show the experimental data and the output of the models for the *X* and *Z* axes, respectively. The fit of the models G_K to the data is computed using the normalized root-mean square error (NRMSE), which are 41% and 59% for the *X* and *Z* axis, respectively. As can be observed in the figures, the fit of the models to the experimental data is satisfying, capturing the major parts of the compliance dynamics. Figure 5 shows the Bode diagram of the identified model along the Z axis, where it can be seen that for stationarity and low frequencies the model is basically a constant gain, as anticipated from a system that adheres to Hooke's Law. For frequencies of the input signal at approximately 12 Hz the dynamics exhibit a significant resonance. In addition, minor resonances are observed as a result of the flexibilities in the attachment of the workpiece to the robot end-effector. This motivates why a static deflection model is not sufficient in a machining scenario; rather a dynamic model is required if dynamic errors with frequencies above the first eigenfrequency of the robot are to be compensated.

Process-Force Model Identification   Intuitively, it is natural to assume that the force required during the milling is related to the amount of material that is being removed. This idea has been used to develop a controller capable of keeping the process force constant for time-varying milling conditions (Sörnmo et al., 2012a). The modeling approach proposed in that paper is adopted here. If it is assumed that the surface of the workpiece is smooth, the amount of material per time unit that is fed to the milling tool is proportional to the feed rate of the workpiece. Consequently, the force will be approximately proportional to the feed rate and can thus be considered as a damping—*i.e.*, as a gain multiplied by the derivative of the input. As an intuitive motivation for this proposal, it is realized that if the workpiece velocity is zero, the force will eventually become zero when all the nearby material has been removed and nothing opposes the rotation of the milling tool. Considering the geometry for the cutting process in the milling scenario shown in Fig. 1, it follows that the force along the Cartesian *X* axis is to be
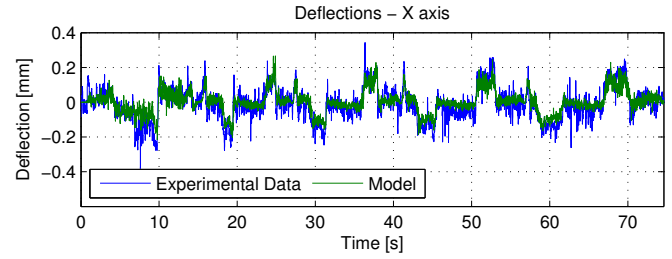

Fig. 3. Measured deflections and model output (G_K) — X axis.
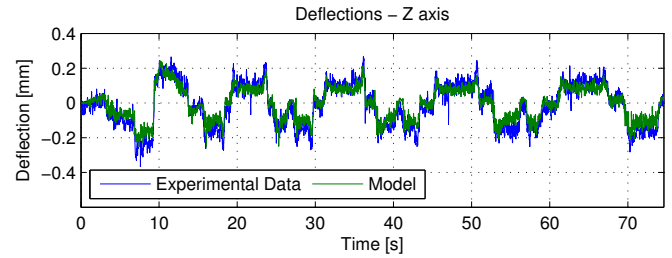

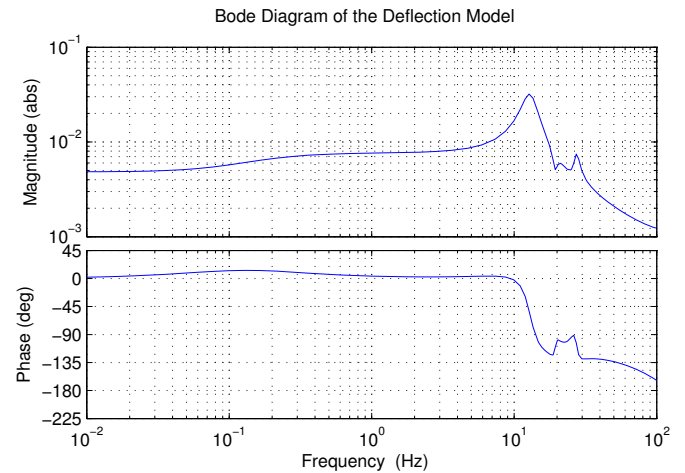Fig. 4. Measured deflections and model output (G_K) — Z axis.

Fig. 5. Bode diagram of the model G_K along the Z axis.

modeled with the velocity along the *Z* axis as input, with a corresponding result for the force along the *Z* axis and the *X* component of the velocity. This observed behavior might be counter-intuitive, but follows directly from the cutting process dynamics. Based on the motivation above, models of the form

$$G_M(s) = K \frac{s}{(1+\tau s)^2}, \qquad (2)$$

where K is a gain, and a double pole with time constant τ (chosen short compared to the dominant milling dynamics) have been included in order to have a strictly proper transfer function and thus avoiding amplification of high-frequency noise in the numerical implementation of the ILC algorithm. It is here assumed that the workpiece is in contact with the tool for the duration of the milling task, and consequently that process forces arise when having nonzero velocities. Figures 6 and 7 depict the measured forces and the output of the models for the *X* and *Z* axes respectively. The NRMSE fit of the models to the data are given by 47% and 56% for the *X* and *Z* axis, respectively. The models exhibit acceptable fit to the experimental data for both axes, which indicate that the models capture the essential dynamics of the system with some minor discrepancies in the static gain for parts of the milling task.
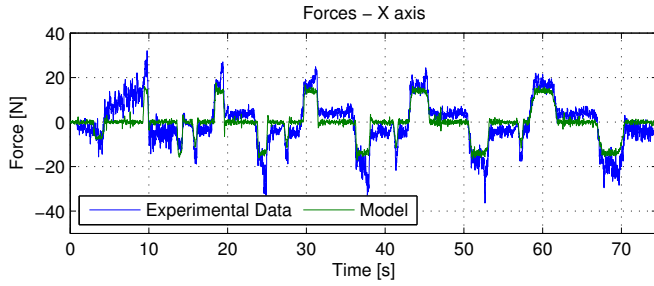
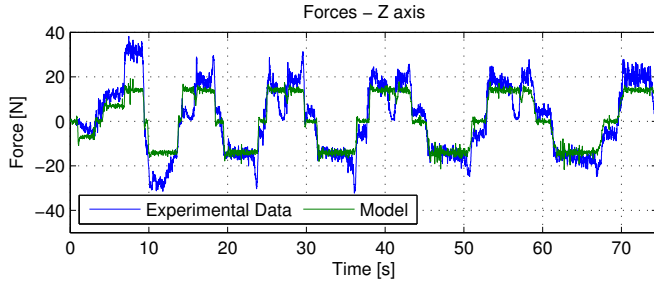Fig. 6. Measured process forces and model output (G_M) — X axis.

Fig. 7. Measured process forces and model output (G_M) — Z axis.
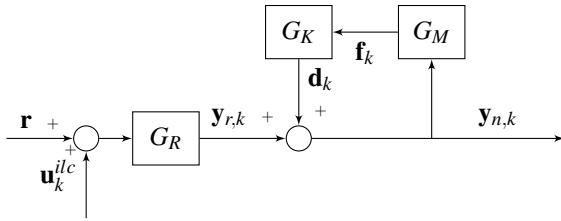

Fig. 8. Block diagram of the ILC-controlled milling system.

2.2 ILC Algorithms

In this section, two different ILC update algorithms are derived and theoretically justified. The first relies on external arm-side measurements in task-space, while the second only relies on end-effector force/torque data.

Arm-Side Measurement ILC   In order to reduce the magnitude of the position errors during the milling task iteratively, the first approach proposed is to apply an ILC algorithm using the arm-side position measurements, provided by an optical tracking system, as the output of the system. The fundamental idea of the ILC algorithm in the milling context is to improve the previous reference path sent to the system by adding a certain quantity to this path, based on the measured error in the previous execution. The update is performed such that the expected output in the next iteration is closer to the desired milling path. Figure 8 shows a block diagram of the considered ILC system. Since the arm-side position is measured, the output $\mathbf{y}_{n,k}$ is directly measurable for each iteration k. For this ILC scheme, the model of the controlled system is

$$\mathbf{y}_{n,k} = G_R\left(\mathbf{r} + \mathbf{u}_k^{ilc}\right) + G_K G_M\,\mathbf{y}_{n,k}. \qquad (3)$$

The terms of this relation may be reorganized into

$$\mathbf{y}_{n,k} = (I - G_K G_M)^{-1} G_R\left(\mathbf{r} + \mathbf{u}_k^{ilc}\right), \qquad (4)$$

in order to cast the model on the form

$$\mathbf{y}_{n,k} = T_r\,\mathbf{r} + T_u\,\mathbf{u}_k^{ilc}, \qquad (5)$$

where $T_r = T_u = (I - G_K G_M)^{-1} G_R$. With this relation, the well-known model-based ILC algorithm, *e.g.*, described in (Norrlöf, 2000), can be applied:

$$\mathbf{u}_{k+1}^{ilc} = Q\left(\mathbf{u}_k^{ilc} + L\,\mathbf{e}_k\right), \qquad (6)$$

where k is the iteration index, the position error is defined as $\mathbf{e}_k = \mathbf{r} - \mathbf{y}_{n,k}$, L is an approximate inverse of the transfer function from $\mathbf{r}$ to $\mathbf{y}_{n,k}$, and Q is a low-pass or bandpass filter with suitably chosen frequency properties. In particular, the choice of Q is based on the desired frequency range for the ILC compensation. A further consideration here is the model accuracy for the frequencies of interest, which may imply a cut-off frequency close to the first eigenfrequency of the system.

In order to make the numerical computations more robust, the ILC update law is reformulated as follows

$$\mathbf{u}_{k+1}^{ilc} = Q\left(\mathbf{u}_k^{ilc} + \tilde{L}\left(I - G_K G_M\right)\mathbf{e}_k\right), \qquad (7)$$

where $\tilde{L}$ is chosen as an approximate inverse of G_R. The relation (7) is obtained by rewriting (6) and explicitly accounting for the structure of the system model (compare with the relation in (3)), but avoiding the inversion of the term related to the milling process dynamics. If G_R contains unstable zeros, the inverse can, for instance, be determined by mirroring these zeros in the unit circle prior to computing the inverse.

Force Measurement ILC   As shown in the previous paragraph, the assumption on explicit arm-side measurements led to a straightforward formulation of an ILC algorithm. Unfortunately, the sensors required for these kinds of measurements are expensive and not commonly available in manufacturing industry today. Consequently, it is of interest to develop a method capable of, to some extent, estimating the arm-side measurements without such a sensor. As stated previously, the position errors that occur because of the milling process-forces are not visible using only the joint-position resolvers of the robot. Hence, to eliminate the need for arm-side measurements, an extra sensor that (together with appropriate models) can estimate the arm-side position errors is required. A wrist-mounted force/torque sensor is a cheaper alternative to high-precision optical tracking systems. Consequently, the proposed second approach is to construct a model-based estimation $\hat{\mathbf{y}}_{n,k}$ of the position deviations using the joint-position resolver measurements and the force sensor measurements according to

$$\hat{\mathbf{y}}_{n,k} = \mathbf{y}_{r,k} + G_K \mathbf{f}_k, \qquad (8)$$

to replace the arm-side position measurements $\mathbf{y}_n$. Using this relation, the error signal used in the ILC algorithm is

$$\mathbf{e}_k = \mathbf{r} - \hat{\mathbf{y}}_{n,k} = \mathbf{r} - \mathbf{y}_{r,k} - G_K \mathbf{f}_k, \qquad (9)$$

and the subsequent update law is then equivalent to (7).

3. RESULTS

In this section, the experimental setup used for evaluation of the ILC algorithms in milling experiments is described. Moreover, experimental results obtained from applying the proposed ILC strategies to milling in aluminium are presented and evaluated.

3.1 Experimental Setup

The experimental platform used to perform the milling experiments comprises an ABB IRB2400 industrial robot (ABB Robotics, 2012) with an S4CPlus control cabinet, a spindle holding the milling tool, a Nikon K600 optical tracking system
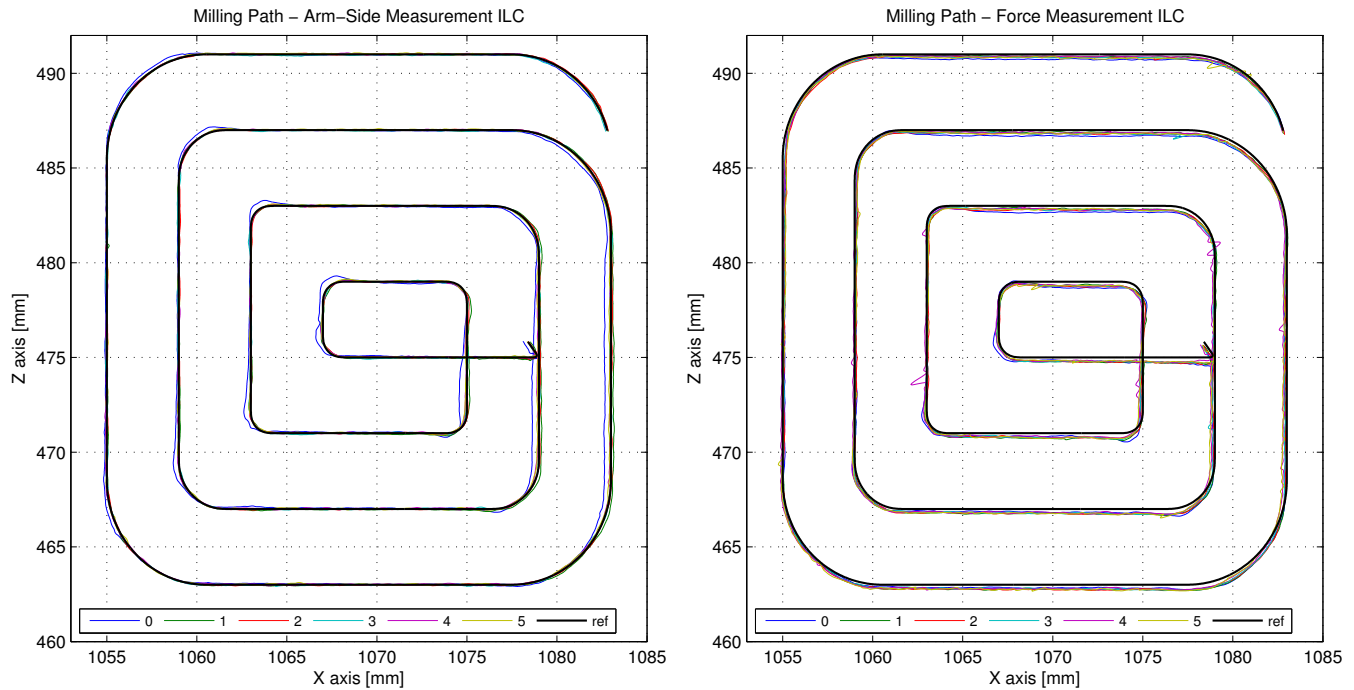
Fig. 9. The reference and measured milling paths for the iterations in the experimental evaluation of the two ILC algorithms (left – arm-side measurement ILC; right – force measurement ILC). The measurements were carried out using the optical tracking system for both figures, but these measurements were only used for evaluation in the force-based ILC algorithm.

(Nikon Metrology, 2010) for task-space measurement of the workpiece position, and a JR3 force/torque sensor of model 100M40A. The optical tracking system has an absolute accuracy of 50–75 μm for the current measurement configuration. Figure 1 shows a photo of the platform, including the definition of the fixed world Cartesian coordinate system used throughout the experiments. Thus, all superscripts denoting the reference frame are omitted for notational convenience. As seen in Fig. 1, a fixed milling spindle was used and the aluminium workpiece (type Al7075) was attached to the robot, which moved along the trajectory required for the milling task to be executed. For low-level access to the robot joint controllers and integration of external sensor data with the robot controller, the ExtCtrl interface (Blomdell et al., 2010) developed at Lund University was employed. Low-level access to the robot controller is beneficial in order to ensure synchronization of the reference and output over the ILC iterations. The specific milling task considered here was to machine a square with a depth-of-cut of 2 mm; the reference milling path used for the experiments is displayed in Fig. 9. With the described experimental setup, the available measurement signals for each of the Cartesian axes were:

- y_r: estimated position of the robot tool, computed from the joint-angle resolvers using forward kinematics;
- y_n: arm-side position of the robot tool measured with the optical tracking system;
- f: force acting on the tool, measured by the wrist-mounted force/torque sensor.

3.2 Experimental Results

The milling experiments were performed on the setup described in the previous paragraph. The results obtained using the two different ILC algorithms are described separately.

Arm-Side Measurement ILC   Figure 10 displays the evolution of the measured position errors with each ILC iteration along the Cartesian coordinate axes (see Fig. 1), while Fig. 11 depicts the normalized magnitude of the errors, defined as the sum of the absolute values of the error in each sample divided by the total error in the first execution (iteration 0). The measured milling paths in each iteration are displayed in the left plot in Fig. 9. As can be observed, the most significant error reduction is obtained already in the first iteration, as can be anticipated by the use of a model-based ILC algorithm. As expected from the direct measurement of the arm-side position of the robot, this approach is capable of removing the offsets present along the *X* and *Y* axes. The *Z* axis also exhibits a significant decrease of the magnitude of the errors, although smaller than the reduction obtained along the other two axes. This may be attributed to the fact that the error magnitude already a priori was smaller, since there was no clearly visible offset error along this axis and only process-induced dynamic errors needed to be compensated.

Force Measurement ILC   Figure 12 displays the errors obtained using the ILC approach with only force measurements, and Fig. 13 shows the evolution of magnitude over iterations. The milling paths (measured using the optical tracking system) in each iteration are displayed in the plot to the right in Fig. 9. It is clear from Fig. 12 that this approach is not capable of detecting and correcting constant offset errors, but is able to reduce the magnitude of the dynamic errors caused by the process forces in the milling task. For this reason, the error decrease achieved with this algorithm is measured as deviations from the offset level. Figure 13 displays the magnitude of the position deviations around the offset, the latter computed as the mean of the error for each axis. In the plot, a decrease of the error can be observed for each axis. The results obtained with this approach are, not surprisingly, less significant than those attained using arm-side measurements, but the method clearly
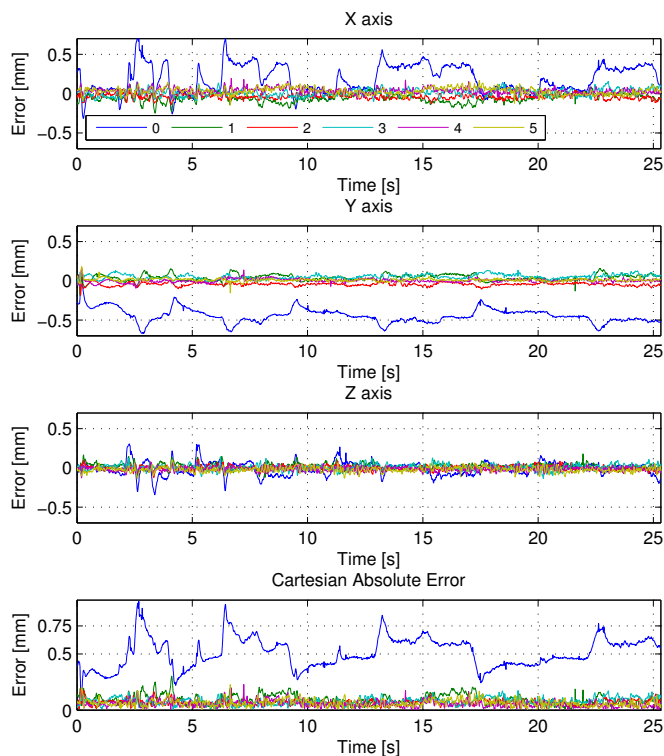
Fig. 10. Errors for each iteration k during milling using the arm-side measurement ILC.
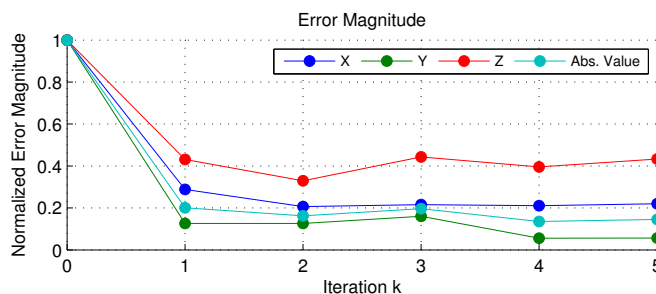

Fig. 11. Magnitude of the error for each ILC iteration k using the arm-side measurement ILC algorithm.

Fig. 12. Errors for each iteration k during milling using the force measurement ILC algorithm. The dashed black lines indicate the offset levels along each axis.

Fig. 13. Magnitude of the deviations around the offset for each ILC iteration k with the force measurement ILC algorithm.

shows its capability of reducing the dynamic component of the errors arising as a result of the process forces. Obviously, considering the absolute error the arm-side measurement ILC algorithm is superior even though a significant decrease of the error is achieved also for the force measurement ILC algorithm.

4. DISCUSSION

As an alternative to previous methods for increasing the accuracy of robotic machining, we have proposed and experimentally verified two ILC algorithms. As mentioned in Sec. 1, the batch-oriented nature of modern production induces significant repetitiveness in the tasks to be performed, thus enabling the possibility of improving the performance of the task from workpiece to workpiece. In addition, the use of model-based ILC enables fast convergence rates, thus minimizing the amount of sub-standard pieces produced during the learning process. According to the error convergence depicted in Figs. 11 and 13, it seems plausible to estimate that a single sub-standard piece for the arm-side ILC and two pieces for the force-based approach are required. Both of the presented ILC algorithms result
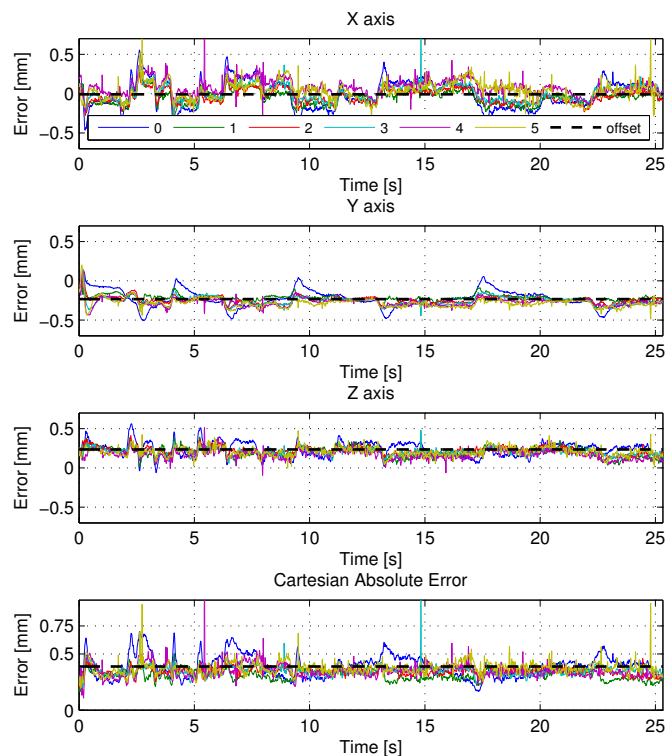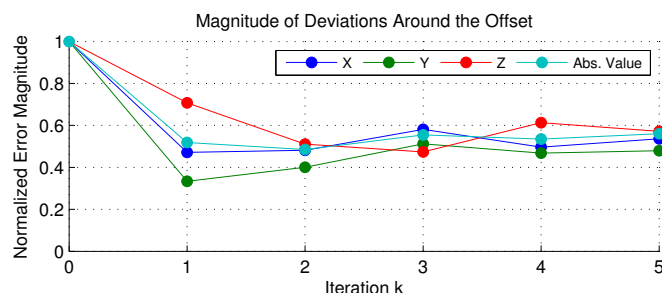
in a significantly increased accuracy. However, the accuracy has not reached the performance offered by CNC machines or that achieved with approaches using additional hardware, such as a macro/micro configuration considered in (Sörnmo et al., 2012b). Still, both methods do improve the accuracy; a decrease of approximately 50% of the uncompensated error is achieved with the force measurement approach. Hence, the results obtained by milling with industrial robots can be improved at a moderate extra cost, particularly using the force-based method that only requires a force/torque sensor.

With the arm-side measurement ILC algorithm, the measurement accuracy of the optical tracking system provides a lower limit on the accuracy that can be achieved in the milling. Thus, since a mean average error of 58 μm was reached, the algorithm can be considered as successful. In addition, to some extent, individual variations in the workpieces used in the experiment and the effect of the wear of the tool also introduced non-repetitive errors and disturbances that the ILC algorithm cannot

compensate. The main limitation of the force-based approach is its inability of reducing constant offsets also present in the free-space motion. However, other methods might be used to correct these kinematic calibration errors, see for instance (Chen et al., 2008). Another shortcoming is the fact that compensation for errors in the model G_K relating the force and the deflection is not accounted for in the ILC scheme, since there is no external measurement of the arm-side position of the robot. Considering that an accurate model is required in order to obtain satisfying results, it is beneficial to complement the system identification approach with a stiffness calibration procedure in order to measure the static and low-frequency properties of the compliance model with high accuracy. In order not to require an external position measurement device, the calibration procedure should keep the arm-side position of the robot tool fixed, and measure the forces emerging when small movements of the joints are performed, see (Lehmann et al., 2013).

Observing Fig. 9 closer, a small number of peaks are observed at certain points along the milling path in the plot for the force measurement ILC approach. This might seem to suggest an instability in the algorithm, as they are apparently not present in the position measurement approach. This is not the case, however, since the reference paths computed by the algorithm do not exhibit this behavior. The most likely explanation for these deviations is the interaction of the aluminium chips emitted from the cutting process with the optical tracking system, as Fig. 10 also exhibits peaks but of lower magnitude. In order to eliminate these stochastic measurement deviations, an outlier detection algorithm should be integrated in the ILC iterations.

A natural question for the future applicability of the method is the generality of the models required for the ILC algorithms. The deflection model G_K obviously depends on the robot in use and the specific configuration, but given that the same limited workspace is used for all milling operations, it can be considered independent of the path traversed in the milling. Hence, a modeling effort should be carried out in order to characterize the robot in the configuration used for the milling operations, but the identified model could be used for multiple machining tasks in a limited workspace. In turn, the process-force model G_M does depend on the milling conditions, so different models are required for different tasks. In this paper, a data-driven modeling approach has been used; however, different models available in the machining literature relating the process forces with the milling conditions could be used as well, see, *e.g.*, (Grote and Antonsson, 2009).

5. CONCLUSIONS

Two model-based ILC-based algorithms have been presented with the aim of reducing the position errors in machining tasks with robots. The first method was based on the measurement of the position of the arm-side of the robot in task space, and the second method used process force measurements. The experimental results obtained from milling experiments in aluminium showed a clear superiority of the algorithm based on arm-side position measurements. An error decrease of approximately 85% was obtained with this algorithm. In turn, the force-based approach was not capable of eliminating constant errors, such as calibration deficiencies, in the position of the robot. Despite this fact, the algorithm was capable of significantly attenuating the dynamic errors caused by the process forces in the milling—approximately 50% decrease of the dynamic position error was achieved—at a lower investment cost in sensors.

REFERENCES

ABB Robotics (2012). ABB IRB2400 Industrial Robot. Data sheet No. PR10034 EN_R7.

Arimoto, S., Kawamura, S., and Miyazaki, F. (1984). Bettering operation of robots by learning. *J. Robotic Systems*, 1(2), 123–140.

Blomdell, A., Dressler, I., Nilsson, K., and Robertsson, A. (2010). Flexible application development and high-performance motion control based on external sensing and reconfiguration of ABB industrial robot controllers. In *Proc. Workshop of "Innovative Robot Control Architectures for Demanding (Research) Applications—How to Modify and Enhance Commercial Controllers", IEEE Int. Conf. Robotics and Automation (ICRA)*, 62–66. Anchorage, AK.

Chen, H., Fuhlbrigge, T., Choi, S., Wang, J., and Li, X. (2008). Practical indstrial robot zero offset calibration. In *Proc, IEEE Int. Conf. Automation Science and Engineering (CASE)*, 516–521. Washington, D.C.

Grote, K. and Antonsson, E. (2009). *Springer Handbook of Mechanical Engineering: Manufacturing Engineering—Machining Processes.* No. 10. Springer Verlag, Berlin Heidelberg, Germany.

Hakvoort, W.B.J., Aarts, R.G.K.M., Van Dijk, J., and Jonker, J.B. (2007). Model-based iterative learning control applied to an industrial robot with elasticity. In *Proc. IEEE Conf. Decision and Control (CDC)*, 4185–4190. New Orleans, LA.

Hogan, N. (1985). Impedance control: An approach to manipulation: Parts I-III. *ASME J. Dynamic Systems, Measurement, and Control*, 107, 1–24.

Johansson, R. (1993). *System Modeling and Identification.* Prentice Hall, Englewood Cliffs, NJ.

Lehmann, C., Olofsson, B., Nilsson, K., Halbauer, M., Haage, M., Robertsson, A., Sörnmo, O., and Berger, U. (2013). Robot joint modeling and parameter identification using the clamping method. In *Proc. IFAC Conf. Manufacturing Modelling, Management, and Control (MIM)*, 843–848. St. Petersburg, Russia.

Ljung, L. (2010). *System Identification Toolbox 7: Users's Guide.* The MathWorks, Inc., Natick, MA.

Miyazaki, F., Kawamura, S., Matsumori, M., and Arimoto, S. (1986). Learning control scheme for a class of robot systems with elasticity. In *Proc. IEEE Conf. Decision and Control (CDC)*, volume 25, 74–79. Athens, Greece.

Nikon Metrology (2010). K-series optical CMM solutions. Data sheet Optical_CMM_EN_0311.

Norrlöf, M. (2002). An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Trans. Robot. Autom.*, 18(2), 245–251.

Norrlöf, M. (2000). *Iterative Learning Control: Analysis, Design, and Experiments.* Ph.D. thesis, Linköping University, Sweden. Thesis No. 653.

Reinl, C., Friedmann, M., Bauer, J., Pischan, M., Abele, E., and von Stryk, O. (2011). Model-based off-line compensation of path deviation for industrial robots in milling applications. In *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM)*, 367–372. Budapest, Hungary.

Sörnmo, O., Olofsson, B., Robertsson, A., and Johansson, R. (2012a). Increasing time-efficiency and accuracy of robotic machining processes using model-based adaptive force control. In *Proc. 10th Int. IFAC Symp. Robot Control (SYROCO)*, 543–548. Dubrovnik, Croatia.

Sörnmo, O., Olofsson, B., Schneider, U., Robertsson, A., and Johansson, R. (2012b). Increasing the milling accuracy for industrial robots using a piezo-actuated high-dynamic micro manipulator. In *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM)*, 104–110. Kaohsiung, Taiwan.

Ugural, A.C. and Fenster, S.K. (2003). *Advanced strength and applied elasticity.* Prentice Hall, Upper Saddle River, NJ.

van Overschee, P. and De Moor, B. (1994). N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1), 75–93.

Wang, J., Zhang, H., and Fuhlbrigge, T. (2009). Improving machining accuracy with robot deformation compensation. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 3826–3831. St. Louis, MO.

Xu, J.X., Panda, S.K., and Lee, T.H. (2010). *Real-time Iterative Learning Control: Design and Applications.* Springer Verlag, London.