

Estimation-based ILC applied to a parallel kinematic robot

Johanna Wallen Axehill, Isolde Dressler, Svante Gunnarsson, Anders Robertsson and Mikael Norrlöf

Linköping University Post Print



N.B.: When citing this work, cite the original article.

Original Publication:

Johanna Wallen Axehill, Isolde Dressler, Svante Gunnarsson, Anders Robertsson and Mikael Norrlöf, Estimation-based ILC applied to a parallel kinematic robot, 2014, Control Engineering Practice, (33).

<http://dx.doi.org/10.1016/j.conengprac.2014.08.008>

Copyright: Elsevier

<http://www.elsevier.com/>

Postprint available at: Linköping University Electronic Press

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-113180>

Estimation-based ILC applied to a parallel kinematic robot

Johanna Wallén Axehill^a, Isolde Dressler^c, Svante Gunnarsson^b, Anders Robertsson^c, Mikael Norrlöf^{ch,d,*}

^aSaab Aeronautics, SE-581 88 Linköping, Sweden,
e-mail: johanna.axehill@saabgroup.com

^bDiv. of Automatic Control, Dept. of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden,
e-mail: {svante,mino}@isy.liu.se

^cDept. of Automatic Control, Lund University, Box 118, SE-221 00 Lund, Sweden,
e-mail: {isolde.dressler, anders.robertsson}@control.lth.se

^dABB Robotics, SE-721 68 Västerås, Sweden

Abstract

Estimation-based iterative learning control (ILC) is applied to a parallel kinematic manipulator known as the Gantry-Tau parallel robot. The system represents a control problem where measurements of the controlled variables are not available. The main idea is to use estimates of the controlled variables in the ILC algorithm, and in the paper this approach is evaluated experimentally on the Gantry-Tau robot. The experimental results show that an ILC algorithm using estimates of the tool position gives a considerable improvement of the control performance. The tool position estimate is obtained by fusing measurements of the actuator angular positions with measurements of the tool path acceleration using a complementary filter.

Keywords: Iterative methods, Learning control, Robotic manipulator, Estimation algorithm, Performance evaluation.

1. Introduction

Robots with parallel kinematic structure have potential of high performance due to high stiffness and high achievable accelerations, however often with the disadvantage of a small workspace [20]. The Gantry-Tau configuration [19], shown in Figure 1 and Figure 5, has a large workspace compared to other parallel robots, while being stiff compared to serial robots. Other advantages are possibilities of machining large components with high accuracy [9], and a modular construction for manufacturing of small lot sizes [11].

To achieve high performance, one way is to rely on high-precision components, detailed models and highly accurate and often expensive measurement devices. Another option could be to use iterative learning control (ILC), see e.g. [2], which is applicable when the system performs the same motion repeatedly.

There are comparatively few contributions with ILC algorithms applied to parallel robots. One example is [1], where ILC algorithms using joint positions are applied to a direct-driven hexapod. In [8], an ILC algo-

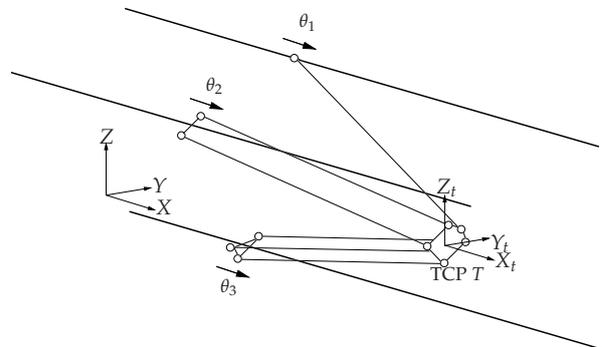


Figure 1: The Gantry-Tau configuration, with cart position vector $\theta = (\theta_1 \ \theta_2 \ \theta_3)^T$ and tool position XYZ .

rithm using joint positions is applied to a planar direct-driven parallel robot, where the ILC algorithm is intended to reduce the low-frequency error components, and the resulting tool position is evaluated by a high-precision laser measurement system. ILC algorithms directly updating motor torques are applied to parallel robots in [6], and in [5] an ILC algorithm using tool-position measurements is applied to a nano-scale robot. To summarise, the applications of ILC mentioned above

*Corresponding author. E-mail address: mino@isy.liu.se, Phone: +46 13 282 704, Fax: +46 13 139 282

are performed on smaller parallel robots with different designs and in many cases stiffer structure compared to the Gantry-Tau robot.

In commercial robot systems typically the motor angular positions are measured, while the objective is to make the tool follow a desired path. Measurements of the actual tool pose is normally not available. An alternative to direct tool measurements is to use additional sensors in combination with some sensor fusion or estimation algorithm to obtain a tool-position estimate. An early approach to estimation-based ILC is presented in [14], which presents a simulation study where measurement of the motor angle is combined with the arm acceleration of a single link robot. The corresponding approach is implemented and evaluated experimentally in [15]. For realistic robot structures the state estimation problems becomes a challenging nonlinear filtering problem. In [3] results are presented based on real data where measurements from motor angles and tool acceleration are fused using Extended Kalman Filter and Particle Filters. The combination of nonlinear state estimation and ILC is evaluated with promising results in [24] using a realistic two degrees-of-freedom simulation model containing mechanical elasticities.

In this paper estimation-based ILC is implemented and evaluated experimentally using the Gantry-Tau parallel kinematics robot. The presentation is a development of the initial work presented in [23] where observer based ILC was first applied to the Gantry-Tau parallel kinematic robot. The paper is organised as follows. Section 2 gives a brief theoretical background to estimation-based ILC, and Section 3 describes the Gantry-Tau robot system, the sensors that will be used in the experiments, and the nominal performance of the robot. In Section 4 the robot tool-position estimation is discussed, and Section 5 specifies the setup for the experiments. Section 6 contains a thorough discussion about the experimental results, and finally, Section 7 concludes the paper.

2. Estimation-based ILC

A key property of the problem studied in the paper is that the controlled variable cannot be directly measured. The situation is schematically described in Figure 2, from [25], where $r(t)$ and $u_k(t)$ denote the reference signal and the ILC input respectively, index k refers to the iteration number. The system T can represent an open loop system as well as a closed loop system with internal feedback. Further details are given in [25]. There are two types of output signals from the system,

$z_k(t)$ denotes the controlled variable and $y_k(t)$ the measured variable. In the robot system in Section 3, $z_k(t)$ is the position of the robot tool in two dimensions. In general the tool position is not directly measurable, therefore the ILC algorithm has to rely on estimates of the controlled variable based on measurements of related quantities. The measured variables are collected in the variable $y_k(t)$.

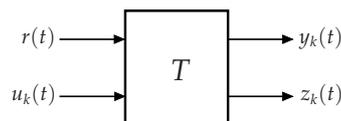


Figure 2: System T with reference $r(t)$, ILC input $u_k(t)$, measured variable $y_k(t)$ and controlled variable $z_k(t)$ at iteration k .

It is assumed that the relationships between the signals in Figure 2 can be described by the equations

$$y_k(t) = T_{ry}(q)r(t) + T_{uy}(q)u_k(t) \quad (1a)$$

$$z_k(t) = T_{rz}(q)r(t) + T_{uz}(q)u_k(t) \quad (1b)$$

where $T_{ry}(q)$, $T_{uy}(q)$, $T_{rz}(q)$, and $T_{uz}(q)$ are stable discrete-time transfer operators and q denotes the shift operator. Using the framework presented in [25] an estimate $\hat{z}(t)$ of the controlled variable can be computed. The estimation approach is illustrated in Figure 3.

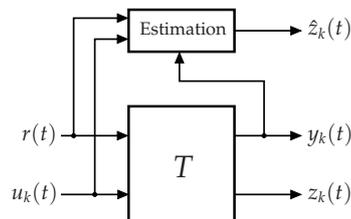


Figure 3: Description of the system T with reference $r(t)$, ILC input $u_k(t)$, measured variable $y_k(t)$ and controlled variable $z_k(t)$ at iteration k . The estimation procedure results in an estimate $\hat{z}_k(t)$ of the controlled variable.

The estimate $\hat{z}_k(t)$ can be generated as

$$\hat{z}_k(t) = F_r(q)r(t) + F_u(q)u_k(t) + F_y(q)y_k(t) \quad (2)$$

with filters $F_r(q)$, $F_u(q)$, and $F_y(q)$. Different ways to choose the filters are discussed in [25] and in Section 4 below.

The ILC algorithms that will be used in the paper are given by

$$u_{k+1}(t) = Q(q)(u_k(t) + L(q)\epsilon_k(t)) \quad (3)$$

where the filters Q and L are possibly non-causal. The error used in the ILC algorithm is the difference between the reference r and the estimate \hat{z}_k in (2) of the controlled variable at iteration k , as in

$$\epsilon_k(t) = r(t) - \hat{z}_k(t) \quad (4)$$

Assume that the system defined by Equation (1) is controlled using the ILC algorithm given by (3), where the estimate $\hat{z}(t)$ is generated according to Equation (2), then the ILC input signal is updated according to

$$u_{k+1}(t) = H(q)u_k(t) + H_r(q)r(t) \quad (5)$$

where

$$H(q) = Q(q)(1 - L(q)(F_u(q) + F_y(q)T_{uy}(q))) \quad (6)$$

$$H_r(q) = Q(q)L(q)(1 - L(q)(F_u(q) + F_y(q)T_{ry}(q))) \quad (7)$$

The stability criterion for the estimation-based ILC algorithm can hence, see [25], be expressed in the frequency domain as

$$|Q(e^{i\omega})(1 - L(e^{i\omega})(F_u(e^{i\omega}) + F_y(e^{i\omega})T_{uy}(e^{i\omega})))| < 1 \quad (8)$$

for all ω .

3. The Robot System

The robot system used in the experiments is shown in Figure 4 and Figure 5. The most relevant variables in the system are presented in the block diagram shown in Figure 4 and the descriptions of the respective variables are presented in Table 1.

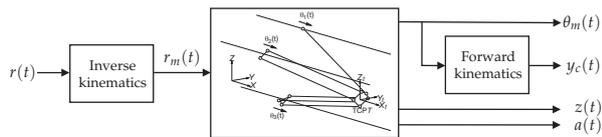


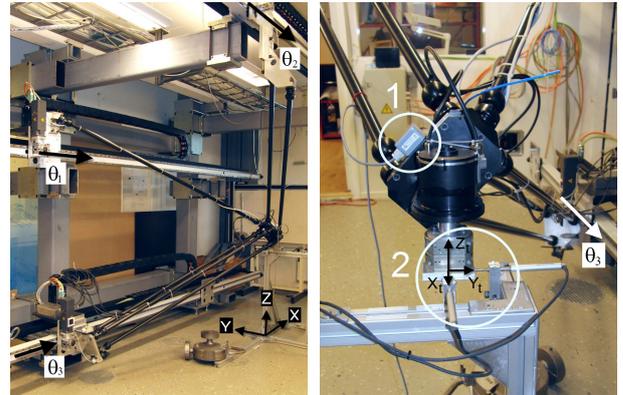
Figure 4: Robot system with relevant control variables.

3.1. The Gantry-Tau robot

The Gantry-Tau robot prototype has three kinematic chains, see Figure 1 and Figure 5. The actuators are of prismatic type, realised by three carts moving along guide-ways, where each cart is attached to an electrical motor via a transmission. The positions of the three carts are represented by the vector $\theta(t)$. The carts are connected to the end-effector plate via link clusters with lengths 1.8 – 2 m. The links belonging to one cluster

Table 1: Definitions of the variables in the system description.

Variable	Description
$r_m(t) \in R^3$	Motor angular position reference
$\theta_m(t) \in R^3$	Measured motor angular position
$\theta(t) \in R^3$	Cart position
$r(t) \in R^2$	Tool position reference
$y_c(t) \in R^2$	Calculated tool position
$z(t) \in R^2$	Measured tool position
$a(t) \in R^2$	Measured tool acceleration



(a) Robot

(b) External sensors

Figure 5: Experimental setup; a) robot structure, b) mounting of the accelerometer (circle 1) and the length gauges (circle 2).

form parallelograms, resulting in constant end-effector orientation, and three translational DOFs. The Gantry-Tau configuration is extended with a 2-DOF serial wrist, see Figure 5b. The additional degrees of freedom are not used in the experiments. The kinematic model of the robot is calibrated and the resulting mean static end-effector positioning error, over the whole workspace, is identified to be $140 \mu\text{m}$.

3.2. Control system

The robot is controlled by an extended ABB IRC5 system, where signals can be read and modified externally as described in [4]. The IRC5 control system operates at 250 Hz sampling rate, and data from external sensors are synchronised with the internal robot system measurements. The movements of the carts are controlled by three separate PID-type controllers. The angular positions of the three motors are represented by the vector $\theta_m(t)$. The motor angular references of the three motors, represented by the vector $r_m(t)$, are generated from the tool position references $r(t)$, using the

kinematic model in [12, 10]. The motors are connected to the carts via transmissions with gear ratio η .

3.3. Sensors and measurements

The variable $y_c(t)$ is computed using the measured angular positions of the motors $\theta_m(t)$, scaled by the gear ratio, and the kinematic model. The accuracy of $y_c(t)$ will depend on the accuracy of the kinematic model and on the actual dynamics from $\theta_m(t)$ to $\theta(t)$, which can include backlash and mechanical flexibilities. The main idea of the paper is to base the ILC algorithm on a tool-position estimate, computed by fusing $y_c(t)$, with information from additional sensors. The additional sensors considered here are accelerometers, but a camera, or a combination of accelerometers and camera, could also be used. For evaluation purposes the experimental setup is equipped with length gauges that measure the tool position. This type of sensor has a relatively small operating range, and a laser scanner would be an alternative, although considerably more expensive.

3-DOF accelerometer. The tool acceleration is measured using an accelerometer denoted MMA7361L from Freescale [13], mounted on the end-effector plate, see Figure 5b. The accelerometer orientation is constant due to constant end-effector orientation. The calculated standard deviation of the measurement noise is about 0.07 m/s^2 , and the output from the accelerometer is represented by the vector $a(t)$.

Length gauges. Two length gauges (position sensors) ST 3078 from Heidenhain [16] mounted on a stand are used for measuring the tool position in XY -direction (horizontal), see Figure 5b. The gauges have a range of 30 mm and accuracy of $\pm 1 \mu\text{m}$, and the output from the sensors is represented by the vector $z(t)$.

3.4. Trajectory

The experiments in the paper are carried out using a test path depicted in Figure 6. The speed 100 mm/s is the actual maximum speed, considering the length of the movement and the acceleration limitations of the motors.

3.5. Nominal performance and repeatability

Figure 6 shows the tool position reference path $r(t)$ and the measured tool position in the XY -direction $z(t)$. Repeatability is a key property when using ILC, since only the repetitive part of the error can be corrected by

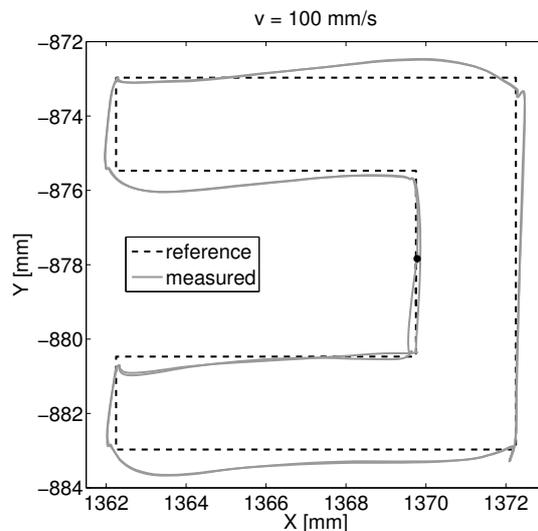


Figure 6: Nominal robot performance, with measured tool behaviour compared to the reference path. The motion starts at the black dot and the motion direction is counter clockwise.

an ILC algorithm. In order to explore the repeatability of the system, five identical experiments were performed on the robot, and the repeatability is evaluated by forming

$$e_{\text{rep},i}^j(t) = \frac{\theta_{m,i}^j(t) - \bar{\theta}_{m,i}(t)}{\eta} \quad (9)$$

where $\theta_{m,i}^j(t)$ denotes the measured angular position of motor $i = 1, \dots, 3$ and experiment $j = 1, \dots, 5$ and $\bar{\theta}_{m,i}(t)$ represents the mean value of the error for the five experiments at time t . To get the cart error in mm the motor angular position is divided by the gear ratio, η . Figure 7 shows the histogram of $\{e_{\text{rep},3}^j\}_{j=1,\dots,5}$ for cart 3 and it is representative for all carts. The majority of samples are within 0.01 mm and the maximum value (0.04 mm) is still small compared to the maximum cart-position error, which is of the order of one millimetre as can be seen in Figure 6.

4. Estimation of tool position

A key component in the estimation-based ILC algorithm studied in the paper is the method for generating the estimate $\hat{z}(t)$ of the controlled variable $z(t)$. The achievable performance is determined by the estimation accuracy of the controlled variable. A general discussion of estimation-based ILC is given in [25], where different ways to generate the estimate is discussed in

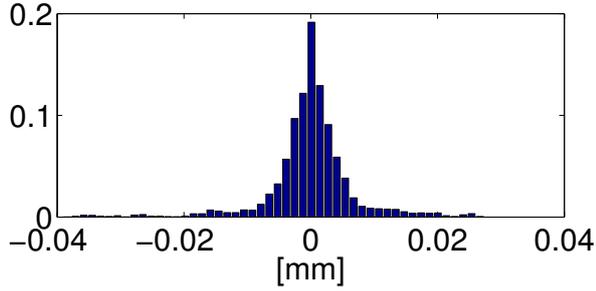


Figure 7: The histogram for the variation of the error $e_{\text{rep},3}^j$ where $j = 1, \dots, 5$ are included and the y-axis is normalised.

terms of performance and other aspects. One obvious alternative is to use a Kalman filter, but this requires accurate models in order to generate sufficiently accurate estimates. In addition, the computational complexity increases rapidly for a more complex robot model. A contribution in the area is [3] where the authors compare the extended Kalman filter and the particle filter when fusing measurements of the tool acceleration with measurements of the motor angular position.

Since the purpose of this paper is to show the feasibility of applying estimation-based ILC to a parallel kinematic robot, complementary filtering, a simpler approach, is used for estimation of the robot tool position. Complementary filtering is an estimation technique for sensor fusion used in, for instance flight control and navigation system design, popular because of its simplicity [17], and it fuses noisy measurements of the same physical variable with different frequency characteristics. Filters are called complementary filters if the sum of the transfer functions is one for all frequencies.

For the robot system considered here, experimental results show that the calculated tool position $y_c(t)$ is a reasonable tool-position estimate for low frequencies. The signal $y_a(t)$, obtained by double integration of the measured tool acceleration $a(t)$, is a reasonably accurate tool-position estimate for higher frequencies. These two signals are fused together by a complementary filter, giving the tool-position estimate

$$\hat{z}(t) = G(q)y_c(t) + (1 - G(q))y_a(t) \quad (10)$$

with a second-order low-pass Butterworth filter $G(q)$, applied using MATLAB's `filtfilt` function. Due to the measurement setup, seen in Figure 5, estimation is only considering the XY -direction since this is the only directions where measurements of the position are available. The cutoff frequency of $G(q)$ is tuned to give the smallest maximum estimation error $e(t) = z(t) - \hat{z}(t)$ for

the path in Figure 6.

From Figure 8 it can be seen that the estimation error $e(t)$ decreases significantly in the Y -direction when using the complementary filter estimate (10) compared to using $\hat{z}(t) = y_c(t)$. This can be explained by the fact that the robot is flexible in this direction and therefore the accelerometer mounted on the end-effector plate provides more additional information than for the stiffer X -direction.

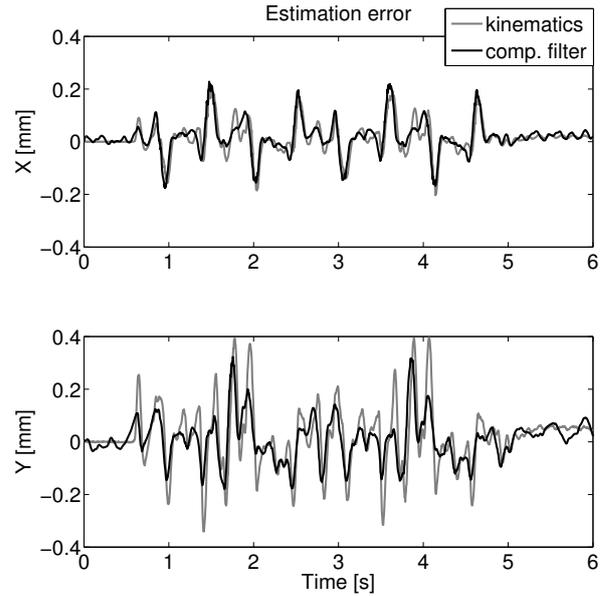


Figure 8: Estimation error $e(t) = z(t) - \hat{z}(t)$ for the test path. Result for calculated tool position $y_c(t)$ from motor angular positions transformed by the forward kinematics (kinematics), compared to results from complementary filter (comp. filter).

In [22] corresponding experiments using a linear Kalman filter are presented. The accuracy is similar to what is obtained using complementary filtering, but it should be kept in mind that the quality of the underlying linear model probably can be improved.

5. Experimental setup

The experiments are carried out using the path shown in Figure 6, and the programmed tool velocity is $v = 100$ mm/s. Experiments with other velocities are presented in [22].

Three types of experiments will be considered, and they differ in the way the error signal driving the update of the ILC input is defined. Recall Equations (3) and (4), i.e.

$$u_{k+1}(t) = Q(q)(u_k(t) + L(q)\epsilon_k(t)) \quad (11)$$

and

$$\epsilon_k(t) = r(t) - \hat{z}_k(t) \quad (12)$$

The experiments are defined as follows:

Case 1: The ILC update is driven by the motor angular position error, i.e.

$$\epsilon_k(t) = r_m(t) - \theta_{m,k}(t)$$

Case 2: The ILC update is driven by the tool position error obtained by using an estimate of the tool position, i.e.

$$\epsilon_k(t) = r(t) - \hat{z}_k(t) \quad (13)$$

where $\hat{z}_k(t)$ is generated using the complementary filter in equation (10)

$$\hat{z}(t) = G(q)y_c(t) + (1 - G(q))y_a(t) \quad (14)$$

Case 3: The ILC update is driven by the the tool position error obtained by using the measured tool position, i.e.

$$\epsilon_k(t) = r(t) - z_k(t)$$

The evaluation is based on the nominal error without ILC, i.e. $k = 0$. The vector of nominal errors for the three motors is given by

$$\begin{pmatrix} e_{m,0}^1 \\ e_{m,0}^2 \\ e_{m,0}^3 \end{pmatrix} = \begin{pmatrix} r_m^1 \\ r_m^2 \\ r_m^3 \end{pmatrix} - \begin{pmatrix} \theta_{m,0}^1 \\ \theta_{m,0}^2 \\ \theta_{m,0}^3 \end{pmatrix} \quad (15)$$

with $e_{m,0}^i$ denoting the motor angular position error, $\theta_{m,0}^i$ being the measured motor angular position for cart i at iteration 0 and r_m^i being the corresponding reference. The cart position error is derived by dividing $e_{m,0}^i$ by the gear ratio η . Analogously to (15), the vector of nominal tool-position errors in the X - and Y -direction is

$$\begin{pmatrix} e_{z,0}^X \\ e_{z,0}^Y \end{pmatrix} = \begin{pmatrix} r^X \\ r^Y \end{pmatrix} - \begin{pmatrix} z_0^X \\ z_0^Y \end{pmatrix} \quad (16)$$

The reduction of the 2-norm of the motor angular position error at iteration k for each of the three carts is given in percentage of the nominal error (15),

$$\bar{e}_{m,k}^1 = 100 \cdot \frac{\|e_{m,k}^1\|_2}{\|e_{m,0}^1\|_2} \quad [\%] \quad (17a)$$

$$\bar{e}_{m,k}^2 = 100 \cdot \frac{\|e_{m,k}^2\|_2}{\|e_{m,0}^2\|_2} \quad [\%] \quad (17b)$$

$$\bar{e}_{m,k}^3 = 100 \cdot \frac{\|e_{m,k}^3\|_2}{\|e_{m,0}^3\|_2} \quad [\%] \quad (17c)$$

The reduction of the tool error is similarly given by

$$\bar{e}_{z,k}^X = 100 \cdot \frac{\|e_{z,k}^X\|_2}{\|e_{z,0}^X\|_2} \quad [\%] \quad (18a)$$

$$\bar{e}_{z,k}^Y = 100 \cdot \frac{\|e_{z,k}^Y\|_2}{\|e_{z,0}^Y\|_2} \quad [\%] \quad (18b)$$

6. Experimental results

6.1. Case 1 — ILC using $\theta_m(t)$

The purpose of this section is to show the performance of the system when applying an ILC algorithm driven by the motor angular position error. For simplicity it is assumed that the couplings between the three motors are relatively small, and hence the system will be treated as three separate SISO systems, i.e.

$$\theta_{m,i}(t) = G_{m,i}(q)r_{m,i}(t), \quad i = 1, 2, 3 \quad (19)$$

where $G_{m,i}(q)$ denotes the transfer operator of motor i and q denotes the shift operator. It is a fairly straightforward problem to identify these systems, and details concerning this step, including model validation, are given in [22]. The system identification gives as a result that all three motors can approximately be described by

$$G_{m,i}(q) = q^{-5} \frac{0.03}{1 - 0.97q^{-1}} \quad (20)$$

with the sampling interval $T_s = 4$ ms. The time delay is caused by internal data communication in the robot control system.

Since the aim of the paper is to concentrate on the qualitative behaviour of the system a simple ILC algorithm is used, and therefore the filter $L(q)$ is chosen as

$$L(q) = \gamma q^\delta \quad (21)$$

with $\delta = 5$ and $\gamma = 0.9$. The filter $Q(q)$ is chosen as a second-order low-pass Butterworth filter with cutoff frequency $f_c = 10$ Hz, and forward-backward-filtering is used in order to give a zero-phase filtering. The same ILC design variables are applied to all motors, since the models (20) of the motor dynamics for the carts are similar to each other. The ILC input signal is added to the motor angular position reference, hence $T_{wy}(q)$ in (1) is equal to $G_{m,i}(q)$ from (20) for the three motors $i = 1, 2, 3$. Using the motor angular positions in the ILC algorithm results in $F_r(q) = F_u(q) = 0$, $F_y(q) = 1$ in (2). Applying the criterion (8) it can be verified that

$$|Q(e^{i\omega})(1 - L(e^{i\omega})T_{wy}(e^{i\omega}))| < 0.95 \quad \forall \omega \quad (22)$$

for all three motors. Further details, also including time domain stability analysis, are given in [22].

Figure 9 shows that the error measure (17) is reduced to nearly 2% for the motors after around five iterations. In the experiments the motor angular position references

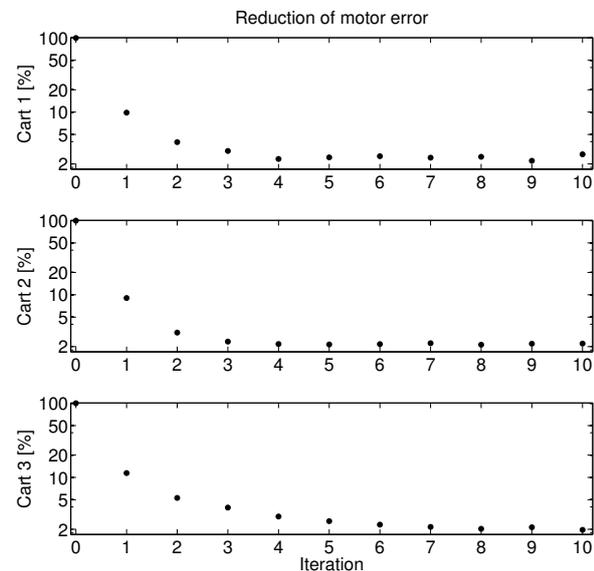


Figure 9: Case 1. Error measure (17) for the three motors as a function of iteration.

$r_m(t)$ are calculated from the tool position reference $r(t)$ using the model of the inverse kinematics and the gear ratio η . Similarly the calculated tool position $y_c(t)$ is determined using the gear ratio, the forward kinematics model, and the measured motor angular positions $\theta_m(t)$. Figure 10 shows the performance of the robot in terms of these quantities, and based on the illustrations in the figure two main observations can be made:

- The calculated tool position $y_c(t)$ follows the tool position reference $r(t)$ very well. This is in agreement with the fact that the motor angular positions $\theta_m(t)$ follow the references $r_m(t)$ more closely thanks to the use of ILC.
- The real tool path error between $r(t)$ and $z(t)$ has been reduced compared the the case without ILC, shown in Figure 6, but there is still a considerable path error.

To examine the repeatability properties, the ILC experiment is repeated five times under as identical conditions as possible. The error measure (17) for each motor and iteration is shown in Figure 11. It can be seen that the behaviour with slightly non-monotone convergence

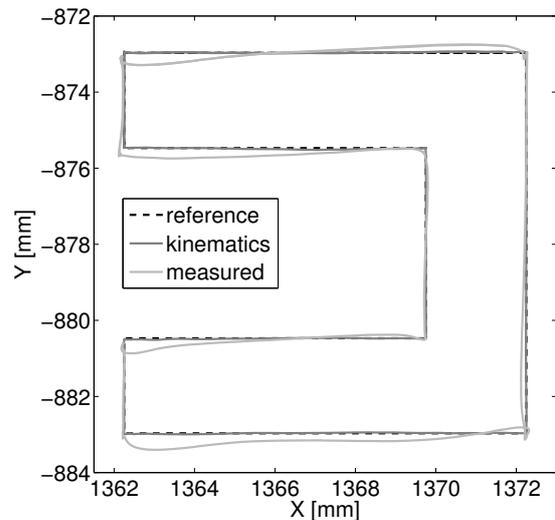


Figure 10: Case 1. Tool performance after 10 iterations. Dashed - tool position reference $r(t)$. Solid dark grey: Tool path computed using kinematic model $y_c(t)$. Solid light grey: Measured tool path $z(t)$.

in Figure 9 may be explained by the variation of the error measure (17) for each iteration. The variation in the resulting error may be even larger due to varying experimental conditions, for example temperature.

6.2. Case 2 — ILC using $\hat{z}(t)$

The overall idea that will be applied here is straightforward, i.e. to apply an ILC algorithm driven by the estimated tool path error given by Equation (13). There are however some issues that need to be considered. First the complementary filter approach for estimating the tool position, Equation (10), needs to be put in the general framework given by (2). Introducing the vector

$$y(t) = \begin{pmatrix} y_c(t) \\ y_a(t) \end{pmatrix} \quad (23)$$

and

$$F_y(q) = \begin{pmatrix} G(q) & 1 - G(q) \end{pmatrix} \quad (24)$$

the estimate can be formulated as

$$\hat{z}(t) = F_y(q)y(t) \quad (25)$$

Second, the ILC inputs will in this case be added to the tool reference signal, and the overall system will then have two inputs and two outputs, i.e. the reference and achieved tool positions in the X- and Y-directions respectively. For simplicity the behaviour in the two directions will be treated as two separate decoupled systems. Given that the estimate of the controlled variable

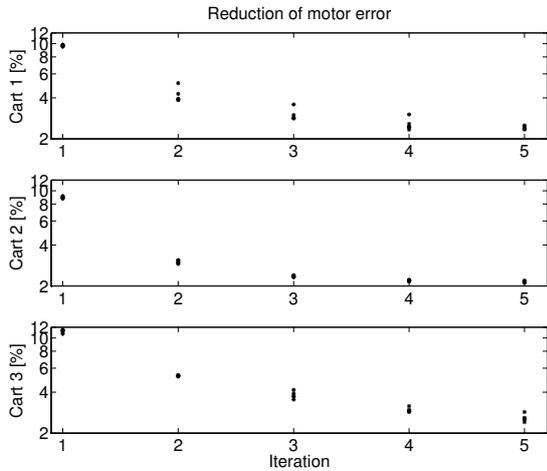


Figure 11: Case 1. Error measure (17) for five experiments to illustrate the repeatability of the error reduction.

is defined by (25) Equation (7) gives

$$H(q) = Q(q)(1 - L(q)F_y(q)T_{wy}(q)) \quad (26)$$

where $T_{wy}(q)$ is the 2×1 transfer operator from the ILC input to the two components in $y(t)$, given by (23).

In order to be able to carry out analysis and design it is necessary to have models of the relationships from ILC input to the components of $y(t)$. Since the ILC signal is added to the reference signal such models can be obtained using system identification. In general this is a challenging task, and some initial results are presented in [22].

Previous work on modelling of the Gantry-Tau robot includes rigid-body dynamics [12, 10] and black-box identification of flexible dynamics [7]. Identification of a slightly different robot prototype of approximately the same size is done in [21] and [18]. The experiments presented in this paper show a notable flexible behaviour, which implies that a rigid-body model is not sufficient. Most of the flexibilities probably originate from the carbon fibre links and the framework, as the joints are very stiff and a comparison of measured motor angular positions and cart positions show considerably less flexibility than the tool motion.

In the identification experiments here the system is excited by applying a PRBS signal to the tool position reference and measuring the tool position $z(t)$ and motor angular positions, from which $y_c(t)$ can be calculated using the kinematic model. Linear black-box models are then estimated using the PRBS signal as input and $z(t)$ and $y_c(t)$ as outputs. More information concerning the identification experiments are given in [22], and the dis-

ussion here will concentrate on the use of the models in the ILC algorithm design.

As a complement to the system identification, pulse-response experiments have been performed in order to determine the resonance frequencies of the robot structure. A force pulse is generated by the impact of a hammer on the end-effector plate in the X - and Y -direction, respectively. The resulting tool position is measured, and the tool vibration can be determined by inspection in the time domain. When the force is applied in the X -direction, a resonance of 7.4 Hz is observed in the X -direction, see Table 2. The tool vibration is 11.4–11.9 Hz in the Y -direction, which after a few oscillations turns into a 7.4 Hz vibration. With the force applied in the Y -direction, the resonance frequency in the X -direction is 10.5 Hz and in the Y -direction 11.5 Hz. The resonance in the Y -direction is larger in magnitude than for the X -direction and dominant for the robot behaviour.

Table 2: Results from pulse-response experiments. The columns represent the impact directions and the rows correspond to the observed vibrations in the X - and Y -directions.

Vibration frequencies	Impact direction	
	X	Y
in X -direction [Hz]	7.4	10.5
in Y -direction [Hz]	7.4, 11.4 – 11.9	11.5

From the identification experiments above it is known that the robot has a dominating resonance frequency at around 11.5 Hz in the Y -direction and not so pronounced resonance frequencies at 7.4 Hz and 10.5 Hz in the X -direction. For ILC to compensate error components around and above these system resonances, it is not sufficient to simply choose a low-pass filter $Q(q)$ with cutoff frequency below the lowest resonance frequency. Instead, the filter design has to be based on a model from the input point of the ILC input signal, here the reference $r(t)$, to the estimate $\hat{z}_k(t)$.

The tuning of $Q(q)$ aims at being robust to large model errors especially around the resonance frequencies, and it is designed from the magnitude of $H(q) = 1 - L(q)F_y(q)T_{wy}(q)$, see Figure 12. Due to measurement noise, learning of the error components up to 30 Hz is chosen, which is above the dominating resonance frequencies of the system. In Figure 12 the robustness properties of the algorithm around the resonance frequencies of the system can be seen, together with the attenuation of frequencies over 30 Hz. The figure also shows that the criterion (8) is fulfilled.

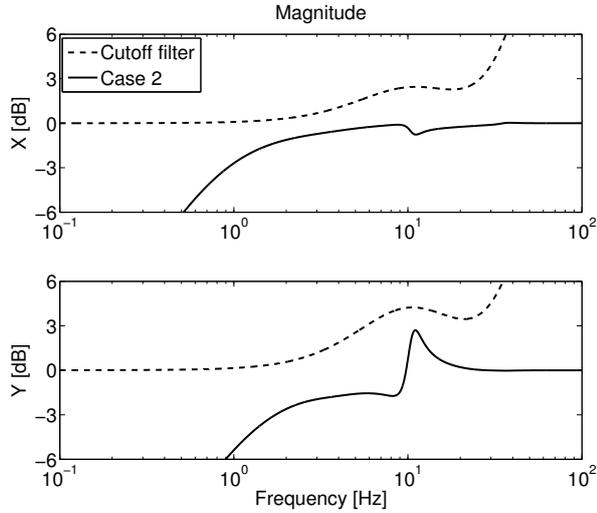


Figure 12: Case 2: The magnitude $|Q^{-1}(e^{i\omega T_s})|$ illustrated together with $|1 - L(e^{i\omega T_s})(F_u(e^{i\omega T_s}) + F_y(e^{i\omega T_s})T_{uy}(e^{i\omega T_s}))|$ in the X- and Y-direction. Learning is cut off at about 30 Hz,

The behaviour of the ILC algorithm applied to the system is then evaluated experimentally, and the resulting tool path, after ten iterations, is shown in Figure 13.

By comparing the resulting tool path shown in Figure 13 to the result for Case 1 shown in Figure 10, it is seen that the tool performance after 10 iterations is improved compared to Case 1, especially in the Y-direction. Most error components in the stiffer X-direction can be compensated using an ILC algorithm with motor angular position measurements. The accelerometer signal gives additional information about the tool position in the Y-direction, so that the performance can be improved in that direction when using tool-position estimates in the ILC algorithm.

6.3. Case 3 — ILC using $z(t)$

Finally, for illustration purposes the ILC algorithm using the tool position is evaluated. This case has close similarities with Case 2, and the design of $Q(q)$ follows the procedure in Case 2, with an algorithm robust to large model errors and with learning up to 30 Hz to avoid high-frequency measurement noise to be included in the learning. For Case 3, with the ILC input signal added to the reference $r(t)$ and $F_r(q) = F_u(q) = 0$, $F_y(q) = 1$ in (2).

The model-based design of $Q(q)$ is followed by some experimental fine-tuning, and the resulting filter $Q(q)$ is almost identical to the filter for Case 2 in the Y-direction, while the gain around the resonance frequency is slightly larger in the X-direction. The result-

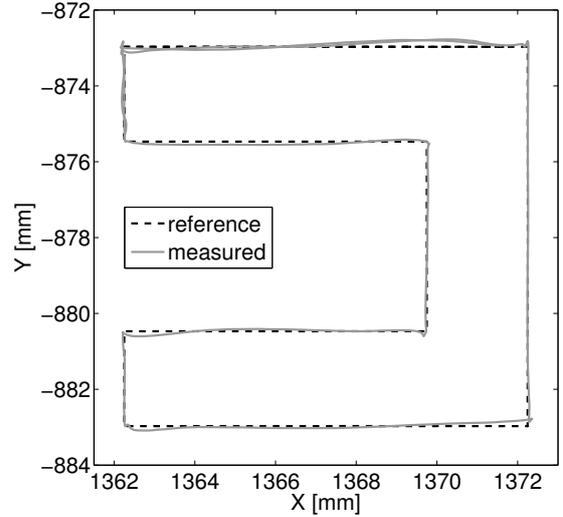


Figure 13: Case 2: Tool performance after 10 iterations; reference path compared to measured tool path.

ing magnitude diagrams are very similar to the diagrams for Case 2 shown in Figure 12, and the corresponding figure is omitted here. The resulting performance is presented in the next section.

6.4. Comparison

Figure 14 shows the resulting performance, in terms of reduction of the tool error measure defined in Equation (18), for the three cases. In addition Table 3 shows the mean value of the error measure (18) over iterations 5 to 10 for the three different cases.

Table 3: Mean value for iterations 5 – 10 of the robot tool error measure (18) for the Cases 1, 2, and 3 in the X- and Y-directions.

	Case 1	Case 2	Case 3
X	13.3 %	12.6 %	10.3 %
Y	36.1 %	23.9 %	24.9 %

Keeping in mind that the results displayed in Figure 13, Figure 14 and in Table 3 have been obtained for one particular reference path some observations can be made. It is clear that Case 2, i.e. estimation based ILC using the estimated tool position, gives better performance than Case 1, i.e. ILC based on the motor angular position error. The biggest improvement is obtained in the Y-direction, which is the direction where the robot shows the highest mechanical flexibility. In

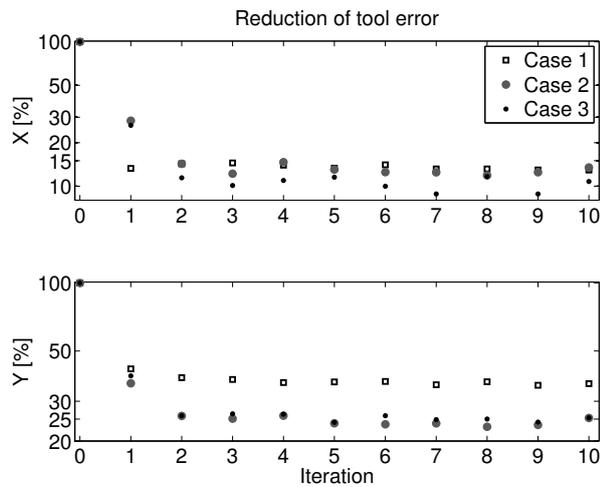


Figure 14: Reduction of the tool error, in terms of the error measure (18), for the Cases 1, 2, and 3 in the X- and Y-direction, respectively.

the Y-direction the performance for Case 2 is slightly better than the performance obtained in Case 3, i.e. ILC based on the measured tool position, but in the X-direction Case 3 gives the best performance. This, somewhat contradictory behaviour, can be explained by the variation seen in Figure 11, the interplay between the quality of the underlying models, and the tuning of the ILC algorithms in the different cases.

7. Conclusions

This paper has presented an approach to estimation based ILC applied to the Gantry-Tau parallel kinematic robot. The motivation for using estimation based ILC is that in practise there are situations when the controlled variable, here the tool position, cannot be measured for practical or economical reasons. The key idea is to generate an estimate of the controlled variable and let the ILC algorithm be driven by the difference between the reference value for the controlled variable and the estimate of the controlled variable. Complementary filtering was chosen for generating the estimate of the tool position because it is a comparatively simple and straightforward method. The estimate is generated by fusing the calculated tool position $y_c(t)$, obtained by combining the measured motor angular positions and the kinematic model, with a high-pass filtered version of $y_a(t)$, the double integrated measured tool acceleration $a(t)$. In addition, comparatively simple ILC algorithms have been used.

Despite the comparatively simple approach the experiments show very clearly that the estimation based ILC algorithm gives considerably improved performance, in terms of tool path error, Figure 13, compared to ILC based on the measured motor angular positions, Figure 10. It should also be noted that already motor based ILC gives improved performance compared to the nominal case without ILC, Figure 6.

There are a number of directions in which the initial results presented in this paper can be developed further. One such direction is improved modelling of the system in general and in particular the dynamics of the arm structure. Access to more accurate models would enable a more systematic approach to the design of the mechanism to generate the estimate of the controlled variable, with the Kalman filter being the most obvious alternative to complementary filtering. More accurate models would also enable a more systematic and model based design of the ILC algorithms. Another interesting research direction is to include other types of affordable sensors like e.g. simple cameras. These topics are all left for future research.

Acknowledgement

This work was supported by ELLIIT, Excellence Center at Linköping – Lund in Information Technology, LCCC Linnaeus Center at Lund University and Vinnova's Industry Excellence Center LINK-SIC at Linköping University, and they are gratefully acknowledged.

References

- [1] Abdellatif, H., Heimann, B., 2010. Advanced model-based control of a 6-DOF hexapod robot: A case study. *IEEE/ASME Trans. Mechatronics* 15 (2), 269–279.
- [2] Ahn, H.-S., Chen, Y., Moore, K. L., Nov. 2007. Iterative learning control: Brief survey and categorization. *IEEE Trans. Systems, Man, Cybernetics – Part C: Appl. Reviews* 37 (6), 1099–1121.
- [3] Axelsson, P., Karlsson, R., Norrlöf, M., 2012. Bayesian State Estimation of a Flexible Industrial Robot. *Control Engineering Practice* 20 (11), 1220–1228.
- [4] Blondell, A., Bolmsjö, G., Brogårdh, T., Cederberg, P., Isaksson, M., Johansson, R., Haage, M., Nilsson, K., Olsson, M., Olsson, T., Robertsson, A., Wang, J., Sep. 2005. Extending an industrial robot controller: Implementation and applications of a fast open sensor interface. *IEEE Robotics Automation Mag.* 12 (3), 85–94.
- [5] Bristow, D. A., Dong, J., Alleyne, A. G., Ferreira, P., Salapaka, S., 2008. High bandwidth control of precision motion instrumentation. *Review Sci. Instr.* 79 (10).
- [6] Burdet, E., Rey, L., Codourey, A., 2001. A trivial and efficient learning method for motion and force control. *Eng. Appl. Artif. Intell.* 14 (4), 487–496.

- [7] Cescon, M., Dressler, I., Johansson, R., Robertsson, A., Jul. 2009. Subspace-based identification of compliance dynamics of parallel kinematic manipulator. In: Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics. Singapore, Singapore, pp. 1028–1033.
- [8] Cheung, J. W. F., Hung, Y. S., 2009. Robust learning control of a high precision planar parallel manipulator. *Mechatronics* 19 (1), 42–55.
- [9] Crothers, P., Freeman, P., Brogårdh, T., Dressler, I., Nilsson, K., Robertsson, A., Zulauf, W., Felder, B., Loser, R., Siercks, K., Mar. 2010. Characterisation of the Tau parallel kinematic machine for aerospace application. *SAE Int. Journal Aerospace* 2 (1), 205–213.
- [10] Dressler, I., 2012. Modeling and control of stiff robots for flexible manufacturing. Dissertation, ISRN LUTFD2/TFRT-1093-SE, available at: <http://www.control.lth.se/Publication/dressler2012phd.html>.
- [11] Dressler, I., Haage, M., Nilsson, K., Johansson, R., Robertsson, A., Brogårdh, T., 2007. Configuration support and kinematics for a reconfigurable Gantry-Tau manipulator. In: IEEE Conference on Robotics and Automation (ICRA). Rome Italy, pp. 2957–2962.
- [12] Dressler, I., Robertsson, A., Johansson, R., 2007. Accuracy of kinematic and dynamic models of a Gantry-Tau parallel kinematic robot. In: IEEE Conference on Robotics and Automation (ICRA). Rome Italy, pp. 883–888.
- [13] Freescale, 2010. Product documentation for accelerometer MMA7361L. URL: http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=KIT3376MMA73x1L, accessed September, 2010.
- [14] Gunnarsson, S., Norrlöf, M., Sep. 2000. Iterative learning control of a flexible mechanical system using accelerometers. In: IFAC 6th symposium on robot control, SYROCO. Vienna, Austria.
- [15] Gunnarsson, S., Norrlöf, M., Rahic, E., Özbek, M., Mar. 2007. On the use of accelerometers in iterative learning control of a flexible robot arm. *International Journal of Control* 80 (3), 363–373.
- [16] Heidenhain, 2010. Product documentation for length gauge ST 3078. URL: <http://www.heidenhain.com>, accessed September, 2010.
- [17] Higgins, Jr., W. T., May 1975. A comparison of complementary and Kalman filtering. *IEEE Trans. Aerospace Electronic Sys.* AES-11 (3), 321–325.
- [18] Hovland, G., Murray, M., Brogårdh, T., Sep. 2007. Experimental verification of friction and dynamic models of a parallel kinematic machine. In: Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics. Zürich, Switzerland, pp. 82–87.
- [19] Johannesson, L., Berbyuk, V., Brogårdh, T., Apr. 2004. Gantry-Tau — a new three degrees of freedom parallel kinematic robot. In: Proc. 4th Chemnitz Parallel Kin. Sem. Chemnitz, Germany, pp. 731–734.
- [20] Merlet, J.-P., 2006. *Parallel Robots*, 2nd Edition. Springer, Dordrecht, The Netherlands.
- [21] Tyapin, I., Hovland, G., Brogårdh, T., Sep. 2008. Kinematic and elastodynamic design optimisation of the 3-DOF Gantry-Tau parallel kinematic manipulator. In: Proc. Workshop on Fundamental issues and future research directions to parallel mechanisms and manipulators. Montpellier, France.
- [22] Wallén, J., 2011. Estimation-based iterative learning control. Dissertations No. 1358, Dept. Electr. Eng., Linköpings universitet, Sweden, available at: <http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-64017>.
- [23] Wallén, J., Dressler, I., Robertsson, A., Norrlöf, M., Gunnarsson, S., Aug. 2011. Observer-based ilc applied to the gantry-tau parallel kinematic robot. In: Proceedings of 18th IFAC World Congress. Milano, Italy, pp. 992–998.
- [24] Wallén, J., Gunnarsson, S., Henriksson, R., Moberg, S., Norrlöf, M., Dec. 2009. ILC applied to a flexible two-link robot model using sensor-fusion-based estimates. In: Proc. IEEE Conf. Decision Control. Shanghai, China, pp. 458–463.
- [25] Wallén, J., Norrlöf, M., Gunnarsson, S., Jan. 2011. A framework for analysis of observer-based ILC. *Asian Journal Control* 13 (1), 3–14.