

MASTER

**Communication-angle iterative learning control with iteration-varying non-equidistant data points
enhancing Piezo-Stepper actuator waveforms**

Aarnoudse, L.I.M.

Award date:
2019

[Link to publication](#)

Disclaimer

This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

Eindhoven University of Technology
Eindhoven, November 12, 2019

Commutation-Angle Iterative Learning Control with Iteration-Varying Non-Equidistant Data Points: Enhancing Piezo-Stepper Actuator Waveforms

Master's Thesis
CST2019.079

Author:

0890579 Leontine Aarnoudse

Supervisors:

Thermo Fisher Scientific Edwin Verschueren
Eindhoven University of Technology Nard Strijbosch
Eindhoven University of Technology Tom Oomen

Committee members:

Eindhoven University of Technology Alessandro Saccon
Eindhoven University of Technology Gert Witvoet

Abstract

Piezo-stepper actuators are used in many nanopositioning systems due to their high resolution, high stiffness, fast response, and the ability to position a mover over an infinite stroke by means of a walking motion. The aim of this thesis is to develop a control approach for attenuating disturbances that are caused by the walking motion and are therefore repeating in the commutation-angle domain. Two new iterative learning control approaches are developed for the commutation-angle domain, that address the iteration-varying and non-equidistant sampling that occurs when the piezo-stepper actuator is driven at varying step frequencies. The first approach uses interpolation of the available iteration-varying data points to learn a disturbance-compensating input signal at an iteration-invariant set of points. The second framework uses suitable basis functions to parameterize input and error signals to obtain continuous expressions. The feasibility of both frameworks is experimentally validated on a piezo-stepper actuator, leading to significant performance improvements.

Contents

Abstract	ii
Nomenclature	v
1 Introduction	1
1.1 Piezo-stepper actuators	1
1.2 Modeling of a piezo-stepper actuator	4
1.3 Iterative learning control for waveform enhancement	6
1.4 Problem formulation	9
2 α-domain ILC	10
2.1 Approach to α -domain ILC	10
2.2 Sampling and approximation	11
2.3 Framework	14
2.4 Convergence of α -domain ILC	16
2.5 Concluding remarks	18
3 α-domain ILC with basis functions	19
3.1 Basis functions in ILC	19
3.2 Framework	20
3.3 Convergence of α -domain ILC with basis functions	25
3.4 Error fitting	26
3.5 Basis function selection	28
3.6 Concluding remarks	34
4 Application to a piezo-stepper actuator: disturbance analysis	35
4.1 Behavior of piezo-stepper actuators	35
4.2 Open-loop clamping and walking performance	37
4.3 Causes of the α -domain disturbance	40
4.4 Directionality	40
4.5 Drive-frequency dependent slope	43
4.6 Concluding remarks	43
5 Application to a piezo-stepper actuator: experiments	44
5.1 Approach to waveform enhancement using α -domain ILC	44
5.2 Experiments: α -domain ILC	47
5.3 Experiments: α -domain ILC with basis functions	52
5.4 Overview of experimental results	57
5.5 Concluding remarks	59
6 Conclusions	60
References	62
A Application to a piezo-stepper actuator: experiments	65
A.1 Influence of the learning gain	65
A.2 Influence of the cost function weights	66
A.3 α -domain ILC with basis functions at high drive frequencies	66
Paper on ‘Commutation-angle iterative learning control with iteration-varying non-equidistant data points: enhancing piezo-stepper actuator waveforms’.	67
Paper on ‘Multi-layer spatial iterative learning control for micro-additive manufacturing’.	74
Declaration concerning the TU/e Code of Scientific Conduct for the Master’s thesis	80

Nomenclature

Number Sets

\mathbb{N}	Set of all natural numbers including 0
\mathbb{R}	Set of all real numbers

Norms and Measures

$\bar{\sigma}(X)$	Maximum singular value of matrix X (equal to matrix induced 2-norm)
$\ x\ _2$	Euclidean vector norm (vector 2-norm)
$\ X\ _{i2}$	Matrix induced 2-norm
RMS(x)	Root Mean Square value of vector x

General Symbols

c	Piezo constant for a piezo shear element	m V^{-1}
c_r	Reference constant for open-loop walking experiments	m rad^{-1}
\bar{d}_α	Sampled α -domain disturbance	m
d_α	α -domain disturbance	m
e	Mover position error	m
\bar{e}	Sampled mover position error	m
f_α	Drive frequency	Hz
f_s	Sampling frequency	Hz
G	Combined piezo shear constant	m V^{-1}
\hat{G}	Estimated combined piezo shear constant	m V^{-1}
h	Sample interval	s
h_0	Direct throughput or gain of the piezo-stepper actuator	m V^{-1}
J	Impulse response matrix for the piezo-stepper actuator	-
j	Iteration index	-
N	Number of sample points	-
s_i	Input voltage for piezo shear element i	V
\bar{u}	Sampled input	V
u_s	Combined input voltage for the shear elements	V
y	Mover position	m
y_d	Desired position of mover	m
y_i	The displacement of a piezo shear element i	m
\bar{y}	Sampled mover position	m
\bar{y}_d	Sampled reference mover position	m
α	Commutation angle	rad
$\bar{\alpha}$	Vector of sampled commutation angle values	rad

Symbols for α -domain ILC Framework

\bar{d}_α^n	α -domain repeating disturbance at the nominal sample points	m
$\bar{d}_{\alpha,j}$	Difference between disturbance estimation and actual disturbance	m
$d_{u,j}$	Difference between estimated response and actual response	m
\bar{e}^n	Error at the nominal sample points	m
f_α^n	Nominal drive frequency	Hz
J^n	Impulse response matrix for the piezo-stepper actuator at the nominal sample points	-
L	Error update filter for α -domain ILC	-

Q	Input update filter for α -domain ILC	-
R_j	Estimation matrix for α -domain ILC	-
S_j	Zero-order hold matrix for α -domain ILC	-
\bar{u}^n	Input at the nominal sample points	V
\bar{y}^n	Mover position at the nominal sample points	m
$\bar{\alpha}^n$	Nominal vector of sampled commutation angle values	rad
β	Learning gain for α -domain ILC	-

Symbols for α -domain ILC with Basis Functions Framework

e_j	Continuous error for iteration j	m
L_ψ	Update filter for error for α -domain ILC with basis functions	-
M	Number of basis functions	-
Q_ψ	Update filter for input for α -domain ILC with basis functions	-
u_j	Continuous input for iteration j	V
$W_e(\alpha)$	Weighting function for error	-
$W_u(\alpha)$	Weighting function for input	-
$W_{\Delta u}(\alpha)$	Weighting function for change in input	-
y_j	Continuous output for iteration j	m
η	Shape parameter for radial basis functions	-
ψ	Basis function vector	-
$\bar{\psi}_j$	Sampled basis function matrix for iteration j	-
ψ_k	Basis function k	-
θ^d	Parameter vector for the disturbance	-
θ_j^e	Parameter vector for error fit of iteration j	-
θ_j^u	Parameter vector for input of iteration j	-
θ^{us}	Parameter vector for the combined shear input	-
θ^{yd}	Parameter vector for the reference	-

1 Introduction

Many nanopositioning systems use piezo-stepper actuators to meet increasing requirements for high precision positioning that arise due to developments in the field of nanotechnology. Applications, such as nano-motion stages [1] and scanning probe microscopy [2], require both the high resolution, high stiffness, and fast response of piezoelectric elements [3] and the large stroke provided by the walking motion. There are various ways to implement this walking motion, see for example walking drives that use longitudinal piezo elements to grab and position a mover [4], ultrasonic motors that combine longitudinal and shear piezo elements [5] and the piezo LEGS motor which uses stacks of piezo elements that form bi-morph legs that can be elongated and deflected [1, 6].

During the walking motion of a piezo-stepper actuator, there are moments of engagement and release between the piezo elements and the mover that can lead to repeating disturbances, for example when the piezo elements in the actuator are not perfectly aligned [2, 7]. The walking motion of a piezo-stepper actuator is implemented using waveforms that describe the mapping from the commutation angle to the input voltage of the piezo elements. The disturbances are repeating with the period of these actuating waveforms and lead to a nonlinear relation between the commutation angle and the mover position, for which control is typically difficult.

Learning control approaches are suitable for the compensation of repeating disturbances, but these control approaches are typically applied to systems for which the references and disturbances are repeating in the temporal domain, and for which the sampling is constant, which is not the case for a piezo-stepper actuator. In industrial implementations, piezo-stepper actuators are driven using different drive frequencies with a constant sampling frequency in the temporal domain. This leads to varying and possibly non-equidistant sampling in the commutation-angle domain in which the disturbances are repeating. Available iterative learning control (ILC) and repetitive control (RC) approaches cannot deal with this type of sampling.

In this thesis, an approach to iterative learning control outside of the temporal domain with iteration-varying non-equidistant sampling is developed. Initially, an ILC framework is developed that uses measurements at iteration-varying sampling points to learn an input signal at an iteration-invariant set of points. This is extended to a framework that uses ILC with basis functions in the commutation-angle domain to parameterize the input and error signals, leading to a continuously defined input signal. This second approach is more suitable for implementation in an industrial setting. The feasibility of both frameworks is validated experimentally using a piezo-stepper actuator, resulting in significant performance improvements.

In this section, the problem considered in this thesis is formulated. In Section 1.1 the functioning of a piezo-stepper actuator is introduced. The actuator is modeled in Section 1.2. Following from this explanation and modeling, the suitability of an iterative learning control approach as well as the limitations of current ILC approaches are investigated in Section 1.3. The problem formulation is given in Section 1.4.

1.1 Piezo-stepper actuators

The piezo-stepper actuator considered in this thesis consists of two groups of piezo elements, each containing one longitudinal element and three shear elements, as shown in Figure 1.1. When the longitudinal element, which is referred to as the clamp, of a group is extended, the shear elements of that group are in contact with the mover. The mover follows the displacement of the connected shear elements. Alternating the two piezo groups results in a walking motion, which leads to an unlimited stroke of the mover.

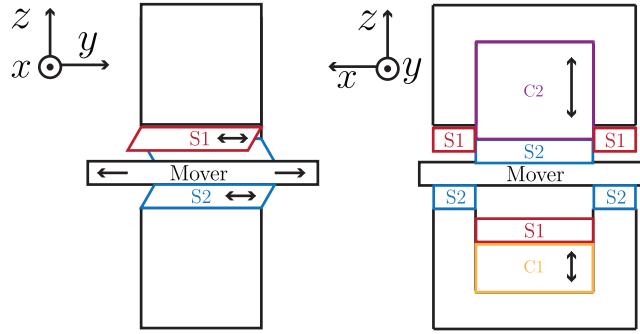


Figure 1.1: Schematic representation a piezo-stepper actuator showing the clamp ('C') and shear ('S') elements of the first (—, orange) and second (—, purple) group.

1.1.1 Actuation

One cycle of the walking motion is referred to as a step and is divided into the following five parts. These parts are illustrated in Figure 1.2.

- Initially, both clamps are extended halfway and both groups are in contact with the mover. The first clamp element is extending while the second is retracting. The two sets of shear elements are moving with equal velocity.
- Once the second set of shear elements loses contact with the mover, these shear elements are reset by moving them backwards.
- After that, the first clamp starts retracting while the second is extending, so that the mover is taken over by the second group.
- Once the first set of shear elements loses contact with the mover, these shear elements are reset.
- Finally, the first clamp starts extending while the second retracts which returns the system to the initial situation.

The cyclic walking motion of the piezo-stepper is implemented using the periodic waveforms that are shown in Figure 1.3. These waveforms map the commutation angle $\alpha \in [0, 2\pi)$ to the input voltage of the piezo elements. The number of steps per second is determined by the drive frequency f_α [Hz]. The piezo-stepper is controlled in open-loop as shown in Figure 1.4. The input is a drive frequency, from which the commutation angle α [rad] is determined. Based on the waveforms the voltage inputs for the shear and clamp elements are generated, resulting in a position output of the mover y [m].

1.1.2 Experiments

The behavior of the piezo-stepper actuator, including effects such as hysteresis, creep and the disturbances introduced by the walking motion, is analyzed in two types of experiments: open-loop walking and open-loop clamping. The desired motion or reference y_d [m] during these experiments differs. The experiments are defined as follows:

Definition 1.1. Open-loop walking experiment: during a walking experiment, both the clamps and shears are actuated using the standard waveforms. Since the derivative of the input voltage to the connected shears is constant, the desired position of the mover is linear, i.e., $y_d(\alpha) = c_r\alpha$ with constant c_r [m rad^{-1}]. The desired linear motion in these experiments is comparable to the desired system behavior in a closed-loop setting.

Definition 1.2. Open-loop clamping experiment: in a clamping experiment, the clamps are controlled using the standard clamp waveforms while the shear elements are not actuated. Since the shear elements are not actuated, the desired position of the mover during this experiment is constant, i.e., $y_d(\alpha) = 0$. These experiments are used to gain insight in the origin and behavior of disturbances in the piezo-stepper actuator.

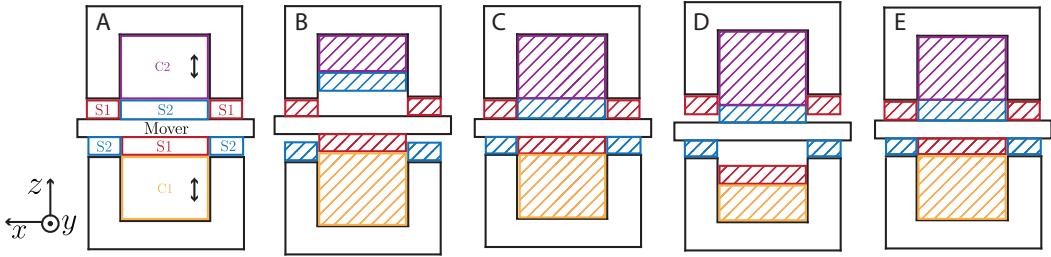


Figure 1.2: The piezo actuator positions a mover by means of a motion reminiscent of walking by alternating the first (—, —) and second (—, —) piezo groups.

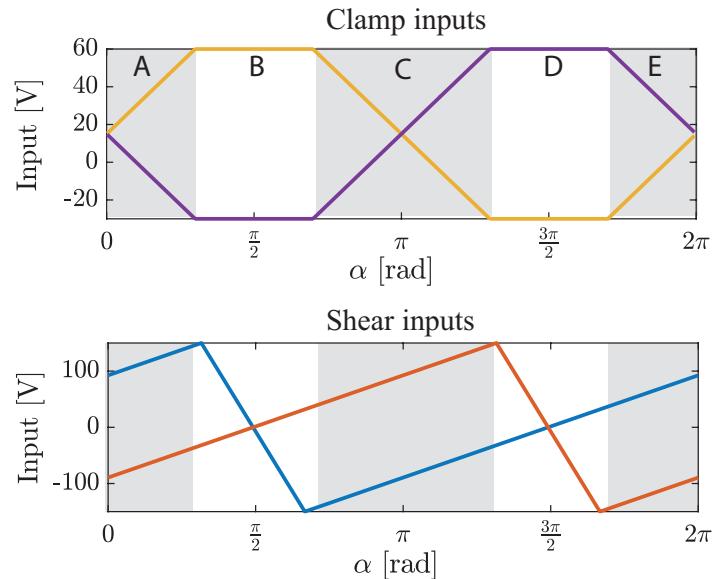


Figure 1.3: The periodic waveforms map the commutation angle α to the input voltage of the piezo elements. The waveforms for clamps 1 (—) and 2 (—) contain regions where both clamps could be in contact with the mover, indicated in gray. In these regions the inputs for the shear elements 1 (—) and 2 (—) have equal derivatives. The five parts into which a step can be divided are indicated.

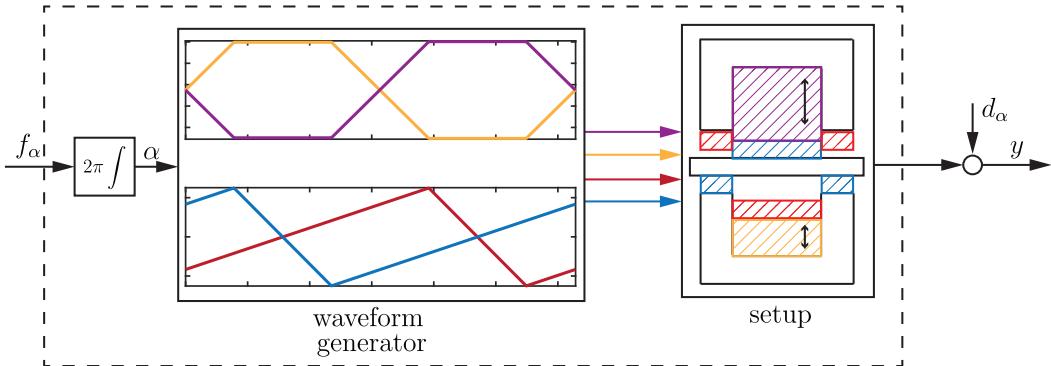


Figure 1.4: Open-loop implementation of the waveforms.

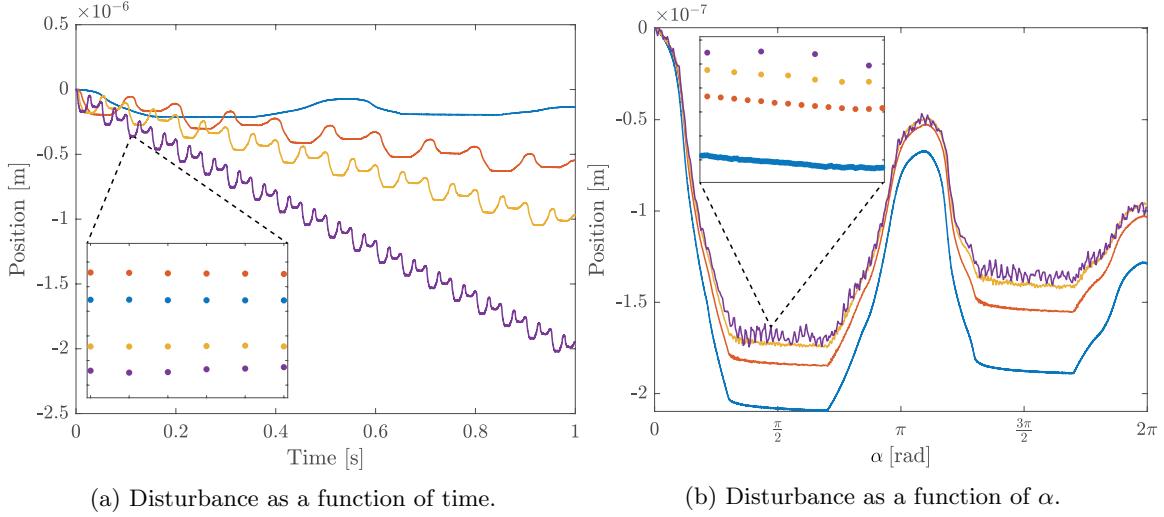


Figure 1.5: Disturbances for a piezo-stepper during open-loop clamping. In the temporal domain (a) the sampling is equidistant (see zoom plot) but the disturbance is not repeating for different drive frequencies. In the α -domain (b) the sampling is non-equidistant for varying drive frequencies, but the disturbances are similar. Drive frequencies: 1 Hz (—), 5 Hz (—), 10 Hz (—), 20 Hz (—).

Different input drive frequencies for the piezo-stepper actuator lead to different sample points in the commutation angle- or α -domain. The system is sampled at a constant frequency in the temporal domain. The drive frequency determines the number of steps within a second, and since the time domain sampling frequency is constant a higher number of steps per second results in fewer data points per step. If the drive frequency changes within one step, the data points in that step are non-equidistant. The difference between the equidistant sampling in the temporal domain and the non-equidistant drive-frequency dependent sampling in the α -domain is illustrated in Figure 1.5.

1.1.3 Disturbances

The walking motion of the piezo-stepper actuator introduces disturbances in the position of the mover that are repeating with the period of the actuating waveforms. During open-loop clamping and walking experiments, the disturbances are shown to occur around moments of engagement and release between the piezo elements and the mover. A possible explanation is imperfect alignment of the piezo elements in the actuator [7, 8]. The disturbances are varying with the drive frequency in the temporal domain, but are repeating in the α -domain, as shown in Figure 1.5. The disturbances, as well as behaviors such as hysteresis and creep, are analyzed further in Chapter 4.

1.2 Modeling of a piezo-stepper actuator

The piezo-stepper actuator is modeled as a gain with a lumped disturbance in the α -domain. Rate-dependent or nonlinear effects such as hysteresis are assumed to be negligible, since these can be compensated using separate feedforward [3], ch.2.11. The time-domain effects and possible compensating strategies are investigated further in Section 4.1. The modeled piezo-stepper actuator is described in the α -domain without any significant time-domain dynamics.

The commutation angle α [rad] is determined by the drive frequency f_α [Hz] and is given by

$$\alpha(t) = 2\pi \int_0^t f_\alpha(\tau) d\tau. \quad (1.1)$$

The system is driven with varying drive frequencies and sampled at a constant frequency f_s [Hz] in the temporal domain. When a single step is taken, it holds that $\alpha(0) = 0$ and $\alpha(T) = 2\pi$, where the duration T [s] of a step depends on the drive frequency. The vector containing the α -values at which a sample is taken during a single step is given by

$$\bar{\alpha} = 2\pi \begin{bmatrix} \int_0^h f_\alpha(\tau) d\tau & \int_0^{2h} f_\alpha(\tau) d\tau & \dots & \int_0^{Nh} f_\alpha(\tau) d\tau \end{bmatrix}^\top \quad (1.2)$$

with sample interval $h = f_s^{-1}$ [s]. The number of samples within a step is given by $N = \lfloor Tf_s \rfloor$.

The displacement of a single shear element y_i [m] is given by

$$y_i(\alpha(t)) = cs_i(\alpha(t)), \quad (1.3)$$

with positive piezo constant c [m V⁻¹] and commutation angle α [rad]. The input voltage s_i [V] is a function of the commutation angle according to the waveforms. An example of the relation between the variables f_α , α , s_i and the mover displacement y [m] is shown in Figure 1.6.

The waveforms are designed to obtain a linear relation between the commutation angle α and the mover displacement y . Therefore, the waveforms of the shear elements are designed such that they have equal derivatives for any α where both clamps could be in contact with the mover, as shown in Fig. 1.3. The part of the step where one of the clamps is completely retracted is used to reset the corresponding shear elements. Because the connected shear elements are always moving with the same velocity, the combination of the two shear inputs $s_1(\alpha)$ and $s_2(\alpha)$ is written as a single input $u_s(\alpha)$ [V], satisfying

$$\frac{\delta u_s(\alpha)}{\delta \alpha} = \begin{cases} \frac{\delta s_1(\alpha)}{\delta \alpha} & \text{if } \alpha \in [\frac{\pi}{3}, \frac{2\pi}{3}] \\ \frac{\delta s_2(\alpha)}{\delta \alpha} & \text{if } \alpha \in [\frac{4\pi}{3}, \frac{5\pi}{3}] \\ \frac{\delta s_1(\alpha)}{\delta \alpha} = \frac{\delta s_2(\alpha)}{\delta \alpha} & \text{otherwise.} \end{cases} \quad (1.4)$$

The corresponding desired mover position is given by

$$y_d(\alpha(t)) = Gu_s(\alpha(t)) \quad (1.5)$$

with piezo shear constant G [m V⁻¹]. During open-loop experiments this linear relation is not obtained due to disturbances. The disturbances are assumed to be relatable to the commutation angle α , and are modeled by the lumped α -domain disturbance $d_\alpha(\alpha)$ [m].

The inputs s_i , $i = 1, 2$ are a function of the commutation angle α through the waveforms, and therefore the total input voltage u_s is also a function of α . Since α itself is a function of time according to (1.1), the input u_s and output y are indirectly a function of time. The position of the mover is described by

$$y(\alpha(t)) = Gu_s(\alpha(t)) + d_\alpha(\alpha(t)). \quad (1.6)$$

When a single step of the piezo-stepper actuator is considered, the system is written in terms of α as

$$y(\alpha) = Gu_s(\alpha) + d_\alpha(\alpha), \quad \alpha \in [0, 2\pi], \quad (1.7)$$

Based on this model, different sampling rates of the α -domain system lead to the same sampled system description and the non-equidistantly sampled system is linear time-invariant, since the system is described as a gain with a disturbance that only depends on the value of α .

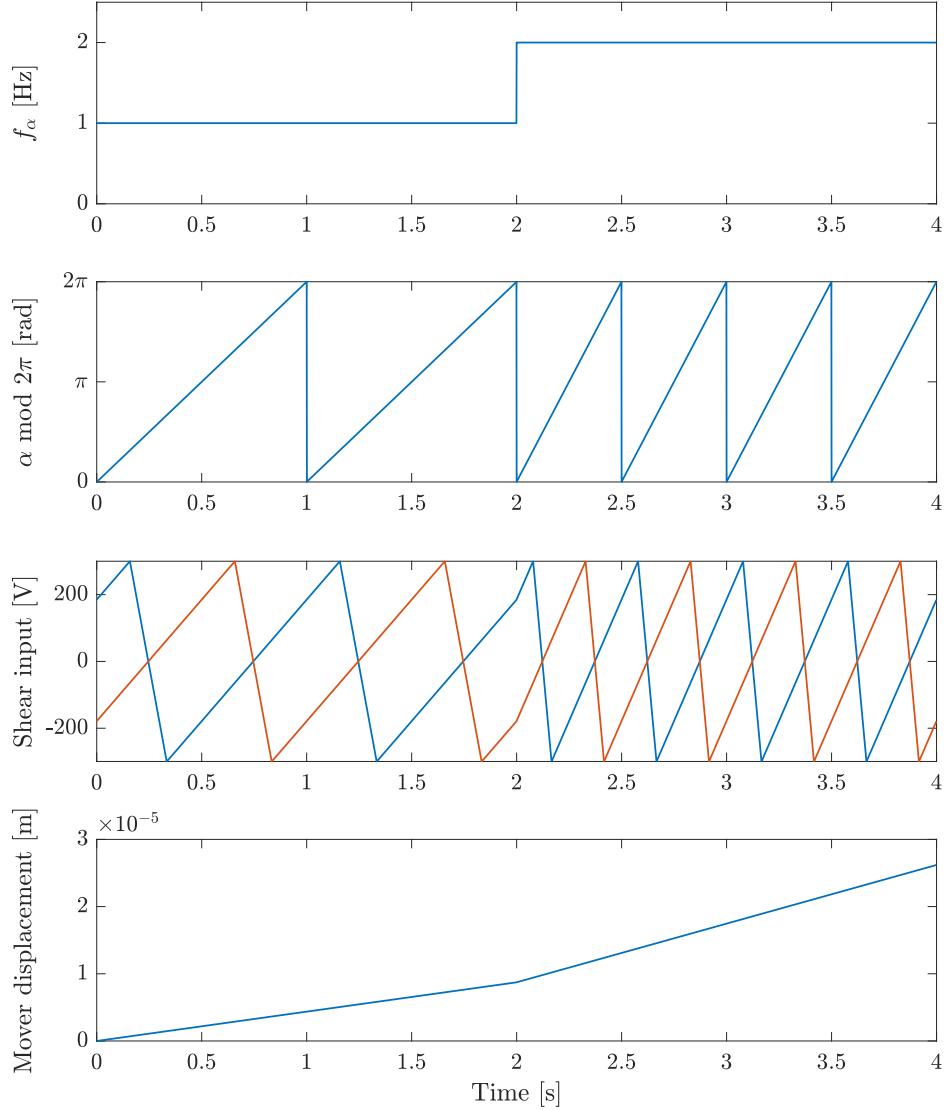


Figure 1.6: Illustration of the relation between a given input drive frequency f_α , the modulus of the corresponding commutation angle $\alpha \bmod 2\pi$, the inputs for the two shear elements s_1 (—) and s_2 (—), and the displacement of the mover y .

1.3 Iterative learning control for waveform enhancement

Learning control approaches are suitable for systems with repeating disturbances such as those observed for a piezo-stepper actuator. In general, approaches such as iterative learning control (ILC) and repetitive control (RC) are aimed at improving performance for a constantly sampled system with a time-domain repeating reference. For systems such as a piezo-stepper actuator, the disturbance and reference are repeating in the α -domain instead of the temporal domain and the sampling in the α -domain is iteration-varying and non-equidistant. Currently available learning control frameworks cannot deal with this type of sampling.

In this section an overview of existing approaches to waveform design for piezo-stepper actuators is given and the suitability of learning-based approaches is investigated. Then, some approaches from literature that consider relevant problems for α -domain ILC, such as iteration-varying and non-equidistant sampling, are explored.

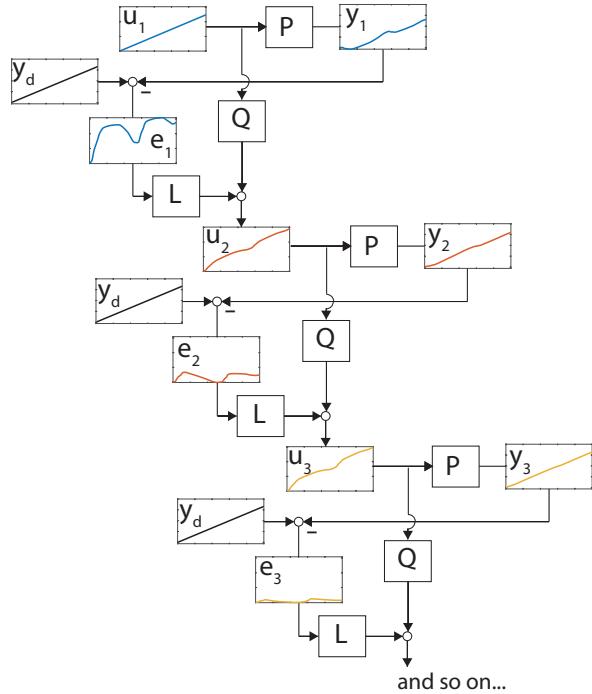


Figure 1.7: Example of an iterative learning control system. The input signal u for plant P is learned over iterations. Comparison of the output y and the iteration-invariant reference y_d results in an error e , which is filtered and added to the filtered input signal, resulting in an improved input signal for the next iteration.

1.3.1 Waveform design and enhancement

Since the main disturbance of the piezo-stepper considered in this thesis is related to the walking motion and is repeating in the α -domain [7, 8], the input waveform can be adapted to compensate for this disturbance. The performance of piezo-stepper actuators depends strongly on the choice of input waveforms, and different methods for designing waveforms are defined in literature. In [2] disturbances at the moment of takeover between two groups of piezo elements are identified and reduced by making both groups move with the same speed during a gradual takeover, as is also the case for the piezo-stepper considered in this thesis. In [9] triangular and trapezoidal waveforms with and without parabolic forms on one side are compared and optimized for a constant velocity. In [6] a coordinate transform is used to define the waveforms for a piezo LEGS motor based on the desired step shape. Data-driven optimization of the waveforms for a motor consisting of four bimorph legs is shown to yield lower velocity errors than a model-driven approach in [1]. However, none of these approaches exploit learning control to design the repeating waveforms.

1.3.2 Iterative learning control

Learning control approaches such as iterative learning control (ILC) can compensate repeating disturbances perfectly, but they may amplify iteration-varying disturbances. In ILC, a feedforward input signal is adapted based on preceding experiments that use the same reference, so that the tracking error is reduced over iterations, as shown in Figure 1.7. ILC is typically applied in the temporal domain to systems that show repeating errors or disturbances for repeating tasks, and can compensate iteration-invariant disturbances perfectly [10,11]. However, typical ILC approaches amplify iteration-varying disturbances [12]. The disturbance for a piezo-stepper actuator is varying in the temporal domain for varying drive frequencies. In the α -domain, the disturbance is repeating. Therefore, temporal domain ILC is not suited for a piezo-stepper actuator and a commutation-angle domain approach is needed.

Existing approaches to ILC outside of the temporal domain depend strongly on assumptions regarding the sampling in the spatial domain. In [13], temporal dynamics are ignored to develop a 2D spatial iterative learning control approach. This approach is used in additive manufacturing, where the output of the system is measured at a set number of discrete points in the spatial domain. In [14], phase-indexed ILC is developed for a walking robot which behaves as an almost periodic system, where it is assumed that in the limit stable periodic behavior is obtained with an unknown period time. However, these approaches cannot cope with the sampling for a piezo-stepper actuator, which is iteration-varying and non-equidistant for varying drive frequencies.

In [8] an approach is presented that uses ILC in the commutation-angle domain to reduce repeating disturbances for a piezo-stepper actuator through waveform enhancement. Experimental results show a significant improvement in performance, both during open-loop clamping experiments and when the waveform learned during clamping is applied in an open-loop walking experiment. However, it is assumed that continuous signals are available which is not feasible in a digital setting. In addition, learning is only applied during open-loop clamping experiments and not during the actual walking motion of the piezo-stepper actuator. For commutation-angle ILC, the iteration-varying non-equidistant sampling needs to be considered explicitly.

Although ILC can be used to compensate hysteresis-caused positioning errors in piezo actuators, inclusion of hysteresis compensation in the commutation-angle ILC framework developed in this thesis is not feasible. In [15], a framework is proposed that uses ILC to compensate a hysteresis-caused positioning error in a piezo system, by dividing the trajectory into parts for which the compensation is learned separately and consecutively. For each iteration, the system is reset to remove hysteresis and initialized using the converged input signals for previous parts of the trajectory. This procedure requires separate design of every possible hysteresis path of the actuator. The piezo-stepper actuator considered in this thesis is driven with varying drive frequencies, and when partial steps are taken in a closed-loop implementation the length of these steps is arbitrary, leading to an unlimited number of possible paths. Therefore, including hysteresis compensation in the ILC framework is not feasible. Instead, it is recommended to compensate the hysteresis using separate feedforward, as is further investigated in Section 4.1.

1.3.3 Repetitive control

Another approach to dealing with repeating disturbances in periodic systems is repetitive control (RC), which considers continuously repeating motion tasks [11]. Several approaches to spatial repetitive control are defined in literature. In [16], a rotational system is reformulated with respect to angular displacement to obtain a quasi-linear parameter-varying system where angular speed is a measurable varying parameter. A controller is designed and implemented using constant angular displacement period spatial sampling. In [17], an approach to multi-rate RC is presented for a system with a fixed time-domain sampling rate. The sampling time of the repetitive controller is adjusted to synchronize signal and control period, and interpolation is used to interface the portions of the system with different sampling times. In [18], delay-varying RC is developed specifically for a walking piezo actuator with bi-morph legs. A repetitive variable, for example the angular orientation of the legs, is measured or observed to adjust the repetitive delay in the RC scheme.

Although they consider some relevant issues, such as varying sampling rates in the angular domain, RC approaches are not entirely suitable for the piezo-stepper actuator. The movement of a piezo-stepper actuator in a closed-loop implementation is not always continuously repeating in the α -domain, since positive and negative drive directions may be used alternately. Although an adapted RC scheme might be considered, an approach based on iterative learning control is deemed more suitable.

1.3.4 Iterative learning control for iteration-varying data points

Existing approaches to learning control for iteration-varying and non-equidistant sampling do not consider the specific sampling situation of commutation-angle ILC. In [19], iteration-varying sampling is considered, which is caused by incomplete trials, for example because a trial is stopped when the system is drifting too far away from its reference. The sampling is equidistant, and it is assumed that a constant sampling length is obtained in the limit. A framework for ILC with non-equidistant sampling is proposed in [20], where the time-stamps of encoders are exploited to eliminate quantization effects. In this approach the output is sampled at a high rate, which is assumed to be a multiple of the lower rate at which the input is sampled. Only part of the sampled output signal is used, resulting in iteration-varying non-equidistant sampling. For a piezo-stepper actuator, the input and output are sampled at the same iteration-varying rate, which in general cannot be related to a constant nominal sampling rate.

1.4 Problem formulation

Piezo-stepper actuators use a walking motion to increase the stroke over which a mover can be positioned, but this walking motion introduces repeating disturbances. These disturbances are repeating in the commutation-angle domain with the period of the actuating waveforms. To reduce the influence of this disturbance and obtain a linear relation between commutation angle and mover position, the waveforms can be adapted to compensate for the repeating disturbance. However, the sampling in the commutation-angle domain is varying and non-equidistant for varying drive frequencies.

Although there exist frameworks for iterative learning control outside of the temporal domain as well as frameworks that consider iteration-varying sampling, there is no framework that is suitable for ILC outside of the temporal domain with iteration-varying non-equidistant sampling. This thesis aims to develop such a framework for commutation-angle iterative learning control, suitable for systems such as a piezo-stepper actuator. This leads to the following three contributions:

1. In Chapter 2, a framework for α -domain ILC with iteration-varying non-equidistant sampling is proposed. In this approach, a disturbance-compensating input signal is defined on an iteration-invariant set of data points. The output at these points is approximated to update the input signal.
2. In Chapter 3 a framework for α -domain ILC with iteration-varying non-equidistant sampling is proposed that uses basis functions to parameterize the input signal. Compared to the first framework, this approach has significant computational and implementational benefits.
3. The feasibility of both frameworks is validated during walking experiments with a piezo-stepper actuator, resulting in significant performance improvements. In Chapter 4 a disturbance analysis of this piezo-stepper actuator is presented. The experimental implementation and results are presented in Chapter 5.

Conclusions and recommendations are given in Chapter 6.

Remark. During the graduation project, a paper has been written based on Chapters 3 and 5. This paper has been submitted to the 21st IFAC World Congress, 2020. In addition, during the graduation project, a paper on Multi-Layer Spatial Iterative Learning Control for Micro-Additive Manufacturing [21] was written and presented at the 8th IFAC Symposium on Mechatronic Systems, Vienna, 2019. Both papers have been added at the end of this document.

2 α -domain ILC

In this chapter a framework is developed for iterative learning control in the α -domain with iteration-varying and possibly non-equidistant sampling. The input signal is defined at an iteration-invariant set of sample points and is implemented using zero-order hold. Because of the iteration-varying sampling, the output is not necessarily measured at this set of sample points. Therefore, the input signal is updated based on an approximated error. The influence of this approximation on the stability and convergence of the ILC scheme is explicitly taken into account.

The general approach to α -domain is explained in Section 2.1. In Section 2.2 the sampling problem that arises when iteration-varying drive frequencies are sampled with a constant time-domain sample frequency is investigated, and the approach to approximating the output at a nominal set of sample points is presented. In Section 2.3 the framework is presented. Stability and convergence conditions are provided in Section 2.4. Lastly, conclusions are given in Section 2.5.

2.1 Approach to α -domain ILC

In this section the approach to α -domain ILC using a lifted framework to learn a sampled input signal and the assumptions used for α -domain ILC are presented. According to Section 1.2, the displacement of the mover of a piezo-stepper actuator actuated by the standard shear waveforms during a single step is described by

$$y(\alpha) = Gu_s(\alpha) + d_\alpha(\alpha), \quad \alpha \in [0, 2\pi]. \quad (2.1)$$

To compensate the disturbance, an additional input signal $u(\alpha)$ is added such that

$$y(\alpha) = G(u_s(\alpha) + u(\alpha)) + d_\alpha(\alpha), \quad \alpha \in [0, 2\pi]. \quad (2.2)$$

This additional input signal is learned using ILC. For the purpose of applying α -domain ILC, the system of (2.2) is sampled and described in lifted form as follows:

$$\bar{y}_j = J(\bar{u}_s + \bar{u}_j) + \bar{d}_\alpha, \quad (2.3)$$

$$\bar{e}_j = \bar{y}_d - \bar{y}_j, \quad (2.4)$$

with sampled mover position \bar{y}_j [m], sampled compensating input \bar{u}_j [V] and sampled α -domain repeating disturbance \bar{d}_α [m] for iteration j . Since the system is assumed to behave as a gain, the impulse response matrix J is of the form $h_0 I$, where $h_0 = G$ is the direct throughput or gain of the piezo system. The sampled error \bar{e}_j [m] is defined as the deviation of the output \bar{y}_j from the sampled desired position \bar{y}_d . For open-loop clamping experiments the continuous reference is $y_d(\alpha) = 0 \forall \alpha \in [0, 2\pi]$. For open-loop walking, the desired motion is a straight line with constant velocity, i.e., $y_d(\alpha) = c_r \alpha$ with $c_r \in \mathbb{R}$.

Lifted ILC is typically used to design an input signal $\bar{u}_j \in \mathbb{R}^{N \times 1}$ at all N points at which the system is sampled. In typical time-domain ILC, the number and location of the samples in $\bar{y}_j \in \mathbb{R}^{N \times 1}$ and \bar{u}_j is constant over iterations, resulting in an iteration-invariant matrix $J \in \mathbb{R}^{N \times N}$. In α -domain ILC, iteration-varying drive frequencies lead to varying lengths of input and output signals, so that the sampled input and output vectors $\bar{u}_j, \bar{y}_j \in \mathbb{R}^{N_j \times 1}$ contain samples for all $\alpha \in \bar{\alpha}_j$, where $\bar{\alpha}_j$ is defined according to (1.2).

For the ILC frameworks, three assumptions are made:

Assumption 2.1. *The initial condition $y_j(0)$ is identical for each iteration j .*

Assumption 2.2. *The length of each iteration is constant in the α -domain, i.e., $\bar{\alpha}_j \in [0, 2\pi]$*

Assumption 2.3. *For each iteration there is a unique mapping $F_j : [0, T_j] \mapsto [0, 2\pi]$ from the time interval $t \in [0, T_j]$ to the commutation-angle interval $\alpha \in [0, 2\pi]$.*

Assumption 2.1 is satisfied by defining the initial position for each iteration to be $y(0) = 0$. Assumption 2.2 is satisfied by choosing the measurement time so that $\alpha(0) = 0$ and $\alpha(T_j) = 2\pi$, which allows varying iteration lengths in the temporal domain. Assumption 2.3 is satisfied when α is continuously increasing or decreasing within an iteration.

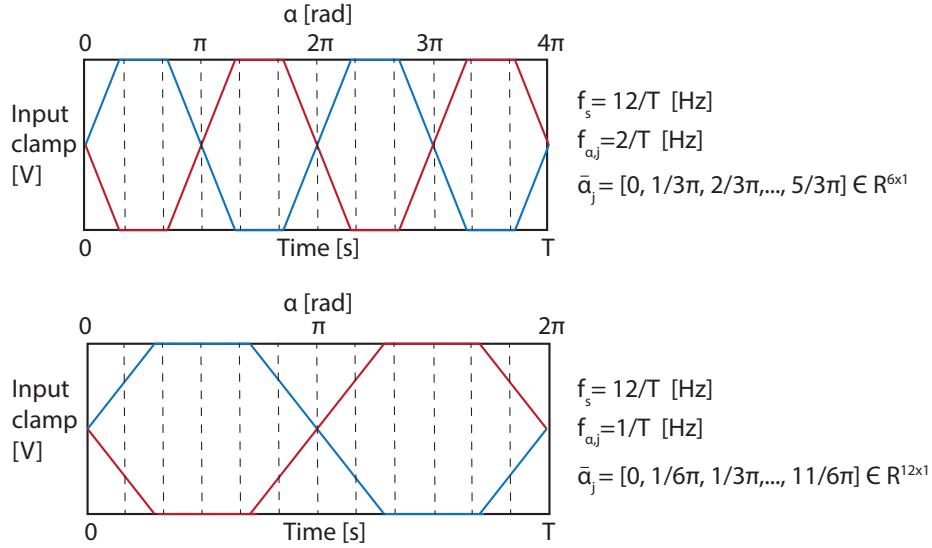


Figure 2.1: For different drive frequencies, the set of points in $[0, 2\pi)$ on which a sample is taken (---) is different.

2.2 Sampling and approximation

For different drive frequencies, the sampling in the α -domain is iteration-varying and non-equidistant. Therefore, the length of the sampled input and output signals is iteration-varying, so that a standard lifted ILC approach is not suitable. Therefore, an approach is presented that learns an input signal at a nominal set of data points by approximating the output at these points based on the measured output.

For different drive frequencies, the number of points on the domain $\alpha \in [0, 2\pi)$ on which the system is sampled is varying. The system is sampled in the temporal domain with a set frequency f_s [Hz]. The rate of change of α is determined by the iteration-varying drive frequency $f_{\alpha,j}$ [Hz]. Together, these frequencies determine the set of α -values $\bar{\alpha}_j$ on which a sample is taken during iteration j , according to (1.2). The set of points $\bar{\alpha}_j$ is iteration-varying as shown in the example in Figure 2.1.

The sampled compensating input signal \bar{u}_j is defined on a nominal set of α -values given by $\bar{\alpha}^n \in \mathbb{R}^{N \times 1}$, corresponding to a nominal drive frequency f_α^n , such that

$$\bar{u}_j(\bar{\alpha}^n) = [u_j(\bar{\alpha}^n(1)) \quad u_j(\bar{\alpha}^n(2)) \quad \dots \quad u_j(\bar{\alpha}^n(N))]^T \in \mathbb{R}^{N \times 1}, \quad (2.5)$$

where $\bar{\alpha}^n(k)$ is the k^{th} element of vector $\bar{\alpha}^n$. To implement \bar{u}_j for iterations with different drive frequencies, i.e., $f_{\alpha,j} \neq f_\alpha^n$ and $\bar{\alpha}_j \neq \bar{\alpha}^n$, zero-order hold is used:

$$u_j(\alpha) = \bar{u}_j(\bar{\alpha}^n(k)) \quad \text{for } \alpha \in [\bar{\alpha}^n(k), \bar{\alpha}^n(k+1)). \quad (2.6)$$

Each iteration the input signal $\bar{u}_j(\bar{\alpha}^n)$ is updated on all N nominal sample points. However, \bar{y}_j does not necessarily contain a sample at each of these points. Comparing $\bar{\alpha}_j$ to $\bar{\alpha}^n$ gives five possible situations:

1. $\bar{\alpha}_j$ and $\bar{\alpha}^n$ are identical when $f_{\alpha,j}$ is equal to the nominal drive frequency f_α^n .
2. $\bar{\alpha}_j$ contains more points than $\bar{\alpha}^n$ when $f_{\alpha,j}$ is smaller than f_α^n , which gives two possibilities:
 - a. Each point in $\bar{\alpha}^n$ is also in $\bar{\alpha}_j$ when $f_{\alpha,j}$ is a divisor of f_α^n .
 - b. Some or all points in $\bar{\alpha}^n$ are not in $\bar{\alpha}_j$.

Algorithm 2.4 Sampling and approximation in α -domain ILC

```

Choose a nominal drive frequency  $f^n$ 
for  $j = 1 : n$  do
    Perform an experiment with  $f_{\alpha,j}$ , implementing  $\bar{u}_j$  using zero-order hold
    Measure the output  $\bar{y}_j(\bar{\alpha}_j)$  at drive-frequency dependent sample points  $\bar{\alpha}_j$ 
    Approximate the output  $\bar{y}_j(\bar{\alpha}^n)$  at nominal sample points
    Determine the error  $\bar{e}_j(\bar{\alpha}^n)$  at the nominal sample points
    Determine the next input  $\bar{u}_{j+1}(\bar{\alpha}^n)$  at the nominal sample points, see Section 2.3
end for

```

3. $\bar{\alpha}_j$ contains less points than $\bar{\alpha}^n$ when $f_{\alpha,j}$ is larger than f_α^n , which gives two possibilities:
 - a. Each point in $\bar{\alpha}_j$ is also in $\bar{\alpha}^n$ when $f_{\alpha,j}$ is a multiple of f_α^n .
 - b. Some or all points in $\bar{\alpha}_j$ are not in $\bar{\alpha}^n$.

For each of these situations, the output at the nominal α -values, $\bar{y}_j(\bar{\alpha}^n)$, is approximated to determine the next input signal $\bar{u}_{j+1}(\bar{\alpha}^n)$. An outline of the approach to updating the input signal is given in Algorithm 2.4.

The output in each of the possible sampling situations is approximated as outlined below, which is illustrated in Figure 2.2. Note that in reality the amount of samples is far larger than what is illustrated here, resulting in smaller deviations between nominal and estimated sample points.

1. $f_{\alpha,j} = f_\alpha^n$, so that $\bar{\alpha}_j = \bar{\alpha}^n$. A sample is taken at all points in $\bar{\alpha}^n$, therefore $\bar{y}_j(\bar{\alpha}^n)$ is known completely and $\bar{u}_{j+1}(\bar{\alpha}^n)$ can be determined.
2. $f_{\alpha,j} < f_\alpha^n$, so that the number of elements in $\bar{\alpha}_j$ is larger than that in $\bar{\alpha}^n$. For this case, there are two possibilities.
 - a. $\frac{f_\alpha^n}{f_{\alpha,j}} \in \mathbb{N}$, so that $\bar{\alpha}^n(k) \in \bar{\alpha}_j \forall k \in \mathbb{N}$. A sample is taken at all points in $\bar{\alpha}^n$, therefore $\bar{y}_j^n(\bar{\alpha}^n)$ is known completely and $\bar{u}_{j+1}(\bar{\alpha}^n)$ can be determined. The extra data is not used.
 - b. $\frac{f_\alpha^n}{f_{\alpha,j}} \notin \mathbb{N}$, so that a sample may be taken at some but not all points in $\bar{\alpha}^n$. For the points that are measured exactly, $\bar{u}_{j+1}(\bar{\alpha}^n)$ is determined directly. For the points in $\bar{\alpha}^n$ for which no exact measurement is available, $\bar{y}_j(\bar{\alpha}^n)$ is approximated by $\bar{y}_j(\bar{\alpha}_j(m))$, where $\bar{\alpha}_j(m)$ is the first measured α -value higher than the nominal one. The extra data is not used.
3. $f_{\alpha,j} > f_\alpha^n$, so that the number of elements in $\bar{\alpha}_j$ is smaller than that in $\bar{\alpha}^n$. For this case, there are two possibilities.
 - a. $\frac{f_{\alpha,j}}{f_\alpha^n} \in \mathbb{N}$, so that $\bar{\alpha}_j(m) \in \bar{\alpha}^n \forall m \in \mathbb{N}$. All sampled points are part of the nominal set, so $\bar{u}_{j+1}(\bar{\alpha}^n)$ can be determined on these points. For the points in the nominal set that are not measured, $\bar{y}_j(\bar{\alpha}^n)$ is approximated using linear interpolation between the measured points and $\bar{u}_{j+1}(\bar{\alpha}^n)$ is determined based on the approximation.
 - b. $\frac{f_{\alpha,j}}{f_\alpha^n} \notin \mathbb{N}$, so that a sample may be taken for some points in $\bar{\alpha}^n$, but not for all and not all sampled points are in the nominal set. For the points in $\bar{\alpha}^n$ for which no exact measurement is available, $\bar{y}_j(\bar{\alpha}^n)$ is approximated by $\bar{y}_j(\bar{\alpha}_j(m))$, where $\bar{\alpha}_j(m)$ is the first measured α -value higher than the nominal one. For the points in the nominal set that are not measured and for which the next measured α -value is equal to or higher than the next point in the nominal set, i.e., $\bar{\alpha}_j(m) < \bar{\alpha}^n(k) < \bar{\alpha}^n(k+1) \leq \bar{\alpha}_j(m+1)$, linear interpolation between the measured points is used to approximate $\bar{y}_j(\bar{\alpha}^n)$ and $\bar{u}_{j+1}(\bar{\alpha}^n)$ is determined based on the approximation.

Remark. Although the sampling is presented as if $f_{\alpha,j}$ is constant within an iteration, this does not have to be the case. In case of a varying drive frequency within an iteration, the data points are non-equidistant. Then, the output signal $\bar{y}_j(\bar{\alpha}^n)$ is determined based on the approach that covers all drive frequencies within the iteration.

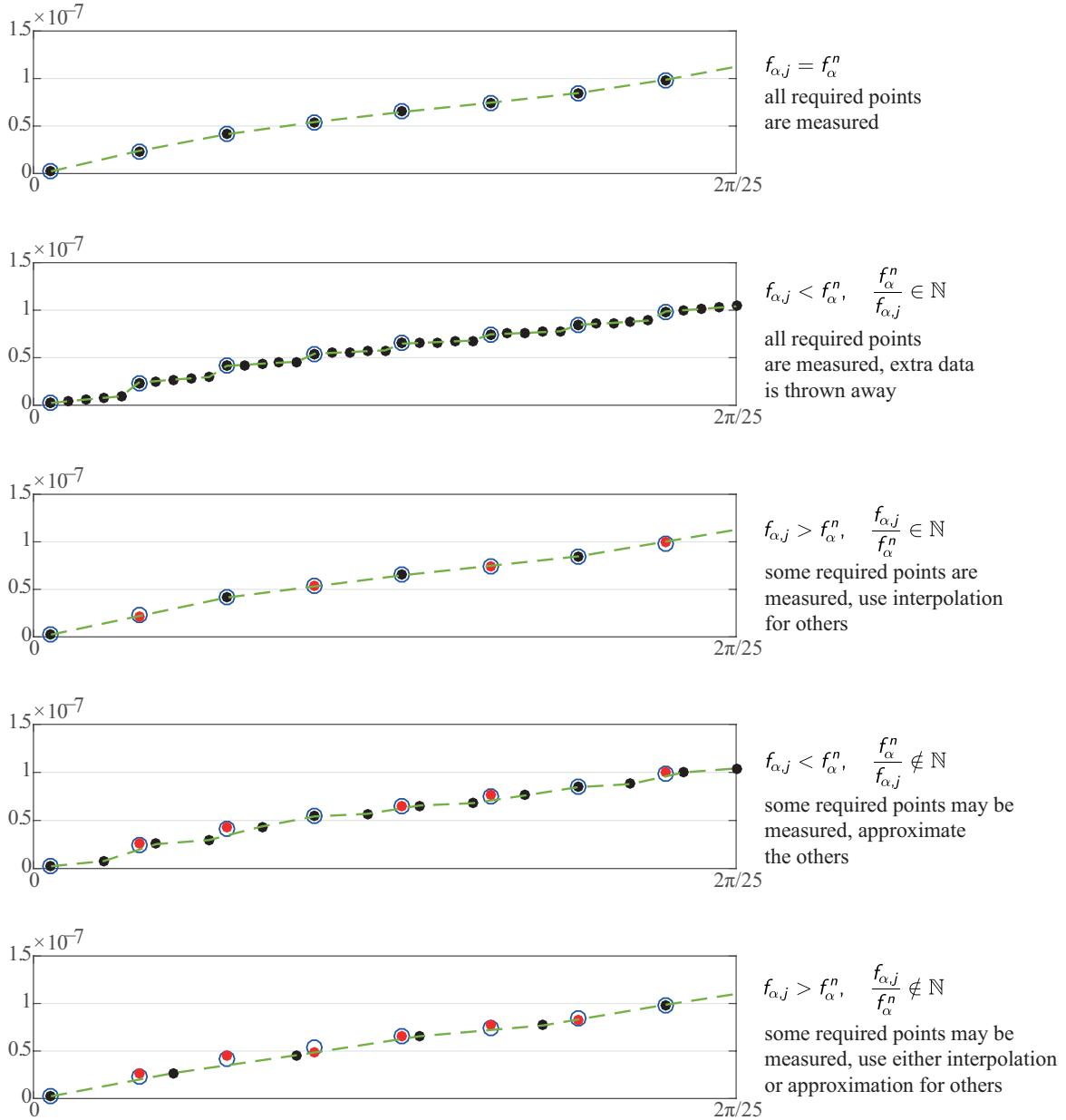


Figure 2.2: Overview of the sampling situations. The samples taken at a certain iteration (\bullet) can be part of the set of nominal sample points (\circlearrowleft). If that is not the case, the value at the nominal sample points is estimated (\bullet). Linear interpolation between the samples is indicated by (- -).

Remark. It might be suggested that linear interpolation should be used for every situation where the output at a nominal sample point $\bar{y}_j(\bar{\alpha}^n(k))$ is not measured exactly. However, due to the zero-order hold based implementation of \bar{u}_j and \bar{u}_s , it holds that

$$\begin{aligned}\bar{y}_j(\alpha) &= h_0(\bar{u}_s(\bar{\alpha}^n(k)) + \bar{u}_j(\bar{\alpha}^n(k))) + d_\alpha(\alpha) \quad \text{for } \alpha \in [\bar{\alpha}^n(k), \bar{\alpha}^n(k+1)) \\ &= \bar{y}_j(\bar{\alpha}^n(k)) + d_\alpha(\alpha) - d_\alpha(\bar{\alpha}^n(k)).\end{aligned}\tag{2.7}$$

Under the assumption that the deviation $(d_\alpha(\alpha) - d_\alpha(\bar{\alpha}^n(k)))$ is small, this approximation is expected to give a more realistic evaluation of the input $\bar{u}_j(\bar{\alpha}^n(k))$ than linear interpolation. This is illustrated in Figure 2.2. The zero-order hold implementation is clearly visible in the second situation, and the difference between the used approximations and the interpolation between the samples is shown especially in the fourth situation.

2.3 Framework

In this approach to α -domain ILC, the input signal is learned at an iteration-invariant set of sample points, by using approximations of the output at these points, and implemented using zero-order hold. In this section an ILC framework is developed that includes all possible sampling situations for α -domain ILC with iteration-varying arbitrary drive frequencies. Iteration-varying matrices are used to describe the system behavior, the zero-order hold implementation of the input signal and the approximation of the output signal at the nominal sample points.

A nominal system is defined, which is obtained if the drive frequency is iteration-invariant and equal to the nominal drive frequency, i.e., $f_{\alpha,j} = f_\alpha^n \forall j \in \mathbb{N}$. This nominal system in lifted form is given by

$$\bar{y}_j^n(\bar{\alpha}^n) = J^n(\bar{u}_s^n(\bar{\alpha}^n) + \bar{u}_j(\bar{\alpha}^n)) + d_\alpha(\bar{\alpha}^n) \quad (2.8)$$

$$\bar{e}_j^n(\bar{\alpha}^n) = \bar{y}_d(\bar{\alpha}^n) - \bar{y}_j^n(\bar{\alpha}^n), \quad (2.9)$$

with $J^n = h_0 I^{N \times N}$ for $\bar{\alpha}^n \in \mathbb{R}^{N \times 1}$. The error \bar{e}^n is defined as the deviation of the measured output \bar{y}^n from the reference \bar{y}_d . The following nominal update law is proposed:

$$\bar{u}_{j+1}^n(\bar{\alpha}^n) = Q \bar{u}_j^n(\bar{\alpha}^n) + \beta L \bar{e}_j^n(\bar{\alpha}^n). \quad (2.10)$$

Where the update filters Q and L and learning gain β are tuned to influence the convergence behavior. The system with iteration-varying drive frequencies is given by

$$\bar{y}_j(\bar{\alpha}^n) = R_j J_j S_j (\bar{u}_s(\bar{\alpha}^n) + \bar{u}_j(\bar{\alpha}^n)) + R_j \bar{d}_{\alpha,j}(\bar{\alpha}_j) \quad (2.11)$$

$$\bar{e}_j(\bar{\alpha}^n) = \bar{y}_d(\bar{\alpha}^n) - \bar{y}_j(\bar{\alpha}^n) \quad (2.12)$$

$$\bar{u}_{j+1}(\bar{\alpha}^n) = Q \bar{u}_j(\bar{\alpha}^n) + \beta L \bar{e}_j(\bar{\alpha}^n). \quad (2.13)$$

In this representation, the matrices S_j and R_j represent respectively the zero-order hold implementation of the input signal, and the sampling and estimation of the output at the nominal sample points $\bar{y}(\bar{\alpha}^n)$. A schematic representation is shown in Figure 2.3. S_j represents the implementation of the input \bar{u}_j , which is defined for $\alpha \in \bar{\alpha}^n$, using zero-order hold. J_j represents the system in lifted form. The measured output at the iteration-varying sample points is given by

$$\bar{y}_j(\bar{\alpha}_j) = J_j S_j (\bar{u}_s(\bar{\alpha}^n) + \bar{u}_j(\bar{\alpha}^n)) + \bar{d}_{\alpha,j}(\bar{\alpha}_j). \quad (2.14)$$

The sampling, approximation and interpolation that is used to obtain the output at the nominal sample points $\bar{y}_j(\bar{\alpha}^n)$ is described by R_j . The size of the matrices is iteration-varying and depends on the size of $\bar{\alpha}_j \in \mathbb{R}^{N_j \times 1}$. The matrices S_j and J_j are of the following form:

$$S_j = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & 0 \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & & \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & & \\ 0 & 0 & \dots & 1 \end{bmatrix} \in \mathbb{R}^{N_j \times N} \quad J_j = \begin{bmatrix} h_0 & 0 & \dots & 0 \\ 0 & h_0 & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \dots & h_0 \end{bmatrix} \in \mathbb{R}^{N_j \times N_j}$$

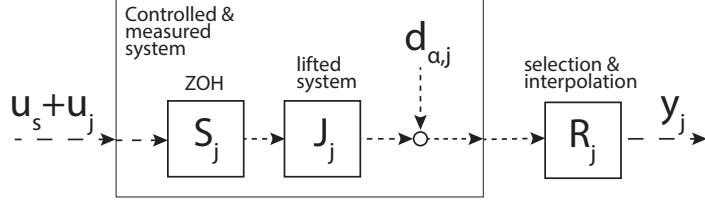


Figure 2.3: Schematic representations of the system with iteration-varying matrices.

For the matrix $R_j \in \mathbb{R}^{N \times N_j}$, different forms are possible, depending on whether interpolation or sampling is needed. Below, two examples of R_j are given for a situation where the measured output is interpolated for $N_j < N$ and where the measured output is sampled for $N_j > N$, respectively. Note that for arbitrary drive frequencies combinations of interpolation, approximation and data selection may occur, resulting in different configurations for R_j . The constant $m_j \in [0, 1]$ is used for interpolation and depends on the ratio between f_α^n and $f_{\alpha,j}$.

$$R_j = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 - m_j & m_j & 0 & \dots & 0 \\ 1 - 2m_j & 2m_j & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & 0 \\ m_j & 1 - m_j & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 1 - m_j & m_j & \dots & 0 \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & 0 & \dots & 1 - m_j \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}^{N \times N_j}$$

Remark. This representation using matrices is given in order to analyze the behavior of the ILC system. In the actual implementation, these matrices are not explicitly used. Instead of using S_j , the input signal is defined for all $\alpha \in [0, 2\pi)$ using zero-order hold for $\alpha \notin \bar{\alpha}^n$. The system output is a sampled signal of size N_j . In the case that $N_j < N$, linear interpolation is used to extend the output. Alternatively, if $N_j > N$, some of the data is removed to reduce the length of the output signal.

The system (2.11) is rewritten to a nominal system with iteration-varying disturbances that describe the deviation of the iteration-varying behavior from the nominal behavior. The system with iteration-varying behaviors interpreted as disturbances is described as

$$\bar{y}_j(\bar{\alpha}^n) = J^n(\bar{u}_s(\bar{\alpha}^n) + \bar{u}_j(\bar{\alpha}^n)) + \bar{d}_\alpha^n(\bar{\alpha}^n) + d_{u,j}(\bar{\alpha}^n) + \tilde{d}_{\alpha,j}(\bar{\alpha}^n), \quad (2.15)$$

or, in matrix form

$$\bar{y}_j(\bar{\alpha}^n) = \begin{bmatrix} h_0(\bar{u}_s(\bar{\alpha}^n(1)) + \bar{u}_j(\bar{\alpha}^n(1))) \\ h_0(\bar{u}_s(\bar{\alpha}^n(2)) + \bar{u}_j(\bar{\alpha}^n(2))) \\ \vdots \\ h_0(\bar{u}_s(\bar{\alpha}^n(N)) + \bar{u}_j(\bar{\alpha}^n(N))) \end{bmatrix} + \begin{bmatrix} \bar{d}_\alpha^n(\bar{\alpha}^n(1)) \\ \bar{d}_\alpha^n(\bar{\alpha}^n(2)) \\ \vdots \\ \bar{d}_\alpha^n(\bar{\alpha}^n(N)) \end{bmatrix} + d_{u,j}(\bar{\alpha}^n) + \tilde{d}_{\alpha,j}(\bar{\alpha}^n). \quad (2.16)$$

A schematic representation of the system in this form is shown in Figure 2.4. The iteration-varying behavior is represented by iteration-varying disturbances $\tilde{d}_{\alpha,j}$ and $d_{u,j}$. The disturbance $\tilde{d}_{\alpha,j}$ is given by the difference between the estimation based on the measured values $R_j \bar{d}_{\alpha,j}$ and the nominal values \bar{d}_α :

$$\tilde{d}_{\alpha,j}(\bar{\alpha}^n) = R_j \bar{d}_{\alpha,j}(\bar{\alpha}_j) - \bar{d}_\alpha(\bar{\alpha}^n). \quad (2.17)$$

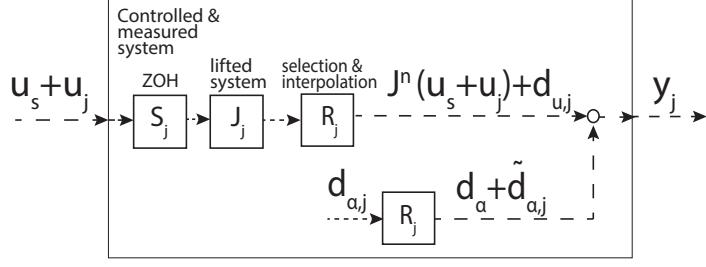


Figure 2.4: Schematic representations of the system with iteration-varying behaviors interpreted as disturbances.

Disturbance $\tilde{d}_{\alpha,j}$ is nonzero when the α -domain repeating disturbance at the nominal sample points $\bar{d}_\alpha(\bar{\alpha}^n)$ is estimated using R_j , either by linear interpolation or by using $\bar{d}_{\alpha,j}(\bar{\alpha}_j(m))$, $\bar{\alpha}_j(m) \in (\bar{\alpha}^n(k), \bar{\alpha}^n(k+1))$ as approximation.

The disturbance $d_{u,j}$ is caused by linear interpolation, when the response to \bar{u}_j for certain α is not measured and is estimated instead. The difference between the nominal response $J^n(\bar{u}_s + \bar{u}_j)$ and the estimated response based on measured values, $R_j J_j S_j(\bar{u}_s + \bar{u}_j)$, is given by

$$d_{u,j}(\bar{\alpha}^n) = R_j J_j S_j(\bar{u}_s(\bar{\alpha}^n) + \bar{u}_j(\bar{\alpha}^n)) - J^n(\bar{u}_s(\bar{\alpha}^n) + \bar{u}_j(\bar{\alpha}^n)). \quad (2.18)$$

2.4 Convergence of α -domain ILC

The convergence and stability of the α -domain ILC system with iteration-varying disturbances is analyzed by considering the convergence conditions for the nominal system behavior and the influence of the iteration-varying behaviors on these conditions. The norms that are used are defined as follows:

- The Euclidean norm of a vector is given by $\|a\|_2 = \sqrt{\sum_i |a_i|^2}$.
- The matrix induced 2-norm is defined as $\|A\|_{i2} = \bar{\sigma}(A)$, where $\bar{\sigma}(A)$ is the largest singular value of matrix A .

The nominal sampled error \bar{e}_j^n and input \bar{u}_j^n are defined as the error and input for the case where $f_{\alpha,j} = f_\alpha^n \forall j \in \mathbb{N}$. The measured error \tilde{e}_j and calculated input \tilde{u}_j deviate from their nominal counterparts by

$$\tilde{e}_j = \bar{e}_j^n - \bar{e}_j \quad (2.19)$$

$$\tilde{u}_j = \bar{u}_j^n - \bar{u}_j. \quad (2.20)$$

The iteration-varying disturbances in (2.17) and (2.18) are assumed to be bounded:

Assumption 2.5. *The difference between the estimated value of the α -domain disturbance and the actual disturbance is bounded, i.e., there exists some finite real constant ρ so that $\|\tilde{d}_{\alpha,j}\|_2 \leq \rho$.*

Assumption 2.6. *The difference between the estimated response to the input and the nominal response is bounded, i.e., there exists some finite real constant μ so that $\|d_{u,j}\|_2 \leq \mu$.*

Using these assumptions, conditions for convergence of the input and error to a bounded region are given in Theorem 2.7. A comparable analysis for general lifted ILC systems is presented in [22].

Theorem 2.7. Consider the system (2.15) with ILC update law (2.13) for which Assumptions 2.5 and 2.6 hold. If the condition

$$\|Q - \beta L J^n\|_{i2} \leq \gamma, \quad (2.21)$$

holds for some $\gamma \in [0, 1)$, then the input signal \bar{u}_j converges to a bounded region around the nominal input \bar{u}_j^n , such that \tilde{u}_j satisfies

$$\lim_{j \rightarrow \infty} \sup \|\tilde{u}_j\|_2 \leq \|\beta L\|_{i2} \frac{\mu + \rho}{1 - \gamma}. \quad (2.22)$$

Similarly, the error \bar{e}_j converges to a bounded region around the nominal error \bar{e}_j^n , such that \tilde{e}_j satisfies

$$\lim_{j \rightarrow \infty} \sup \|\tilde{e}_j\|_2 \leq \left(\|\beta L\|_{i2} \frac{\|J^n\|_{i2}}{1 - \gamma} + 1 \right) (\mu + \rho). \quad (2.23)$$

Remark. As a result of Theorem 2.7, the iteration-invariant filters Q and L can be designed so that the nominal system is monotonically convergent. For these filters, the system with iteration-varying disturbances converges to a bounded set.

Proof. The ILC system in the representation where iteration-varying behaviors caused by interpolation and approximation are interpreted as disturbances is given by

$$\bar{y}_j = J^n(\bar{u}_s + \bar{u}_j) + \bar{d}_\alpha^n + d_{u,j} + \tilde{d}_{\alpha,j} \quad (2.24)$$

$$\bar{e}_j = \bar{y}_d^n - \bar{y}_j \quad (2.25)$$

$$\bar{u}_{j+1} = Q\bar{u}_j + \beta L\bar{e}_j. \quad (2.26)$$

Substituting (2.19), (2.20) and (2.24) in (2.26) gives

$$\begin{aligned} \bar{u}_{j+1} &= Q\bar{u}_j + \beta L(\bar{y}_d^n - J^n\bar{u}_s - J^n\bar{u}_j - \bar{d}_\alpha^n - d_{u,j} - \tilde{d}_{\alpha,j}) \\ &= (Q - \beta L J^n)\bar{u}_j + \beta L(\bar{y}_d^n - J^n\bar{u}_s - \bar{d}_\alpha^n) - \beta L(d_{u,j} + \tilde{d}_{\alpha,j}) \\ &= (Q - \beta L J^n)\bar{u}_j^n - (Q - \beta L J^n)\tilde{u}_j + \beta L(\bar{y}_d^n - J^n\bar{u}_s - \bar{d}_\alpha^n) - \beta L(d_{u,j} + \tilde{d}_{\alpha,j}). \end{aligned} \quad (2.27)$$

Subtracting nominal update law (2.10) from (2.27) gives

$$\tilde{u}_{j+1} = (Q - \beta L J^n)\tilde{u}_j + \beta L(d_{u,j} + \tilde{d}_{\alpha,j}). \quad (2.28)$$

Using the triangle inequality, it holds that

$$\|\tilde{u}_{j+1}\|_2 \leq \|(Q - \beta L J^n)\tilde{u}_j\|_2 + \|\beta L(d_{u,j} + \tilde{d}_{\alpha,j})\|_2. \quad (2.29)$$

Since a matrix induced p -norm is compatible with its corresponding vector p -norm due to the multiplicative property, [23] pp. 534 Theorem A.3, it holds that

$$\|(Q - \beta L J^n)\tilde{u}_j\|_2 \leq \|Q - \beta L J^n\|_{i2} \|\tilde{u}_j\|_2. \quad (2.30)$$

Using (2.30), condition (2.21) and the bounds from Assumptions 2.5 and 2.6 are substituted in (2.29) to give

$$\|\tilde{u}_{j+1}\|_2 \leq \gamma \|\tilde{u}_j\|_2 + \|\beta L\|_{i2} (\|d_{u,j}\|_2 + \|\tilde{d}_{\alpha,j}\|_2) \quad (2.31)$$

$$\lim_{j \rightarrow \infty} \sup \|\tilde{u}_j\|_2 \leq \gamma \lim_{j \rightarrow \infty} \sup \|\tilde{u}_j\|_2 + \|\beta L\|_{i2} (\mu + \rho). \quad (2.32)$$

From which it follows that

$$\lim_{j \rightarrow \infty} \sup \|\tilde{u}_j\|_2 \leq \|\beta L\|_{i2} \frac{\mu + \rho}{1 - \gamma}. \quad (2.33)$$

Thus, \bar{u}_j converges to a bounded region around its nominal counterpart. Similarly, \bar{e}_j is given by

$$\bar{e}_j = \bar{y}_d^n - \bar{y}_j = \bar{y}_d^n - J^n\bar{u}_s - J^n\bar{u}_j - \bar{d}_\alpha^n - d_{u,j} - \tilde{d}_{\alpha,j} \quad (2.34)$$

$$= \bar{y}_d^n - J^n\bar{u}_s - J^n\bar{u}_j^n + J^n\tilde{u}_j - d_\alpha - d_{u,j} - \tilde{d}_{\alpha,j}. \quad (2.35)$$

Subtracting (2.9) from (2.35) gives

$$\tilde{e}_j = J^n \tilde{u}_j - d_{u,j} - \tilde{d}_{\alpha,j}, \quad (2.36)$$

which leads to the inequality

$$\|\tilde{e}_j\|_2 \leq \|J^n \tilde{u}_j\|_2 + \|d_{u,j}\|_2 + \|\tilde{d}_{\alpha,j}\|_2. \quad (2.37)$$

Substituting (2.33) and the bounds from Assumptions 2.5 and 2.6 in (2.37) gives

$$\lim_{j \rightarrow \infty} \sup \|\tilde{e}_j\|_2 \leq \left(\|\beta L\|_{i2} \frac{\|J^n\|_{i2}}{1-\gamma} + 1 \right) (\mu + \rho). \quad (2.38)$$

Thus, \bar{e}_j converges to a bounded region around its nominal counterpart. ■

2.5 Concluding remarks

A new ILC framework is developed that can compensate repeating disturbances in the α -domain, in which the sampling is iteration-varying and possibly non-equidistant. A disturbance-compensating input is defined and updated on an iteration-invariant set of sample points, and applied using zero-order hold. Convergence of the system to a bounded region can be shown under certain conditions.

The framework developed in this chapter has two disadvantages. First of all, the input signal that is designed in this way is bumpy due to measurement noise, even when the learning gain is low so that the input is learned over multiple iterations. Secondly, the amount of parameters to be learned is large, especially when compared to the minimum number of available samples. For a piezo-stepper actuator in a closed-loop implementation, the desired maximum drive frequency is estimated at 300 Hz which gives 33 samples on $\alpha \in [0, 2\pi]$. This requires either a small number of points on which the input signal is defined, or a large amount of interpolation and estimation, both of which are expected to lead to inaccurate results for drive frequencies that vary over a large range. In addition, implementing the disturbance-compensating input requires using a lookup table, which is computationally heavy and inefficient and can be problematic in an industrial implementation. Because of these disadvantages, an alternative approach using basis function parameterization is proposed in Chapter 3.

3 α -domain ILC with basis functions

In this chapter a framework for iterative learning control with basis functions in the α -domain is presented. The input signal is defined on a continuous domain by a set of basis functions multiplied with scaling parameters that are learned iteratively. This framework has several advantages compared to the framework of Chapter 2. First of all, the number of parameters that is learned is reduced significantly, since the number of parameters required to describe the input signal is lower than the number of sample points that is needed when the input signal is learned directly. However, all measured points are still taken into account in the update law and no data is thrown away. Secondly, the learned signal is defined continuously over the interval $\alpha \in [0, 2\pi)$ and it is smooth, whereas when the input is learned directly using the α -domain ILC of Chapter 2 the resulting signal is often bumpy.

In this chapter α -domain ILC with basis functions is presented. An overview of the use of basis functions in ILC, as well as a comparison with the proposed approach, is given in Section 3.1. In Section 3.2 a framework is proposed. The convergence criteria are presented in Section 3.3. In Section 3.4 an approach to fitting the error using basis functions is presented. Suitable basis functions for a piezo-stepper actuator are selected in Section 3.5. Lastly conclusions are given in Section 3.6.

3.1 Basis functions in ILC

Basis functions can be used to parameterize the input and output signals of a system to obtain certain properties. In general, the basis functions used in ILC are iteration-invariant and ILC is used to learn a set of parameters for these functions. In this section, an overview of some applications of basis functions in ILC is given, and these applications are compared to the proposed framework for α -domain ILC with basis functions.

Examples of applications of basis functions in ILC include the projection of the error on a basis to learn only the repetitive part of the error, and the parameterization of the input signal to achieve extrapolation possibilities. A framework for basis function ILC that encompasses manipulation of both the input and output signals is presented in [24]. In [25] a large set of orthonormal sine and cosine basis functions is used to parameterize the error signal, so that only the repetitive part is learned. Sine and cosine basis functions are used because they can model the part of the error that is to be compensated, namely a force ripple. In [26] the feedforward signal is chosen to be a function of the reference, using polynomial basis functions, to obtain an error that is invariant under the choice of the reference. ILC is used to learn the parameters for this feedforward parameterization using a norm-optimal approach. In [27] rational basis functions are used for feedforward parameterization, so that both poles and zeros can be optimized. These approaches all use a lifted framework with sampled signals with an iteration-invariant length.

Although there exist frameworks for basis function ILC that parameterize the input, the error, or both, the purpose and implementation of these parameterizations differs from that for α -domain ILC with basis functions. In α -domain ILC, both the input and the error are parameterized to obtain continuous descriptions that do not depend on the iteration-varying non-equidistant sampling, resulting in an iteration-invariant framework. Parameterization of the error signal is used in [25], where the output is projected on a basis to reduce the influence of iteration-varying behaviors like noise. There is a strong relation between the expected error dynamics and the chosen basis. The disturbance considered in α -domain ILC can be approximated by different types of basis functions, as investigated in Section 3.5. In [26, 27] input signals are parameterized to obtain extrapolation possibilities for the reference, which is not needed α -domain ILC.

The purpose of using basis functions in α -domain ILC is to obtain a continuous framework to which ILC can be applied, to deal with the iteration-varying non-equidistant sampling. Therefore, the lifted frameworks for basis function ILC found in literature, such as the approach presented in [24], which encompasses the approaches of [25, 26], are not applicable. If such a lifted framework is used for α -domain ILC, this causes the size of the basis function matrix, as well as that of the input and error vectors, to be iteration-varying. This is prevented by using a continuous framework instead.

3.2 Framework

Iterative learning control is used to learn a parameterized input signal using a continuous framework for α -domain ILC with basis functions. Since the framework requires a continuous description of the error signal, a fit of the sampled error is made. This continuous signal is then used to update the parameters for the input signal. In this section, the approach to obtaining a continuous system description is presented. Then, an update law is proposed.

3.2.1 Obtaining a continuous system description

Since the sampling in the α -domain is iteration-varying and non-equidistant for varying drive frequencies, the input and error signals are parameterized using basis functions to obtain continuous expressions to which ILC can be applied.

The displacement of the mover of a piezo-stepper actuator during a single step is described by

$$y(\alpha) = G(u_s(\alpha) + u(\alpha)) + d_\alpha(\alpha), \quad \alpha \in [0, 2\pi], \quad (3.1)$$

with the standard input to the shear elements $u_s(\alpha)$ to which the disturbance-compensating input $u(\alpha)$ is added. For α -domain ILC with basis functions, it is assumed that continuous signals are available. The continuous ILC system is given by

$$y_j(\alpha) = h_0(u_s(\alpha) + u_j(\alpha)) + d_\alpha(\alpha) \quad (3.2)$$

$$e_j(\alpha) = y_d(\alpha) - y_j(\alpha). \quad (3.3)$$

The piezo-stepper is assumed to behave as a gain, with piezo gain h_0 , disturbance-compensating input u_j , α -domain repeating disturbance d_α and reference y_d . The input is constructed using basis functions as follows:

$$u_j(\alpha) = \psi(\alpha)^T \theta_j^u. \quad (3.4)$$

Where the basis function vector ψ , containing M basis functions, and parameter vector θ_j^u are given by

$$\psi(\alpha) = [\psi_1(\alpha) \quad \psi_2(\alpha) \quad \dots \quad \psi_M(\alpha)]^T \in \mathbb{R}^{M \times 1} \quad (3.5)$$

$$\theta_j^u = [\theta_{1,j}^u \quad \theta_{2,j}^u \quad \dots \quad \theta_{M,j}^u]^T \in \mathbb{R}^{M \times 1} \quad (3.6)$$

such that

$$u_j(\alpha) = \psi(\alpha)^T \theta_j^u = \psi_1(\alpha) \theta_{1,j}^u + \psi_2(\alpha) \theta_{2,j}^u + \dots + \psi_M(\alpha) \theta_{M,j}^u. \quad (3.7)$$

The sampled error signal is parameterized using the same set of functions that forms the basis for the input u_j , since the system is assumed to behave as a gain. Each iteration, the error $\bar{e}_j(\bar{\alpha}_j)$ is sampled on the points $\bar{\alpha}_j \in \mathbb{R}^{N_j \times 1}$ according to (1.2). For the piezo-stepper system given by (3.2) and (3.3) it is assumed that the output and error are known for all α . Therefore, a continuous approximation of the error for all $\alpha \in [0, 2\pi]$ is made by fitting the sampled error signal using basis functions. An outline of the approach to α -domain ILC with basis functions for n iterations is shown in Algorithm 3.2.

It is assumed that the basis functions can be scaled to describe the α -domain disturbance, reference, and the standard shear input exactly, which can be satisfied by using suitable basis functions.

Assumption 3.1. *The basis functions can be scaled to describe the α -domain disturbance, reference and shear input exactly, i.e., $d_\alpha = \psi^T \theta^d$, $y_d = \psi^T \theta^{yd}$, and $u_s(\alpha) = \psi(\alpha)^T \theta^{us}$.*

Using (3.2), (3.3), (3.4) and Assumption 3.1, the error is written as

$$e_j(\alpha) = \psi^T(\alpha)(\theta^{yd} - h_0(\theta^{us} + \theta_j^u) - \theta^d) = \psi^T(\alpha)\theta_j^e. \quad (3.8)$$

The approach to finding the parameters θ_j^e in (3.8) based on the measured error signal is presented in Section 3.4.

Algorithm 3.2 Obtaining continuous signals for α -domain ILC with basis functions

Choose a basis for the input u (Section 3.5)

for $j = 1 : n$ **do**

 Perform an experiment with drive frequency $f_{\alpha,j}$
 N_j sample points given by $\bar{\alpha}_j \in \mathbb{R}^{N_j \times 1}$ according to (1.2)

for $i = 1 : N_j$ **do**

 Sampled input according to (3.4): $\bar{u}_j(\bar{\alpha}_j(i)) = \psi^T(\bar{\alpha}_j(i))\theta_j^u$

 Sampled output according to (3.2): $\bar{y}_j(\bar{\alpha}_j(i)) = h_0(\bar{u}^s(\bar{\alpha}_j(i)) + \bar{u}_j(\bar{\alpha}_j(i))) + \bar{d}_\alpha(\bar{\alpha}_j(i))$

 Sampled error according to (3.3): $\bar{e}_j(\bar{\alpha}_j(i)) = \bar{y}_d(\bar{\alpha}_j(i)) - \bar{y}_j(\bar{\alpha}_j(i))$

end for

 Obtain total sampled error $\bar{e}_j \in \mathbb{R}^{N_j \times 1}$

 Approximate continuous $e_j(\alpha) = \psi^T(\alpha)\theta_j^e$ using a least squares fit of the sampled error \bar{e}_j . Find best fitting parameters θ_j^e (Section 3.4)

 Use continuous $e_j(\alpha)$ in the ILC framework to obtain continuous $u_{j+1}(\alpha) = \psi^T(\alpha)\theta_{j+1}^u$ (Section 3.2)

end for

3.2.2 Update law

A continuous ILC update law is developed to determine the update parameters θ_{j+1}^u for iteration $j+1$, using the continuously defined input and error signals of iteration j . A cost function is used to determine the optimal parameters for the input signal. The input for iteration $j+1$ is given by

$$u_{j+1}(\alpha) = \psi^T(\alpha)\theta_{j+1}^u, \quad (3.9)$$

where the optimal vector of parameters θ_{j+1}^u is the minimizer of cost function \mathcal{J} , given by

$$\begin{aligned} \mathcal{J}(\theta_{j+1}^u) &= \int_0^{2\pi} W_e(\alpha)e_{j+1}(\alpha)^2 d\alpha + \int_0^{2\pi} W_u(\alpha)u_{j+1}(\alpha)^2 d\alpha \\ &\quad + \int_0^{2\pi} W_{\Delta u}(\alpha)(u_{j+1}(\alpha) - u_j(\alpha))^2 d\alpha. \end{aligned} \quad (3.10)$$

The weights $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta u}(\alpha)$ are positive functions that are tuned to obtain certain performance and robustness properties. $W_e(\alpha)$ influences the performance of the learning, $W_u(\alpha)$ influences the robustness against model uncertainty and $W_{\Delta u}(\alpha)$ influences the attenuation of iteration-varying disturbances.

The following theorem gives the optimal update for the vector of input parameters θ_j^u .

Theorem 3.3. Consider the system described by (3.2) and (3.3) for which the input $u_{j+1}(\alpha) = \psi^T(\alpha)\theta_{j+1}^u$ of iteration $j+1$ is constructed using any given set of linearly independent basis functions $\psi(\alpha)$. The cost function (3.10) with positive weight functions $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta u}(\alpha)$ leads to an optimal update of the parameters θ_{j+1}^u given by

$$\begin{aligned}\theta_{j+1}^u &= Q_\psi \theta_j^u + L_\psi \theta_j^e, \quad \text{with} \\ Q_\psi &= \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} \\ &\quad \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \\ L_\psi &= \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} \int_0^{2\pi} W_e(\alpha) h_0 \psi(\alpha) \psi^T(\alpha) d\alpha.\end{aligned}\tag{3.11}$$

For scalar weights $W_e(\alpha) = W_e$, $W_u(\alpha) = W_u$ and $W_{\Delta u}(\alpha) = W_{\Delta u}$ for all $\alpha \in [0, 2\pi]$ this reduces to

$$\begin{aligned}\theta_{j+1}^u &= Q_\psi \theta_j^u + L_\psi \theta_j^e, \quad \text{with} \\ Q_\psi &= \frac{h_0^2 W_e + W_{\Delta u}}{h_0^2 W_e + W_u + W_{\Delta u}} I \\ L_\psi &= \frac{W_e h_0}{h_0^2 W_e + W_u + W_{\Delta u}} I.\end{aligned}\tag{3.12}$$

Proof. Using (3.2), (3.3), (3.4) and (3.8) the error for iteration $j+1$ is written as a function of the error for iteration j as

$$\begin{aligned}e_{j+1}(\alpha) &= y_d(\alpha) - y_{j+1}(\alpha) = y_d(\alpha) - h_0(u_s(\alpha) + u_{j+1}(\alpha)) - d_\alpha(\alpha) \\ &= e_j(\alpha) - h_0(u_{j+1}(\alpha) - u_j(\alpha)) = e_j(\alpha) - h_0 \psi^T(\alpha)(\theta_{j+1}^u - \theta_j^u).\end{aligned}\tag{3.13}$$

Substituting (3.4) and (3.13) in (3.10) gives

$$\begin{aligned}\mathcal{J}(\theta_{j+1}^u) &= \int_0^{2\pi} W_e(\alpha) (e_j(\alpha) - h_0 \psi^T(\alpha)(\theta_{j+1}^u - \theta_j^u))^2 d\alpha + \int_0^{2\pi} W_u(\alpha) (u_{j+1}(\alpha))^2 d\alpha + \\ &\quad \int_0^{2\pi} W_{\Delta u}(\alpha) (u_{j+1}(\alpha) - u_j(\alpha))^2 d\alpha.\end{aligned}\tag{3.14}$$

Which is rewritten as follows:

$$\begin{aligned}\mathcal{J}(\theta_{j+1}^u) &= \int_0^{2\pi} W_e(\alpha) e_j^2(\alpha) d\alpha - 2h_0 \int_0^{2\pi} W_e(\alpha) e_j(\alpha) \psi^T(\alpha)(\theta_{j+1}^u - \theta_j^u) d\alpha + \\ &\quad h_0^2 \int_0^{2\pi} W_e(\alpha) ((\psi^T(\alpha)\theta_{j+1}^u)^2) d\alpha - 2h_0^2 \int_0^{2\pi} W_e(\alpha) \psi^T(\alpha) \theta_{j+1}^u \psi^T(\alpha) \theta_j^u d\alpha + \\ &\quad h_0^2 \int_0^{2\pi} W_e(\alpha) (\psi^T \theta_j^u)^2 d\alpha + \int_0^{2\pi} W_u(\alpha) (\psi^T(\alpha)(\theta_{j+1}^u))^2 d\alpha + \int_0^{2\pi} W_{\Delta u}(\alpha) ((\psi^T(\alpha)\theta_{j+1}^u)^2) d\alpha - \\ &\quad \int_0^{2\pi} 2W_{\Delta u}(\alpha) \psi^T(\alpha) \theta_{j+1}^u \psi^T(\alpha) \theta_j^u d\alpha + \int_0^{2\pi} W_{\Delta u}(\alpha) (\psi^T \theta_j^u)^2 d\alpha.\end{aligned}\tag{3.15}$$

This expression contains several multiplications of basis functions. These multiplications are given by

$$(\psi^T \theta_j^u)^2 = \sum_{k=1}^M \left(\psi_k \theta_{k,j}^u \sum_{l=1}^M \psi_l \theta_{l,j}^u \right), \quad \psi^T \theta_{j+1}^u \psi^T \theta_j^u = \sum_{k=1}^M \left(\psi_k \theta_{k,j+1}^u \sum_{l=1}^M \psi_l \theta_{l,j}^u \right). \tag{3.16}$$

The first order necessary condition for optimality is used to find the stationary points of \mathcal{J} :

$$\frac{\partial \mathcal{J}(\theta_{j+1}^u)}{\partial \theta_{j+1}^u} = 0. \quad (3.17)$$

Using (3.16), (3.17) is written as

$$\begin{aligned} \frac{\partial \mathcal{J}(\theta_{j+1}^u)}{\partial \theta_{j+1}^u} &= \frac{\partial}{\partial \theta_{j+1}^u} \left(-2h_0 \int_0^{2\pi} W_e(\alpha) e_j(\alpha) \psi^T(\alpha) d\alpha (\theta_{j+1}^u) + \right. \\ &\quad \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) (\psi^T(\alpha) \theta_{j+1}^u)^2 d\alpha \\ &\quad \left. - 2 \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi^T(\alpha) \theta_{j+1}^u \psi^T(\alpha) \theta_j^u d\alpha \right). \end{aligned} \quad (3.18)$$

The derivative of a function to a vector is given by

$$\frac{\partial \mathcal{J}(\theta_{j+1}^u)}{\partial \theta_{j+1}^u} = \begin{bmatrix} \frac{\partial \mathcal{J}(\theta_{j+1}^u)}{\partial \theta_{1,j+1}^u} \\ \frac{\partial \mathcal{J}(\theta_{j+1}^u)}{\partial \theta_{2,j+1}^u} \\ \vdots \\ \frac{\partial \mathcal{J}(\theta_{j+1}^u)}{\partial \theta_{M,j+1}^u} \end{bmatrix}. \quad (3.19)$$

And the integral of a vector of functions is

$$\int_0^{2\pi} e_j(\alpha) \psi^T(\alpha) d\alpha = \begin{bmatrix} \int_0^{2\pi} e_j(\alpha) \psi_1(\alpha) d\alpha & \int_0^{2\pi} e_j(\alpha) \psi_2(\alpha) d\alpha & \dots & \int_0^{2\pi} e_j(\alpha) \psi_M(\alpha) d\alpha \end{bmatrix}. \quad (3.20)$$

The derivative of \mathcal{J} to a single parameter is given by

$$\begin{aligned} \frac{\partial \mathcal{J}(\theta_{k,j+1}^u)}{\partial \theta_{k,j+1}^u} &= -2h_0 \int_0^{2\pi} W_e(\alpha) e_j(\alpha) \psi_k(\alpha) d\alpha + \\ &\quad 2 \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi_k(\alpha) \psi^T(\alpha) d\alpha \theta_{j+1}^u - \\ &\quad 2 \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi_k(\alpha) \psi^T(\alpha) d\alpha \theta_j^u. \end{aligned} \quad (3.21)$$

Using this expression, the complete derivative is written as

$$\begin{aligned} \frac{\partial \mathcal{J}(\theta_{j+1}^u)}{\partial \theta_{j+1}^u} &= -2h_0 \int_0^{2\pi} W_e(\alpha) e_j(\alpha) \psi(\alpha) d\alpha + \\ &\quad 2 \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_{j+1}^u - \\ &\quad 2 \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_j^u = 0. \end{aligned} \quad (3.22)$$

Which is rewritten to the standard matrix form for a system of linear equations $Ax = b$:

$$\begin{aligned} \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_{j+1}^u &= \\ \int_0^{2\pi} h_0 W_e(\alpha) e_j(\alpha) \psi(\alpha) d\alpha + \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_j^u. \end{aligned} \quad (3.23)$$

Substituting the error parameterization of (3.8) results in the expression

$$\begin{aligned} \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_{j+1}^u &= \\ \int_0^{2\pi} h_0 W_e(\alpha) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_j^e + \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_j^u, \end{aligned} \quad (3.24)$$

such that

$$\theta_{j+1}^u = \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} \left(\int_0^{2\pi} h_0 W_e(\alpha) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_j^e + \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \theta_j^u \right). \quad (3.25)$$

The combination of weighting functions is written as

$$W(\alpha) = (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)). \quad (3.26)$$

To solve (3.25) uniquely, the matrix with basis function integrals needs to be invertible. This symmetric matrix is given by

$$\begin{aligned} \int_0^{2\pi} W(\alpha) \psi(\alpha) \psi^T(\alpha) d\alpha &= \\ \begin{bmatrix} \int_0^{2\pi} W(\alpha)(\psi_1(\alpha))^2 d\alpha & \int_0^{2\pi} W(\alpha)\psi_1(\alpha)\psi_2(\alpha) d\alpha & \dots & \int_0^{2\pi} W(\alpha)\psi_1(\alpha)\psi_M(\alpha) d\alpha \\ \int_0^{2\pi} W(\alpha)\psi_2(\alpha)\psi_1(\alpha) d\alpha & \int_0^{2\pi} W(\alpha)(\psi_2(\alpha))^2 d\alpha & \dots & \int_0^{2\pi} W(\alpha)\psi_2(\alpha)\psi_M(\alpha) d\alpha \\ \vdots & \vdots & & \vdots \\ \int_0^{2\pi} W(\alpha)\psi_M(\alpha)\psi_1(\alpha) d\alpha & \int_0^{2\pi} W(\alpha)\psi_M(\alpha)\psi_2(\alpha) d\alpha & \dots & \int_0^{2\pi} W(\alpha)(\psi_M(\alpha))^2 d\alpha \end{bmatrix}, \end{aligned} \quad (3.27)$$

which is the Gramian matrix for a set of functions $\sqrt{W(\alpha)}\psi_k(\alpha)$, $k \in [1, 2, \dots, M]$. According to Gram's criterion for linearly independent functions, this matrix is positive definite if and only if the functions $\sqrt{W(\alpha)}\psi_k(\alpha)$ are linearly independent [28], pp.246-248. The chosen basis functions $\psi_k(\alpha)$ are linearly independent. Therefore, the set of functions $\sqrt{W(\alpha)}\psi_k(\alpha)$ is linearly independent if $\sqrt{W(\alpha)} > 0 \forall \alpha \in [0, 2\pi]$. If one of the weight functions is positive for all $\alpha \in [0, 2\pi]$ and the others are non-negative, this condition is met and a unique optimal θ_{j+1}^u is found.

Equation (3.25) is written as an update law of the form

$$\theta_{j+1}^u = Q_\psi \theta_j^u + L_\psi \theta_j^e, \quad \text{with} \quad (3.28)$$

$$\begin{aligned} Q_\psi &= \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \\ L_\psi &= \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} \int_0^{2\pi} W_e(\alpha) h_0 \psi(\alpha) \psi^T(\alpha) d\alpha. \end{aligned}$$

When the weights $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta,u}(\alpha)$ are chosen as scalars, i.e., $W_e(\alpha) = W_e$, $W_u(\alpha) = W_u$ and $W_{\Delta,u}(\alpha) = W_{\Delta,u}$ for all $\alpha \in [0, 2\pi]$, the filters are written as

$$\begin{aligned} Q_\psi &= \left((h_0^2 W_e + W_u + W_{\Delta u}) \int_0^{2\pi} \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} (h_0^2 W_e + W_{\Delta u}) \int_0^{2\pi} \psi(\alpha) \psi^T(\alpha) d\alpha \quad (3.29) \\ L_\psi &= \left((h_0^2 W_e + W_u + W_{\Delta u}) \int_0^{2\pi} \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} W_e(\alpha) h_0 \int_0^{2\pi} \psi(\alpha) \psi^T(\alpha) d\alpha, \end{aligned}$$

such that the update law is reduced to

$$\begin{aligned} \theta_{j+1}^u &= Q_\psi \theta_j^u + L_\psi \theta_j^e, \quad \text{with} \quad (3.30) \\ Q_\psi &= \frac{h_0^2 W_e + W_{\Delta u}}{h_0^2 W_e + W_u + W_{\Delta u}} I \\ L_\psi &= \frac{W_e h_0}{h_0^2 W_e + W_u + W_{\Delta u}} I. \end{aligned}$$

■

3.3 Convergence of α -domain ILC with basis functions

To avoid large learning transients and ensure convergence to a unique input signal, the ILC system needs to be monotonically convergent. A sequence of parameter vectors $\{\theta_j^u\}_{j \in \mathbb{N}}$ is monotonically convergent in the 2-norm towards a fixed parameter vector θ^{u*} if

$$\|\theta^{u*} - \theta_{j+1}^u\|_2 \leq \gamma \|\theta^{u*} - \theta_j^u\|_2 \quad \forall j \in \mathbb{N}, \quad (3.31)$$

with $0 \leq \gamma < 1$. For an update law of the form

$$\theta_{j+1}^u = Q_\psi \theta_j^u + L_\psi \theta_j^e, \quad (3.32)$$

the sequence of parameter vectors is monotonically convergent in the 2-norm if the following condition is met [10].

$$\bar{\sigma}(Q_\psi - L_\psi h_0) < 1. \quad (3.33)$$

Conditions for monotonic convergence of α -domain ILC with basis functions are given by the following theorem.

Theorem 3.4. Consider the system described in (3.2) and (3.3) with input (3.4). For the update law (3.11), with non-negative weight functions $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta,u}(\alpha)$ of which at least $W_e(\alpha)$ or $W_u(\alpha)$ is positive for all $\alpha \in [0, 2\pi]$, and linearly independent basis functions ψ , the sequence of parameter vectors $\{\theta_j^u\}_{j \in \mathbb{N}}$ is monotonically convergent in the 2-norm towards a fixed parameter vector θ^{u*} .

Proof. The system behavior is described by the scalar h_0 and the basis functions are scaled linearly with the parameters. Therefore the update law for the input signal parameters is written as

$$\theta_{j+1}^u = Q_\psi \theta_j^u + L_\psi \theta_j^e = (Q_\psi - L_\psi h_0) \theta_j^u + L_\psi (\theta^{yd} - h_0 \theta^{us} - \theta^d). \quad (3.34)$$

The sequence of parameter vectors $\{\theta_j^u\}_{j \in \mathbb{N}}$ is monotonically convergent in the 2-norm towards a fixed parameter vector θ^{u*} for $\bar{\sigma}(Q_\psi - L_\psi h_0) < 1$. It holds that

$$Q_\psi - L_\psi h_0 = \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta,u}(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} \int_0^{2\pi} W_{\Delta,u}(\alpha) \psi(\alpha) \psi^T(\alpha) d\alpha, \quad (3.35)$$

and $\bar{\sigma}((A + B)^{-1}B) < 1$ for $A > 0$, $B \geq 0$.

Therefore, the sequence of parameter vectors is monotonically convergent if the matrix $\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha$ is positive definite. This is the Gramian matrix for the set of functions $\sqrt{h_0^2 W_e(\alpha) + W_u(\alpha)} \psi_k(\alpha)$, $k \in [1, 2, \dots, M]$, which is positive definite if and only if the functions in the set are linearly independent. Since the basis functions $\psi_k(\alpha)$ are linearly independent and since the weighting functions $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta,u}(\alpha)$ are non-negative, the Gramian is positive definite if $W_e(\alpha)$ or $W_u(\alpha)$ is positive for all $\alpha \in [0, 2\pi]$. Then, the condition $\bar{\sigma}(Q_\psi - L_\psi h_0) \leq \gamma < 1$ holds and the sequence of parameter vectors $\{\theta_j^u\}_{j \in \mathbb{N}}$ is monotonically convergent in the 2-norm towards a fixed parameter vector θ^{u*} . ■

The fixed parameter vector that is obtained for $j \rightarrow \infty$ is given by:

$$\begin{aligned} \theta^{u*} &= L_\psi (\theta^{yd} - h_0 \theta^{us} - \theta^d) (I - Q_\psi + L_\psi h_0)^{-1} \\ &= \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} \int_0^{2\pi} h_0 W_e(\alpha) \psi(\alpha) \psi^T(\alpha) d\alpha (\theta^{yd} - h_0 \theta^{us} - \theta^d) \end{aligned} \quad (3.36)$$

The error θ_∞^e that is obtained for $j \rightarrow \infty$ is given by:

$$\begin{aligned} \theta^{e*} &= \theta^{yd} - h_0 \theta^{us} - \theta^d - h_0 \theta^{u*} \\ &= \left(I - \int_0^{2\pi} h_0^2 W_e(\alpha) \psi(\alpha) \psi^T(\alpha) d\alpha \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha)) \psi(\alpha) \psi^T(\alpha) d\alpha \right)^{-1} \right) (\theta^{yd} - h_0 \theta^{us} - \theta^d) \end{aligned} \quad (3.37)$$

When the weight functions are chosen to be constant scalars, these expressions reduce to:

$$\theta^{u*} = \frac{h_0 W_e}{h_0^2 W_e + W_u} I (\theta^{yd} - h_0 \theta^{us} - \theta^d) \quad (3.38)$$

$$\theta^{e*} = \left(I - \frac{h_0^2 W_e}{h_0^2 W_e + W_u} I \right) (\theta^{yd} - h_0 \theta^{us} - \theta^d) \quad (3.39)$$

Remark. It is possible to choose the weights $W_u(\alpha)$ and $W_{\Delta u}(\alpha)$ equal to zero. When $W_u(\alpha) = 0 \forall \alpha \in [0, 2\pi]$, $Q_\psi = I$. When both $W_u(\alpha) = 0 \forall \alpha \in [0, 2\pi]$ and $W_{\Delta u}(\alpha) = 0 \forall \alpha \in [0, 2\pi]$, $L_\psi = h_0^{-1} I$. In addition, when $W_u(\alpha) = 0 \forall \alpha \in [0, 2\pi]$ the converged error and parameter vector are given by respectively $\theta^{e*} = 0$ and $\theta^{u*} = h_0^{-1} (\theta^{yd} - h_0 \theta^{us} - \theta^d)$.

Remark. The sequence of parameter vectors of α -domain ILC system with basis functions is monotonically convergent if the quality of the error fit is not influenced by the number of samples. It is assumed that the error is described exactly by the weighted basis functions, $e_j = \psi \theta_j^e$, so that the difference between fit and measured signal is negligible for any number of samples larger than the number of basis functions (Assumption 3.1). Under these assumptions, the quality of the linear least squares fit described in Section 3.4 is not influenced by the number of samples provided that a unique solution exists. Therefore, there is no difference in quality of the fit between iterations which could influence the convergence.

3.4 Error fitting

In this section, the sampled error is parameterized to obtain a continuous description that is used in the framework of Section 3.2. The same basis functions that form the basis for the input signal are used, since the system is assumed to behave as a gain. Therefore, functions that give a good description of the error signal are a suitable basis for the input signal. The selection of these functions is described in Section 3.5. An outline of the process of obtaining continuous signals is shown in Algorithm 3.2 in Section 3.2.

To obtain a continuous description of the error $e_j(\alpha) = \psi^T(\alpha) \theta_j^e$ in (3.3), the measured data is fitted on the basis functions ψ using a least squares fit, so that for each basis function one optimal scaling parameter is found. The error $\bar{e}_j(\bar{\alpha}_j)$ is a sampled signal, the length of which is determined by the time-domain sample frequency and the drive frequency. The points at which the error is sampled for iteration j are given by $\bar{\alpha}_j \in \mathbb{R}^{N_j \times 1}$ according to (1.2). The number of parameters, and therefore the number of basis functions M , is limited by the length of the error signal.

The optimal vector of parameters $\theta_j^e \in \mathbb{R}^{M \times 1}$ that provides the best fit for \bar{e}_j is determined using a least squares fit. The following cost function is minimized:

$$\mathcal{J}_e = \sum_{i=1}^{N_j} (\bar{e}_j(i) - \psi^T(\bar{\alpha}_j(i)) \theta_j^e)^2 \quad (3.40)$$

The following theorem gives the optimal parameter vector θ_j^e for the fit of the sampled error.

Theorem 3.5. Consider the sampled error vector \bar{e}_j of the system described by (3.2) and (3.3) for which a fit over a continuous domain is given by $e_j = \psi(\alpha)\theta_j^e$. If the basis functions in $\psi(\alpha)$, as well as their sampled counterparts in $\bar{\psi}_j$, are linearly independent, then the parameter vector that gives the unique least-squares optimal fit in terms of the cost function (3.40) for \bar{e}_j is given by:

$$\theta_j^e = (\bar{\psi}_j \bar{\psi}_j^T)^{-1} \bar{\psi}_j \bar{e}_j. \quad (3.41)$$

Proof. The sampled basis function matrix $\bar{\psi}_j \in \mathbb{R}^{M \times N_j}$ contains the values of the M basis functions on the N_j sample points $\bar{\alpha}_j$ of iteration j and is given by

$$\bar{\psi}_j = [\bar{\psi}_1(\bar{\alpha}_j) \quad \bar{\psi}_2(\bar{\alpha}_j) \quad \dots \quad \bar{\psi}_M(\bar{\alpha}_j)]^T = \begin{bmatrix} \psi_1(\bar{\alpha}_j(1)) & \psi_2(\bar{\alpha}_j(1)) & \dots & \psi_M(\bar{\alpha}_j(1)) \\ \psi_1(\bar{\alpha}_j(2)) & \psi_2(\bar{\alpha}_j(2)) & \dots & \psi_M(\bar{\alpha}_j(2)) \\ \vdots & \vdots & & \vdots \\ \psi_1(\bar{\alpha}_j(N)) & \psi_2(\bar{\alpha}_j(N)) & \dots & \psi_M(\bar{\alpha}_j(N)) \end{bmatrix}^T. \quad (3.42)$$

Using this representation, the cost function is rewritten to

$$\mathcal{J}_e = (\bar{e}_j - \bar{\psi}_j^T \theta_j^e)^T (\bar{e}_j - \bar{\psi}_j^T \theta_j^e). \quad (3.43)$$

Using the first order necessary condition for optimality, the following solution is found for the optimal parameters θ_j^e :

$$\frac{\partial \mathcal{J}_e}{\partial \theta_j^e} = -2\bar{\psi}_j \bar{e}_j + 2\bar{\psi}_j \bar{\psi}_j^T \theta_j^e = 0 \quad (3.44)$$

$$\theta_j^e = (\bar{\psi}_j \bar{\psi}_j^T)^{-1} \bar{\psi}_j \bar{e}_j, \quad (3.45)$$

with

$$\bar{\psi}_j \bar{\psi}_j^T = \begin{bmatrix} \psi_1(\bar{\alpha}_j)^2 & \psi_1(\bar{\alpha}_j)\psi_2(\bar{\alpha}_j) & \dots & \psi_1(\bar{\alpha}_j)\psi_M(\bar{\alpha}_j) \\ \psi_2(\bar{\alpha}_j)\psi_1(\bar{\alpha}_j) & \psi_2(\bar{\alpha}_j)^2 & \dots & \psi_2(\bar{\alpha}_j)\psi_M(\bar{\alpha}_j) \\ \vdots & \vdots & & \vdots \\ \psi_M(\bar{\alpha}_j)\psi_1(\bar{\alpha}_j) & \psi_M(\bar{\alpha}_j)\psi_2(\bar{\alpha}_j) & \dots & \psi_M(\bar{\alpha}_j)^2 \end{bmatrix}, \quad (3.46)$$

which is the Gramian matrix of the sampled representation of the basis functions $\bar{\psi}_j^T$. The Gramian is always positive semi-definite, and if the vectors $\psi_k(\bar{\alpha}_j)$, $k = 1, 2, \dots, M$ in $\bar{\psi}_j^T$ are linearly independent it is positive definite and therefore invertible [28], pp.246-248. Then, the unique stationary point that minimizes the convex cost function and gives an optimal fit of the measured error is found to be

$$\theta_j^e = (\bar{\psi}_j \bar{\psi}_j^T)^{-1} \bar{\psi}_j \bar{e}_j. \quad (3.47)$$

■

Remark. The number of parameters to be fitted is always chosen lower than the minimum number of samples that is obtained during an iteration to prevent linear dependency of the sampled basis function vectors in $\bar{\psi}_j^T$ and singularity of its Gramian. For a singular Gramian, there are infinitely many solutions for the fit. These solutions give a reasonable fit on the sampled data points, but outside of these points no guarantees can be given for the quality of the fit. Implementing such a fit in a later iteration where the number of samples is higher could be problematic. When the sampled basis functions are linearly independent and there are more data points than parameters, a unique least squares optimal fit exists.

3.5 Basis function selection

In this section suitable basis functions for α -domain ILC are selected. The basis functions that are used to describe the input and error should be linearly independent to ensure that a unique solution is found for the cost functions. Possible choices for a set of linearly independent basis functions include polynomial basis functions, Fourier series and radial basis functions. The suitability of these functions depends on the system. In this section, the considered system is the piezo-stepper actuator presented in Chapter 1.

Basis functions for iterative learning control are in general chosen based on prior knowledge regarding the origin and shape of the disturbance or reference. In [25] the output is projected on sine-based basis functions, since the disturbance is a force ripple which can be modeled by this type of functions. When basis functions are used to define the feedforward input as a function of the reference, see e.g. [26], typical choices for basis functions are derivatives of the reference, i.e., the velocity, acceleration, jerk and so on. In α -domain ILC, basis functions are used to approximate continuous descriptions of the sampled errors and to parameterize the input signals. Since the system is assumed to behave as a gain, basis functions that approximate the shape of the error signal well are also suited to parameterize the input signal, therefore only a single set of basis functions is needed. These functions are selected based on how well they approximate the shape of the error signal.

Several types of basis functions are considered, since the error signal contains both curved and straight regions so that there is no obviously suitable choice of basis functions. The different types of functions are compared using an experimentally obtained error signal. First, the suitability of radial basis functions is explained. Secondly, the alternatives, Fourier series and polynomial functions, are explored. Then, the selection of suitable basis functions is described and lastly the linear independence of the sampled basis function vectors is investigated.

Remark. The maximum number of basis functions that is considered is 33, since the maximum desired drive frequency of the piezo-stepper actuator is estimated at 300 Hz with a time-domain sample frequency of 10 kHz. This results in 33 sample points on the domain $\alpha \in [0, 2\pi]$.

3.5.1 Radial basis functions

The first type of functions that is considered for the parameterization of the input and error signals for a piezo-stepper actuator is radial basis functions. A radial basis function is a function for which the value only depends on the distance from a center point c : $\psi(x, c) = \psi(\|x - c\|)$. A set of M basis functions is given by $\psi_k(x) = \psi(\|x - c_k\|)$, $k \in [1, 2, \dots, M]$, where each ψ_k is called a kernel. There are several standard radial basis functions for which the kernels are linearly independent, for example those given in Table 3.1, where $r = \|x - c\|$ and η is a shape tuning parameter [29], pp. 1-4.

Radial basis functions are a type of kernel functions that is used in the fields of machine learning, function recovery from scattered data and meshless solving of partial differential equations, see e.g. [29–31]. A kernel is a function that describes a measure of similarity between two objects, which is typically symmetric and non-negative [30], ch. 14. Recently, kernel techniques from the field of machine learning have been applied in the field of system identification and control. This results in kernel-based regularized identification techniques that are used to identify finite impulse response models of causal systems [32], where kernels are used to impose stability aspects. In [33], kernels are used to encode non-causality, by using non-causal rational orthonormal basis functions to identify finite impulse response models of inverse systems for feedforward control.

In α -domain ILC the purpose of the basis functions is to provide a continuous approximation of the part of the sampled error that is caused by the α -domain repeating disturbance, for which the sampling may be non-equidistant. Temporal-domain system dynamics are not taken into account, and there is no need for imposing system properties such as stability or non-causality. Therefore, the role of basis functions and kernel methods in α -domain ILC shows more similarities to kernel methods regarding function recovery of scattered data and meshless function approximation than to impulse response estimation related methods.

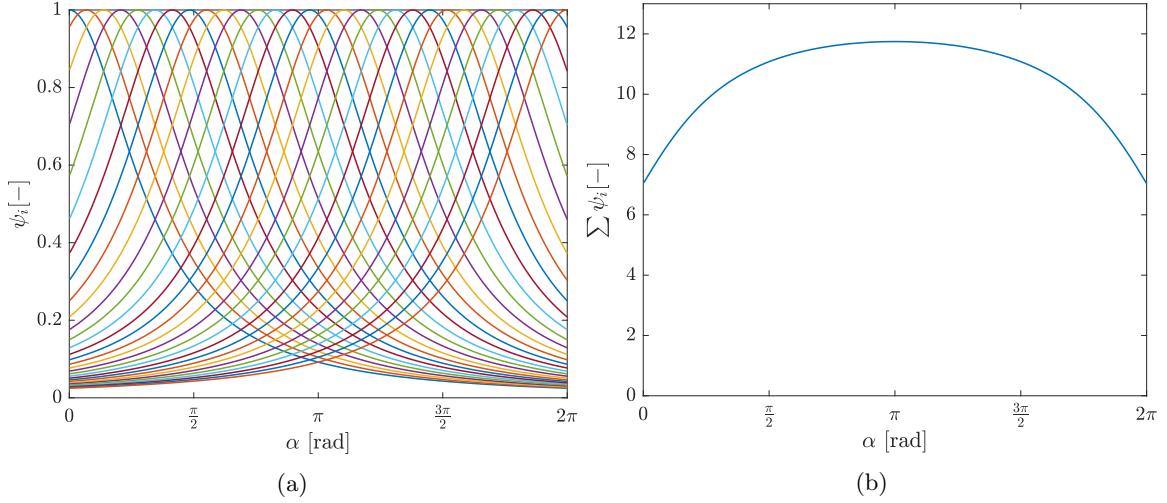


Figure 3.1: Inverse quadratic radial basis functions (a) and their summation (b).

Radial basis functions are expected to be suitable basis functions for α -domain ILC, since they are suited for approximation of functions of any shape, even when the data are scattered in their domain [34], ch. 1. Therefore, they are expected to give good approximations of a non-equidistantly sampled error in α -domain ILC. Since the error is only defined on the domain $\alpha \in [0, 2\pi]$ a limited amount of radial basis functions with center points divided over $\alpha \in [0, 2\pi]$ can be used for approximation.

To analyze the suitability of radial basis functions for α -domain ILC, several fits of the error are made. A set of 30 inverse quadratic radial basis functions with equidistant center points in the domain $\alpha \in [0, 2\pi]$ and shape factor $\eta = 1$, shown in Figure 3.1, is used to fit an error signal with 1000 samples. The original signal and the fit are shown in Figure 3.2a. In Figure 3.2b, the difference between the error signal and the fit is plotted. The root mean square of this difference is approximately 1.5×10^{-9} .

Reducing the amount of samples used for the basis function fit causes a small increase in the error, as shown in Figure 3.3 where the same 30 Gaussian radial basis functions are fitted using a down-sampled error signal of 50 equidistant samples. The difference between the fit and the error signal is determined at the original 1000 points and has a RMS value of approximately 1.6×10^{-9} .

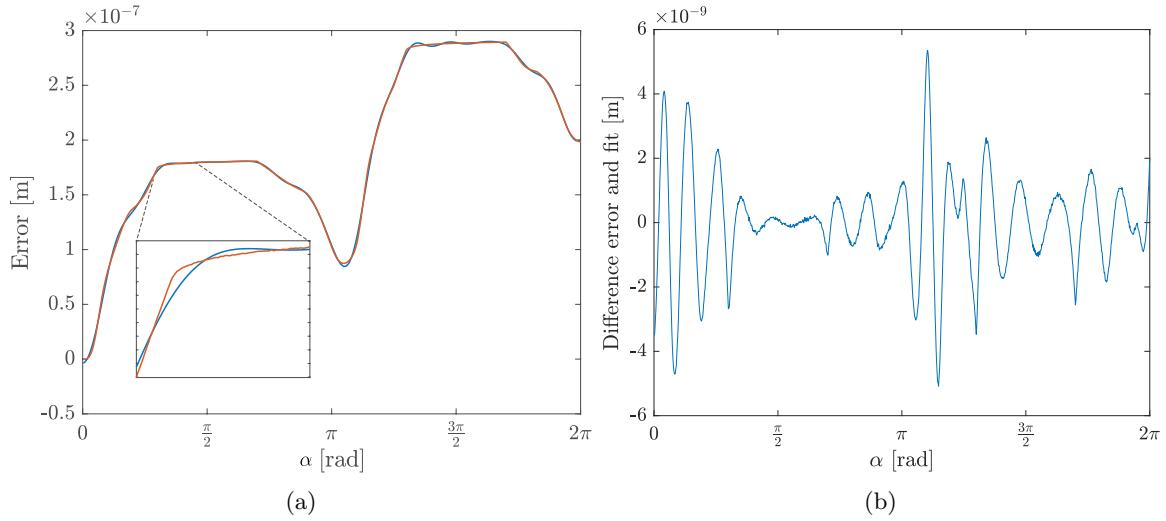
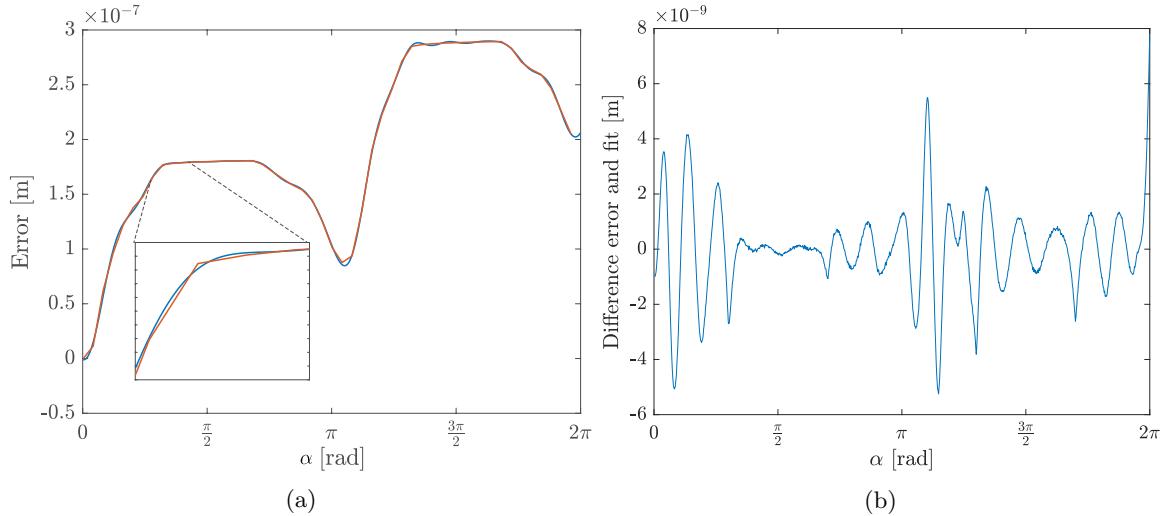
The difference between the fit and the measured error is negligibly small, and the number of sample points barely influences the quality of the fit. The differences between the two fits for $N = 1000$ and $N = 50$ are small, as shown in Figure 3.4. The largest differences occur around the first and last data point, since the impact of missing sample points is the largest there. The RMS value of the difference is around 5.4×10^{-10} .

The other types of radial basis functions that are considered; Gaussian, multiquadric and inverse multiquadric radial basis functions, give comparable results. The RMS value for the difference between the fit and the signal for $N = 1000$ is between 1.4×10^{-9} and 1.6×10^{-9} for all four types of radial basis functions. Therefore, the radial basis functions are selected based on the complexity of the computation of $u_j(\alpha) = \psi^T(\alpha)\theta_j^u$, as this is relevant for implementation in a closed-loop system.

Remark. For radial basis functions, the continuous kernels are linearly independent, and when the centers c are chosen equal to the sampled data points x the sampled basis function vectors are linearly independent as well. In most applications of radial basis functions either the center points or the data points, or even both, can be chosen freely. For this application, the center points are set while the sample points are iteration-varying. Linear independence of the basis function vector for this situation is investigated in Subsection 3.5.4.

Table 3.1: Linearly independent radial basis functions, $r = \|x - c\|$.

Name	$\psi(r)$
Gaussian	$\psi(r) = \exp(-(\eta r)^2)$
Multiquadric	$\psi(r) = \sqrt{1 + (\eta r)^2}$
Inverse multiquadric	$\psi(r) = (\sqrt{1 + (\eta r)^2})^{-1}$
Inverse quadratic	$\psi(r) = (1 + (\eta r)^2)^{-1}$

Figure 3.2: Comparison between the fit (—) using 30 inverse quadratic radial basis functions and the error signal (—) of $N = 1000$ samples (a), and the corresponding difference between signal and fit (b).Figure 3.3: Comparison between the fit (—) using 30 inverse quadratic radial basis functions and the error signal (—) of $N = 50$ samples (a), and the corresponding difference between signal and fit at 1000 points (b).

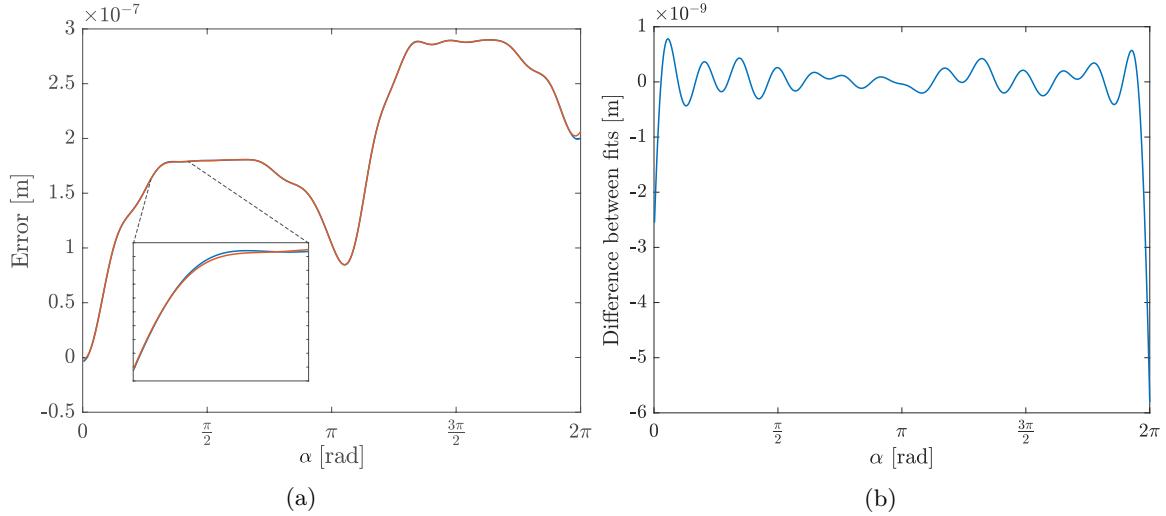


Figure 3.4: Comparison between fits of error signals with a length of 1000 (—) and 50 (—) samples (a), and the difference between the two fits at 1000 points (b).

3.5.2 Alternatives

Two alternatives to radial basis functions are polynomial basis functions and Fourier series. Fourier series can give close approximations of signals, but they have disadvantages for this application. Given the shape of the error signal, Fourier-series based functions are expected to give a good approximation. However, computational efficiency in calculating the input signal is important in an industrial setting. When Fourier-series based functions are used, calculating the input for a certain α -value during operation involves calculating sines, which is significantly more computationally demanding than calculating for example powers.

A fit is made using a 15th order Fourier series, as shown in Figure 3.5. This series contains 31 functions for which a scaling parameter is fitted: an offset, 15 sines and 15 cosines. Since the functions themselves are iteration-invariant, an optimal base period for the fits is determined beforehand. The RMS value of the difference between the signal and fit is 2.0×10^{-9} , which is larger than the values obtained for radial basis functions.

Polynomial basis functions are also shown to provide less accurate fits. In Figure 3.6, the result of a fit of the error signal using 30 polynomial basis functions is shown. The RMS value of the difference between the signal and the fit is approximately 2.4×10^{-9} . Since the results for fits with both Fourier series and polynomial basis functions are worse than those for radial basis functions, radial basis functions are preferred for the application of α -domain ILC on the piezo-stepper actuator.

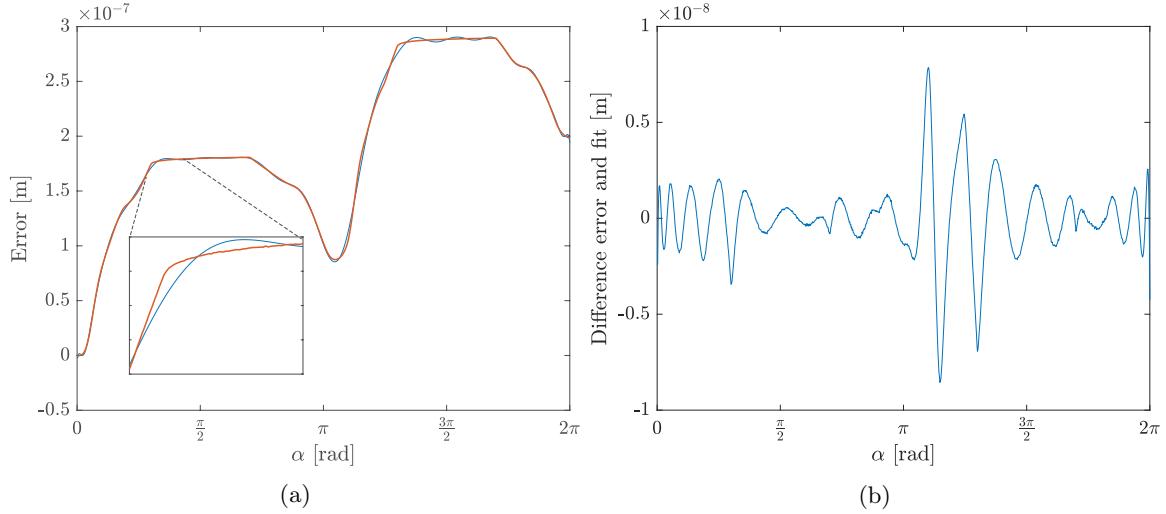


Figure 3.5: Comparison between the fit (—) using 31 Fourier-series based functions and the error signal (—) of $N = 1000$ samples (a), and the corresponding difference between signal and fit (b).

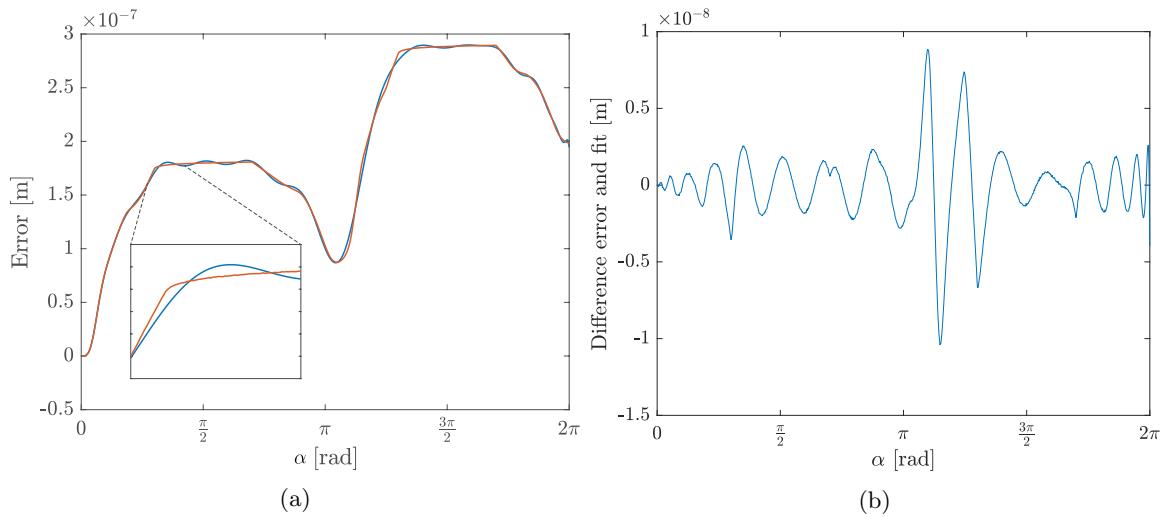


Figure 3.6: Comparison between the fit (—) using 30 polynomial basis functions and the error signal (—) of $N = 1000$ samples (a), and the corresponding difference between signal and fit (b).

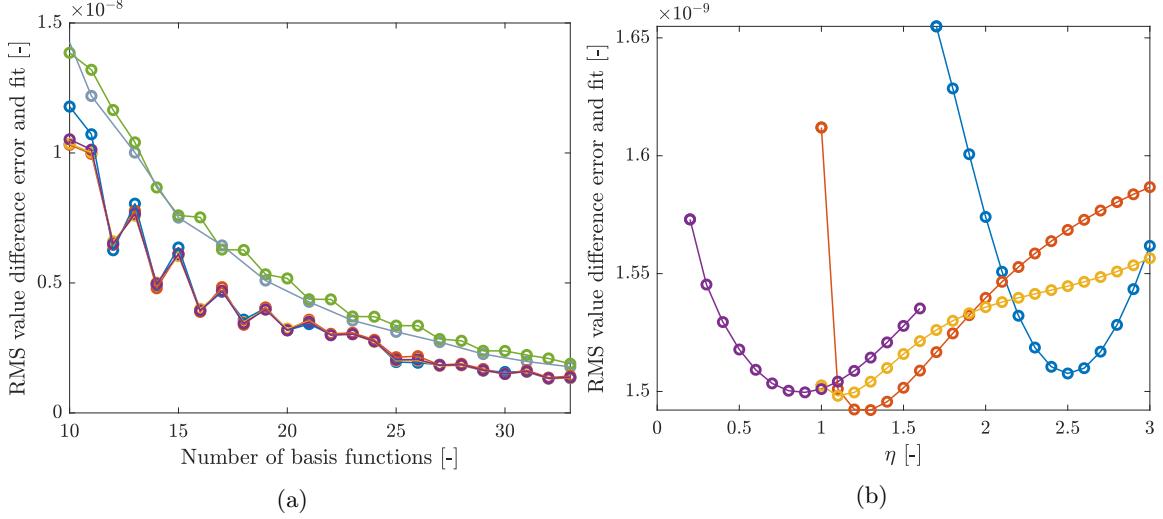


Figure 3.7: The quality of the fit depends on the type and number of (radial) basis functions (a) and the value of the shape factor η (b). An error signal of 1000 samples is used for the fits. Types of basis functions: Gaussian (blue circle), multiquadric (orange circle), inverse multiquadric (yellow circle), inverse quadratic (purple circle), polynomial (green circle) and Fourier series (red circle).

3.5.3 Selection

To select suitable radial basis functions and shape parameters, different types of radial basis functions with varying numbers and shape factors are compared. The functions are used to fit a measured error signal with a length of 1000 samples, and the resulting RMS values of the difference between the error signal and the fit are shown in Figure 3.7a. Radial basis functions are shown to give better fits than Fourier series and polynomial basis functions if the number of functions is equal. For radial basis functions even numbers of functions give better fits, and increasing the number of functions improves the fit as well. However, the differences between the types of radial basis functions are small.

The influence of the shape factor is shown in Figure 3.7b for fits with 30 radial basis functions. In the regions that are plotted, the differences are small but for shape factor values outside these regions the quality of the fit reduces fast and in some cases the sampled basis function matrices become close to singular.

Since the differences in quality of fit between the types of radial basis functions that are considered are small, the type that requires the least amount of calculations to determine $u_j(\alpha) = \psi(\alpha)\theta_j$ is used. For the set of functions considered here, that type is inverse quadratic radial basis functions:

$$\psi_k(\alpha) = \frac{1}{1 + (\eta\|\alpha - c_k\|)^2}, \quad k = 1, 2, \dots, 30 \quad (3.48)$$

Where the centers c_k are equidistant points on $[0, 2\pi]$. Based on Figure 3.7, 30 inverse quadratic functions with a shape factor of $\eta = 1$ are used in the experiments.

3.5.4 Linear independence of the sampled basis function vectors

According to Theorem 3.5 the sampled basis function vectors $\bar{\psi}_j = [\psi_1(\bar{\alpha}_j) \quad \psi_2(\bar{\alpha}_j) \quad \dots \quad \psi_M(\bar{\alpha}_j)]^T$ need to be linearly independent in order for the unique least-squares optimal fit of the sampled error vector to exist. Although the continuous kernels are linearly independent [29, 31], this does not need to hold for the sampled kernels.

For a set of radial basis functions it holds that the sampled kernels are linearly independent if the centers of the functions, or trial centers, are chosen equal to the sample points, also referred to as test points [34], pp. 11-35. Then, the matrix $\bar{\psi}_j^T$ is square, symmetric and positive definite. This is a useful property for many applications of radial basis function interpolation, where the test points can be chosen freely. In the case of α -domain ILC, the iteration-invariant trial centers can be chosen but the sample points are iteration-varying and do not coincide with the trial centers for all iterations. This results in an unsymmetric matrix $\bar{\psi}_j^T$. In most cases, the number of basis functions M is lower than the number of sample points N_j , so that $\bar{\psi}_j^T \in \mathbb{R}^{N_j \times M}$ is also non-square.

For a unique least-squares optimal fit of the sampled error to exist, the unsymmetric, non-square matrix $\bar{\psi}_j^T \in \mathbb{R}^{N_j \times M}$ needs to have full column rank M . Unsymmetric interpolation problems with radial basis function kernels, where the trial centers and test points do not coincide, have been used successfully in a variety of applications in the field of solving partial differential equations. However, in [35] it is shown that a general proof of the non-singularity of the linear systems of unsymmetric interpolation problems is impossible. Numerical evidence shows that cases of singularity can be constructed with effort, but that they are rare. According to literature, the linear systems are non-singular if the sampling is ‘fine enough’ in a sense that is not yet quantified further [36, 37].

It is expected that for the application considered in this thesis, the sampled kernels are linearly independent for all sampling situations. The density of the test points is always higher than that of the trial centers since the number of equidistant samples generated at the maximum drive frequency is higher than the number of radial basis functions. In addition, the trial centers are equidistant over the domain $[0, 2\pi)$. However, guarantees that the kernels are indeed linearly independent for all used sampling situations may or may not exist and are outside of the scope of this thesis. Therefore, it is suggested to check the rank of $\bar{\psi}_j^T$ for each fit. In the experiments described in Section 5.3, $\bar{\psi}_j^T$ always has full column rank.

3.6 Concluding remarks

A new framework for α -domain ILC is presented that can compensate repeating disturbances in the α -domain, by using basis functions to parameterize iteration-varying non-equidistantly sampled input and error signals. This parameterization allows for a continuous, smooth description of the input signal, the parameters for which are found using an optimal ILC update law. The sequence of parameter vectors is monotonically convergent to a fixed parameter vector if the basis functions that are used are linearly independent.

In this approach, all measured data of an iteration is used without need for approximation, and the amount of parameters that is learned is small. Therefore, this framework is preferable to the standard α -domain ILC framework of Chapter 2.

The framework can be used to compensate the α -domain repeating disturbance for the piezo-stepper actuator presented in Chapter 1, which is modeled as a gain with an α -domain repeating disturbance. For this application, inverse quadratic radial basis functions are suitable for parameterization of the input and error signals. After an analysis of the disturbance in Chapter 4, the feasibility of the two new frameworks for α -domain ILC is experimentally validated using a piezo-stepper actuator in Chapter 5.

4 Application to a piezo-stepper actuator: disturbance analysis

In this chapter, the piezo-stepper actuator introduced in Chapter 1 is analyzed with regard to behaviors such as hysteresis and creep in the piezo elements and the disturbances that are visible during open-loop walking and clamping experiments. It is determined whether the actuator can indeed be modeled as a gain with a dominant α -domain repeating disturbance, for which the ILC frameworks presented in this thesis are suitable.

In Section 4.1 some behaviors of piezo-stepper actuators are investigated. To analyze the disturbances in the system, experiments that show the open-loop clamping and walking behavior are presented in Section 4.2. In Section 4.3 possible causes of the disturbance are explored. In Section 4.4 the behaviors for positive and negative drive frequencies are compared and the directionality of the system is investigated. Rate-dependent effects are investigated in Section 4.5. Lastly, concluding remarks are given in Section 4.6.

4.1 Behavior of piezo-stepper actuators

Piezoelectric actuators are suitable for application in nano-positioning due to their high resolution, high stiffness and fast response. However, the actuators also show behaviors such as hysteresis and creep. In this section an overview of the piezoelectric effect, these behaviors and their influence on the response of a piezo-stepper actuator is given.

Piezoelectric actuators are based on the piezoelectric effect that occurs in certain crystalline lattices. These crystalline lattices have the unique characteristic that they deform under pressure, leading to a separation of the centers of gravity of the positive and negative charges which creates a dipole moment. The dipole moment induces an electric charge across the surface of the material. Conversely, applying a voltage to the material leads to mechanical strain in the lattice and deformation of the material. A simple model illustrating these effects is shown in Figure 4.1.

Two behaviors for piezoelectric actuators that influence the response significantly are hysteresis and creep. Hysteresis can be modeled as a nonlinearity between the applied voltage and the resulting mechanical position [38]. The cause of hysteresis in piezoelectric materials is believed to be irreversible losses caused by interaction of similarly oriented electric dipoles when an electric field is applied [3], pp. 17-39. The amount of hysteresis varies depending on the amplitude of the input voltage and its frequency, although it is often modeled as being rate-independent for applications with constant input frequencies [39]. The hysteresis in one of the shear elements of the piezo-stepper actuator presented in Chapter 1 is shown in Figure 4.2.

Methods of dealing with hysteresis include stepsize modulation, hysteresis compensation and charge control. Since hysteresis is amplitude-dependent, reducing the stepsize or amplitude reduces the amount of hysteresis in the system. However, this reduces the already small stroke of the actuator significantly. To compensate hysteresis via feedforward several models can be used, e.g. a simple polynomial model, a Maxwell slip model or a classical Preisach model which is the most widely used approach [39]. These models are rate-independent, but the rate-dependent behavior can be taken into account by cascading a hysteresis model with a model of the linear dynamics [38]. Alternatively, using charge control rather than voltage control is known to reduce the hysteresis effect significantly [40]. This is not widely implemented, because implementing charge control with the available charge/current amplifiers leads to difficulties [39].

Creep is a rate-dependent deformation of piezoelectric materials due to a constant electric field, and is visible as a low-frequency drift after a step change in input voltage. Since the time constant for creep is typically several minutes, its influence is mostly visible in the low-frequency and static positioning ability [3], pp. 17-39. For high input frequencies the influence of creep is negligible, but it may become visible as a rate-dependent effect when high and low frequencies are compared. Creep

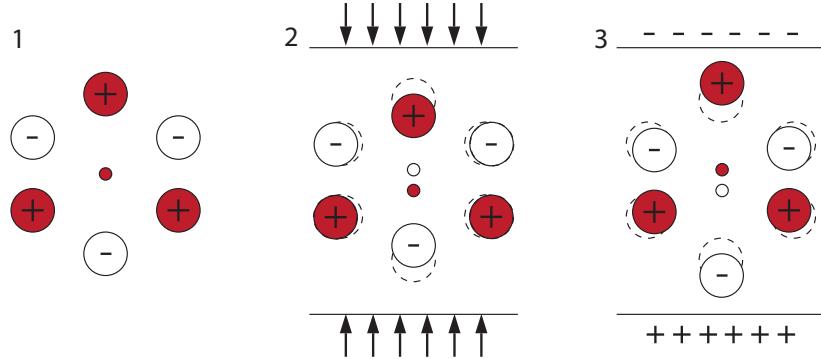


Figure 4.1: A simple model illustrating the piezoelectric effect. The large circles represent positive and negative ions, and the small circles represent their centers of gravity. 1) shows the equilibrium state, 2) shows how mechanical stress induces an electric dipole and 3) shows how an applied voltage produces mechanical strain. (Figure is based on [3].)

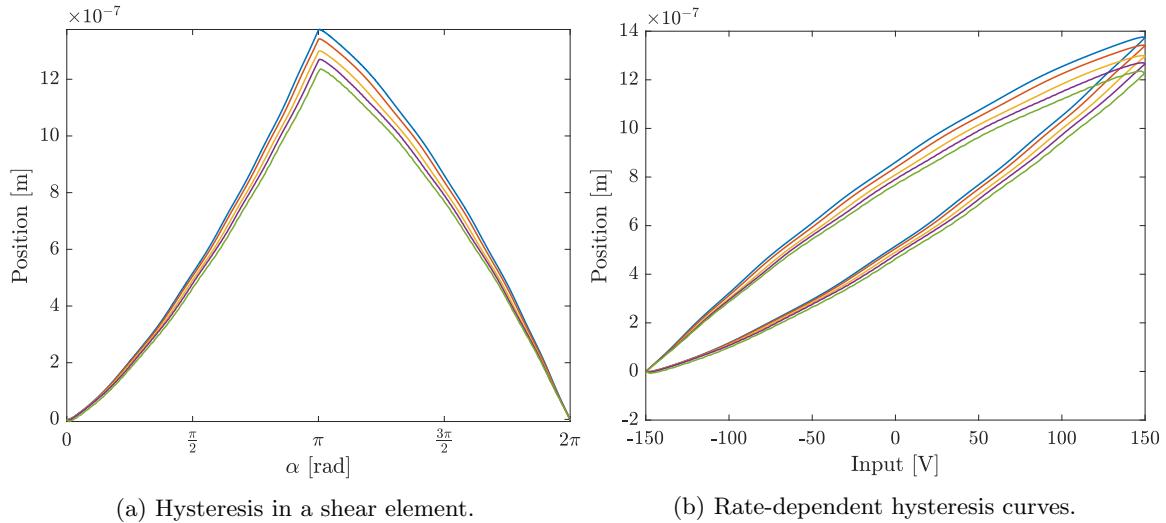


Figure 4.2: Rate-dependent hysteresis in a shear element of the piezo-stepper actuator. Drive frequencies: 1 Hz (—), 2 Hz (—), 5 Hz (—), 10 Hz (—), 20 Hz (—).

can be modeled as a linear effect and model-based feedforward can be used to reduce its influence. Model-based feedforward using cascaded models of rate-dependent linear effects and rate-independent hysteresis can improve the performance of piezoelectric actuators significantly [38].

Hysteresis and rate-dependent behavior are present in the shear elements of the piezo-stepper actuator considered in this thesis. Therefore, they are expected to be visible when ILC is used to compensate the α -domain repeating disturbance. When complete steps are taken for each iteration, the hysteresis is iteration-invariant and it is expected to be compensated by the input that is learned using ILC. However, when the extension is made to partial steps and changes of sign within a step, which is required for closed-loop implementation of the ILC framework, hysteresis compensation will be required. The rate-dependent behavior is expected to influence the ILC convergence behavior when an input signal is learned over varying drive frequencies, as is discussed further in Section 4.5. Compensation of these behaviors is outside of the scope of this thesis, but it is recommended for future work.

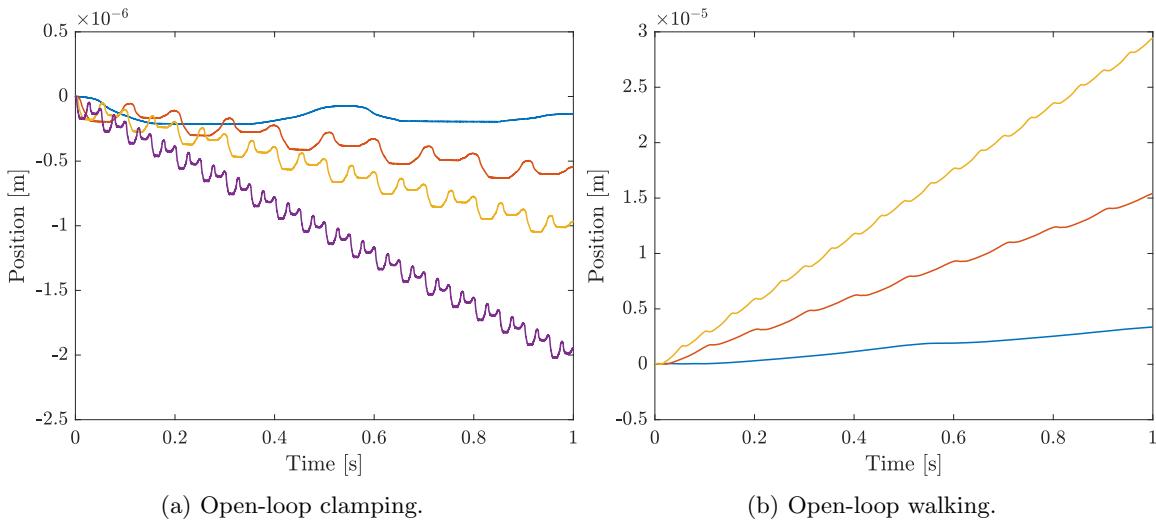


Figure 4.3: The position during open-loop clamping (a) and walking experiments (b) shows significant disturbances. Drive frequencies: 1 Hz (—), 5 Hz (—), 10 Hz (—), 20 Hz (—).

4.2 Open-loop clamping and walking performance

In this section, the performance of the piezo-stepper actuator during open-loop clamping and walking experiments, as well as the repetitive behavior of the disturbances seen during these experiments, is analyzed.

For the piezo-stepper actuator introduced in Chapter 1, the position and velocity during a step is analyzed. The position is measured directly using an optical encoder [7]. To determine the velocity a numerical derivative of the position signal is used. The resulting velocity signal contains high-frequent noise, therefore it is filtered using a zero-phase low-pass filter. The result is averaged over windows with a length of 2π rad. The amount of windows and the number of samples within a window are determined by the drive frequency, as each measurement has a length of 10 s and a time-domain sample frequency of 10 kHz. Note that for negative drive frequencies, the value of α is decreasing in time.

The position of the mover during open-loop clamping and walking experiments at different drive frequencies shows disturbances, as shown in Figure 4.3. For a single drive frequency, this disturbance is repeating in the temporal domain, but for varying drive frequencies the disturbance in the temporal domain is varying. Therefore, the disturbances are analyzed in the α -domain.

4.2.1 Open-loop clamping

The disturbance during open-loop clamping experiments for different drive frequencies is shown to be repeating in the α -domain. In these experiments, the desired position is $y_d(\alpha) = 0 \forall \alpha \in [0, 2\pi]$. The deviation from the desired position shows a similar shape for the drive frequencies ranging from 1 Hz to 20 Hz, as shown in Figure 4.4a. The velocity profiles for these frequencies show even more similarities, and the main difference is in the height of the peaks in velocity as shown in Figure 4.4b.

For negative drive frequencies ranging from -1 Hz to -20 Hz the position and velocity plots are also similar, as shown in Figure 4.5. However, the signals for different signs of the drive frequency are different on $\alpha \in [0, 2\pi)$. The shapes show some similarities that are analyzed further in Section 4.3, where the cause of the repeating disturbance is investigated.

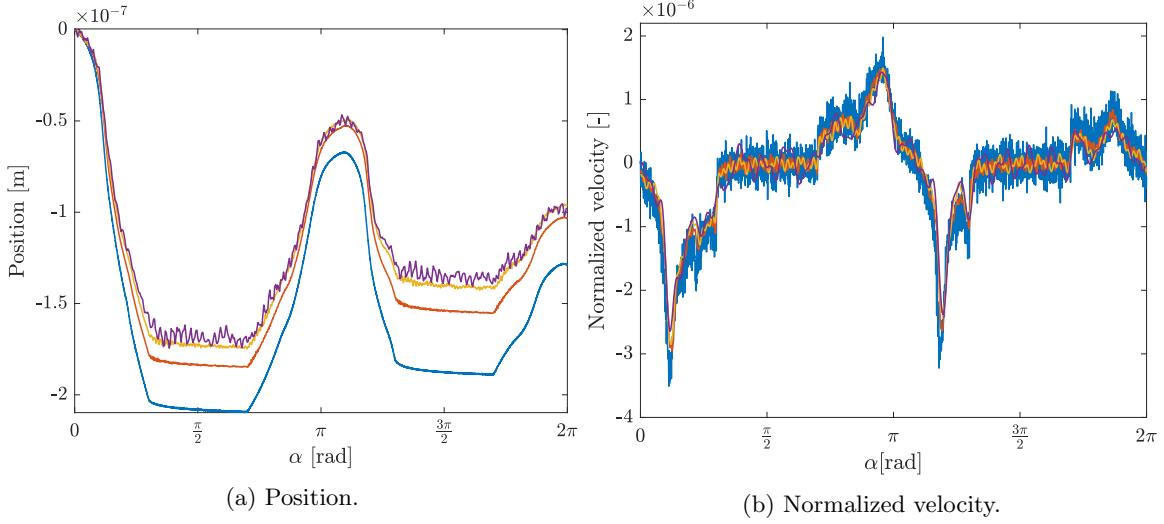


Figure 4.4: α -domain repeating disturbances in the position (a) and velocity (b) during an open-loop clamping experiment for $f_\alpha > 0$. Drive frequencies: 1 Hz (—), 5 Hz (—), 10 Hz (—), 20 Hz (—).

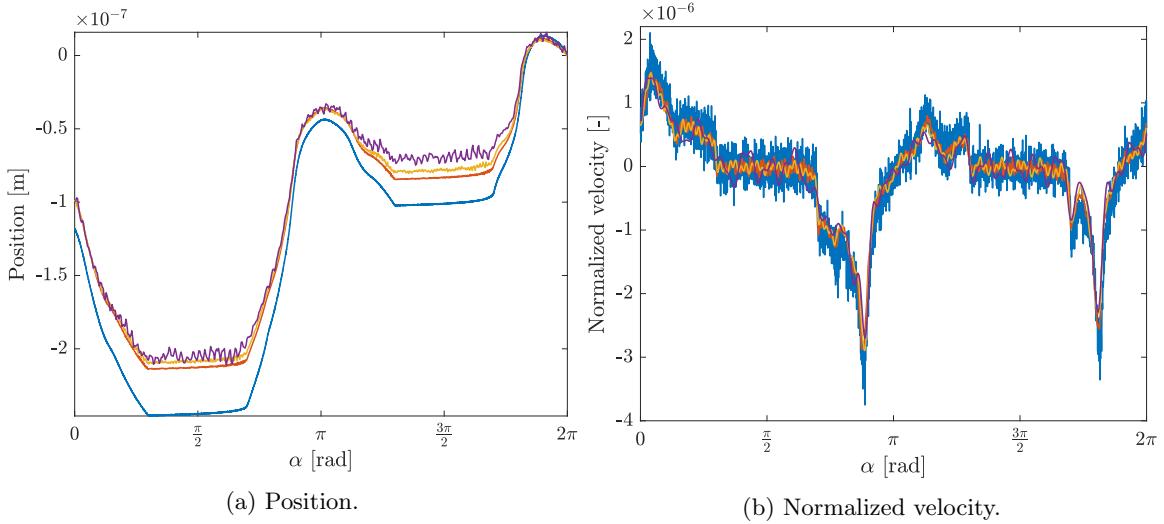


Figure 4.5: α -domain repeating disturbances in the position (a) and velocity (b) during an open-loop clamping experiment for $f_\alpha < 0$. Drive frequencies: -1 Hz (—), -5 Hz (—), -10 Hz (—), -20 Hz (—).

4.2.2 Open-loop walking

During open-loop walking experiments the disturbances are also shown to be repeating in the α -domain for different drive frequencies. In these experiments the desired position is a straight line given by $y_d(\alpha) = c_r\alpha$ for some constant α . The error, which is defined as the deviation of the position from this line, is shown to be repeating in α in Figure 4.6 and Figure 4.7 for positive drive frequencies ranging from 1 Hz to 10 Hz and negative drive frequencies ranging from -1 Hz to -10 Hz respectively. In walking experiments, the position and velocity disturbances for positive drive frequencies are larger than those for negative drive frequency. This difference is analyzed further in Section 4.4.

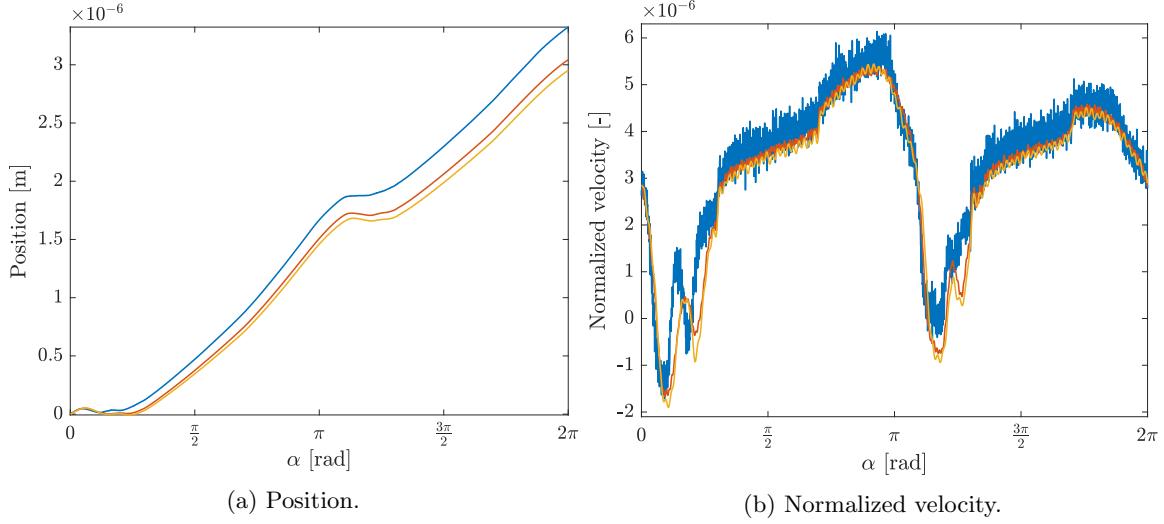


Figure 4.6: α -domain repeating disturbances in the position (a) and velocity (b) during an open-loop walking experiment for $f_\alpha > 0$. Drive frequencies: 1 Hz (—), 5 Hz (—), 10 Hz (—).

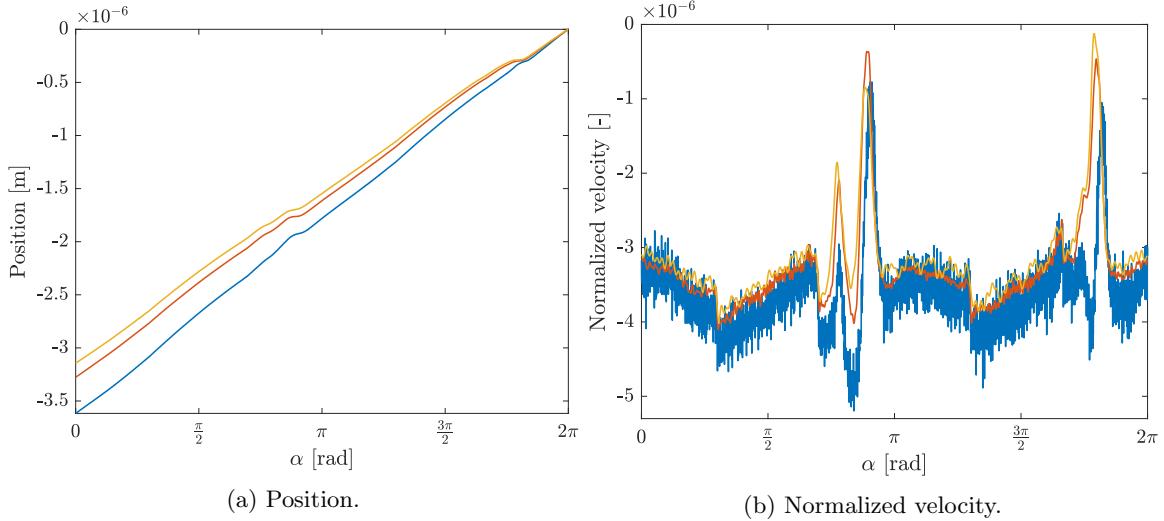


Figure 4.7: α -domain repeating disturbances in the position (a) and velocity (b) during an open-loop walking experiment for $f_\alpha < 0$. Drive frequencies: -1 Hz (—), -5 Hz (—), -10 Hz (—).

Hysteresis is visible in the position and velocity signals for α -values where a single group of piezo elements is engaged, e.g. around $\alpha = \frac{\pi}{2}$. In the open-loop clamping experiments the velocity is zero for commutation angles where the clamp input is constant, but in open-loop walking the velocity is increasing or decreasing, depending on the direction of the drive frequency. Even though the derivative of the input voltage is constant, the mover velocity is not constant. Rate-dependent behaviors are also visible, as a smaller drive frequency results in a larger step size. These rate-dependent behaviors are analyzed further in Section 4.5.

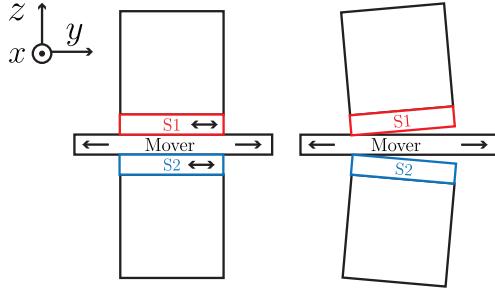


Figure 4.8: Tilted clamps are a possible cause of the α -domain repeating disturbance.

Table 4.1: Sitoff and liftoff voltages for the piezo actuator.

Piezo group	Sitoff voltage	Liftoff voltage
1	-9.5 V	-20.5 V
2	-16 V	-23.5 V

4.3 Causes of the α -domain disturbance

The main disturbance during both clamping and walking experiments is repeating in the α -domain and shows distinct peaks at certain α -values. This suggests that the disturbance is caused by the movement of the clamping piezo elements, which is periodic in α . More specific, a possible cause of the disturbance is a titled positioning of the clamps, which is illustrated in Figure 4.8. This would lead to a change in position whenever the clamps are engaging with or retracting from the mover. The moments of first engagement and total retraction of the clamps are referred to as respectively sitoff and liftoff. The approximated voltages at which sitoff and liftoff occur for this specific piezo-stepper actuator are based on experiments presented in [7] and are given in Table 4.1.

The locations of the peaks in the velocities correspond to the sitoff and liftoff moments of the clamps, as shown in Figure 4.9 and 4.10 for respectively positive and negative drive frequencies during open-loop walking and clamping. The largest peak in velocity occurs just after the sitoff of the first piezo group in both Figure 4.9a and 4.10a. The other peak, which is slightly smaller, occurs just after the sitoff of the second group. The largest disturbances occur during the take-over, between the sitoff of one piezo group and the liftoff of the other. The same connection between the disturbances and the clamp movements is seen in Figure 4.9b and 4.10b.

4.4 Directionality

The disturbances around the sitoff and liftoff moments during open-loop clamping experiments do not depend on the sign of the drive frequency. This can be seen when the sitoff and liftoff moments are aligned by flipping the velocity signal of an open-loop clamping experiment with $f_\alpha < 0$ over the y -axis and shifting it with π , as is done in Figure 4.11. The observation supports the idea that the disturbances are caused by a tilted clamp. Since the sitoff and liftoff moments occur at different values of α for different signs of the drive frequency, the behavior is directional in the α -domain. As a result, an input signal $u(\alpha)$ that compensates for the α -domain repeating error will always be dependent on the sign of the drive frequency.

In walking experiments the velocity signals for positive and negative drive frequencies are not similar, see Figure 4.12. The locations of the peaks can be related to the sitoff and liftoff moments, but the disturbances in velocity for positive drive frequencies are larger. Because of these differences, the compensating input signals for positive and negative drive frequencies have to be designed separately.

Compensating signals are designed separately for positive and negative drive frequencies, and for clamping and walking experiments. For positive drive frequencies the shape of the velocity signal during clamping and walking is similar except for the hysteresis effects seen during walking. This suggests that it could be possible to design a compensating signal based on the clamping experiments and apply it during walking experiments to improve the performance, as was also shown in [8]. For negative drive frequencies clamping and walking experiments give different velocity signals so that a signal that improves the performance during clamping with $f_\alpha < 0$ is not expected to improve the performance during walking.

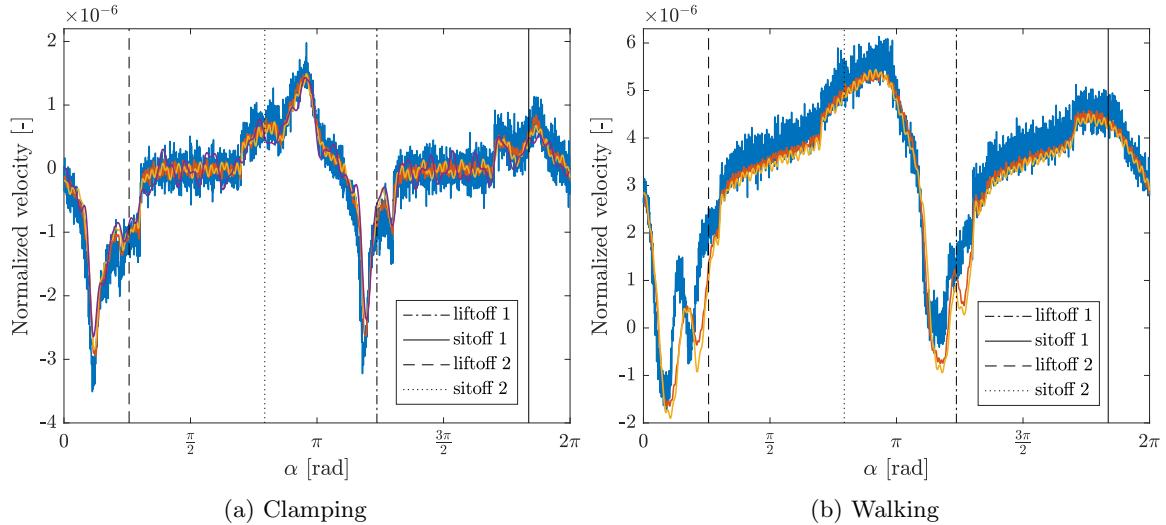


Figure 4.9: Velocity signals during open-loop clamping and walking experiments for $f_\alpha > 0$ compared to the sitoff and liftoff moments of the clamps. Drive frequencies: 1 Hz (—), 5 Hz (—), 10 Hz (—), 20 Hz (—).

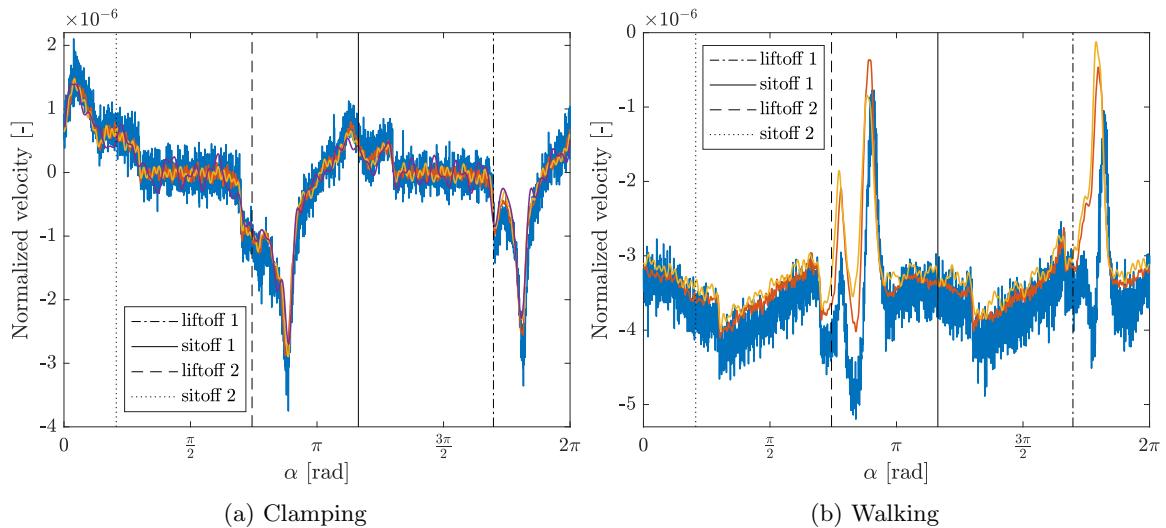


Figure 4.10: Velocity signals during open-loop clamping and walking experiments for $f_\alpha < 0$ compared to the sitoff and liftoff moments of the clamps. Note that for $f_\alpha < 0$ the value of α is decreasing in time. Drive frequencies: -1 Hz (—), -5 Hz (—), -10 Hz (—), -20 Hz (—).

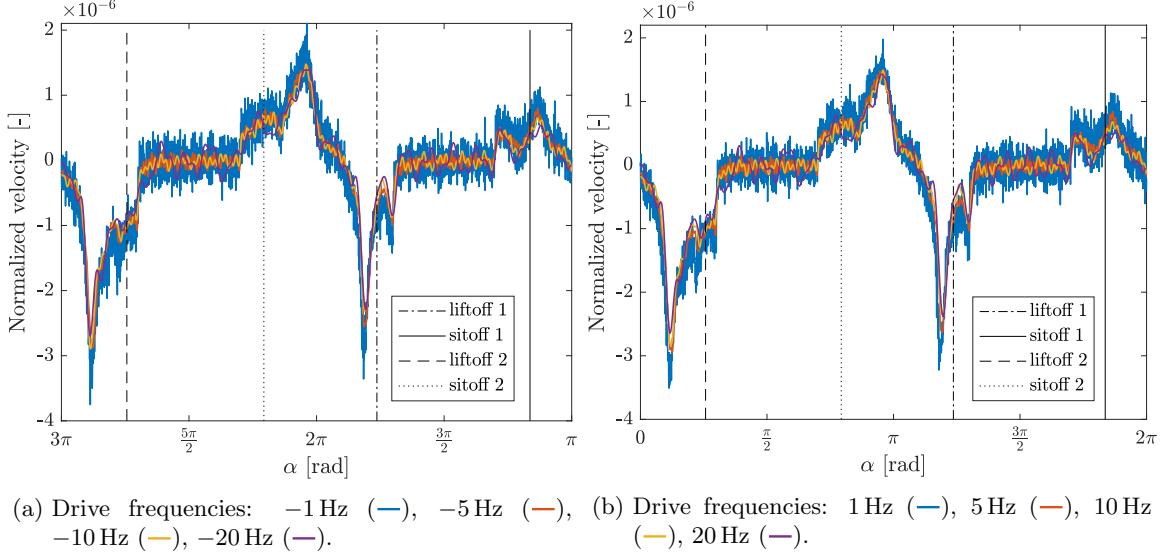


Figure 4.11: The velocity signal of a clamping experiment for negative drive frequencies is flipped and shifted with π (a) to obtain a signal almost identical to that measured for positive drive frequencies (b).

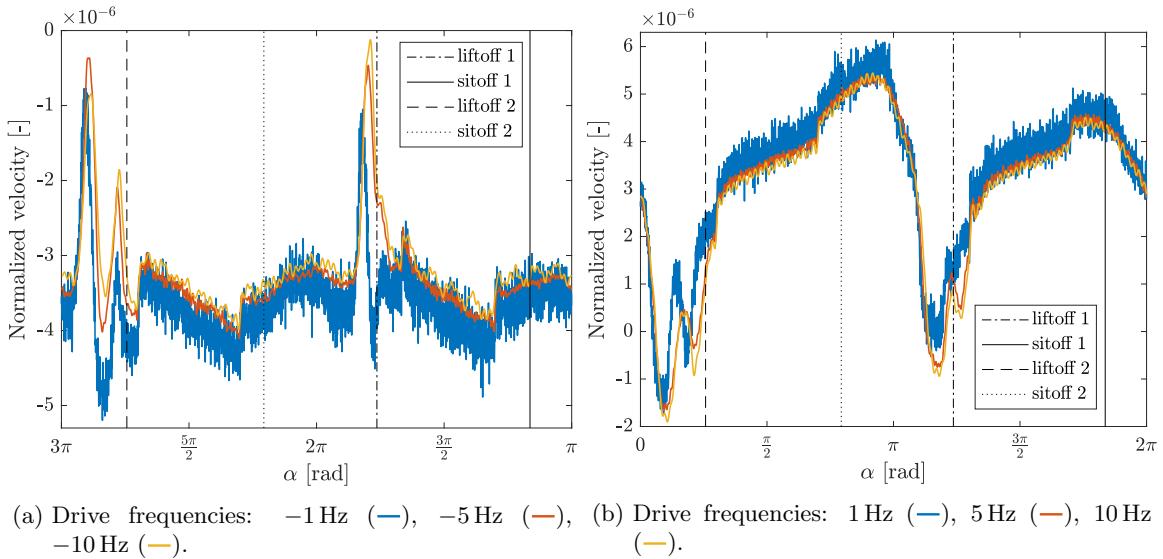
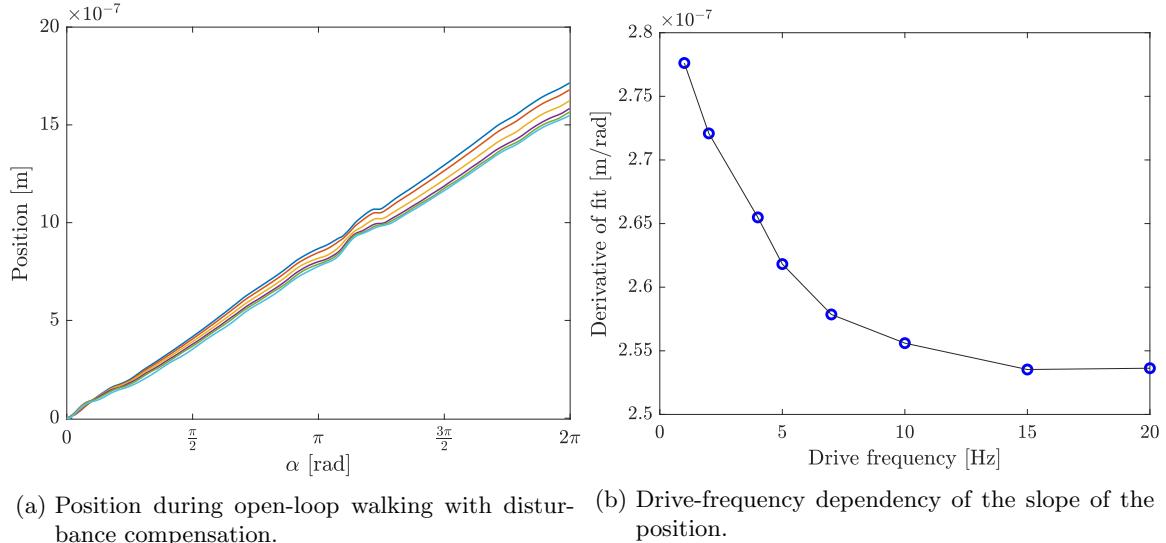


Figure 4.12: The velocity signal of a walking experiment for negative drive frequencies is flipped and shifted with π (a), resulting in a signal different from that for positive drive frequencies (b).



(a) Position during open-loop walking with disturbance compensation. (b) Drive-frequency dependency of the slope of the position.

Figure 4.13: The slope of the position in the α -domain is rate-dependent. Drive frequencies: 1 Hz (—), 2 Hz (—), 5 Hz (—), 10 Hz (—), 15 Hz (—), 20 Hz (—).

4.5 Drive-frequency dependent slope

The slope of a position signal in the α -domain shows dependency on the drive frequency because of the rate-dependent hysteresis described in Section 4.1. This variation in step size is visible in e.g. Figure 4.2a, 4.6a and 4.7a. Since the rate dependence is expected to influence the convergence behavior of α -domain ILC during open-loop walking, it is analyzed.

The difference in slope is visible when a tilt-compensating signal that is learned during open-loop clamping [8] is applied during open-loop walking for different drive frequencies, as shown in Figure 4.13a. The slope is determined by making a linear fit of the position signals. In Figure 4.13b the slope is plotted as a function of the drive frequency, which shows that for higher drive frequencies the slope reduces. A lower drive frequency results in a slightly larger step for the same input signal.

Since the rate-dependent behavior is not repeating in the α -domain, it acts as an iteration-varying disturbance that limits the α -domain ILC performance. It is recommended to compensate this behavior via feedforward to improve the results obtained by α -domain ILC. Hysteresis and creep compensation are outside of the scope of this thesis, but it is expected that the experimental results in Chapter 5 can be improved by compensating the rate-dependent behavior.

4.6 Concluding remarks

Based on open-loop clamping and walking experiments it is concluded that the dominant disturbance is indeed repeating in the α -domain. For different drive frequencies the measured position and normalized velocity as a function of α are comparable. The peaks in the disturbance can be related to the moments of engagement and release of the clamping elements, suggesting that the disturbance is caused by a tilted positioning of the clamps. The disturbances are directional, meaning that they depend on the sign of the drive frequency. It is also observed that the size of a step is influenced slightly by the drive frequency.

The piezo-stepper can be modeled as a gain with a dominant α -domain disturbance, which means that the system is suitable for experimental validation of the frameworks of Chapter 2 and 3. The experiments and their results are presented in Chapter 5.

5 Application to a piezo-stepper actuator: experiments

In this chapter the α -domain ILC frameworks are applied to a piezo-stepper actuator that is modeled as a gain with a dominant α -domain repeating disturbance. In the experimental validation, positive and negative drive frequencies as well as open-loop clamping and walking situations are considered separately, since the disturbance analysis of Chapter 4 shows that there is no single disturbance-compensating signal for all situations.

The general experimental approach to waveform enhancement using α -domain ILC is explained in Section 5.1. In Section 5.2 experimental results using the ILC framework of Chapter 2 are presented. The ILC approach of Chapter 3 that uses basis functions is validated in Section 5.3. An overview of the experimental results is given in Section 5.4 and lastly concluding remarks are given in Section 5.5.

5.1 Approach to waveform enhancement using α -domain ILC

In this section the implementation of the learned input signal by means of enhancement of the standard waveforms of the shear elements is explained. Then, an overview of the implementation of both frameworks for α -domain ILC is given.

ILC is applied to the piezo-stepper actuator during open-loop clamping and open-loop walking experiments. For open-loop clamping, the non-compensating input to the shear elements is 0 V and the reference $y_d(\alpha) = 0$ for all $\alpha \in [0, 2\pi]$. For open-loop walking the goal is to get a constant velocity, therefore the position reference is a straight line given by $y_d(\alpha) = c_r\alpha$. The constant $c_r \in \mathbb{R}$ is determined by estimating the maximum displacement of the mover in one step of 2π rad.

The shear elements have a stroke of approximately 3.0×10^{-6} m with an input amplitude of 250 V. Therefore, the estimated inverse shear model is given by $\hat{G}^{-1} = 1.67 \times 10^8$ V m $^{-1}$. Each shear element is connected for $\frac{5}{3}\pi$ rad of each step, during which the voltage is increased from -150 V to 150 V. The movement rate of the mover is therefore approximated by 3.4×10^{-7} m rad $^{-1}$. However, this movement rate does not account for the hysteresis effects in the piezo elements. To ensure that the reference is feasible, the slope of the reference is chosen lower: $c_r = 3 \times 10^{-7}$ m rad $^{-1}$.

5.1.1 Waveform enhancement

For implementation, the input signal $u_j(\alpha)$, $\alpha \in [0, 2\pi]$ that is designed using α -domain ILC is separated into separate inputs $s_{comp,i,j}(\alpha)$, $i = 1, 2$ for the two sets of shear elements.

To prevent jumps in the input when the piezo-stepper is actuated for more than one step, the inputs are designed so that $s_{comp,i,j}(0) = s_{comp,i,j}(2\pi)$, $i = 1, 2$. This is accomplished by using the range of α where one clamp is retracted completely to reset the corresponding shear element. In addition, the shear waveforms are defined so that the velocities of the shear elements are identical for any α where both clamps could be in contact with the mover, to prevent additional disturbances during takeover. Using these design guidelines, a learned disturbance-compensating input is divided into waveforms $s_{comp,i,j}(\alpha)$, $i = 1, 2$ for the two groups of shear elements, satisfying

$$\frac{\delta u_j(\alpha)}{\delta \alpha} = \begin{cases} \frac{\delta s_{comp,1,j}(\alpha)}{\delta \alpha} & \text{if } \alpha \in [\frac{\pi}{3}, \frac{2\pi}{3}] \\ \frac{\delta s_{comp,2,j}(\alpha)}{\delta \alpha} & \text{if } \alpha \in [\frac{4\pi}{3}, \frac{5\pi}{3}] \\ \frac{\delta s_{comp,1,j}(\alpha)}{\delta \alpha} = \frac{\delta s_{comp,2,j}(\alpha)}{\delta \alpha} & \text{otherwise.} \end{cases} \quad (5.1)$$

The separate shear inputs are added to the standard shear waveforms, resulting in enhanced waveforms. The approach to waveform enhancement is shown in Figure 5.1.

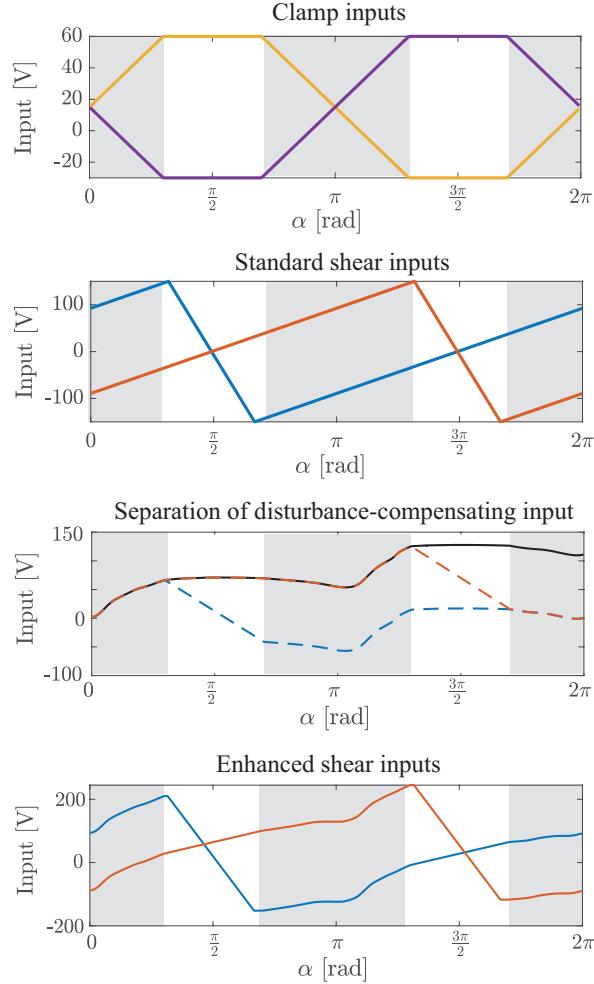


Figure 5.1: Approach to waveform enhancement using a learned disturbance-compensating input signal. In the waveforms of clamp elements 1 (—) and 2 (—), regions of α where both clamps could be in contact with the mover are indicated in gray (a). The standard inputs for shear element 1 (—) and (—) have equal derivatives in this region (b). The learned input signal $u(\alpha)$ (—) is shown in (c) and is divided into separate waveforms for the shear elements 1 (---) and 2 (---). The resulting compensating inputs are added to the standard waveforms, resulting in the new enhanced waveforms for the shear elements (d). Note that regions where one of the groups is not in contact with the mover are used to reset the corresponding shear elements.

5.1.2 Implementation of the ILC frameworks

The steps taken during a series of ILC iterations for the ILC frameworks without and with basis functions are described in respectively Algorithm 5.1 and 5.2. In the next sections, the proposed frameworks are used during both open-loop clamping and open-loop walking experiments to learn enhanced piezo-stepper waveforms that compensate the α -domain repeating disturbance.

Remark. Each experiment lasts several seconds and contains multiple steps of the piezo-stepper actuator. In the algorithms the data of each experiment is separated into windows containing one step each, after which the data of these windows is averaged. This approach reduces the influence of varying disturbances that do not depend on the drive frequency, such as sensor noise. The averaging does not influence the update laws and convergence conditions given in Theorems 2.7, 3.3 and 3.4.

Algorithm 5.1 Waveform enhancement using iterative learning control

Choose a nominal drive frequency f^n

for $j = 1 : n$ **do**

if $j = 1$ **then**

 Standard shear inputs: $s_i(\alpha)$, $i = 1, 2$

else

 Enhanced shear inputs: $s_i(\alpha) + s_{comp,i,j}(\alpha)$, $i = 1, 2$

end if

Perform an open-loop walking or clamping experiment of several steps with $f_{\alpha,j}$

Separate data into windows of one step using a peak-finder function

Determine the error at the nominal sample points for each window using approximation and interpolation

Find \bar{e}_j by averaging the error of the windows

if $j = 1$ **then**

 Input update according to (2.13): $\bar{u}_{j+1} = \beta L \bar{e}_j$

else

 Input update according to (2.13): $\bar{u}_{j+1} = Q \bar{u}_j + \beta L \bar{e}_j$

end if

Divide \bar{u}_{j+1} into sampled waveforms $\bar{s}_{comp,1,j+1}$ and $\bar{s}_{comp,2,j+1}$

end for

Algorithm 5.2 Waveform enhancement using iterative learning control with basis functions

Choose a basis ψ , see Section 3.5

for $j = 1 : n$ **do**

if $j = 1$ **then**

 Standard shear inputs: $s_i(\alpha)$, $i = 1, 2$

else

 Enhanced shear inputs: $s_i(\alpha) + s_{comp,i,j}(\alpha)$, $i = 1, 2$

end if

Perform an open-loop walking or clamping experiment for several steps with $f_{\alpha,j}$

Separate data into windows of one step using a peak-finder function

Fit the error of each window using basis functions according to Section 3.4

Find $e_j = \psi^T \theta_j^e$ by averaging the fits of each window

if $j = 1$ **then**

 Parameter update according to (3.11): $\theta_{j+1} = L \theta_j^e$

else

 Parameter update according to (3.11): $\theta_{j+1} = Q_\psi \theta_j + L_\psi \theta_j^e$

end if

Input update according to (3.4) : $u_{j+1} = \psi^T \theta_{j+1}$

Divide u_{j+1} into waveforms $s_{comp,1,j+1}$ and $s_{comp,2,j+1}$

end for

5.2 Experiments: α -domain ILC

In this section α -domain ILC is applied during open-loop walking according to Algorithm 5.1. First, results for open-loop walking with arbitrary drive frequencies varying between 1 Hz and 11 Hz are presented. Then, results for open-loop walking with negative drive frequencies are presented.

5.2.1 α -domain ILC during open-loop walking with arbitrary drive frequencies

α -domain ILC is applied during an open-loop walking experiment with completely arbitrary drive frequencies, so that both estimation and interpolation of the measured output is required to update the input signal $\bar{u}_j(\bar{\alpha}^n)$ at the nominal sample points. The nominal drive frequency is chosen as 1 Hz. Since the system is sampled at 10 kHz, $\bar{\alpha}^n \in \mathbb{R}^{N \times 1}$ contains $N = 10000$ equidistant points on the domain $\alpha \in [0, 2\pi]$. The ILC update law (2.13) is used. The filters are chosen as $Q = I$ and $L = \hat{G}^{-1} = 1.67 \times 10^8$, and the learning gain is chosen to be $\beta = 0.3$. Increasing the learning gain increases the convergence speed, but it also increases the sensitivity to iteration-varying effects significantly. In Appendix A.1 the influence of the learning gain is shown.

During the experiments the error and input signal converge, as shown in Figure 5.2 and Figure 5.3a respectively. In sixteen iterations the root mean square value of the error is reduced by a factor 20 from 3.4×10^{-7} to 1.7×10^{-8} . However, the convergence is not monotonic. At iterations 4 and 8, and especially at iteration 17, a change in drive frequency causes an increase in the error. Note that between these frequencies, the input signal converges towards its final value (between iteration 3 and 4) or stays approximately equal (between iteration 16 and 17), as shown in Figure 5.3b. Results in Appendix A.1 for different values of β show that although the error can increase when the drive frequency changes, the error does converge to a bounded region.

The increase in error that occurs when the drive frequency changes is caused by the drive-frequency dependent slope in the position shown in Section 4.5. This is visible in Figure 5.4a where the errors for iterations before and after a change of drive frequency are plotted. When the slope is removed from the results by fitting and subtracting a linear trend line, the α -domain repeating part of the error decreases between iteration 3 and 4, and stays approximately the same between iterations 16 and 17, since the error at iteration 16 is small. This is shown in Figure 5.4b.

The deviation from a linear trend line converges monotonically over the iterations, as shown in Figure 5.5. This is determined by fitting a trend line through the position of each iteration and subtracting it from the position signal. The deviation is reduced by a factor 8 in seventeen iterations and shows monotonic convergence even at iterations where the total error is increasing. This leads to the conclusion that the α -domain repeating part of the disturbance is compensated, and that the results obtained with α -domain ILC can be improved significantly when the rate-dependent part of the hysteresis is compensated, for example by using a separate feedforward.

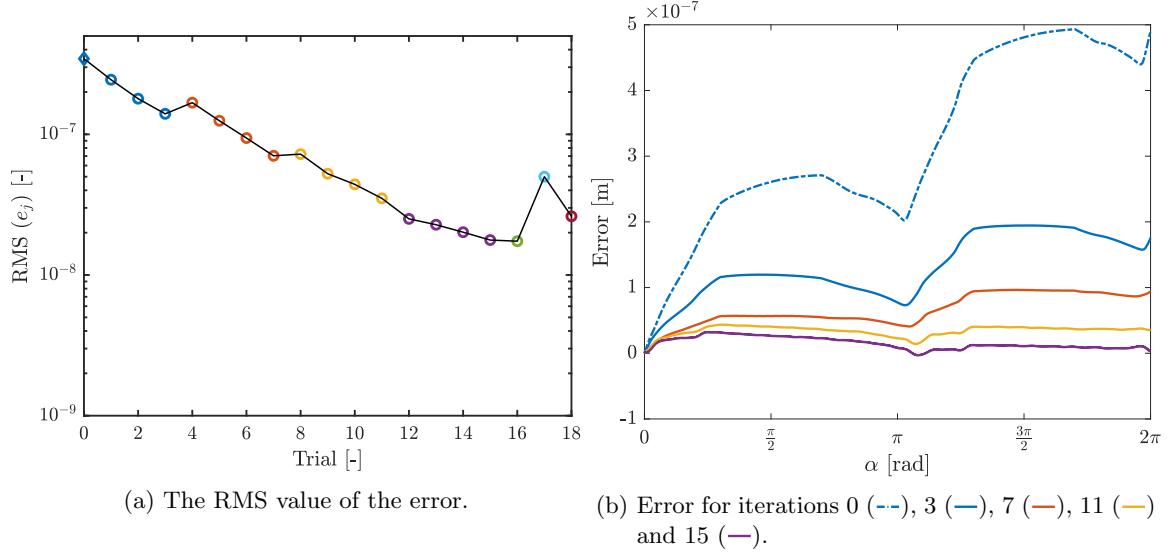


Figure 5.2: α -domain ILC convergence for open-loop walking with arbitrary drive frequencies. Subsequent drive frequencies: 1 Hz (blue circle), 7 Hz (orange circle), 11 Hz (yellow circle), 10 Hz (purple circle), 6 Hz (green circle), 2 Hz (blue circle), 9 Hz (orange circle).

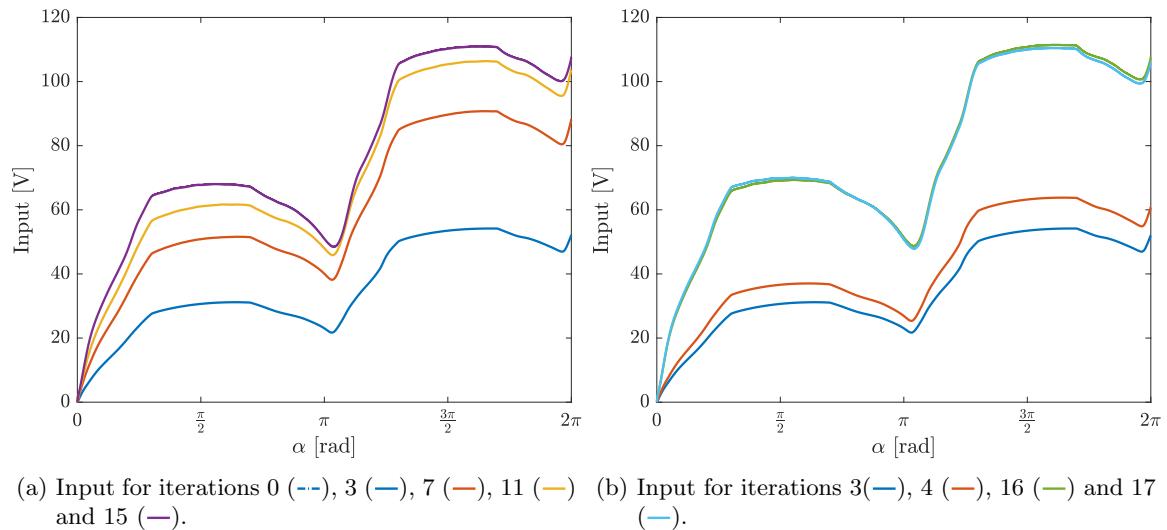


Figure 5.3: Convergence of input signal over iterations.

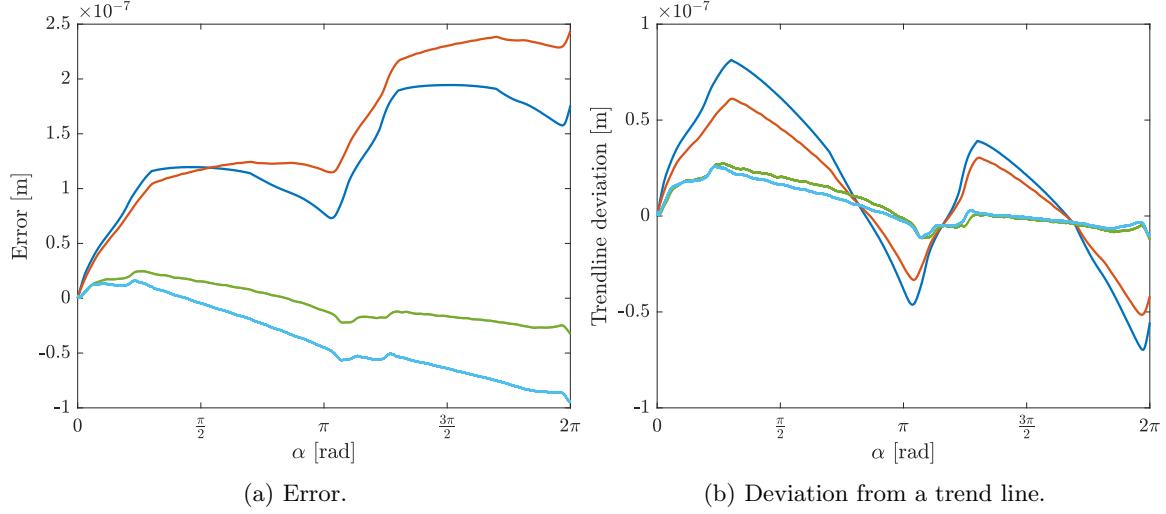


Figure 5.4: A change in drive frequency between iterations 3 (—) and 4 (—) and between iterations 16 (—) and 17 (—) causes an increase in error because of a drive-frequency dependent slope. However, the deviation from a linear trend line reduces between these iterations.

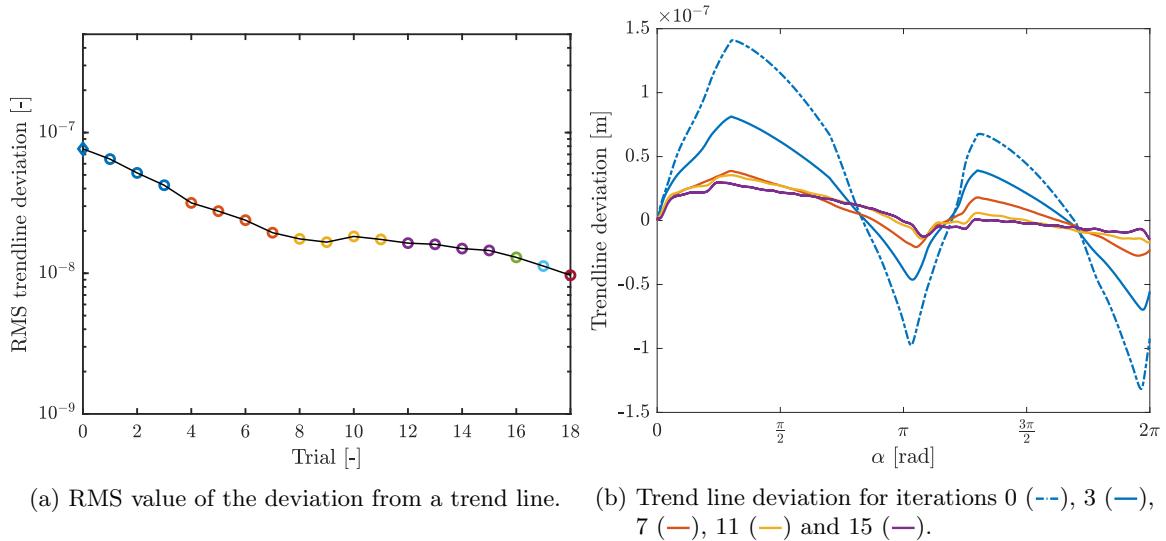


Figure 5.5: The deviation from a linear trend line shows convergence for arbitrary drive frequencies. Subsequent drive frequencies: 1 Hz (○), 7 Hz (○), 11 Hz (○), 10 Hz (○), 6 Hz (○), 2 Hz (○), 9 Hz (○).

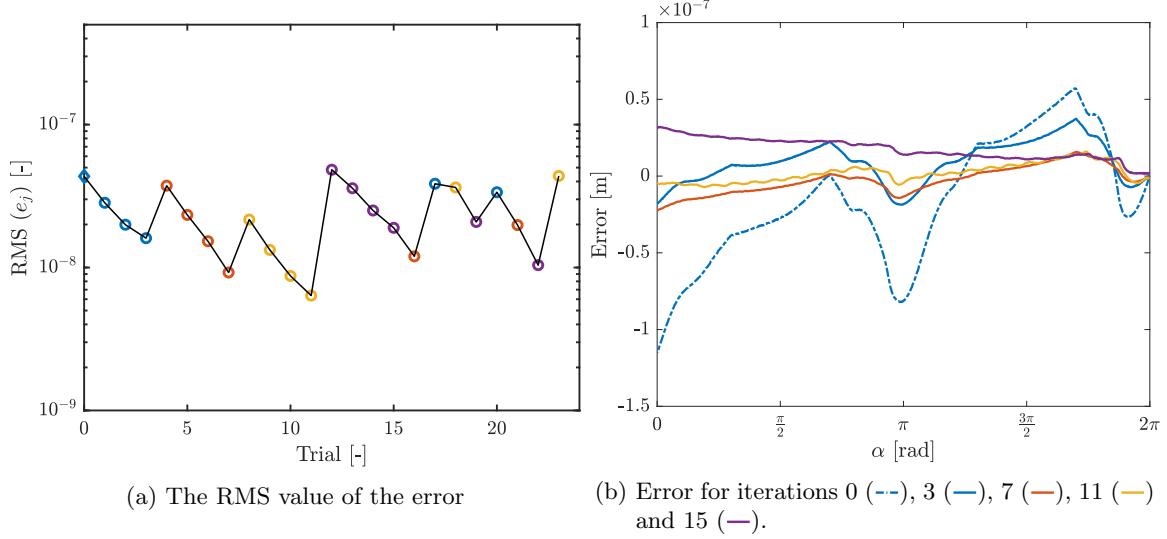


Figure 5.6: Error development over iterations during open-loop walking for negative drive frequencies with $\beta = 0.3$. Subsequent drive frequencies: -1 Hz (\circlearrowleft), -5 Hz (\circlearrowright), -10 Hz (\circlearrowuparrow), -2 Hz (\circlearrowdownarrow).

5.2.2 Negative drive frequencies

Since the disturbance is directional in the α -domain, compensating signals for negative drive frequencies need to be learned separately, as mentioned in Chapter 4. For negative drive frequencies, the disturbances during open-loop walking are significantly smaller, which influences the learning behavior for these frequencies.

In Figure 5.6 the development of the error over iterations is shown for an experiment with negative drive frequencies and a learning gain of $\beta = 0.3$. The initial error is already in the range of the converged error for positive drive frequencies. For iteration-invariant drive frequencies the error converges further, but when the drive frequency changes the error jumps back to approximately the initial value. This could lead to the conclusion that for negative drive frequencies ILC does not improve the performance. However, in Figure 5.7a it is shown that the position signals during later iterations are far more linear than those in the first iteration. These results are comparable to those for positive drive frequencies shown in Figure 5.7b.

The deviations from a linear trend line for open-loop walking experiments with positive and negative drive frequencies show comparable convergence behavior, as shown in Figure 5.8. Therefore, it is concluded that ILC can improve the performance for negative drive frequencies significantly when the drive-frequency dependent slope is compensated. As explained in Section 4.5, this requires hysteresis compensation which is outside of the scope of this thesis. For now, results for positive drive frequencies are presented as these give clearer performance improvements.

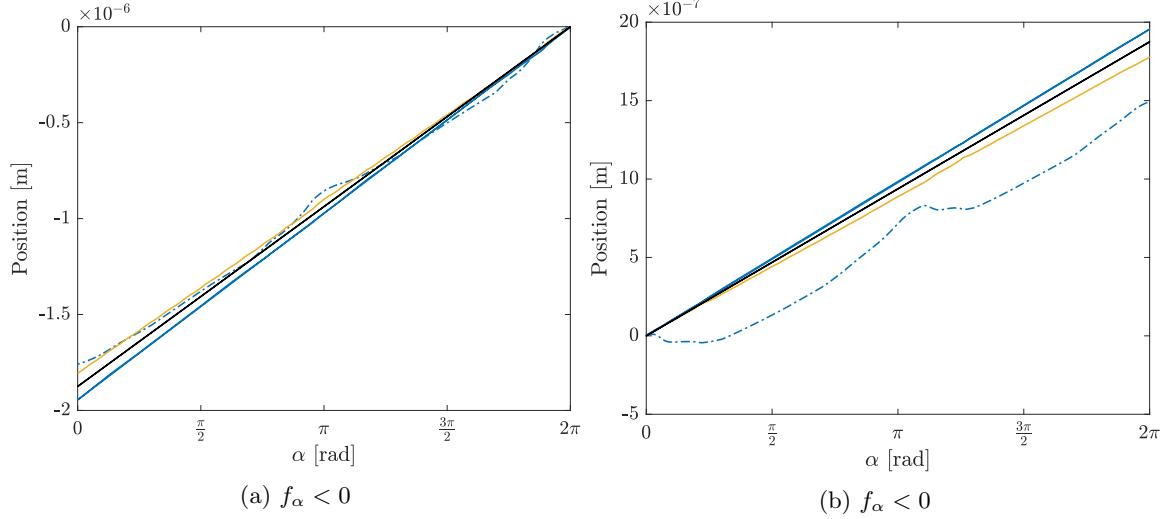


Figure 5.7: Position for positive and negative drive frequencies compared ($\beta = 0.3$). Iterations 0 (---), 17 (—), and 18 (—) are compared to the reference (—).

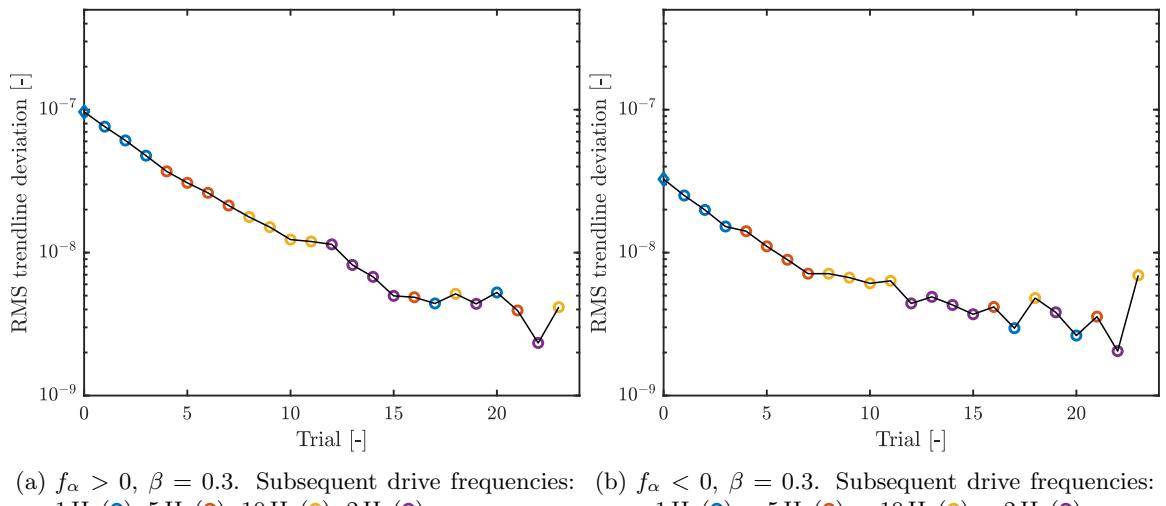


Figure 5.8: RMS values of the deviation from a linear trend line for positive (a) and negative (b) drive frequencies.

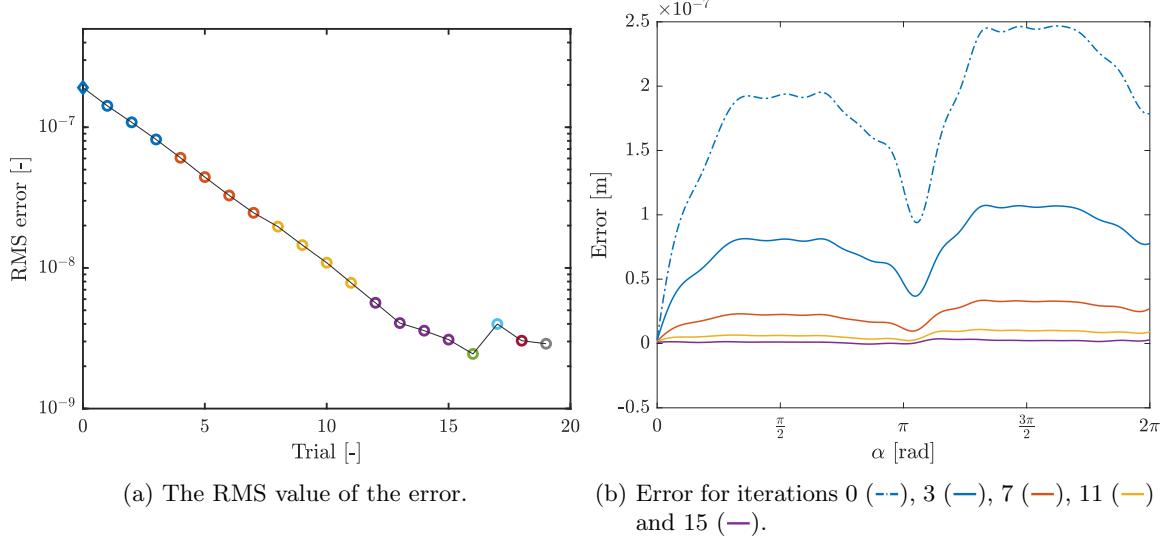


Figure 5.9: Error convergence during an open-loop clamping experiment with $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 3.6 \times 10^{-17}$. Subsequent drive frequencies: 1 Hz (blue circle), 7 Hz (orange circle), 11 Hz (yellow circle), 10 Hz (purple circle), 6 Hz (green circle), 2 Hz (blue circle), 9 Hz (red circle), 15 Hz (blue circle).

5.3 Experiments: α -domain ILC with basis functions

In this section α -domain ILC with basis functions is applied during open-loop clamping and walking for positive drive frequencies according to Algorithm 5.2. First, experimental results for open-loop clamping and walking with arbitrary, iteration-varying drive frequencies are presented. Secondly, experiments in which the drive frequency changes within steps and iterations are presented, to show the feasibility of the framework for non-equidistant sampling. Thirdly, an experiment in which the error is defined as the deviation from an iteration-varying trend line is presented to illustrate the ILC performance possibilities for a system without the rate-dependent effects of Section 4.5.

A set of 30 inverse quadratic basis functions with equidistant center points on the domain $\alpha \in [0, 2\pi)$ is used for all experiments presented in this section. The choice of weights influences the convergence behavior. Increasing W_u increases the robustness against model uncertainty, but it also increases the converged error value. Increasing $W_{\Delta u}$ decreases the sensitivity to iteration-varying behaviors, but it also decreases the convergence speed. In Appendix A.2 some examples of different cost function weights are given. For all experiments the weighting functions are chosen as constant scalars, i.e., $W_e(\alpha) = W_e \forall \alpha \in [0, 2\pi)$ etc..

5.3.1 Open-loop clamping and walking with arbitrary drive frequencies

During an open-loop clamping experiment the error converges monotonically to a bounded region, as shown in Figure 5.9. The RMS value of the error is reduced by a factor 50 in thirteen iterations. The weights of the cost function for this experiment are $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 3.6 \times 10^{-17}$. The drive frequencies vary between 1 Hz and 15 Hz and are arbitrary. During clamping, the influence of the drive-frequency dependent slope is limited, so that a change in drive frequency does not result in an increase in error.

Several experiments are done during open-loop walking with varying cost function weights and arbitrary drive frequencies that vary between 1 Hz and 9 Hz. In Figure 5.10, the averaged RMS value of the error and the fitted average error are shown for the weights $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 4.68 \times 10^{-17}$. A change in drive frequency results in an increase in error because of rate dependence of the step size (Section 4.5).

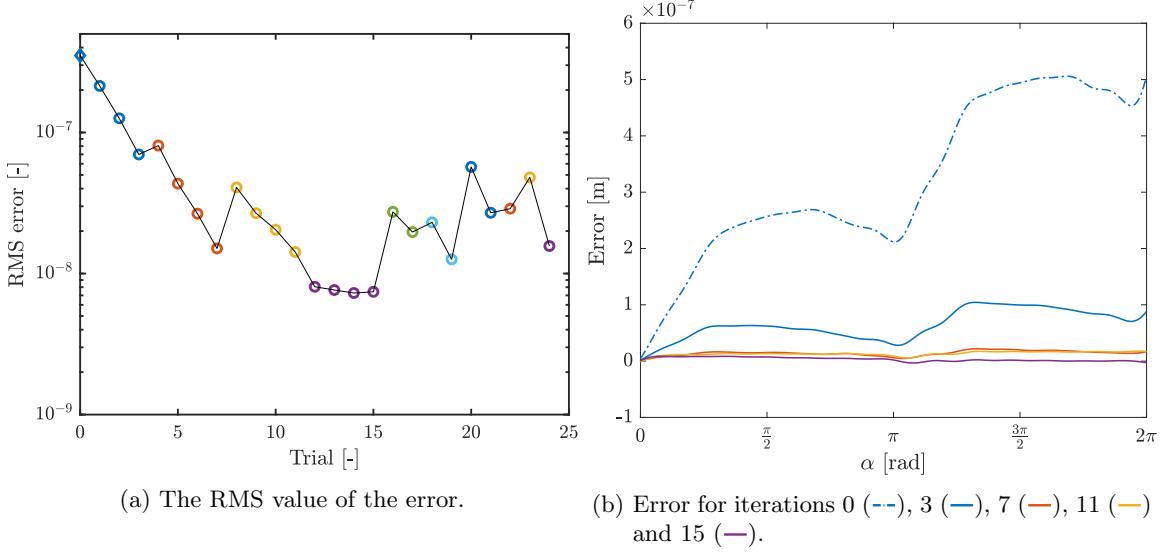


Figure 5.10: Error convergence during an open-loop walking experiment with $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 4.68 \times 10^{-17}$. Subsequent drive frequencies: 1 Hz (blue circle), 3 Hz (orange circle), 7 Hz (yellow circle), 5 Hz (purple circle), 9 Hz (green circle), 4 Hz (cyan circle).

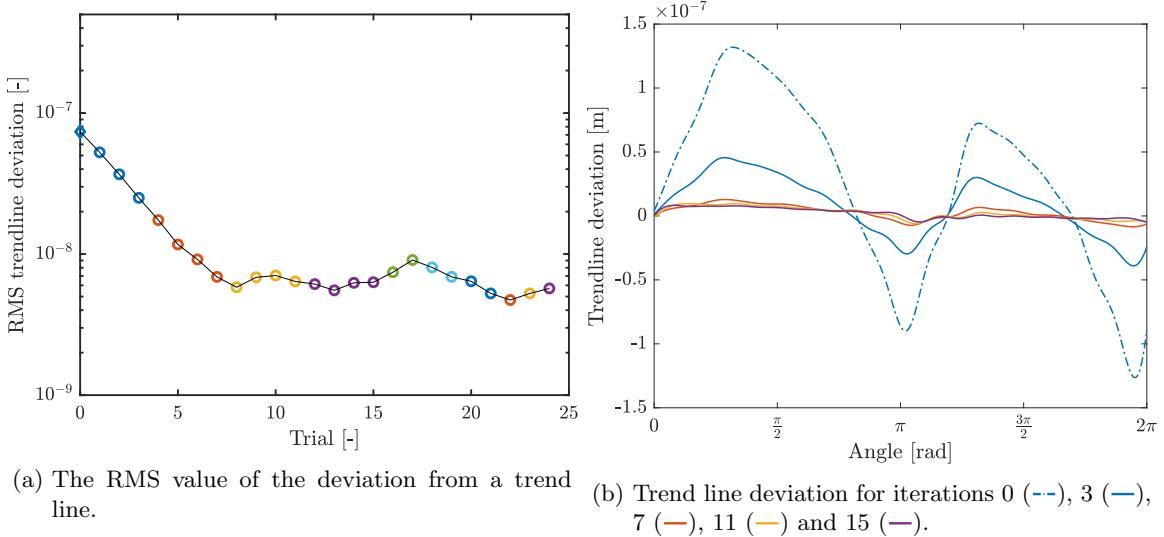


Figure 5.11: Deviation from a linear trend line during an open-loop walking experiment with $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 4.68 \times 10^{-17}$. Subsequent drive frequencies: 1 Hz (blue circle), 3 Hz (orange circle), 7 Hz (yellow circle), 5 Hz (purple circle), 9 Hz (green circle), 4 Hz (cyan circle).

In Figure 5.11, the deviation from a linear trend line is considered instead. The trend line is determined separately for each period by fitting a linear function through the position data. The resulting fit is subtracted from the data to remove the f_α -dependent slope from the results. In this figure, it is shown that the convergence is monotonic and that the deviation from a linear trend line is reduced significantly. Therefore, it is concluded that the α -domain repeating disturbance is compensated.

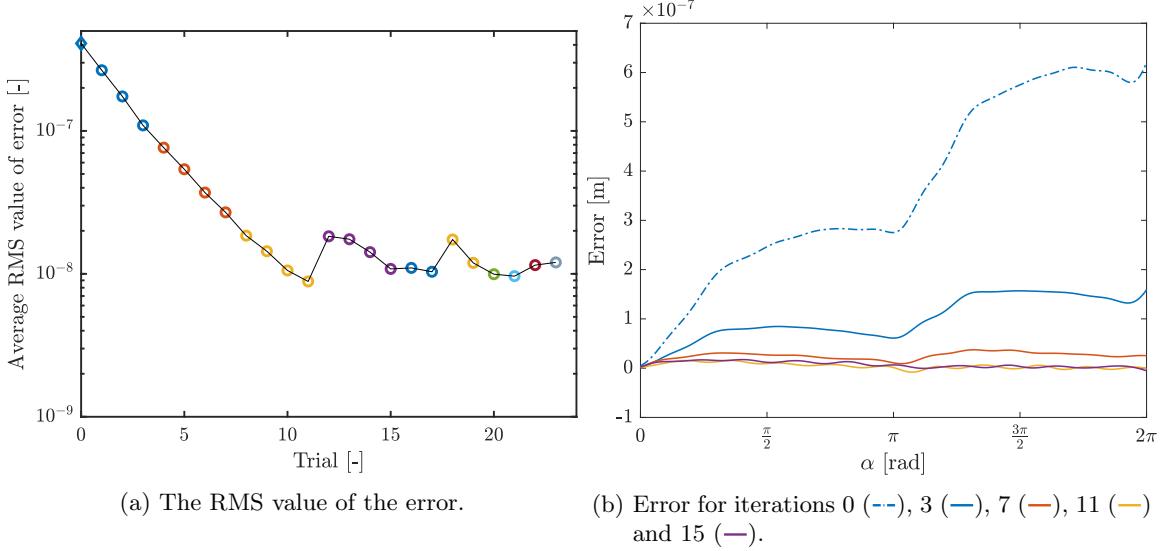


Figure 5.12: Error convergence during an open-loop walking experiment with $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 4.68 \times 10^{-17}$. Subsequent drive frequencies: 30 Hz (blue circle), 35 Hz (orange circle), 25 Hz (yellow circle), 28 Hz (purple circle), 22 Hz (green circle), 20 Hz (cyan circle), 27 Hz (red circle), 32 Hz (blue circle).

The rate-dependence of the step size decreases for higher drive frequencies, as is shown in the analysis in Section 4.5. Therefore, α -domain ILC with basis functions is applied at drive frequencies that vary between 20 Hz and 35 Hz. As expected, the convergence of the error observed in these experiments is closer to monotonicity than that for lower drive frequencies, as shown in Figure 5.12. However, the influence of the drive-frequency dependent slope is still visible. In Appendix A.3 additional results are shown for experiments with a larger range of drive frequencies, where $f_\alpha \in [20, 50]\text{Hz}$.

5.3.2 Open-loop walking with change of drive frequency within step

The framework for α -domain ILC with basis functions is applied during open-loop walking while the drive frequency changes within iterations and within steps. This results in iteration-varying and non-equidistant sampling, which also occurs when the piezo-stepper actuator is used in a closed-loop system where the drive frequency is the control variable. While a constant value of the drive frequency during one step or iteration can be realized in open-loop experiments, it is not a realistic representation of the circumstances in which α -domain ILC should be applicable. The drive frequency changes for this experiment are given in Table 5.1.

The error converges monotonically when α -domain ILC with basis functions is applied with varying drive frequencies within steps and iterations, as shown in Figure 5.13. The influence of the drive-frequency dependent slope on the averaged error is reduced, because the error is now averaged over the different drive frequencies used during one iteration. However, this does not solve the problem of rate dependency, because the size of the steps for different drive frequencies is still varying.

The position of the mover during iterations 0 and 9 is shown in Figure 5.14. The variation of drive frequency within the iterations is visible, as well as the effect of the compensating waveforms on the position. For the first iteration the α -domain repeating disturbances are clearly visible, whereas for the last iterations the position is a straight line.

This experiment shows that the framework for α -domain ILC with basis functions can be applied in a closed-loop setting where the drive frequency varies within steps and iterations. However, in a closed-loop setting partial steps and changes of the sign of the drive frequency within a step can occur. To use the framework in such a setting, hysteresis compensation is required. In addition, compensating waveforms for positive and negative drive frequencies need to be learned in parallel.

Table 5.1: Drive frequency changes within an iteration.

Iteration	t of change drive frequency	Drive frequencies
0, 1	[0, 2, 4, 6] s	[1, 2, 3, 4] Hz
2, 3	[0, 2.5, 4.5, 6.5] s	[1, 3, 7, 6] Hz
4, 5	[0, 2.5, 4.5, 6.5] s	[1, 6, 3, 8] Hz
6, 7	[0, 2.5, 4.5, 6.5] s	[1, 2.5, 4.5, 5.5] Hz
8, 9	[0, 2.5, 4.5, 6.5] s	[1, 3.8, 5.1, 3.6] Hz

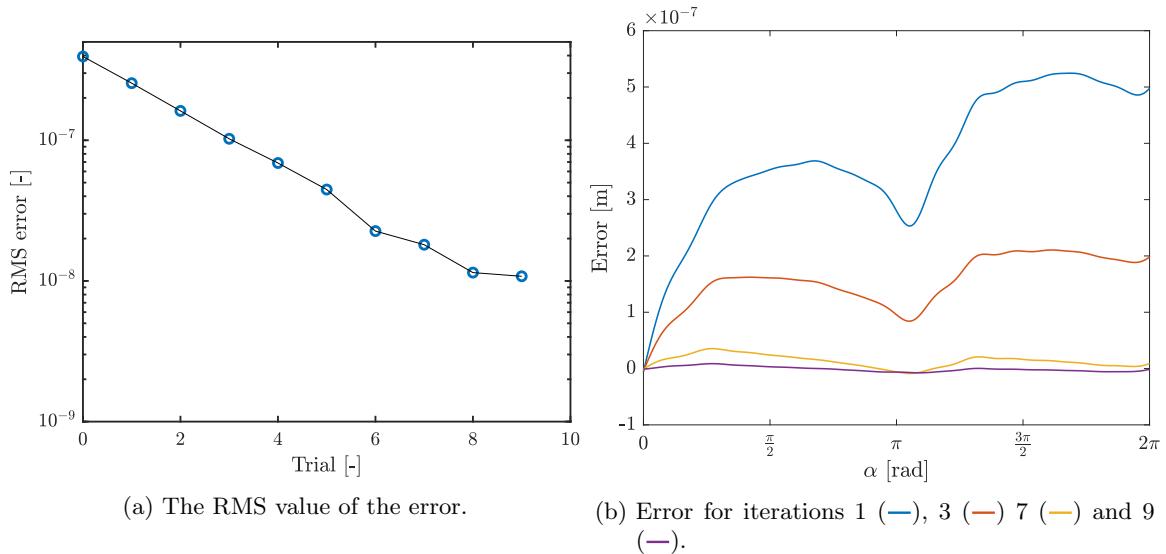


Figure 5.13: Error convergence during open-loop walking experiments with drive frequencies that vary within iterations and steps. $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 4.68 \times 10^{-17}$.

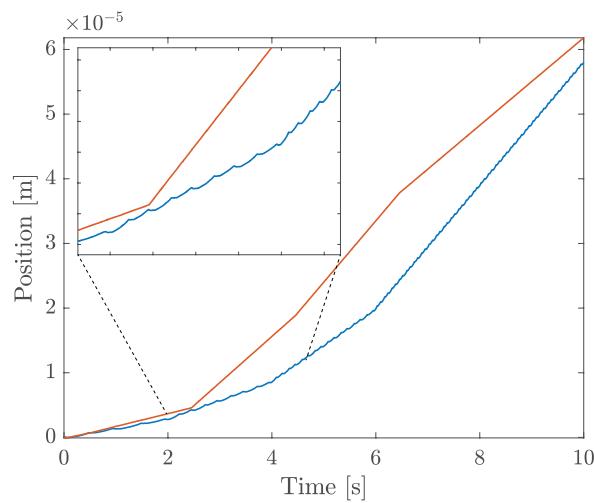


Figure 5.14: Position of the mover over time for iteration 0 (—) and 9 (—) of an open-loop walking experiment with changing drive frequencies within steps and iterations.

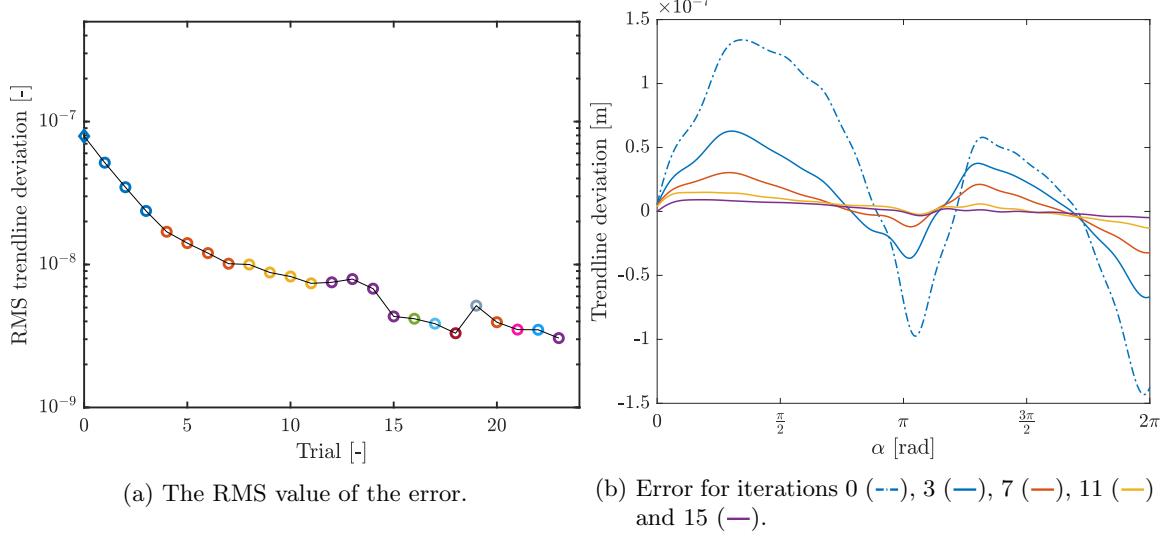


Figure 5.15: The error is shown to converge during an open-loop clamping experiment using α -domain ILC with basis functions and weights $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 4.68 \times 10^{-17}$ when the deviation from a linear trend line used as error. Subsequent drive frequencies: 1 Hz (blue circle), 7 Hz (orange circle), 11 Hz (yellow circle), 10 Hz (purple circle), 6 Hz (green circle), 2 Hz (cyan circle), 9 Hz (red circle), 15 Hz (blue circle), 5 Hz (purple circle), 12 Hz (blue circle).

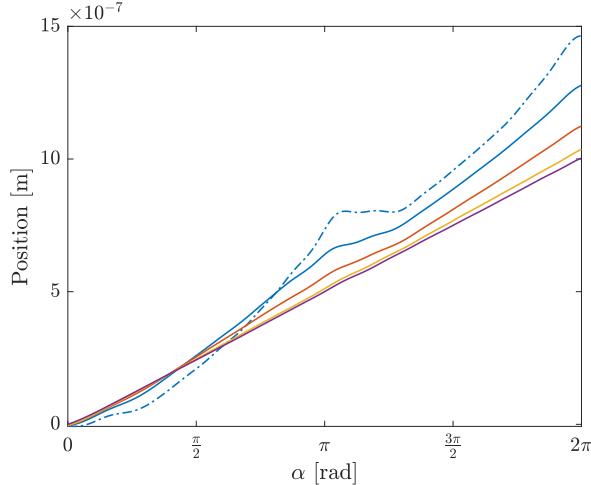


Figure 5.16: Position of the mover for iterations 0 (dashed blue), 3 (solid blue), 7 (solid orange), 11 (solid yellow), and 15 (solid purple).

5.3.3 Open-loop walking with a varying reference slope

To illustrate what performance could be achieved when the rate-dependent behavior is compensated, α -domain ILC with basis functions is applied to reduce the deviation from an iteration-varying linear trend line directly. Instead of defining the error as the deviation from an iteration-invariant reference, the error is now defined as the deviation from a straight line where the slope of the line is allowed to be varying. The deviation from such a varying linear trend line is shown to converge in Figure 5.15.

Remark. This approach is not suited for closed-loop implementation since the slope of the reference is not prevented from going to zero. In fact, in Figure 5.16 the results of this experiment show that the slope of the straight line that is obtained is lower than that of the reference line used earlier. However, this experiment shows that when there is no rate-dependent behavior the proposed framework can obtain convergence of the error that is close to monotonic, resulting in a significant improvement in performance.

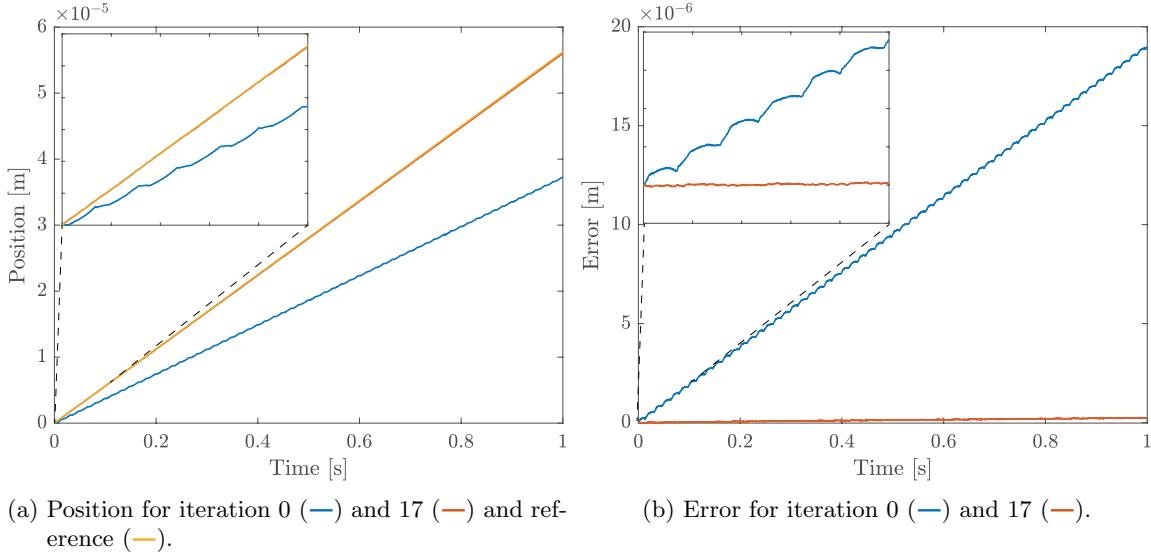


Figure 5.17: Position of the mover (a) and error (b) over time for iteration 0 (—) and 17 (—) of an open-loop walking experiment for a drive frequency of 30 Hz

5.4 Overview of experimental results

The frameworks for α -domain ILC can compensate the repeating disturbance for a piezo-stepper actuator, resulting in significant performance improvements. In Figure 5.17 the improvements in position and error are shown in the temporal domain for α -domain ILC with basis functions.

The two approaches to α -domain ILC converge to approximately the same converged error values during open-loop walking with arbitrary iteration-varying drive frequencies, as shown in Figure 5.18. In both cases, the position is improved significantly, as shown in Figure 5.19. The differences in transient behavior are explained by the choice of the learning gain for standard α -domain ILC and the choice of cost function weights for α -domain ILC with basis functions, which influence the convergence behavior. To determine the best update law parameters for the piezo-stepper actuator and get a better view of the achievable results, compensation of the rate-dependent behavior is required.

In Table 5.2 a summary of the most important results is given. The two approaches to α -domain ILC converge to the same bound on the error and the deviation from a linear trend line. The error during walking is reduced with a factor 6 even when the drive-frequency dependent slope causes an increase in error when the drive frequency changes. When the slope is removed from the position data, it is shown that the deviation from a straight trend line is reduced by a factor 8, and when the deviation from a linear trend line is used as error metric while learning, its RMS-value is reduced by a factor 15 over iterations. For higher drive frequencies, where the rate-dependence is smaller, the error is reduced by a factor 22. During clamping, when the drive-frequency dependent slope has less influence, the error is reduced by a factor 50.

The framework for α -domain ILC with basis functions is the preferred approach because of its computational benefits. The differences between the converged errors of the two methods are small, but the framework without basis functions is limited in the drive frequencies that can be used and it is computationally heavy. Experimental results show that α -domain ILC with basis functions can be used for drive frequencies up to at least 50 Hz (Appendix A.3), and that the framework can deal with variation of the drive frequency within iterations and steps.

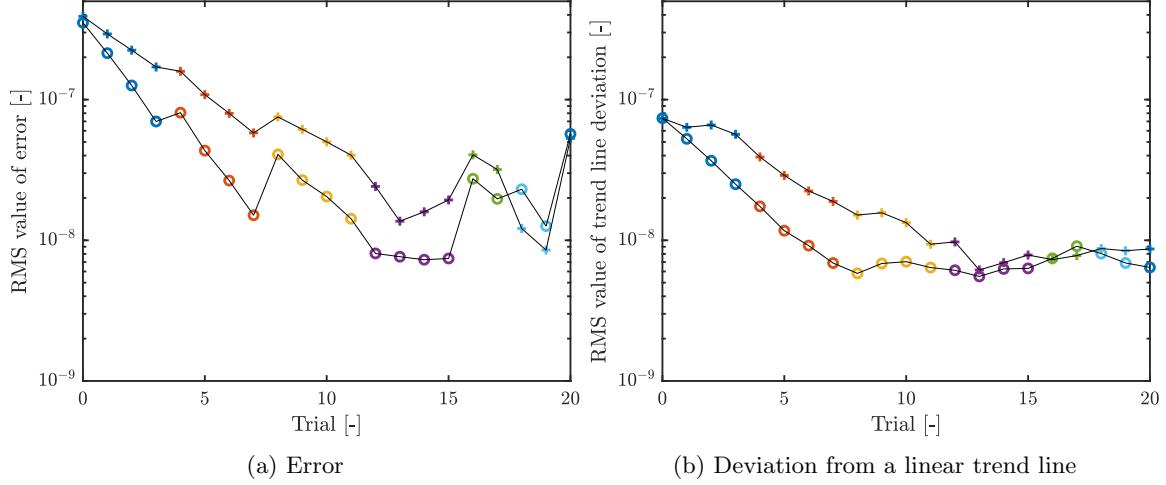


Figure 5.18: Comparison between α -domain ILC with (o) and without (+) basis functions. For the approach with basis functions, weights $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 4.68 \times 10^{-17}$ are used. For the approach without basis functions, learning gain $\beta = 0.3$. Subsequent drive frequencies: 1 Hz(o), 3 Hz(o), 7 Hz(o), 5 Hz(o), 9 Hz(o), 4 Hz(o).

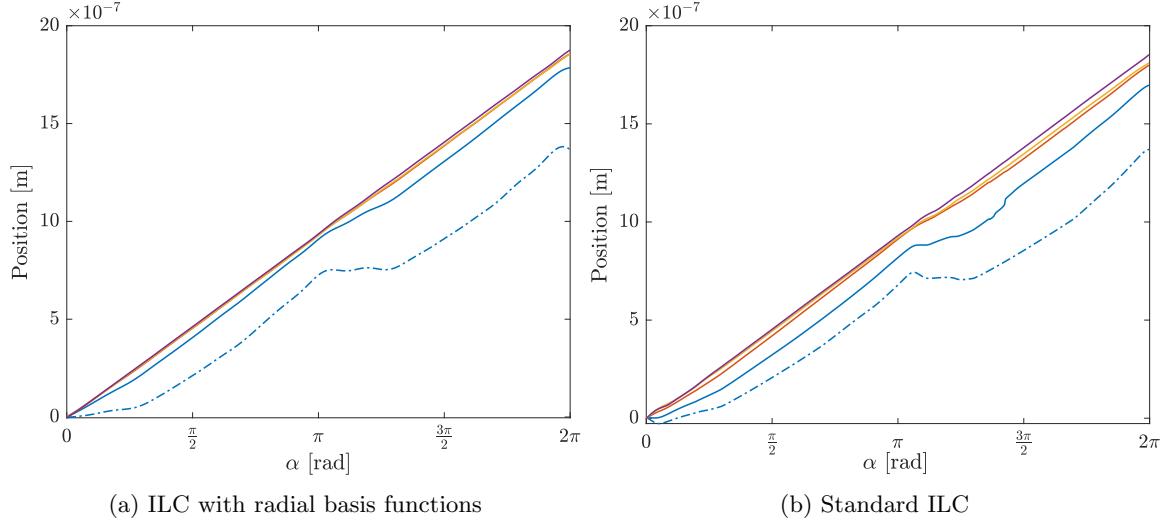


Figure 5.19: Position development over iterations for α -domain ILC with (a) and without (b) basis functions for iterations 0 (---), 3 (—), 7 (—), 11 (—), and 15 (—).

Table 5.2: Summary of results.

Approach	Initial value RMS(e)	Converged value RMS(e)
Walking, standard ILC	3.5×10^{-7}	5.5×10^{-8}
Walking, standard ILC, trend line deviation	7.7×10^{-8}	9.7×10^{-9}
Clamping, basis function ILC	2.0×10^{-7}	4.0×10^{-9}
Walking, basis function ILC	3.5×10^{-7}	5.5×10^{-8}
Walking, basis function ILC, trend line deviation	7.4×10^{-8}	9.1×10^{-9}
Walking, basis function ILC, variation within step	3.9×10^{-7}	1.1×10^{-8}
Walking, basis function ILC, reducing trend line deviation directly	7.9×10^{-8}	5.0×10^{-9}
Walking, basis function ILC with high f_α	4.1×10^{-7}	1.8×10^{-8}

5.5 Concluding remarks

Experiments show that both standard α -domain ILC and α -domain ILC with basis functions can compensate the α -domain repeating disturbance for a piezo-stepper actuator, improving the performance of the actuator significantly. The error converges to a bounded region during open-loop clamping and walking experiments with iteration-varying drive frequencies. A large part of this error is caused by a drive-frequency dependent slope in the position. When this slope is removed from the measurement data, the convergence of the deviation from a linear trend line converges to a small bounded region for both positive and negative drive frequencies.

α -domain ILC with basis functions has several advantages compared to the standard α -domain ILC framework, such as the continuous definition of the input signal and the small number of parameters that is learned. Experiments show that when basis functions are used, learning is possible at drive frequencies up to 50 Hz. In addition, it is possible to change the drive frequency within steps and iterations and update the input signal based on the resulting non-equidistantly sampled position signal.

The performance can be improved even further if the rate-dependent hysteresis in the system is compensated. Hysteresis compensation is also required for implementation of the framework in a closed-loop setting, where partial steps are taken and where the sign of the drive frequency can change within a step. Therefore, compensation of the rate-dependent hysteresis to improve the performance even further and allow implementation of the framework in a closed-loop setting is recommended for future work.

6 Conclusions

Two new frameworks for α -domain iterative learning control are presented that are capable of compensating disturbances that are repeating in the α -domain for a piezo-stepper actuator, while coping with iteration-varying and non-equidistant measurement and actuation points. In the first approach, measurements at iteration-varying non-equidistant sampling points are used to approximate the output at an iteration-invariant nominal set of points, at which the input signal is learned. Zero-order hold is used to implement the input signal at sample points outside of the nominal set. Conditions for convergence of the input and error to a bounded region are constructed by interpreting the variations caused by the approximations as bounded iteration-varying disturbances.

The second approach uses basis functions to parameterize the input and error signals. This has several advantages, such as an improved smoothness of the input signal which is now defined over a continuous domain and a significant reduction in the number of parameters to be learned. A norm-optimal update law for the input parameters is provided. In addition, monotonic convergence of the sequence of input parameters is shown when the norm-optimal update law and a linearly independent set of basis functions are used. Radial basis functions, for example of the inverse quadratic type, are shown to be suitable for use with the piezo-stepper actuator considered in this thesis.

The feasibility of both frameworks is validated experimentally using a piezo-stepper actuator. This actuator is modeled as a gain with disturbances. Since the dominant disturbances are shown to be repeating in the α -domain, the newly developed approaches to α -domain ILC can be applied. The piezo-stepper actuator waveforms are enhanced by learning a disturbance-compensating signal that is added to the standard input waveforms, leading to significant performance improvements during both open-loop clamping and walking experiments. Compensation of the error caused by the walking motion leads to a linear relation between commutation angle and mover position, which improves the positioning accuracy and reduces the complexity of closed-loop control in an industrial setting.

Recommendations

Several recommendations are given to enable the implementation of the α -domain ILC framework on a piezo-stepper in a closed-loop setting, and to improve some theoretical aspects of the framework.

- **Compensation of rate-dependent hysteresis**

Compensation of the rate-dependent hysteresis is expected to increase the performance of commutation-angle domain ILC. The iteration-varying behavior in the piezo-stepper actuator is reduced by removing the drive-frequency dependent slope. In addition, hysteresis compensation makes it possible to use ILC in a closed-loop setting where both the drive frequency and the sign of the drive frequency are allowed to vary within steps, and where partial steps may be taken.

- **Implementation in a closed-loop setting**

For implementation of α -domain ILC in a closed-loop setting where the drive frequency is the control variable, it needs to be possible to take partial steps and to change the sign of the drive frequency within a step. This requires parallel learning of enhanced waveforms for positive and negative drive frequencies. In addition, these waveforms need to be designed so that switching between them is possible without introducing new disturbances.

- **Selection of optimal basis functions**

For a piezo-stepper with hysteresis compensation in a closed-loop setting, the variables used in α -domain ILC with basis functions can be optimized. This includes selecting the cost function weights to meet certain performance criteria, selecting the number and type of basis functions that is used, and, in case of radial basis functions, selecting the shape parameters and center points. To determine the number of basis functions, it might be possible to use the general framework for optimization-based ILC presented in [12]. Within this framework, a cost function can be defined that takes into account the number of basis functions. However, this framework considers a sampled input and output at an iteration-invariant set of sample points. In case of

α -domain ILC, minimization of the number of basis functions should consider both the quality of the fit of the error, which can be determined at a finite number of sample points, and the factors taken into account in the continuous cost function presented in Chapter 3.

- **Influence of sampling on the quality of the fit**

In this thesis, it is assumed that the selected basis function can describe the measured error perfectly, so that the number and spread of the samples in the measured error signal does not influence the quality of the fit, provided that the number of parameters that is fitted is lower than or equal to the number of samples. Instead of making this assumption, the relation between the presence of measurement noise, the number and spread of samples and the quality of the fit might be investigated to quantify the influence of iteration-varying sampling on the convergence properties of the ILC framework.

References

- [1] R. Merry, M. Maassen, M. Van De Molengraft, N. Van De Wouw, and M. Steinbuch, "Modeling and waveform optimization of a nano-motion piezo stage," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 4, pp. 615–626, 2011.
- [2] M. Den Heijer, V. Fokkema, A. Saedi, P. Schakel, and M. J. Rost, "Improving the accuracy of walking piezo motors," *Rev. Sci. Instrum.*, vol. 85, no. 5, p. 055007, 2014.
- [3] A. Fleming and K. Leang, *Design , Modeling and Control of Nanopositioning Systems*. Cham: Springer, 2014.
- [4] E. Shamoto and T. Moriwaki, "Development of a "walking drive" ultraprecision positioner," *Precis. Eng.*, vol. 20, no. 2, pp. 85–92, 1997.
- [5] Y. Egashira, K. Kosaka, S. Takada, T. Iwabuchi, T. Baba, S. Moriyama, T. Harada, K. Nagamoto, A. Nakada, H. Kubota, and T. Ohmi, "0.69 nm resolution ultrasonic motor for large stroke precision stage," in *Proc. IEEE Conf. Nanotechnol.*, (Maui, USA), pp. 397–402, 2001.
- [6] T. Uzunovic, E. Golubovic, and A. Sabanovic, "Piezo LEGS driving principle based on coordinate transformation," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 3, pp. 1395–1405, 2015.
- [7] P. Tacx, "Performance analysis and feasibility of advanced control for a walking piezo actuator," 2018.
- [8] N. Strijbosch, P. Tacx, E. Verschueren, and T. Oomen, "Commutation angle iterative learning control: enhancing piezo-stepper actuator waveforms," in *IFAC Symp. Mechatron. Syst.*, (Vienna, Austria), 2019.
- [9] T. Y. Jiang, T. Y. Ng, and K. Y. Lam, "Optimization of a piezoelectric ceramic actuator," *Sensors and Actuators*, vol. 84, pp. 81–94, 2000.
- [10] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "Survey of iterative learning control: a learning-based method for high-performance tracking control," *IEEE Control Syst.*, vol. 26, no. 3, pp. 96–114, 2006.
- [11] L. Blanken, R. de Rozario, J. van Zundert, S. Koekebakker, M. Steinbuch, and T. Oomen, "Advanced feedforward and learning control for mechatronic systems," in *Proc. 3th DSPE Conf. Precis. Mechatronics*, (Sint Michielsgestel, The Netherlands), pp. 79–86, 2016.
- [12] T. Oomen and C. R. Rojas, "Sparse iterative learning control with application to a wafer stage: achieving performance, resource efficiency, and task flexibility," *Mechatronics*, vol. 47, pp. 134–147, 2017.
- [13] D. Hoelzle and K. Barton, "On spatial iterative learning control via 2-D convolution: stability analysis and computational efficiency," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, pp. 1504–1512, 2016.
- [14] F. Kong, A. M. Boudali, and I. Manchester, "Phase-indexed ILC for control of underactuated walking robots," in *2015 IEEE Conf. Control Appl.*, (Sydney, Australia), pp. 1467–1472, 2015.
- [15] K. Leang and S. Devasia, "Design of hysteresis-compensating iterative learning control for piezo-positioners: application to atomic force microscopes," *Mechatronics*, vol. 16, no. 3-4, pp. 141–158, 2006.
- [16] C. Chen and G. Chiu, "Spatially periodic disturbance rejection with spatially sampled robust repetitive control," *J. Dyn. Syst. Meas. Control*, vol. 130, no. 2, 2008.
- [17] Z. Cao and G. F. Ledwich, "Adaptive repetitive control to track variable periodic signals with fixed sampling rate," *IEEE/ASME Trans. Mechatronics*, vol. 7, no. 3, pp. 378–384, 2002.

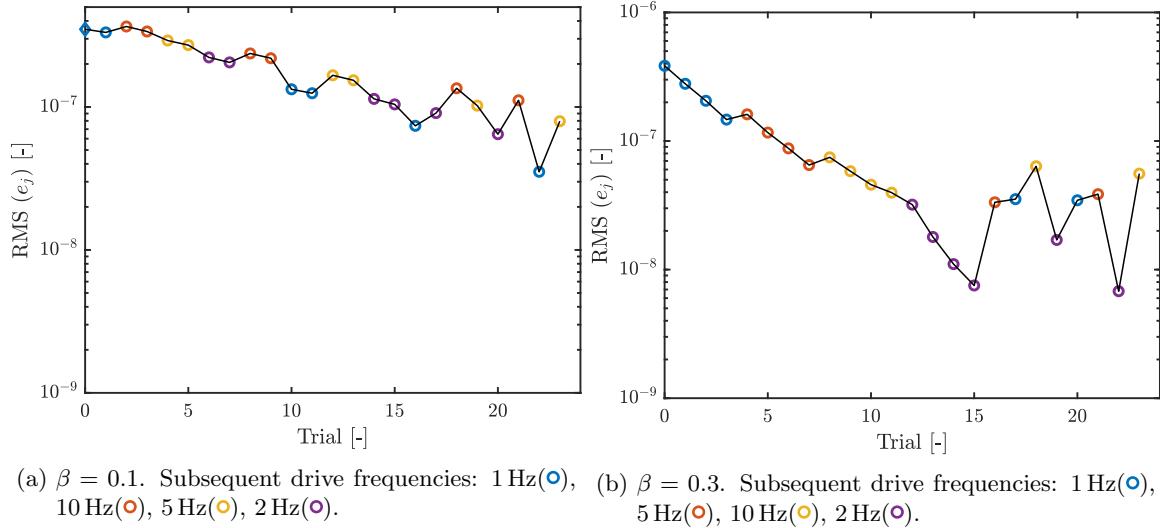
- [18] R. J. E. Merry, D. J. Kessels, W. P. M. H. Heemels, M. J. G. Van De Molengraft, and M. Steinbuch, “Delay-varying repetitive control with application to a walking piezo actuator,” *Automatica*, vol. 47, no. 8, pp. 1737–1743, 2011.
- [19] X. Li and D. Shen, “Two novel iterative learning control schemes for systems with randomly varying trial lengths,” *Syst. Control Lett.*, vol. 107, pp. 9–16, 2017.
- [20] N. Strijbosch and T. Oomen, “Beyond quantization in iterative learning control : exploiting time-varying time-stamps,” in *Proc. 2019 Am. Control Conf.*, (Philadelphia, PA, USA), pp. 2984–2989, 2019.
- [21] L. Aarnoudse, C. Pannier, Z. Afkhami, T. Oomen, and K. Barton, “Multi-layer spatial iterative learning control for micro-additive manufacturing,” in *8th IFAC Symp. Mechatron. Syst.*, (Vienna, Austria), 2019.
- [22] B. Altin and K. Barton, “Learning control of linear iteration varying systems with varying references through robust invariant update laws,” in *Proc. 2015 Am. Control Conf.*, (Chicago, USA), pp. 4880–4885, 2015.
- [23] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control*. Wiley, 2 ed., 2005.
- [24] J. Van De Wijdeven and O. H. Bosgra, “Using basis functions in iterative learning control: analysis and design theory,” *Int. J. Control*, vol. 83, no. 4, pp. 661–675, 2010.
- [25] S. Mishra and M. Tomizuka, “Projection-based iterative learning control for wafer scanner systems,” *IEEE/ASME Trans. Mechatronics*, vol. 14, no. 3, pp. 388–393, 2009.
- [26] J. Bolder, T. Oomen, S. Koekebakker, and M. Steinbuch, “Using iterative learning control with basis functions to compensate medium deformation in a wide-format inkjet printer,” *Mechatronics*, vol. 24, no. 8, pp. 944–953, 2014.
- [27] J. Van Zundert, J. Bolder, and T. Oomen, “Optimality and flexibility in Iterative Learning Control for varying tasks,” *Automatica*, vol. 67, pp. 295–302, 2016.
- [28] F. Gantmacher, *The Theory of Matrices*. New York: Chelsea Publishing Company, 1959.
- [29] C. S. Chen, Y. C. Hon, and R. A. Schaback, *Scientific Computing with Radial Basis Functions*. 2007.
- [30] K. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge: The MIT Press, 2012.
- [31] R. Schaback and H. Wendland, “Kernel techniques: from machine learning to meshless methods,” *Acta Numer.*, pp. 543–639, 2006.
- [32] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, “Kernel methods in system identification, machine learning and function estimation: a survey,” *Automatica*, vol. 50, no. 3, pp. 657–682, 2014.
- [33] L. Blanken, I. Van den Meijdenberg, and T. Oomen, “Inverse system estimation for feedforward: a kernel-based approach for non-causal systems,” in *18th IFAC Symp. Syst. Identification, Stock.*, pp. 1050–1055, 2018.
- [34] M. Buhmann, *Radial Basis Functions: Theory and Implementations*. Cambridge: Cambridge University Press, 2009.
- [35] Y. C. Hon and R. Schaback, “On unsymmetric collocation by radial basis functions,” *Appl. Math. Comput.*, vol. 119, pp. 177–186, 2001.
- [36] R. Schaback, “Convergence of unsymmetric kernel-based meshless collocation methods,” *SIAM J. Numer. Anal.*, vol. 45, no. 1, pp. 333–351, 2007.
- [37] R. Schaback, “Unsymmetric meshless methods for operator equations,” *Numer. Math.*, vol. 114, pp. 629–651, 2010.

- [38] D. Croft, G. Shed, and S. Devasia, “Creep, hysteresis, and vibration compensation for piezoactuators: atomic force microscopy application,” *J. Dyn. Syst. Meas. Control*, vol. 123, pp. 35–43, 2001.
- [39] S. Devasia, E. Eleftheriou, and S. Moheimani, “A Survey of control issues in nanopositioning,” *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 5, pp. 802–823, 2007.
- [40] C. Newcomb and I. Flinn, “Improving the linearity of piezoelectric ceramic actuators,” *Electron. Lett.*, vol. 18, no. 11, pp. 442–444, 1982.

A Application to a piezo-stepper actuator: experiments

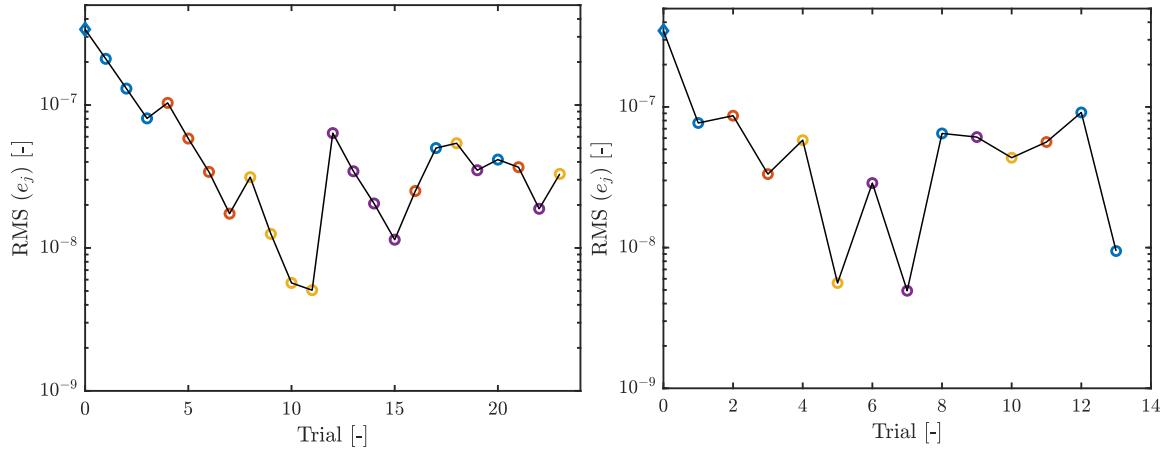
A.1 Influence of the learning gain

In this section, some convergence results are shown for varying values of the learning gain β . The influence of the value of β on the error convergence is analyzed during open-loop walking experiments with a limited set of drive frequencies that are all divisors of the nominal drive frequency. The input signal is learned on a set of $N = 1000$ points, and the drive frequencies $f_\alpha \in [1, 2, 5, 10]$ Hz all generate samples on these points. The learning gains $\beta = 1$, $\beta = 0.5$, $\beta = 0.3$ and $\beta = 0.1$ are compared in the Figures A.1 and A.2. A high learning gain results in fast convergence, but the influence of the drive-frequency dependent slope when the drive frequency changes is large. For a lower learning gain the increases in error when the drive frequency changes are smaller, but the convergence speed reduces.



(a) $\beta = 0.1$. Subsequent drive frequencies: 1 Hz(○), 5 Hz(○), 10 Hz(○), 2 Hz(○). (b) $\beta = 0.3$. Subsequent drive frequencies: 1 Hz(○), 5 Hz(○), 10 Hz(○), 2 Hz(○).

Figure A.1: Influence of learning gain β on convergence.



(a) $\beta = 0.5$. Subsequent drive frequencies: 1 Hz(○), 5 Hz(○), 10 Hz(○), 2 Hz(○). (b) $\beta = 1$. Subsequent drive frequencies: 1 Hz(○), 5 Hz(○), 10 Hz(○), 2 Hz(○), 5 Hz(○).

Figure A.2: Influence of learning gain β on convergence.

A.2 Influence of the cost function weights

For α -domain ILC with basis functions the cost function weights W_e , W_u and $W_{\Delta u}$ are the tuning parameters for the convergence behavior. In Figure A.3, the averaged RMS values of the deviation from a linear trend line are shown for two different sets of cost function weights. Increasing the value of W_u increases the final error value after convergence, and decreasing $W_{\Delta u}$ increases the sensitivity to iteration-varying disturbances.

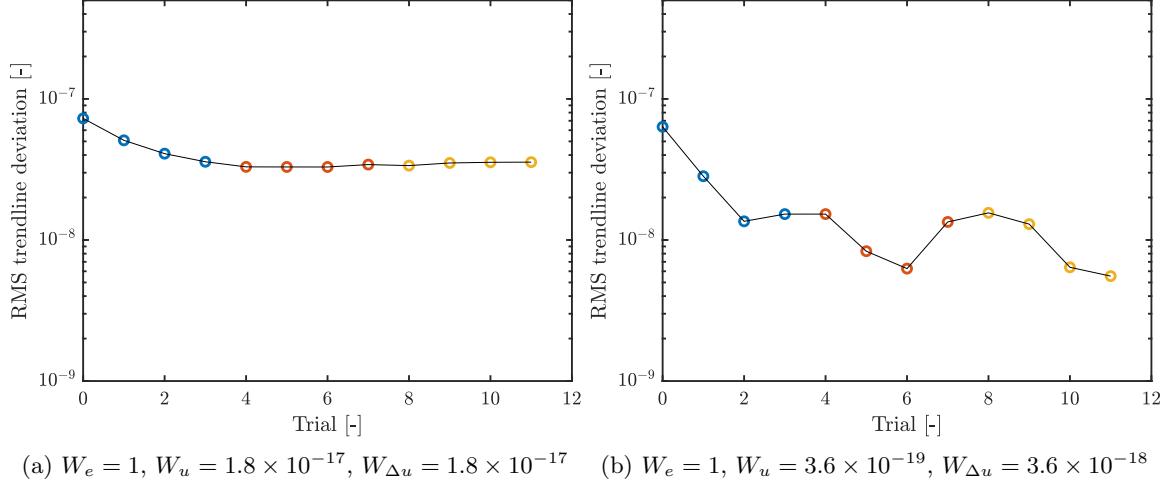


Figure A.3: Convergence for different cost function weights. Subsequent drive frequencies: 1 Hz(○), 3 Hz(○), 7 Hz(○).

A.3 α -domain ILC with basis functions at high drive frequencies

α -domain ILC with basis functions is applied during walking at drive frequencies ranging from 20 Hz to 50 Hz. As expected, the convergence of the error, which is shown in Figure A.4, is closer to monotonicity than that for lower drive frequencies, since the influence of the rate-dependent behavior is smaller at these frequencies. However, the influence from the drive-frequency dependent slope is still visible, for example between iteration 17 and 18. In addition, a vibration that might be related to the experimental setup is visible in the error signal at later iterations.

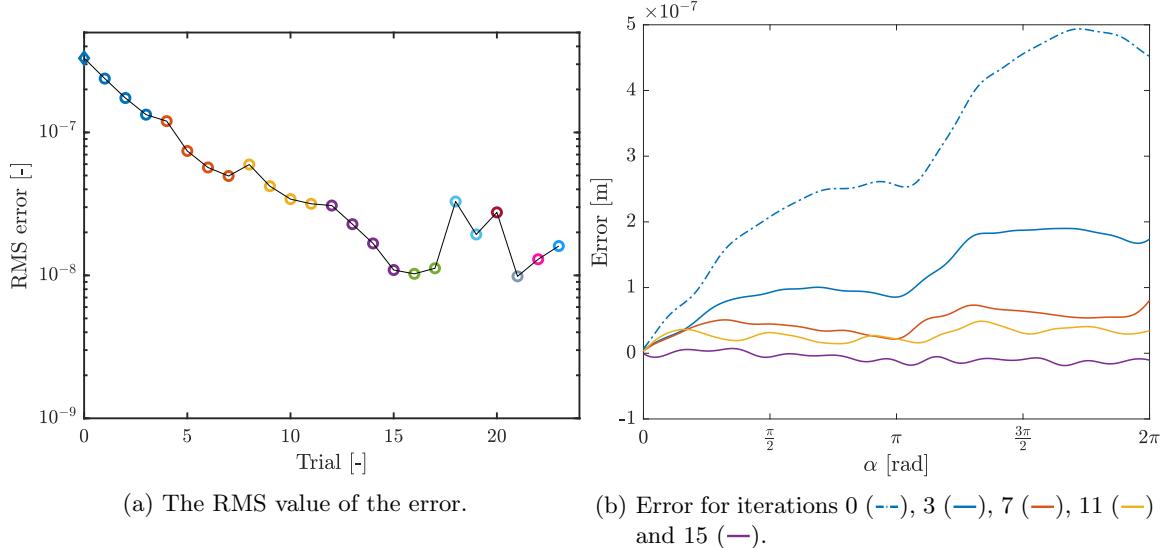


Figure A.4: The error is shown to converge during an open-loop clamping experiment using α -domain ILC with basis functions. $W_e = 1, W_u = 0, W_{\Delta u} = 4.68 \times 10^{-17}$. Subsequent drive frequencies: 20 Hz(○), 30 Hz(○), 50 Hz(○), 25 Hz(○), 32 Hz(○), 44 Hz(○), 22 Hz(○), 27 Hz(○), 38 Hz(○), 40 Hz(○).

Commutation-Angle Iterative Learning Control with Iteration-Varying Non-Equidistant Data Points: Enhancing Piezo-Stepper Actuator Waveforms

Leontine Aarnoudse * Nard Strijbosch * Edwin Verschueren **
Tom Oomen *

* Eindhoven University of Technology, Department of Mechanical Engineering, P.O. Box 513, 5600 MB Eindhoven (e-mail:
l.i.m.aarnoudse@student.tue.nl, n.w.a.strijbosch@tue.nl)
** Thermo Fisher Scientific, Eindhoven, The Netherlands

Abstract: Piezo-stepper actuators are used in many nanopositioning systems due to their high resolution, high stiffness, fast response, and the ability to position a mover over an infinite stroke by means of motion reminiscent of walking. The aim of this paper is to develop a control approach for attenuating disturbances that are caused by the walking motion and are therefore repeating in the commutation-angle domain. A new iterative learning control approach is developed for the commutation-angle domain, that addresses the iteration-varying and non-equidistant sampling that occurs when the piezo-stepper actuator is driven at varying drive frequencies by parameterizing the input and error signals. Experimental validation of the framework on a piezo-stepper actuator leads to significant performance improvements.

Keywords: Iterative Learning Control, Feedforward Control, Motion Control Systems, Piezo Actuators, Micromechantronic Systems

1. INTRODUCTION

Many nanopositioning systems use piezo-stepper actuators to meet increasing requirements for high precision positioning that arise due to developments in the field of nanotechnology. Applications, such as nano-motion stages (Merry et al. (2011)) and scanning probe microscopy (Den Heijer et al. (2014)), require the high resolution, high stiffness, and fast response of the piezoelectric elements as well as a large mover stroke which is provided by a motion that resembles walking. There are various ways to implement this walking motion, for example by walking drives (Shamoto and Moriawaki (1997)) or bi-morph legs (Uzunovic et al. (2015)).

During the walking motion of a piezo-stepper actuator, engagement and release between the piezo elements and the mover can lead to repeating disturbances (Den Heijer et al. (2014); Strijbosch et al. (2019)). The piezo-stepper actuator is actuated using waveforms that describe the mapping from the commutation angle to the input voltage of the piezo elements. The disturbances are repeating with the period of these actuating waveforms and lead to a nonlinear relation between commutation angle and mover position, for which control typically is difficult.

For varying velocities, the error profile caused by these disturbances is varying. In industrial implementations, piezo-

stepper actuators are driven using varying drive frequencies with a constant sampling frequency in the temporal domain. Since the disturbances are repeating with the period of the actuating waveform, they are varying in the temporal domain for varying drive frequencies. In the commutation-angle domain, in which the waveforms are repeating, the disturbances are repeating. However, the sampling in the commutation-angle domain is varying and possibly non-equidistant.

Learning control approaches such as iterative learning control (ILC) can compensate iteration-invariant disturbances perfectly, but they may amplify iteration-varying disturbances. In ILC, a feedforward input signal is modified based on preceding experiments that use the same reference, so that the tracking error is reduced over iterations (Bristow et al. (2006)). Iteration-invariant disturbances are compensated perfectly, but typical ILC approaches amplify trial-varying disturbances significantly (Oomen and Rojas (2017)). Therefore, temporal domain ILC is not suited for a piezo-stepper actuator and commutation-angle domain approach is needed instead.

Existing approaches to ILC outside of the temporal domain depend strongly on assumptions regarding the sampling in the spatial domain. In Hoelzle and Barton (2016), a 2D spatial ILC framework for micro-additive manufacturing is developed, in which the output of the system is measured at an iteration-invariant number of discrete points in space. In Kong et al. (2015), phase-indexed ILC is developed for a walking robot that behaves almost

* This work is part of the research programme VIDI with project number 15698, which is (partly) financed by the Netherlands Organization for Scientific Research (NWO).

periodically, for which it is assumed that in the limit stable periodic behavior is obtained. These approaches cannot cope with iteration-varying non-equidistant sampling.

In Strijbosch et al. (2019) an approach is presented that uses ILC in the commutation-angle domain to reduce the repeating disturbances for a piezo-stepper actuator through waveform enhancement. However, it is assumed that continuous-time signals are available which is not feasible in a digital setting. In addition, learning is only applied during so-called ‘clamping’ experiments and not during the actual walking motion of the piezo-stepper actuator. For commutation-angle ILC, the iteration-varying non-equidistant sampling needs to be considered explicitly.

Existing approaches to learning control for iteration-varying and non-equidistant sampling do not consider the specific sampling situation of commutation-angle ILC. In Li and Shen (2017), iteration-varying sampling caused by incomplete trials is considered, where the sampling is equidistant and it is assumed that a constant sampling length is obtained in the limit. A framework for ILC with non-equidistant sampling is proposed in Strijbosch and Oomen (2019). In this approach the output is sampled at a high rate, which is assumed to be a multiple of the lower rate at which the input is sampled. Only part of the sampled output signal is used, resulting in iteration-varying non-equidistant sampling. For a piezo-stepper actuator, the input and output signals are sampled at the same iteration-varying rate, which in general cannot be related to a constant nominal sampling rate.

Although there exist frameworks for ILC outside of the temporal domain as well as frameworks that consider iteration-varying sampling, there is no framework that is suitable for ILC outside of the temporal domain with iteration-varying non-equidistant sampling. This paper aims to develop such a framework for commutation-angle iterative learning control, suitable for systems such as a piezo-stepper actuator. This leads to the following two contributions:

- (1) A framework for commutation-angle domain ILC with iteration-varying and non-equidistant sampling, that uses basis functions to parameterize the input signal, is proposed.
- (2) The framework is implemented during walking experiments with a piezo-stepper actuator, resulting in significant performance improvements.

This paper is organized as follows. In Section 2 the functioning of a piezo-stepper actuator is explained and the problem formulation is given. In Section 3 the proposed framework for commutation-angle domain ILC is presented. In Section 4 the framework is experimentally validated using a piezo-stepper actuator and lastly, conclusions are given in Section 5. All proofs are given in the appendices.

2. PROBLEM FORMULATION

2.1 Piezo-stepper actuators

Piezo-stepper actuators consist of a combination of longitudinal and shear elements in varying configurations,

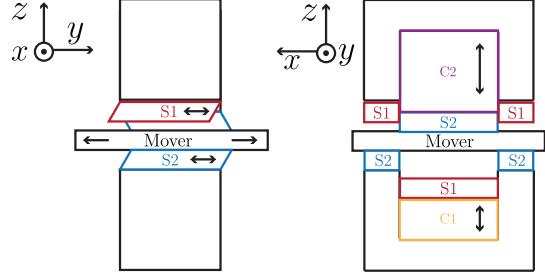


Fig. 1. Schematic representation of a piezo-stepper actuator showing the clamp ('C') and shear ('S') elements of the first (—, —) and second (—, —) group.

see e.g. Shamoto and Moriwaki (1997); Uzunovic et al. (2015); Naikwad et al. (2016). The piezo-stepper actuator considered in this paper consists of two groups of piezo elements, each containing one longitudinal or ‘clamp’ element and three shear elements, as shown in Fig. 1. When the clamp element of a group is extended, the corresponding shear elements are in contact with the mover. The mover follows the displacement of the connected shear elements. Alternating the two piezo groups results in a walking motion, which leads to an unlimited stroke of the mover.

The cyclic walking motion of the piezo-stepper is implemented using the periodic waveforms shown in Fig. 2. These waveforms map the commutation angle $\alpha \in [0, 2\pi]$ [rad] to the input voltage of the piezo elements. The number of steps per second is determined by the drive frequency f_α [Hz]. This drive frequency is integrated to obtain the commutation angle, i.e.,

$$\alpha(t) = 2\pi \int_0^t f_\alpha(\tau) d\tau. \quad (1)$$

The system is driven with varying drive frequencies and sampled at a constant frequency f_s [Hz] in the temporal domain. For a single step, the vector containing the α -values at which a sample is taken is given by

$$\bar{\alpha} = 2\pi \left[\int_0^h f_\alpha(\tau) d\tau \int_0^{2h} f_\alpha(\tau) d\tau \dots \int_0^{Nh} f_\alpha(\tau) d\tau \right]^\top \quad (2)$$

with sample interval $h = f_s^{-1}$ [s]. The number of samples within a step is given by $N = \lfloor T f_s \rfloor$, where for T [s] it holds that $\alpha(T) = 2\pi$. This is implemented in an open-loop setting.

2.2 Modeling

The piezo-stepper actuator is modeled as a gain with a lumped disturbance in the α -domain. It is assumed that rate-dependent effects such as creep and hysteresis are negligible, since these can be compensated using a separate feedforward (Fleming and Leang (2014), ch. 2,11). The modeled piezo-stepper actuator is described in the α -domain without any significant time-domain dynamics.

The displacement of a single shear element y_i [m] is given by

$$y_i(\alpha(t)) = c s_i(\alpha(t)), \quad i = 1, 2, \quad (3)$$

with positive piezo constant c [m V^{-1}] and shear inputs s_i [V], $i = 1, 2$.

The waveforms are designed to obtain a linear relation

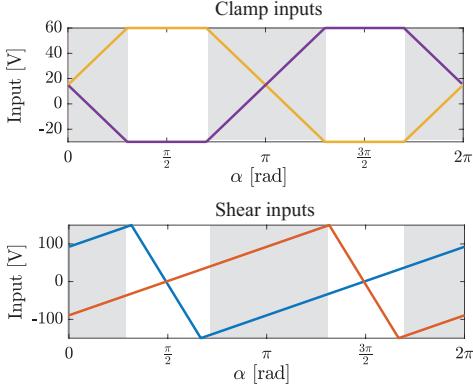


Fig. 2. The waveforms for clamps 1 (—) and 2 (—) contain regions where both clamps could be in contact with the mover, indicated in gray. In these regions the inputs for the shear elements 1 (—) and 2 (—) have equal derivatives.

between the commutation angle α and the mover displacement y . Therefore, the waveforms of the shear elements are designed such that they have equal derivatives for any α where both clamps could be in contact with the mover, as shown in Fig. 2. The part of the step where one of the clamps is completely retracted is used to reset the corresponding shear elements. Because the connected shear elements are always moving with the same velocity, the combination of the two shear inputs $s_1(\alpha)$ and $s_2(\alpha)$ is written as a single input $u_s(\alpha)$ [V], satisfying

$$\frac{\delta u_s(\alpha)}{\delta \alpha} = \begin{cases} \frac{\delta s_1(\alpha)}{\delta \alpha} & \text{if } \alpha \in [\frac{\pi}{3}, \frac{2\pi}{3}] \\ \frac{\delta s_2(\alpha)}{\delta \alpha} & \text{if } \alpha \in [\frac{4\pi}{3}, \frac{5\pi}{3}] \\ \frac{\delta s_1(\alpha)}{\delta \alpha} = \frac{\delta s_2(\alpha)}{\delta \alpha} & \text{otherwise.} \end{cases} \quad (4)$$

The corresponding desired mover position is given by

$$y_d(\alpha) = h_0 u_s(\alpha) \quad (5)$$

with piezo constant h_0 [m V⁻¹]. During open-loop experiments the desired linear relation between commutation angle and mover position is not obtained due to disturbances. These disturbances are assumed to be relatable to the commutation angle, and are modeled by a lumped disturbance $d_\alpha(\alpha)$ [m]. Therefore, the position of the mover with disturbances is described by

$$y(\alpha(t)) = h_0 u_s(\alpha(t)) + d_\alpha(\alpha(t)). \quad (6)$$

When a single step of the piezo-stepper actuator is considered, the system is written in terms of α as

$$y(\alpha) = h_0 u_s(\alpha) + d_\alpha(\alpha), \quad \alpha \in [0, 2\pi]. \quad (7)$$

2.3 Repeatability of the disturbances

During the walking motion, the desired linear relation between commutation angle and mover position is not obtained, since the position of the piezo-stepper actuator shows disturbances that are repeating with the period of the actuating waveform. In Fig. 3a and 3b, the disturbance is plotted in the temporal domain and the α -domain, respectively, for different drive frequencies. In the temporal domain the sampling is equidistant and equal for different drive frequencies, but the disturbance is drive-frequency dependent. In the α -domain, the disturbance is repeating for different drive frequencies. However, the number of samples within a step and the distance between

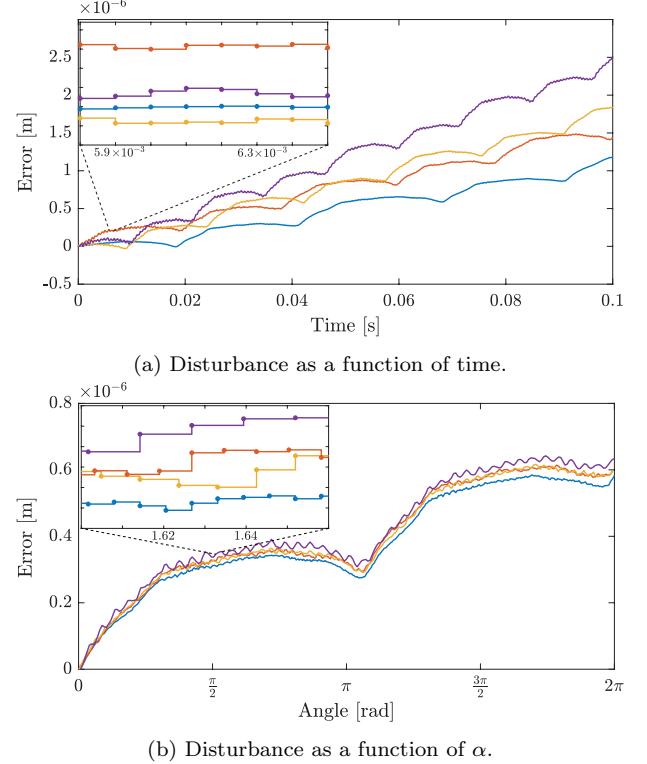


Fig. 3. Disturbances for a piezo-stepper during open-loop walking with drive frequencies 20 Hz (—), 25 Hz (—), 30 Hz (—) and 40 Hz (—). In the temporal domain (a) the sampling is equidistant (see zoom plot) but the disturbance is not repeating for different drive frequencies. In the α -domain (b) the sampling is non-equidistant for varying drive frequencies, but the disturbances are similar.

the samples in the α -domain, given by (2), is varying for varying drive frequencies.

This paper aims to develop a framework for α -domain iterative learning control with iteration-varying and non-equidistant sampling for systems with a dominant α -domain repeating disturbance. This framework can be used to increase the performance and reduce the influence of the α -domain repeating disturbance for a piezo-stepper actuator using the waveform enhancement method presented in Strijbosch et al. (2019).

3. α -DOMAIN ILC WITH BASIS FUNCTIONS

In this section, the framework for α -domain iterative learning control with basis functions is presented and convergence conditions are given.

3.1 Assumptions

The following assumptions are made for α -domain ILC.

Assumption 1. The initial condition $y_j(0)$ is identical for each iteration j .

Assumption 2. The length of each iteration is constant in the α -domain, i.e., $\alpha_j \in [0, 2\pi]$.

Assumption 3. For each iteration there is a unique mapping $F_j : [0, T_j] \mapsto [0, 2\pi]$ from the time interval $t \in [0, T_j]$ to the commutation-angle interval $\alpha \in [0, 2\pi]$.

Assumption 4. The basis functions $\psi(\alpha)$ can be scaled to describe the α -domain disturbance and reference exactly, i.e., $d_\alpha(\alpha) = \psi^\top(\alpha)\theta^d$ and $y_d(\alpha) = \psi^\top(\alpha)\theta^{yd}$.

Assumption 1 is satisfied by defining the initial position for each iteration to be $y(0) = 0$. Assumption 2 is satisfied by choosing the measurement time so that $\alpha(0) = 0$ and $\alpha(T_j) = 2\pi$, which allows varying iteration lengths in the temporal domain. Assumption 3 is satisfied when α is continuously increasing or decreasing within an iteration. Assumption 4 is satisfied by using suitable basis functions.

3.2 Approach

Iterative learning control in the α -domain is used to compensate the α -domain repeating disturbance at iteration-varying drive frequencies. ILC cannot be applied directly to the sampled input and output signals, since the sampling is iteration-varying. The vector containing the α -values at which a sample is taken in iteration j is given by $\bar{\alpha}_j \in \mathbb{R}^{N_j \times 1}$ according to (2). Since $\bar{\alpha}_j$ is iteration-varying, and the sample points may be non-equidistant, the input and error signals are parameterized using basis functions to obtain continuous descriptions. ILC is applied to these continuous signals, and the learned input signal is sampled for implementation. The continuous system to which ILC is applied is given by

$$y_j(\alpha) = h_0(u_s(\alpha) + u_j(\alpha)) + d_\alpha(\alpha), \quad (8)$$

$$e_j(\alpha) = y_d(\alpha) - y_j(\alpha), \quad (9)$$

with standard shear input $u_s(\alpha) = \psi(\alpha)^\top \theta^{us}$, disturbance-compensating input $u_j(\alpha)$ and error $e_j(\alpha)$. The compensating input is constructed using basis functions as

$$u_{j+1}(\alpha) = \psi^\top(\alpha)\theta_{j+1}^u. \quad (10)$$

The basis function vector ψ , containing M linearly independent basis functions, and parameter vector θ_j^u are given by

$$\psi(\alpha) = [\psi_1(\alpha) \ \psi_2(\alpha) \ \dots \ \psi_M(\alpha)]^\top \in \mathbb{R}^{M \times 1}, \quad (11)$$

$$\theta_j^u = [\theta_{1,j}^u \ \theta_{2,j}^u \ \dots \ \theta_{M,j}^u]^\top \in \mathbb{R}^{M \times 1}. \quad (12)$$

An outline of the approach to α -domain ILC for n iterations is given in Algorithm 1.

Algorithm 1.

Choose a basis ψ , see Section 4

for $j = 1 : n$ **do**

 Perform an experiment for one step with $f_{\alpha,j}$

 Find θ_j^e using a least squares fit as described in Section 3.3

 Update the input parameters θ_{j+1}^u as described in Section 3.4

 Update the input according to (10): $u_{j+1} = \psi^\top \theta_{j+1}^u$

 Divide u_{j+1} into waveforms $s_{1,j+1}$ and $s_{2,j+1}$ according to Section 4.2

end for

3.3 Error parameterization

The sampled error signal is parameterized using the same set of functions that forms the basis for the input u_j , since the system is assumed to behave as a gain. For each iteration, the error is sampled for all $\alpha \in \bar{\alpha}_j$, resulting in the sampled error signal $\bar{e}_j(\bar{\alpha}_j)$. Using (8), (9), (10) and

Assumption 4, the parameterized error for iteration j is given by

$$e_j(\alpha) = \psi^\top(\alpha)(\theta^{yd} - h_0(\theta^{us} + \theta_j) - \theta^d) = \psi^\top(\alpha)\theta_j^e. \quad (13)$$

The vector of parameters $\theta_j^e \in \mathbb{R}^{M \times 1}$ that provides the optimal fit for \bar{e}_j at the sample points $\bar{\alpha}_j$ is determined using a least squares fit, where the following cost function is minimized:

$$\mathcal{J}_e(\theta_j^e) = \sum_{i=1}^{N_j} (\bar{e}_j(i) - \psi^\top(\bar{\alpha}_j(i))\theta_j^e)^2 \quad (14)$$

The following theorem gives the optimal parameter vector θ_j^e for the fit of the sampled error.

Theorem 5. Consider the sampled error vector \bar{e}_j of the system described by (8) and (9) for which a fit over a continuous domain is given by $e_j = \psi^\top(\alpha)\theta_j^e$. If the sampled basis functions in $\bar{\psi}_j$ are linearly independent, the parameter vector that gives the unique least-squares optimal fit in terms of the cost function (14) for \bar{e}_j is given by:

$$\theta_j^e = (\bar{\psi}_j \bar{\psi}_j^\top)^{-1} \bar{\psi}_j \bar{e}_j, \text{ with} \quad (15)$$

$$\bar{\psi}_j = [\psi_1(\bar{\alpha}_j) \ \psi_2(\bar{\alpha}_j) \ \dots \ \psi_M(\bar{\alpha}_j)]^\top. \quad (16)$$

3.4 ILC update law

A continuous ILC update law is developed to determine the input parameters θ_{j+1}^u for iteration $j+1$, using the continuously defined input and error signals of iteration j . To find the optimal input parameters, the continuous cost function \mathcal{J} is minimized, which is given by

$$\begin{aligned} \mathcal{J}(\theta_{j+1}^u) = & \int_0^{2\pi} (W_e(\alpha)e_{j+1}(\alpha)^2 + W_u(\alpha)u_{j+1}(\alpha)^2 \\ & + W_{\Delta u}(\alpha)(u_{j+1}(\alpha) - u_j(\alpha))^2) \, d\alpha. \end{aligned} \quad (17)$$

The weights $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta u}(\alpha)$ are non-negative functions that are tuned to obtain certain performance and robustness properties. $W_e(\alpha)$ influences the performance of the learning, $W_u(\alpha)$ influences the robustness against model uncertainty and $W_{\Delta u}(\alpha)$ influences the attenuation of iteration-varying disturbances.

The following theorem gives the optimal update for the vector of input parameters θ_j^u .

Theorem 6. Consider the system described by (8) and (9) for which the input $u_{j+1}(\alpha) = \psi^\top(\alpha)\theta_{j+1}^u$ of iteration $j+1$ is constructed using any given set of linearly independent basis functions $\psi(\alpha)$. The cost function (17) with non-negative weight functions $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta u}(\alpha)$, of which at least one is positive for all $\alpha \in [0, 2\pi]$, leads to an optimal update of the parameters θ_{j+1}^u given by

$$\theta_{j+1}^u = Q_\psi \theta_j^u + L_\psi \theta_j^e, \text{ with} \quad (18)$$

$$\begin{aligned} Q_\psi = & \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) \right. \\ & \left. + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^\top(\alpha) \, d\alpha \right)^{-1} \\ & \int_0^{2\pi} (h_0^2 W_e(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^\top(\alpha) \, d\alpha \end{aligned}$$

$$L_\psi = \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) \right. \\ \left. + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^\top(\alpha) d\alpha \right)^{-1} \\ \int_0^{2\pi} W_e(\alpha) h_0 \psi(\alpha) \psi^\top(\alpha) d\alpha.$$

3.5 Monotonic convergence

To avoid large learning transients and ensure convergence to a unique input signal, the ILC system needs to be monotonically convergent. Conditions for monotonic convergence of α -domain ILC with basis functions are given by the following theorem.

Theorem 7. Consider the system described in (8) and (9) with input (10). For the update law (18), with non-negative weight functions $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta,u}(\alpha)$ of which at least $W_e(\alpha)$ or $W_u(\alpha)$ is positive for all $\alpha \in [0, 2\pi]$, and linearly independent basis functions ψ , the sequence of parameter vectors $\{\theta_j^u\}_{j \in \mathbb{N}}$ is monotonically convergent in the 2-norm towards a fixed parameter vector θ^{u*} .

4. EXPERIMENTAL RESULTS

To validate the framework for α -domain ILC with basis functions, it is applied during a series of walking experiments with a piezo-stepper actuator. First, suitable basis functions are selected. Then, experimental results are presented.

4.1 Basis function selection

The input and error signals are parameterized using a set of 30 inverse quadratic radial basis functions. Basis functions in ILC are typically chosen based on prior knowledge regarding the disturbance or reference, such as the origin (Bolder et al. (2014)) or the shape (Mishra and Tomizuka (2009)). In α -domain ILC, the basis functions are chosen based on how well they describe the sampled error signal. The M inverse quadratic radial basis functions are linearly independent (Schaback and Wendland (2006), ch. 5) and given by

$$\psi_k(\alpha) = \frac{1}{1 + (\|\alpha - c_k\|)^2}, \quad k = 1, 2, \dots, M, \quad (19)$$

where the center points c_k of the radial basis functions are divided equidistantly over the domain $[0, 2\pi]$.

The set of 30 inverse quadratic radial basis functions is used to fit error signals with 1000 and 50 equidistant samples, as shown in Fig. 4. For a fit using 1000 samples, the root mean square (RMS) of the difference between sampled error signal and fit is approximately 2.4×10^{-9} . The RMS value of the difference between this fit and a fit using a downsampled signal of 50 samples is approximately 8.6×10^{-10} . Therefore, it is concluded that the basis functions describe the error well and that the influence of the number of available sample points on the quality of the fit is negligible, provided that the number of samples is larger than or equal to the number of basis functions.

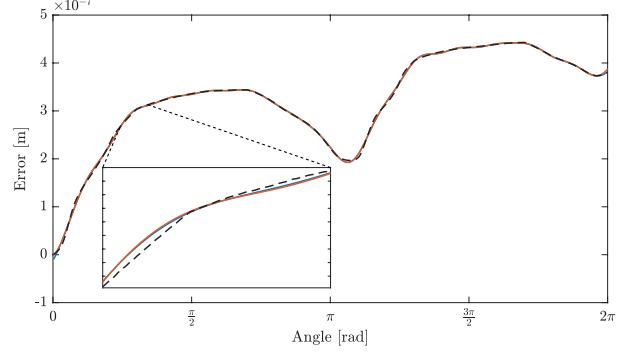


Fig. 4. Comparison between the sampled error signal (---) and fits using 30 inverse quadratic radial basis functions and 1000 (—) or 50 (—) samples.

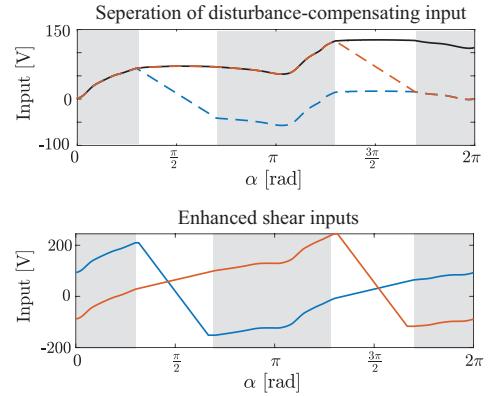


Fig. 5. Waveform enhancement using a learned disturbance-compensating input signal. Regions where both groups could be in contact with the mover are indicated in gray. A compensating input signal (—) is divided into two inputs for the shear elements 1 (---) and 2 (---). These inputs are added to the standard waveforms, resulting in enhanced waveforms for shear elements 1 (—) and 2 (—).

4.2 Experimental results

The input signal $u_j(\alpha)$, which is learned using ILC, is separated into two inputs $s_{i,j}$, $i = 1, 2$ for the two groups of shear elements, satisfying (4). These shear inputs are added to the standard waveforms, resulting in enhanced waveforms as is shown in Fig. 5.

In the experiments scalar cost function weights are used, i.e., $W_e(\alpha) = W_e \forall \alpha \in [0, 2\pi]$ etc.. The desired linear relation between the commutation angle α and the position of the mover y is described by the reference $y_d(\alpha) = 3 \times 10^{-7} \alpha$.

During an open-loop walking experiment with iteration-varying drive frequencies ranging between 20 – 35 Hz, the error is reduced significantly and converges to a bounded region, as shown in Fig. 6 and 7. At iterations 12 and 18, a change in drive frequency causes an increase of the RMS value of the error. This is caused by rate-dependent behaviors in the piezo shear elements, which can be compensated using feedforward (Croft et al. (2001)).

The improvements in the temporal domain are shown in

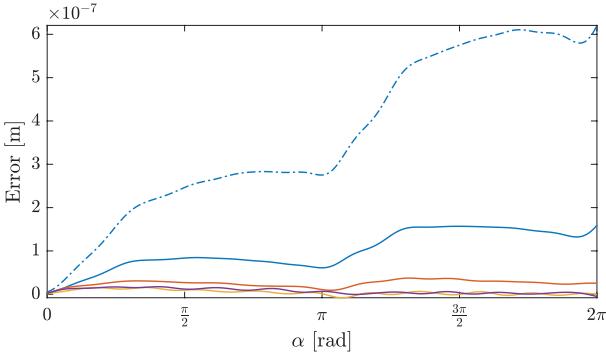


Fig. 6. Error convergence over iterations 0 (30 Hz, ---), 3 (30 Hz, —), 7 (35 Hz, —), 11 (25 Hz, —) and 15 (28 Hz, —).

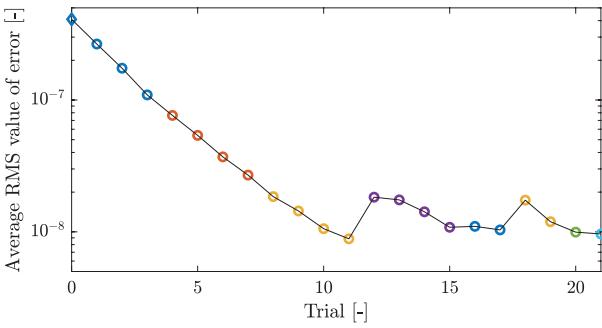


Fig. 7. Convergence of the RMS value of the error during an open-loop walking experiment with $W_e = 1$, $W_u = 0$ and $W_{\Delta u} = 4.7 \times 10^{-17}$. Subsequent drive frequencies: 30 Hz (blue circle), 35 Hz (orange circle), 25 Hz (yellow circle), 28 Hz (purple circle), 22 Hz (green circle), 20 Hz (blue circle).

Fig. 8, where the position of the mover for iterations 0 and 17 is compared. It is shown that the repeating disturbance is compensated, so that a linear relation between commutation angle and mover position is obtained.

5. CONCLUSION

A new framework for α -domain iterative learning control is presented that is capable of fully mitigating repeatable disturbances in the α -domain for a piezo-stepper actuator, while coping with iteration-varying and non-equidistant measurement and actuation points. Basis functions are used to parameterize the input and error signals and obtain continuous descriptions. These continuous descriptions are used in an optimal ILC update law. Compensation of the α -domain repeating disturbances for a piezo-stepper actuator during walking experiments results in a linear relation between commutation angle and mover position. This improves the positioning accuracy and reduces the complexity of closed-loop control in an industrial setting.

REFERENCES

- Bolder, J., Oomen, T., Koekebakker, S., and Steinbuch, M. (2014). Using iterative learning control with basis functions to compensate medium deformation in a wide-format inkjet printer. *Mechatronics*, 24(8), 944–953.
- Bristow, D.A., Tharayil, M., and Alleyne, A.G. (2006). Survey of iterative learning control: a learning-based method for high-performance tracking control. *IEEE Control Syst.*, 26(3), 96–114.
- Croft, D., Shed, G., and Devasia, S. (2001). Creep, hysteresis, and vibration compensation for piezoactuators: atomic force microscopy application. *J. Dyn. Syst. Meas. Control*, 123, 35–43.
- Den Heijer, M., Fokkema, V., Saedi, A., Schakel, P., and Rost, M.J. (2014). Improving the accuracy of walking piezo motors. *Rev. Sci. Instrum.*, 85(5), 055007.
- Fleming, A. and Leang, K. (2014). *Design, Modeling and Control of Nanopositioning Systems*. Springer, Cham.
- Gantmacher, F. (1959). *The Theory of Matrices*. Chelsea Publishing Company, New York.
- Gunnarsson, S. and Norrlöf, M. (2001). On the design of ILC algorithms using optimization. *Automatica*, 37(12), 2011–2016.
- Hoelzle, D. and Barton, K. (2016). On spatial iterative learning control via 2-D convolution: stability analysis and computational efficiency. *IEEE Trans. Control Syst. Technol.*, 24(4), 1504–1512.
- Kong, F., Boudali, A.M., and Manchester, I. (2015). Phase-indexed ILC for control of underactuated walking robots. In *2015 IEEE Conf. Control Appl.*, 1467–1472. Sydney, Australia.
- Li, X. and Shen, D. (2017). Two novel iterative learning control schemes for systems with randomly varying trial lengths. *Syst. Control Lett.*, 107, 9–16.
- Merry, R., Maassen, M., Van De Molengraft, M., Van De Wouw, N., and Steinbuch, M. (2011). Modeling and waveform optimization of a nano-motion piezo stage. *IEEE/ASME Trans. Mechatronics*, 16(4), 615–626.
- Mishra, S. and Tomizuka, M. (2009). Projection-based iterative learning control for wafer scanner systems. *IEEE/ASME Trans. Mechatronics*, 14(3), 388–393.
- Naikwad, S., Vandervelden, R., and Hosseini, S.H. (2016). A novel hybrid self-sensing method for force estimation in a piezo-stepper actuator. In *Proc. 4th Int. Conf. Control. Mechatronics Autom.*, 162–167. Barcelona, Spain.
- Oomen, T. and Rojas, C.R. (2017). Sparse iterative learning control with application to a wafer stage: achieving performance, resource efficiency, and task flexibility. *Mechatronics*, 47, 134–147.
- Schaback, R. and Wendland, H. (2006). Kernel techniques: from machine learning to meshless methods. *Acta Numer.*, 543–639.
- Shamoto, E. and Moriwaki, T. (1997). Development of a “walking drive” ultraprecision positioner. *Precis. Eng.*,

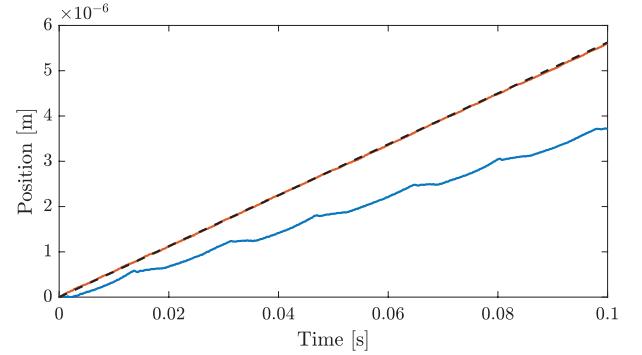


Fig. 8. Position of the mover over time for iteration 0 (—) and 17 (—) and reference (---) with $f_\alpha = 30$ Hz.

- method for high-performance tracking control. *IEEE Control Syst.*, 26(3), 96–114.
- Croft, D., Shed, G., and Devasia, S. (2001). Creep, hysteresis, and vibration compensation for piezoactuators: atomic force microscopy application. *J. Dyn. Syst. Meas. Control*, 123, 35–43.
- Den Heijer, M., Fokkema, V., Saedi, A., Schakel, P., and Rost, M.J. (2014). Improving the accuracy of walking piezo motors. *Rev. Sci. Instrum.*, 85(5), 055007.
- Fleming, A. and Leang, K. (2014). *Design, Modeling and Control of Nanopositioning Systems*. Springer, Cham.
- Gantmacher, F. (1959). *The Theory of Matrices*. Chelsea Publishing Company, New York.
- Gunnarsson, S. and Norrlöf, M. (2001). On the design of ILC algorithms using optimization. *Automatica*, 37(12), 2011–2016.
- Hoelzle, D. and Barton, K. (2016). On spatial iterative learning control via 2-D convolution: stability analysis and computational efficiency. *IEEE Trans. Control Syst. Technol.*, 24(4), 1504–1512.
- Kong, F., Boudali, A.M., and Manchester, I. (2015). Phase-indexed ILC for control of underactuated walking robots. In *2015 IEEE Conf. Control Appl.*, 1467–1472. Sydney, Australia.
- Li, X. and Shen, D. (2017). Two novel iterative learning control schemes for systems with randomly varying trial lengths. *Syst. Control Lett.*, 107, 9–16.
- Merry, R., Maassen, M., Van De Molengraft, M., Van De Wouw, N., and Steinbuch, M. (2011). Modeling and waveform optimization of a nano-motion piezo stage. *IEEE/ASME Trans. Mechatronics*, 16(4), 615–626.
- Mishra, S. and Tomizuka, M. (2009). Projection-based iterative learning control for wafer scanner systems. *IEEE/ASME Trans. Mechatronics*, 14(3), 388–393.
- Naikwad, S., Vandervelden, R., and Hosseini, S.H. (2016). A novel hybrid self-sensing method for force estimation in a piezo-stepper actuator. In *Proc. 4th Int. Conf. Control. Mechatronics Autom.*, 162–167. Barcelona, Spain.
- Oomen, T. and Rojas, C.R. (2017). Sparse iterative learning control with application to a wafer stage: achieving performance, resource efficiency, and task flexibility. *Mechatronics*, 47, 134–147.
- Schaback, R. and Wendland, H. (2006). Kernel techniques: from machine learning to meshless methods. *Acta Numer.*, 543–639.
- Shamoto, E. and Moriwaki, T. (1997). Development of a “walking drive” ultraprecision positioner. *Precis. Eng.*,

20(2), 85–92.

Strijbosch, N., Tacx, P., Verschueren, E., and Oomen, T. (2019). Commutation angle iterative learning control: enhancing piezo-stepper actuator waveforms. In *IFAC Symp. Mechatron. Syst.* Vienna, Austria.

Strijbosch, N. and Oomen, T. (2019). Beyond quantization in iterative learning control : exploiting time-varying time-stamps. In *Proc. 2019 Am. Control Conf.*, 2984–2989. Philadelphia, PA, USA.

Uzunovic, T., Golubovic, E., and Sabanovic, A. (2015). Piezo LEGS driving principle based on coordinate transformation. *IEEE/ASME Trans. Mechatronics*, 20(3), 1395–1405.

Appendix A. PROOF OPTIMAL FIT

In this appendix the proof for Theorem 5 is given.

Proof. For the minimum of $\mathcal{J}_e(\theta_j^e)$, it holds that

$$\frac{\partial \mathcal{J}_e(\theta_j^e)}{\partial \theta_j^e} = 0. \quad (\text{A.1})$$

Using this first order necessary condition for optimality, the optimal parameters θ_j^e in terms of cost function (14) are found to be

$$\theta_j^e = (\bar{\psi}_j \bar{\psi}_j^\top)^{-1} \psi_{d,j} \bar{e}_j, \quad (\text{A.2})$$

where $\bar{\psi}_j \bar{\psi}_j^\top$ is the Gramian matrix of the sampled representation of the basis functions $\bar{\psi}_j^\top$. The Gramian is always positive semi-definite, and if the vectors $\psi_k(\bar{\alpha}_j)$, $k = 1, 2, \dots, M$ in $\bar{\psi}_j^\top$ are linearly independent it is positive definite and therefore invertible (Gantmacher (1959), pp.246–248). Then, the unique stationary point that minimizes the convex cost function and gives an optimal fit of the measured error is given by (A.2).

Appendix B. PROOF OPTIMAL UPDATE LAW

In this appendix the proof for Theorem 6 is given.

Proof. For the minimum of $\mathcal{J}(\theta_{j+1}^u)$ it holds that

$$\frac{\partial \mathcal{J}(\theta_{j+1}^u)}{\partial \theta_{j+1}^u} = 0. \quad (\text{B.1})$$

Using this first order necessary condition for optimality, following along the lines of Gunnarsson and Norrlöf (2001), the optimal update of the parameters θ_{j+1}^u in terms of cost function (17) is found to be given by

$$\theta_{j+1}^u = Q_\psi \theta_j^u + L_\psi \theta_j^e, \quad (\text{B.2})$$

with Q_ψ and L_ψ according to (18). Both Q_ψ and L_ψ contain the inverse of the matrix

$$\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^\top(\alpha) d\alpha.$$

This is the Gramian matrix for a set of functions

$$\sqrt{h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)} \psi_k(\alpha), k \in [1, 2, \dots, M],$$

which is positive definite and therefore invertible if and only if these functions are linearly independent (Gantmacher (1959), pp. 246–248). The set of weighted functions is linearly independent if the chosen basis functions ψ_k are linearly independent, all weight functions $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta u}$ are non-negative, and at least one of the weight functions is positive for all $\alpha \in [0, 2\pi]$. Then, the optimal update of the input parameters is given by (18).

Appendix C. PROOF MONOTONIC CONVERGENCE

The definition of monotonic convergence is given in the following lemma.

Lemma 8. A sequence of parameter vectors $\{\theta_j^u\}_{j \in \mathbb{N}}$ is monotonically convergent in the 2-norm towards a fixed parameter vector θ^{u*} if

$$\|\theta^{u*} - \theta_{j+1}^u\|_2 \leq \gamma \|\theta^{u*} - \theta_j^u\|_2 \quad \forall j \in \mathbb{N}, \quad (\text{C.1})$$

with $0 \leq \gamma < 1$. For an update law of the form

$$\theta_{j+1}^u = Q_\psi \theta_j^u + L_\psi \theta_j^e, \quad (\text{C.2})$$

the sequence of parameter vectors is monotonically convergent in the 2-norm if the following condition is met (Bristow et al. (2006)).

$$\bar{\sigma}(Q_\psi - L_\psi h_0) < 1. \quad (\text{C.3})$$

Using Lemma 8, the proof for Theorem 7 is given.

Proof. The update law for the input signal parameters is written as

$$\theta_{j+1}^u = (Q_\psi - L_\psi h_0) \theta_j^u + L_\psi (\theta^{yd} - h_0 \theta^{us} - \theta^d). \quad (\text{C.4})$$

It holds that

$$Q_\psi - L_\psi h_0 = \left(\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha) + W_{\Delta u}(\alpha)) \psi(\alpha) \psi^\top(\alpha) d\alpha \right)^{-1} \int_0^{2\pi} W_{\Delta u}(\alpha) \psi(\alpha) \psi^\top(\alpha) d\alpha, \quad (\text{C.5})$$

and $\bar{\sigma}((A + B)^{-1}B) < 1$ for $A > 0$, $B \geq 0$. Therefore, condition (C.3) is met if the matrix

$$\int_0^{2\pi} (h_0^2 W_e(\alpha) + W_u(\alpha)) \psi(\alpha) \psi^\top(\alpha) d\alpha$$

is positive definite. Since the basis functions ψ_k are linearly independent, this Gramian matrix is positive definite for non-negative weight functions $W_e(\alpha)$, $W_u(\alpha)$ and $W_{\Delta u}(\alpha)$ of which at least $W_e(\alpha)$ or $W_u(\alpha)$ is positive for all $\alpha \in [0, 2\pi]$. Then, the condition $\bar{\sigma}(Q_\psi - L_\psi h_0) < 1$ holds and the sequence of parameter vectors $\{\theta_j^u\}_{j \in \mathbb{N}}$ is monotonically convergent in the 2-norm towards a fixed parameter vector θ^{u*} .

Multi-Layer Spatial Iterative Learning Control for Micro-Additive Manufacturing*

Leontine Aarnoudse * Christopher Pannier **
Zahra Afkhami ** Tom Oomen * Kira Barton **

* Dept. of Mechanical Engineering, Eindhoven University of
Technology, 5600 MB Eindhoven, The Netherlands (e-mail:
l.i.m.aarnoudse@student.tue.nl).

** Dept. of Mechanical Engineering, University of Michigan, Ann
Arbor, MI 48109, USA (e-mail: *pannier@umich.edu*)

Abstract: Spatial iterative learning control (SILC) has been used in the control of additive manufacturing systems that can be described by their spatial dynamics. Since the current framework is limited to single-layer parts, the aim of this paper is to provide an approach to multi-layer SILC using learning in the layer-to-layer dimension. Mathematical formulation of a multi-layer SILC controller is provided, and active feedback control is demonstrated to reduce the error accumulation over the iterations. Simulation results using a model of high-resolution e-jet printing verify performance improvements for the proposed framework.

Keywords: Learning Control, Micro-Additive Manufacturing, Control Applications, Feedback Control, Spatial Domain

1. INTRODUCTION

Micro-additive manufacturing systems based on jet printing are used to fabricate functional electronics, sensors, and applications in the field of biotechnology, for example lab-on-a-chip devices (Onses et al. (2015)). Performance of these devices depends on the accurate construction of the device geometry. Traditional feedback control cannot be used for control of the printed geometry because real-time surface sensing is not feasible at this length scale, however the printed surface can be scanned after curing a layer. Commercial inkjet additive manufacturing systems lack topography feedback and use a subtractive milling process, although new concepts for closed-loop topography control are emerging in research (Sithi-Amorn et al. (2015); Guo et al. (2017)). Because of the lack of real-time feedback, new strategies for purely additive control of the printed part geometry are needed to improve manufacturing yield and precision.

Spatial iterative learning control (SILC) is a model-based feedforward technique for deposited topography control. In SILC, two-dimensional convolution is used to describe the spatial spread of build material using the coupling of each discrete point in the x,y-plane to its neighbors (Hoelzel and Barton (2016)). Models of drop formation (Carter et al. (2014)) and spread (Pannier et al. (2017)) have been developed, but layer-to-layer dynamics are not well understood. Therefore plant models are based on system identification for a limited set of topographies (Wang et al. (2018b)). The SILC framework considers the part geometry to be a heightmap and has been successfully

applied to electrohydrodynamic jet (e-jet) printing for single-layer prints (Wang et al. (2018a)).

Although multi-layer micro-scale additive manufacturing is increasingly important, at present SILC is formulated for single layers only. Single-layer parts are limited in the maximum height that can be achieved, whereas a multi-layer framework allows for more freedom in reference geometry design and hence more part applications. Many micro-additive manufacturing device applications consist of multiple layers, and material deposition precision can be improved by dividing a structure in multiple layers.

The aim of this paper is to extend SILC to a multi-layer framework. This extension is limited to parts consisting of layers with identical heightmaps and materials. Since the multi-layer parts that are considered consist of repeating layers, learning within the part is possible. Small variances in the printed layer are captured through an iteration-and spatially-varying impulse response that describes the system behavior as a function of the previous layer topography. Robustness for iteration-varying system behavior has been examined for broad applications (Altin et al. (2017); Meng and Moore (2017)). These conditions are reviewed for the system considered here. An important consideration in multi-layer SILC is the accumulation of the total topography error from layer-to-layer. A method for reducing this accumulation is through the use of a feedback pass. The multi-layer SILC framework is designed to enable activation of such a mechanism. The contribution of this paper is threefold:

- (1) A framework for multi-layer SILC using learning within the part is presented and the iteration-varying system behavior is addressed.

* This work is supported by the U.S. National Science Foundation under Grant Nos. GRFP-1256260 and CMMI-1434693.

- (2) Feedback is implemented in separate passes, allowing correction of the accumulated error.
- (3) The proposed framework with the feedback implementation is validated in simulations.

This paper is organized as follows. In Section 2 notation is introduced. In Section 3 the design problem is defined. The multi-layer SILC framework, as well as a way to implement feedback passes to correct the accumulated error, is presented in Section 4. In Section 5 the numerical simulations and their results are presented. Conclusions and future directions are presented in Section 6.

2. PRELIMINARIES

We take \mathbb{Z} to represent the set of integers and \mathbb{N} to represent the set of nonnegative integers. For any matrix \mathbf{M} , $\|\mathbf{M}\| = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}$ denotes the Frobenius norm. The following definitions are used.

- A plant is said to be bounded input, bounded output (BIBO) stable if given an input function with finite magnitude $|f(x, y, t)| < \alpha$ for $t > 0$ and all x and y , the output has a finite magnitude $|g(x, y, t)| < \beta$ for $t > 0$ and all x and y , with $\alpha, \beta \in \mathbb{R}^+$.
- If $\lim_{k \rightarrow \infty} \|x_k - v_k\| = 0$ for any norm, x is said to converge to v .
- If $\lim_{k \rightarrow \infty} \sup \|x_k - v_k\| < \infty$ for any norm, x is said to converge to a bounded neighborhood of v
- An ILC update law with updates f_j is said to be monotonically convergent in the l_2 -norm if $\|f_{\infty} - f_{j+1}\| \leq k \|f_{\infty} - f_j\| \quad \forall j \in \mathbb{N}$ for some $k \in [0, 1]$.
- A matrix \mathbf{H} lies in a neighborhood of a nominal matrix $\bar{\mathbf{H}}$ if there exists a finite real constant ρ so that $\|\mathbf{H} - \bar{\mathbf{H}}\| < \rho$.

3. PROBLEM DEFINITION

The aim of this paper is to develop an SILC approach to additively manufacture multi-layer parts consisting of identical layers of a single material. First, the plant model for the micro-additive manufacturing system is presented. Secondly, the suitability of SILC for this type of system is explained.

3.1 Plant model

Consider an additive manufacturing system as described in Hoelzle and Barton (2016). The plant is causal in time and non-causal in space, meaning that an applied input at a spatial coordinate will influence the output at advanced coordinates. The spatiotemporal dynamics of the plant can be simplified to be purely spatial, and the system is bounded input, bounded output (BIBO) stable. The additive manufacturing system is modelled using a heightmap evolution model (Pannier et al. (2019)), resulting in the spatially non-causal convolution equation

$$g_{k+1}(x, y) = g_k(x, y) + \sum_{\substack{m \in \mathbb{Z} \\ n \in \mathbb{Z}}} h_{k+1}^{(m,n)}(x - m, y - n) f_{k+1}(m, n) \quad (1)$$

where input f_k is an array of drop sizes, output g_k is a heightmap and $h_k^{(m,n)}$ is the impulse response at location

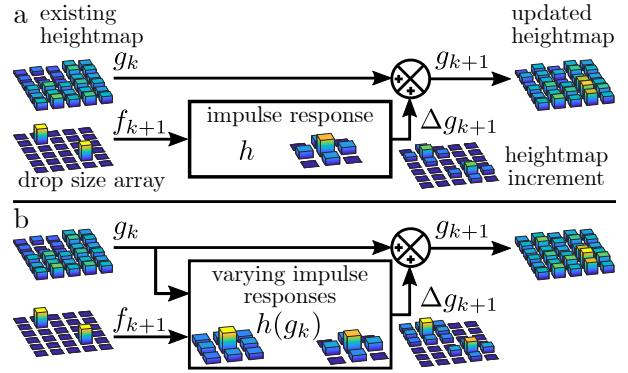


Fig. 1. Heightmap evolution model for a constant impulse response (a) and a topography-dependent impulse response (b).

(m, n) for each layer k . All 2D signals have finite support. The impulse response depends strongly on the surface on which the material is printed. Since layers are printed on top of each other, h_{k+1} is a function of the previous layer topography g_k as shown in Fig. 1. The system can be written in lifted form

$$\mathbf{g}_{k+1} = \mathbf{g}_k + \mathbf{H}(\mathbf{g}_k) \mathbf{f}_{k+1} \quad (2)$$

where \mathbf{f}_k and \mathbf{g}_k are the row-wise vectorizations of f_k and g_k . In the multi-layer SILC approach we will consider a part consisting of multiple layers. The system is described in terms of the height change of these layers $\Delta \mathbf{g}_k$ instead of the total height map \mathbf{g}_k :

$$\Delta \mathbf{g}_{k+1} = \mathbf{g}_{k+1} - \mathbf{g}_k = \mathbf{H}(\mathbf{g}_k) \mathbf{f}_{k+1} \quad (3)$$

3.2 Spatial Iterative Learning Control

Iterative learning control is a control method whereby the feedforward input signal is modified based on preceding experiments that use the same reference, so that the norm of the tracking error is reduced. In traditional ILC, the assumption of constant plant and reference over iterations is crucial.

The main difference between SILC and temporal ILC is that SILC is based on spatial coordinates. This makes it suitable for additive manufacturing systems that use raster trajectories. In these trajectories two points can be close in space, yet distant in time. In SILC, two-dimensional convolution is used to describe the spatial dynamics as the spread of material using the coupling of each coordinate to its neighbors. The spatial distribution of material is a function of the heightmap of the neighboring locations. SILC can be described in a form similar to the lifted ILC framework for temporal systems.

4. MULTI-LAYER SPATIAL ITERATIVE LEARNING CONTROL

For a multi-layer part consisting of repeating layers, SILC learning over these layers is possible. In this vertical learning approach to multi-layer SILC, each layer is an iteration of the SILC system, as shown in Fig. 2. The topography error accumulates over layers, causing iteration-varying system behavior as well as an accumulated deviation from the desired final height h_{des} . This section provides the

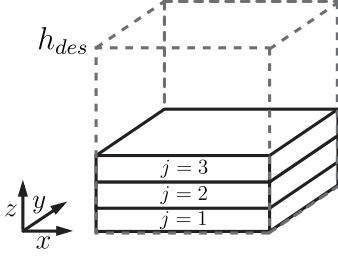


Fig. 2. Multi-layer part consisting of repeating topography across layers.

multi-layer SILC framework for vertical learning, the conditions for stability and convergence and the controller design. To address the accumulation of the topography error, the implementation of feedback printing passes is proposed and the controller choice for this feedback is discussed.

4.1 Framework

The desired topography Δg_d is identical for all n layers. Since the part has a target total topography $h_{des} = n\Delta g_d$, the number of iterations within one part is limited. This is different from conventional ILC, where there is no limit to the number of iterations. To apply vertical SILC, it is assumed that a single material is used for each layer, and that the total thickness of previous layers does not influence the spatial impulse response of the system. Since each layer k is one iteration of the SILC loop, the system dynamics can be rewritten as follows in terms of the iterations j :

$$\Delta g_{j+1} = H(g_j)f_{j+1} \quad (4)$$

The feedforward input signal f_{j+1} is calculated from the filtered input signal f_j and error e_j from the previous iteration. The multi-layer SILC update law (Fig. 3) is given by:

$$f_{j+1} = L_f f_j + L_e e_j = (L_f - L_e H(g_{j-1}))f_j + L_e \Delta g_d \quad (5)$$

With filters L_f and L_e . The error is defined as

$$e_j = \Delta g_d - \Delta g_j \quad (6)$$

where Δg_d is the desired output for a single layer and Δg_j is the actual output for that layer.

In addition to the iteration-varying plant matrix, a nominal system is defined. The nominal system behavior \bar{H} is the plant matrix that results from the desired topography: $\bar{H} = H(g_d)$. For the nominal system, it holds that $H_j = \bar{H} \quad \forall j \in \mathbb{N}$.

Because of the finite target height of the part, the number of iterations within one part is limited. However, the multi-layer SILC can be extended to a multi-layer, multi-part SILC framework. A second part can take advantage of the learning from the previous part, by starting at an iteration $j > 1$ using the last SILC update from the previous part.

4.2 Stability & convergence

The iteration-varying system behavior influences the stability and convergence conditions of the update law (5). Under certain conditions regarding the boundedness of the

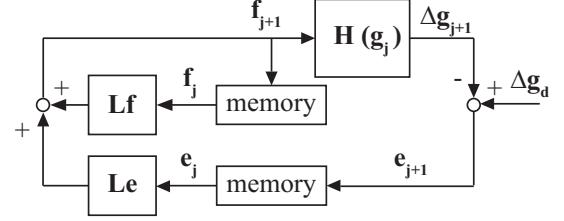


Fig. 3. Block diagram of SILC.

variations, Altin et al. (2017) have shown stability and convergence of the update law.

It is assumed that the plant matrices $H_{j+1} = H(g_j)$ lie in a neighborhood of the nominal plant \bar{H} for all $j \in \mathbb{N}$. The filters L_f and L_e are selected to meet the robustness condition

$$\|L_f - L_e H_j\| \leq \gamma < 1, \quad \forall j \in \mathbb{N}. \quad (7)$$

In the case of nominal system behavior, the robustness condition (7) guarantees monotonic convergence. If these assumptions hold, the SILC update law (5) is asymptotically and BIBO stable. The input and error of the iteration-varying system converge to a bounded neighborhood of the nominal input and error \bar{f} and \bar{e} , and the system output Δg_j is bounded. If H converges to \bar{H} , then f and e converge to \bar{f} and \bar{e} .

In the multi-layer additive manufacturing application considered in this work, layers are printed directly on top of each other. As such, heightmap errors accumulate from one layer to the next, unlike SILC across multiple parts (Hoelzle and Barton (2016)). The framework and stability analysis provided in this work hold for a spatially- and iteration-varying model, with the assumption that the model variations are small so that the system converges to a small neighborhood of the nominal system behavior.

4.3 Norm-optimal SILC

The update law used for the system described in the previous two sections should be robust against plant uncertainties (Altin et al. (2017)). One approach to selecting the filters is norm-optimal SILC. In norm-optimal SILC, the following cost function is minimized:

$$\mathcal{J} = e_{j+1}^T Q e_{j+1} + f_{j+1}^T S f_{j+1} + (f_{j+1} - f_j)^T R (f_{j+1} - f_j) \quad (8)$$

where the weight matrices Q , S and R are user-defined positive definite matrices to obtain performance and robustness objectives such as attenuation of trial-varying disturbances (R). These matrices are commonly chosen to be scaled identity matrices: $Q = qI$, $S = sI$ and $R = rI$. The norm-optimal filters L_f and L_e that minimize this cost function for the nominal plant \bar{H} are given by:

$$L_f = (\bar{H}^T Q \bar{H} + S + R)^{-1} (\bar{H}^T Q \bar{H} + R) \quad (9)$$

$$L_e = (\bar{H}^T Q \bar{H} + S + R)^{-1} \bar{H}^T Q \quad (10)$$

The derivation of these learning filters follows along the lines of the derivation in Gunnarsson and Norrlöf (2001). The final error in the case of nominal system behavior without disturbances is given by:

$$e_\infty = (I - \bar{H}(\bar{H}^T Q \bar{H} + S)^{-1} \bar{H}^T Q) g_d \quad (11)$$

The final error and the convergence behavior depend on the weighting matrices. To obtain the desired SILC

behavior, the weighting matrices for \mathbf{L}_f and \mathbf{L}_e can be tuned.

4.4 Feedback implementation

As discussed, additive manufacturing involves layers printed directly on top of each other, allowing printing errors not compensated by SILC to accumulate from layer-to-layer. Despite the lack of in-layer monitoring, the total layer height can be measured at the end of each printed layer cycle. Once the total topography error reaches a critical amount, a *feedback* iteration can be implemented. To compensate for actuator saturation limits and improve the sensitivity of the feedback correction, the feedback error compensation is combined with the most recent SILC input. Importantly, this combined layer does not contribute to the *learning* process, hence $g_j = g_{k+1}$. Since the feedback pass is implemented outside of the SILC loop, it can be considered a partial system reset. The open-loop nature of the system to which SILC is applied does not change.

The total topography error is defined as

$$\mathbf{e}_{fb,k} = \sum_{n=1}^k \Delta \mathbf{g}_{d,n} - \mathbf{g}_k \quad (12)$$

and is used to determine whether a feedback pass is required. Note that this error is different from the layer-wise SILC error \mathbf{e}_j (6). If the Frobenius norm of the total topography error $\|\mathbf{e}_{fb}\|$ of an ILC layer is larger than the threshold value e_t , a feedback action is introduced between the iterations. This feedback action is given by:

$$\mathbf{u}_{k+1} = \mathbf{f}_{j+1} + \mathbf{C} \mathbf{e}_{fb,k} \quad (13)$$

where \mathbf{f}_{j+1} is the most recent SILC signal as defined in (5) and $\mathbf{C} \mathbf{e}_{fb,k}$ is the feedback signal. The topography update for the feedback pass is given by:

$$\mathbf{g}_j = \mathbf{g}_{k+1} = \mathbf{g}_k + \mathbf{H}(\mathbf{g}_k) \mathbf{u}_{k+1} \quad (14)$$

Note that \mathbf{g}_{k+1} acts as an replacement of \mathbf{g}_k . After this feedback action, the SILC resumes according to (5).

4.5 Controller choice

The goal of the feedback controller \mathbf{C} is to reduce the past error as much as possible with a single input, because each input costs one SILC iteration, and the number of iterations within one part is limited by the total height of the part as is shown in Fig. 2. Since the feedback action can be seen as a system reset that occurs outside of the actual SILC loop, the choice of \mathbf{C} does not influence the convergence or stability properties of the SILC loop.

One option is to use the inverse of the estimated spatial impulse response of the system. However, the spatial impulse response on a certain topography is typically difficult to estimate. A second option is to use a simple proportional controller. These controllers typically require multiple passes to reduce the error to zero, but a single pass with a simple controller is expected to reduce the error enough to get below the threshold value. A suitable proportional gain of the controller can be determined experimentally.

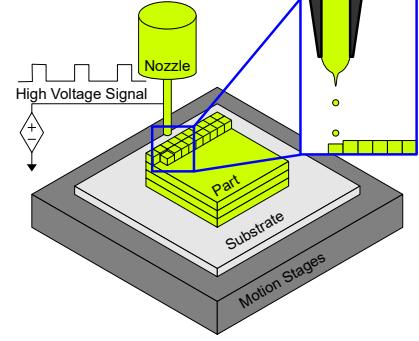


Fig. 4. Schematic representation of e-jet printing.

Table 1. Update law simulation parameters

Norm-optimal weights	$\ \mathbf{L}_f - \mathbf{L}_e \bar{\mathbf{H}}\ $
$q = 1, s = 1, r = 1$	0.54
$q = 1, s = 0.2, r = 1$	0.85
$q = 1, s = 0.2, r = 2$	0.92

5. SIMULATIONS

To determine the feasibility of the proposed multi-layer SILC framework, it is applied to an e-jet printing system in simulation. In this section the simulated e-jet system and the simulation parameters are presented. Then, the results are evaluated based on the traditional ILC criteria of convergence and final topography error. Lastly, alternative performance criteria regarding the total part are presented.

5.1 System setup

An e-jet printer is a microscale additive manufacturing system consisting of a conducting nozzle and substrate, a high-voltage power supply and translational stages as shown in Fig. 4 (Barton et al. (2010)). A pulsed voltage potential between the nozzle and the substrate is used to extract drops from the nozzle. The pulse width of the voltage potential is used to modulate the drop volume (Mishra et al. (2010)).

In the simulation, a layer is printed (4) and measured to determine the error (6) as well as the next iteration input (5). To determine whether a feedback pass is necessary, the norm of the total topography error (12) is compared to the error threshold value.

The linear regression method presented in (Pannier et al. (2019)) is used to generate the topography-dependent nominal model $\bar{\mathbf{H}}$ and simulated physical plant $\mathbf{H}(g_j)$. The spatial impulse response of a droplet deposited on a flat layer of thinly deposited ink is chosen as the nominal behavior on which the learning filters are based. After one part has been printed, the data from the last SILC iteration from the previous part is used to determine the new input. As a result the input converges over multiple parts.

5.2 SILC update law and feedback law design

The reference used for the simulations is a layer with a flat inner part of 256×256 pixels with a height of $0.5 \mu\text{m}$ and a

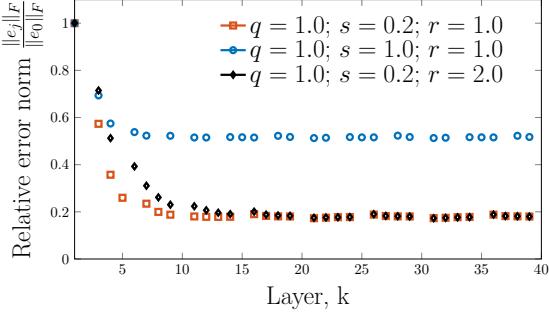


Fig. 5. SILC convergence for a system with feedback passes. The SILC behavior can be influenced by tuning the cost function weighting matrices $\mathbf{Q} = q\mathbf{I}$, $\mathbf{S} = s\mathbf{I}$ and $\mathbf{R} = r\mathbf{I}$.

two-pixel wide border with decreased height as the spread of ink prevents the printing of large height increments. One part consists of ten of these layers, resulting in a total average part height of 4.85 μm . For each simulation four parts are printed, totalling 40 layers.

To simulate the variation of the e-jet system, noise is added to the input signal. The noise is normally distributed with a mean of 0 and a standard deviation of 0.03, approximately 1/8th of the mean value of the nominal input $\bar{\mathbf{f}}$. The reason for choosing normally distributed noise is the assumption that smaller errors in the printing process are more common than larger errors. Since additive manufacturing systems cannot remove material in the previously printed layers, any negative values in the input signal are set to zero.

Different learning filter weights are used to determine their influence on the convergence behavior of the system. All filters are selected to satisfy $\|\mathbf{L}_f - \mathbf{L}_e \bar{\mathbf{H}}\| \leq \gamma < 1$ as shown in Table 1. This guarantees monotonic convergence for the nominal system. However, the values of \mathbf{H}_j are iteration-varying so convergence is not guaranteed. To deal with these variations, learning weights that give robust filters are used.

The controlled system behavior with and without feedback is compared. For the feedback signal, a simple proportional controller with a gain of 2 is used. The threshold value for the Frobenius norm of the error is $e_t = 80$, which is approximately 2/3rd of the initial error. To ensure that each part contains a minimum number of SILC learning passes, a second feedback pass in a row is not permitted. An exception is the the last pass, which is always a feedback pass.

5.3 Evaluation of convergence and final error

All update laws satisfy the monotonic convergence condition for nominal system behavior. Increasing the value of s increases the final error \mathbf{e}_∞ (11), while lowering it increases γ and ultimately leads to an unstable update law due to presence of disturbances. Increasing r reduces the convergence speed, and decreasing it results in instability. The convergence behavior for the different weights in Table 1 is shown in Fig. 5.

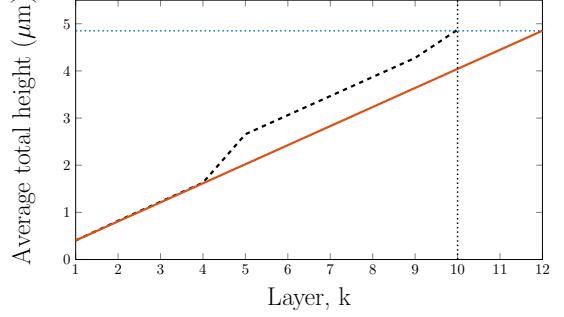


Fig. 6. With feedback (---), the target height h_{des} (...) of 4.85 μm is reached after the target number of ten layers (...). Without feedback (—), twelve layers are needed to reach this height, exceeding the target number of layers ($q = 1$, $s = 0.2$, $r = 1$).

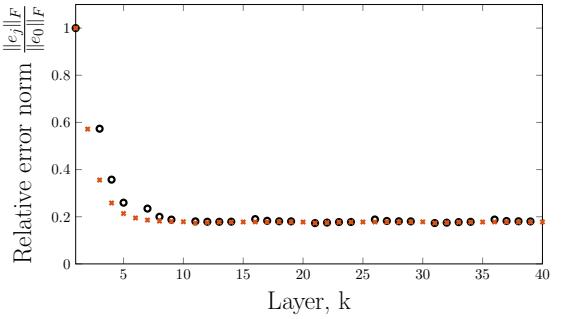


Fig. 7. With feedback (○), more layers are required for the single layer topography error \mathbf{e}_j to converge than without feedback (×) ($q = 1$, $s = 0.2$, $r = 1$).

Two different error metrics are considered. The error for a single layer \mathbf{e}_j is reduced over iterations using SILC and converges to a nonzero final value. As a result, the total topography error \mathbf{e}_{fb} after the target number of ten passes is nonzero, and the final height of the part is lower than the target height. Introducing feedback passes reduces the total topography error, as shown in Fig. 6. Since feedback layers cannot be used for learning, in the approach with feedback the convergence of the single layer error \mathbf{e}_j requires more layers. This is shown in Fig. 7. The converged error value is not influenced by the feedback.

5.4 Alternative performance criteria

The goal of this multi-layer SILC is to print a part, for which the important parameters are the final height of the part and the roughness of the surface. Even if the final error for a layer is nonzero, the final height can still be reached using extra layers as is shown in Fig. 6. The final height of the part can be approximated by the mean value of the final topography. The roughness of the surface is determined by the standard deviation σ of the middle part of the last layer. In Table 2, the values of these criteria, as well as the number of layers required to reach the final height, are given for the different norm-optimal update laws with and without feedback. All values are based on the third printed part because all learning strategies have converged at that point.

Table 2. Update laws compared using AM-specific criteria

Approach	Final height [μm]	σ [μm]	Layers [-]
NO + fb: $q = 1.0, s = 1.0, r = 1.0$	4.85	0.043	10
NO + fb: $q = 1.0, s = 0.2, r = 1.0$	4.87	0.034	10
NO + fb: $q = 1.0, s = 0.2, r = 2.0$	4.87	0.033	10
NO: $q = 1.0, s = 1.0, r = 1.0$	4.85	0.019	20
NO: $q = 1.0, s = 0.2, r = 1.0$	4.85	0.015	12
NO: $q = 1.0, s = 0.2, r = 2.0$	4.85	0.016	12
Target	4.85	0	10

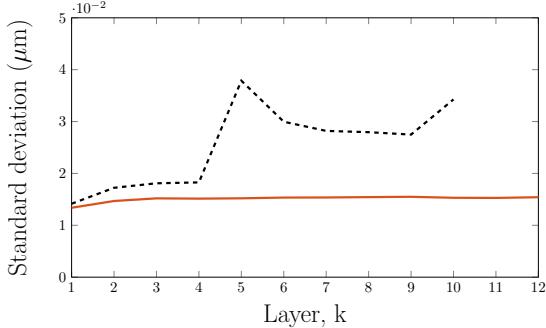


Fig. 8. With feedback (--) the standard deviation of the topography is increased significantly compared to the case without feedback (—) ($q = 1, s = 0.2, r = 1$).

When feedback is used, the target height is reached in the target number of layers. Without feedback, the required number of layers depends on the value of the converged error. The influence of the update law on the final height is small.

The standard deviation over layers within one part for cases with and without feedback is shown in Fig. 8. Without feedback, the standard deviation is asymptotic and reaches its final value after approximately four iterations. Feedback passes at $k = 5$ and $k = 10$ of part three cause an increase in the standard deviation. After the feedback pass, the standard deviation decreases. Since update laws with a higher converged error result in more feedback passes, the standard deviation, and therefore the surface roughness of the final part, is higher for these update laws (Table 2).

The simulations show the feasibility of the vertical multi-layer SILC approaches with and without feedback. Feedback avoids an increase in the number of layers; however, feedback increases the standard deviation of the final layer. The reduction in convergence speed and monotonicity caused by feedback is negligible.

6. CONCLUSIONS

This work shows that for multi-layer parts with repeating layers, it is possible to learn within the part by defining each layer as one SILC iteration. Simulations show that the addition of feedback allows the part to reach the target height in the target number of iterations. However, adding feedback also increases the variation in layer topography.

This approach can be used to manufacture multi-layer parts using microscale additive manufacturing. In future work, we aim to define learning in parts that consist of

both repeating and unique layers, since many optical, acoustic and electronic applications can be built that way. The implementation of LPV models as those in Pannier et al. (2019) in SILC, for example by using iteration-varying learning filters, offers interesting perspectives as well.

REFERENCES

- B. Altin, J. Willems, T. Oomen, and K. Barton. Iterative Learning Control of Iteration-Varying Systems via Robust Update Laws with Experimental Implementation. *Control Engineering Practice*, vol. 62, 36–45, May 2017.
- K. Barton, S. Mishra, K.A. Shorter, A. Alleyne, P. Ferreira, and J. Rogers. A desktop electrohydrodynamic jet printing system. *Mechatronics*, vol. 20, 611–616, 2010.
- W. Carter, G.C. Popell, J. Samuel, and S. Mishra. A Fundamental Study and Modeling of the Micro-Droplet Formation Process in Near-Field Electrohydrodynamic Jet Printing. *J. of Micro- and Nano-Manufacturing*, vol. 2, June 2014.
- S. Gunnarsson and M. Norrlöf. On the Design of ILC Algorithms Using Optimization. *Automatica*, vol. 37, 2011–2016, 2001.
- Y. Guo, J. Peters, T. Oomen, and S. Mishra. Distributed Model Predictive Control for Ink-Jet 3D Printing. *IEEE Int. Conf. on Advanced Intelligent Mechatronics*, July 2017.
- D.J. Hoelzle and K.L. Barton. On Spatial Iterative Learning Control via 2-D Convolution: Stability Analysis and Computational Efficiency. *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, July 2016.
- D. Meng and K.L. Moore. Robust Iterative Learning Control for Nonrepetitive Uncertain Systems. *IEEE Trans. Autom. Control*, vol. 62, no. 2, Feb. 2017.
- S. Mishra, K.L. Barton, A.G. Alleyne, P.M. Ferreira, and J.A. Rogers. High-speed and drop-on-demand printing with a pulsed electrohydrodynamic jet. *J. of Micromechanics and Micro-Eng.*, vol. 20, Aug. 2010.
- M.S. Onses, E. Sutanto, P.M. Ferreira, A.G. Alleyne, and J.A. Rogers. Mechanisms, Capabilities, and Applications of High-Resolution Electrohydrodynamic Jet Printing. *Small*, vol. 1, issue 34, June 2015.
- C. Pannier, M. Wu, D. Hoelzle, and K. Barton. LPV Models for Jet-Printed Heightmap Control. *Amer. Control Conf.* (Accepted), 2019.
- C.P. Pannier, M. Diagne, I.A. Spiegel, D.J. Hoelzle, and K. Barton. A Dynamical Model of Drop Spreading in Electrohydrodynamic Jet Printing. *J. of Manufacturing Science and Engineering*, vol. 139, Nov. 2017.
- P. Sitthi-Amorn, J.E. Ramos, Y. Wang, J. Kwan, J. Lan, W. Wang, and W. Matusik. MultiFab: A Machine Vision Assisted Platform for Multi-Material 3D Printing. *ACM Trans. Graphics*, vol. 34, 2015.
- Z. Wang, C. Pannier, K. Barton, and D.J. Hoelzle. Application of Robust Monotonically Convergent Spatial Iterative Learning Control to Microscale Additive Manufacturing. *Mechatronics*, vol. 56, 2018a.
- Z. Wang, P.M. Sammons, C.P. Pannier, K. Barton, and D.J. Hoelzle. System Identification of a Discrete Repetitive Process Model for Electrohydrodynamic Jet Printing. *Amer. Control Conf.*, 2018b.

Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

I have read the TU/e Code of Scientific Conductⁱ.

I hereby declare that my Master's thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date

..... 11/11/2019

Name

..... L.I.M. aarnoudse

ID-number

..... 0890579

Signature

..... 

Submit the signed declaration to the student administration of your department.

ⁱ See: <http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/>

The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.

More information about scientific integrity is published on the websites of TU/e and VSNU