Spatial Iterative Learning Control: Systems with Input Saturation

Merid Lješnjanin*,†, Ying Tan†, Denny Oetomo† and Christopher T. Freeman‡

Abstract— **This paper proposes a novel Iterative Learning Control (ILC) framework for *spatial* tracking. Spatial tracking means that the *temporal* component is *not* fixed which violates the standing assumption on time intervals in *traditional* ILC. The proposed framework allows for the length of the time interval to *change* with each iteration. To relate the spatial information from the past to the present iteration, the concept of *spatial projection* is proposed. A class of nonlinear uncertain systems with input saturation is chosen for demonstration. An a appropriate ILC control law, exploiting the *spatial projection* idea, is proposed and the corresponding convergence analysis, based on the Composite Energy Function, is carried out. It is shown that spatial tracking is achieved under appropriate assumptions related to spatial projection and provided that the desired trajectory is realizable within the saturation bound. Finally, simulation results illustrate the predicted convergence.**

I. Introduction

Iterative Learning Control (**ILC**) is a control framework which focuses on *repeatable* task(s). A classical example is car assembly which includes several repeatable tasks performed by a robot, such as welding and painting. There are many ways these tasks can be executed. Assuming (or given) a *desired* motion profile, the goal of ILC is then to improve the corresponding *transient* behavior and achieve *perfect* tracking. The ILC framework does this by exploiting the repeatability feature via application of an appropriate *learning* mechanism. *Knowledge* acquired through *learning* enables construction of a control law which effectively and efficiently[1] accomplishes the transient and tracking goals. There is a vast literature associated with ILC, spanning both, the theory and the practice; for instance, cf.: survey papers [1]–[3], books [4]–[6] and references therein.

A specific property in ILC analysis and controller design is that the considered models evolve over a two-dimensional temporal space. In particular, the dynamics of interest evolve over the discrete-time or the continuous-time while its repeatability evolves over the iteration-time[2]. The majority of the literature on ILC analysis and design uses a standing assumption that at each iteration, the dynamics of interest evolves over a *fixed* finite discrete-time or continuous-time interval. Loosely speaking this means that a given task *must* be completed within this *fixed* time interval. To illustrate why this causes a problem for certain applications, imagine drawing a circle of a fixed radius. Assuming *fixed* finite time intervals, this corresponds to drawing the circle within that time interval during each iteration. Indeed, in many applications this assumption is valid; e.g., one can program a robot to perform this task. However, in human motor learning, or in rehabilitation of a stroke patient, this assumption cannot be satisfied; [7] demonstrates this even for healthy subjects. For instance, in stroke rehabilitation, during each session, a patient has to repeat the same task (movement) a sufficient number of times. Now, imagine that the rehabilitation process includes drawing a circle of a fixed radius without robotic assistance. It is unreasonable to expect that the patient will execute this task (movement) in a *fixed* time interval every iteration (trial). Depending on the effects of the injury, including factors such as pain, range of motion, level of concentration and fatigue, and the stage of the rehabilitation, it is more likely to expect that the duration of the time interval will vary with each iteration (trial). It is also natural to expect that over sufficiently many iterations the duration of the time interval will converge to some value. Thus, for applications such as stroke rehabilitation, where the *spatial* component of the corresponding task is *fixed* while its *temporal* component may *vary*, the standing assumption on the time intervals needs to be relaxed. One relaxation, that seems natural, is to assume *bounded* instead of *fixed* time intervals; notice also that the former subsumes the latter. This assumption then enables problem formulation so that the emphasis is on the spatial component in the corresponding ILC analysis and design.

Related industrial research on Spatial ILC (SILC), is largely driven by the particular mechanical applications. For instance, these include switched reluctance motors [8], nonlinear rotary systems [9] and micro-additive manufacturing [10]. Some theoretical results can be found in [11], [12] and [13]. The first reference deals with a problem of directly computing control using given temporal relation between already learned trajectories, while the second considers the problem of minimization of the total tracking error of the repeated tasks with output constraints. The last reference explores the freedom of not specifying temporal information related to the spatial movement and shows some significant practical yields.

This paper aims to address this need by providing a SILC framework for a class of nonlinear uncertain systems with

*The corresponding author.

†University of Melbourne, Australia, x@unimelb.edu.au, x∈ {merid.ljesnjanin, yingt, doetomo}.

‡University of Southampton, UK, cf@ecs.soton.ac.uk.

[1] Note that other control methodologies can be used to solve the problems ILC addresses. However, they usually do not exploit the specific properties associated with the corresponding problems, such as repeatability, and thus result in a less efficient and effective control laws. In addition, ILC can enforce perfect tracking in the presence of significant modeling uncertainties.

[2] Indeed, if a discrete-time corresponds to the set of non-negative integer numbers, then this discrete-time and the iteration-time are the same. Otherwise, the equidistance between points belonging to a discrete-time might not equal to one.

input saturation. Notice that input saturation is ubiquitous in real world applications, while theoretically, the corresponding control problem is challenging and of broad interest; cf. [14] and references therein. Thus, the corresponding SILC results have potential for high impact, both, theoretically and practically. The underpinning component of the proposed framework is an idea of *spatial projection* which is tightly related to *appropriate* temporal rescaling. To the best of authors' knowledge the only reference that utilizes a similar idea in the ILC setting is reported in [15]. Namely, in [15], ILC and time-scale transformation are used to identify added mass, drag and buoyancy in the dynamics of the underwater robots. The considered application is rather specific, and moreover no general analysis and design framework is provided. After introduction of the *spatial projection*, the present paper proposes an appropriate SILC controller and carries out the corresponding convergence analysis using appropriate Composite Energy Function (CEF) method. It is shown that under the appropriate assumptions related to the *spatial projection* and the assumptions akin to those utilized in [14], spatial tracking is achieved. Finally, simulation results demonstrate the convergence, even under disturbances introduced by computer implementation.

The paper is organized as follows. In the sequel, the mathematical preliminaries and notational conventions are provided which is followed with the formulation of the problem in Section II. Then, in Section III, the corresponding assumptions are provided and main results are stated. This is followed with Section IV, in which the simulations results demonstrate the claims of the main results. Finally, Section V hosts the concluding remarks.

Preliminaries: Symbols $\mathbb{Z}$ and $\mathbb{R}$, respectively, denote the set of integer and real numbers. A set $\mathbb{D} \in \{\mathbb{Z}, \mathbb{R}\}$ which elements are $\diamond$-bounded, $\diamond \in \{\leq, <, >, \geq\}$, by an element $a \in \mathbb{D}$, is defined as $\mathbb{D}_{\diamond a} := \{x \in \mathbb{D} : x \diamond a\}$. The set of natural numbers is then $\mathbb{N} := \mathbb{Z}_{>0}$ while $\mathbb{N}_0 := \mathbb{Z}_{\geq 0}$. The Cartesian product between sets $\mathbb{D}_j,\ j \in \{1, \ldots, d\},\ d \in \mathbb{N}$, is denoted as $\mathbb{D}_1 \times \cdots \times \mathbb{D}_d$. However, when $\mathbb{D}_j = \mathbb{D},\ \forall j \in \{1, \ldots, d\},\ d \in \mathbb{N}$, the shorthand notation $\mathbb{D}^d$ is used. Its elements are denoted as ordered d-tuples, i.e., $(x_1, \ldots, x_d)$ where $x_j \in \mathbb{D},\ \forall j \in \{1, \ldots, d\},\ d \in \mathbb{N}$. Moreover, throughout the document, this notation is used to denote column vectors. Standard notation for p norm is used throughout the document. In particular, $\|x\|_p^p := \sum_{j=1}^{n} |x_j|^p,\ n \in \mathbb{N},\ p \in [1, \infty)$, where $|\cdot|$ denotes the standard Euclidean norm. In addition, for a given $[0, \mathsf{T}] \mapsto x(t) \in \mathbb{R}^n,\ \mathsf{T} > 0$, its the supremum norm is defined as $\|x\|_{\mathsf{s}} := \max_{t \in [0,\mathsf{T}]} \|x(t)\|_1$, while its time-weighted norm is defined as $\|x\|_\lambda := \max_{t \in [0,\mathsf{T}]} e^{-\lambda \cdot t} \cdot \|x(t)\|_1$, where $\lambda > 0$.

II. Problem formulation

For clarity reasons, first an ILC model for a class of nonlinear uncertain systems with input saturation is presented[3] . This is followed with a discussion on why such a model is not applicable for the applications with *non-fixed* time intervals. The discussion sets the stage for the introduction of an appropriate *spatial projection* mapping. This mapping then assists in the reasoning behind the proposed SILC model. Finally, this spatial projection mapping is later used in the construction of an appropriate SILC control law.

So, consider a dynamical model evolving over a two dimensional temporal space[4],

$$\dot{x}(i,t) = f\big(t, x(i,t)\big) + B\big(t, x(i,t)\big) \operatorname{sat}\big(u(i,t), \bar{u}\big), \quad (1)$$

where[5],

$$\big(i,t\big) \in \mathbb{N}_0 \times [0, \mathsf{T}],\ \mathsf{T} \in \mathbb{R}_{>0}, \quad (2)$$

is an element of a two dimensional temporal space, while,

$$\begin{aligned} &\operatorname{sat}\big(u(i,t), \bar{u}\big) \\ &\qquad := \big(\operatorname{sat}\big(u_1(i,t), \bar{u}_1\big), \ldots, \operatorname{sat}\big(u_m(i,t), \bar{u}_m\big)\big), \end{aligned} \quad (3)$$

with,

$$\begin{aligned} &\operatorname{sat}\big(u_j(i,t), \bar{u}_j\big) \\ &\qquad := \operatorname{sgn}\big(u_j(i,t)\big) \cdot \min\{\bar{u}_j, \|u_j(i,t)\|_1\},\ \bar{u}_j > 0, \end{aligned} \quad (4)$$

for each $j \in \{1, \ldots, m\}$ and each $\big(i,t\big) \in \mathbb{N}_0 \times [0, \mathsf{T}]$. Variables $x \in \mathbb{B}_x \subseteq \mathbb{R}^n$ and $u \in \mathbb{B}_u \subseteq \mathbb{R}^m$, respectively, are the model state and input, while $(n, m) \in \mathbb{N}^2$. Further, $f : \mathbb{R}_{\geq 0} \times \mathbb{B}_x \to \mathbb{B}_x$ is an *unknown* model state mapping while $B : \mathbb{R}_{\geq 0} \times \mathbb{B}_x \to \mathbb{B}_x \times \mathbb{B}_u$, is a *known* input matrix. Correspondingly, consider a model of a desired behavior,

$$\dot{x}_{\mathsf{d}}(t) = f\left(t, x_{\mathsf{d}}(t)\right) + B\left(t, x_{\mathsf{d}}(t)\right) u_{\mathsf{d}}(t), \quad (5)$$

where,

$$\|u_{\mathsf{d}}(t)\|_1 \leq \bar{u},\ \forall t \in [0, \mathsf{T}], \quad (6)$$

while $x_{\mathsf{d}} \in \mathbb{B}_{x_{\mathsf{d}}} \subseteq \mathbb{R}^n$ and $u_{\mathsf{d}} \in \mathbb{B}_{u_{\mathsf{d}}} \subseteq \mathbb{R}^m$, respectively, are the desired behavior state and input.

Remark 1 (Input bound): As reported in [14], the whole premise behind the successful application of an ILC paradigm, for uncertain nonlinear systems with input saturation, is that inequality (6) is satisfied. Namely, if satisfied, it is possible to construct an appropriate ILC control law (cf. (7)) so that then after sufficiently many iterations, an input that approximates u_{d} well enough, is generated (cf. [14, Theorem 2]). □

A standard ILC objective is to construct an ILC control law so that desired behavior is *learned* with respect to a corresponding performance index after sufficiently many iterations. More precisely, the goal is to achieve $\lim_{i\to\infty} \Delta x(i,t) = (0, \ldots, 0)$, where $\Delta x(i,t) = x_{\mathsf{d}}(t) - x(i,t)$. Depending on the properties of the corresponding system different ILC control laws can be used, e.g., cf. [4]–[6]. For instance, in [14], the following ILC control law is

[3] Note that considered system is borrowed from [14], however, the present paper uses notation which emphasizes that the system evolves over a two dimensional temporal space.

[4] Note that $\dot{x}(i,t) := dx(i,t)/dt$.

[5] In the sequel, this temporal notation, akin to the one encountered in hybrid systems, is changed to a notation accustomed to the ILC literature. At this stage, the present notational choice is chosen for clarity.

used,

$$\begin{aligned} u(i+1,t) &:= \mathrm{sat}\left(u(i,t),\bar{u}\right) \\ &\quad + \kappa \cdot B\left(t,x(i+1,t)\right)^{\top} \Delta x(i+1,t), \end{aligned} \tag{7}$$

where $\kappa \in \mathbb{R}$ is the *learning control gain*.

Remark 2: Note that in (7), the ILC control law uses the state error at the current iteration $i+1$ (i.e., time domain feedback) instead of the state error from the previous iteration i (i.e., time domain feedforward). Also, since the system (1) has the relative degree one, contraction mapping method would require the derivative information of the state error. This would then require for (7) to incorporate that quantity. However, with appropriate assumptions and CEF based approach, as documented in [14], it can be shown that the proposed ILC algorithm (7) achieves perfect tracking. □

Now, consider a scenario where the duration of *each* trial is not *fixed* (in this case (2) is no longer satisfied). This, for instance, can relate to the *rate* at which the corresponding dynamics evolves not being *fixed*. For instance, consider a scenario similar to the rehabilitation process, mentioned in Section I. Recall that in the rehabilitation process, during each treatment (session), a patient repeats (sufficiently many times) a given movement. Recall also that due to many factors, there is no guarantee that each repetition of a given movement is performed at the same rate and thus within the same time interval. Now, since in this case the task is not repeatable over a *fixed* time interval (i.e., (2) is not satisfied), the traditional ILC cannot be applied directly. Thus, one needs a more general framework. In particular, one needs a framework which focuses on the *spatial* information. There are different ways to handle spatial information. This paper focuses on the *spatial projection* mapping defined below.

Definition 1 (Spatial Projection): Consider $s(t) \in \mathbb{R}^l,\ l \in \mathbb{N}$, where $t \in [0,T],\ T > 0$. Given $\epsilon > 0$, a spatial projection mapping is defined as,

$$\sigma_\epsilon\left(s(t)\right) := s(\epsilon \cdot \tau), \tag{8}$$

where $\tau \in \left[0, \frac{T}{\epsilon}\right]$. □

Remark 3 (Spatial Projection): Indeed, the spatial projection is achieved via *temporal rescaling* (*rate* ϵ) and the corresponding manipulation of the original time domain. However, the following example provides the reasoning behind the idea of *spatial projection* and it also serves as a prelude to the introduction of the model used for SILC. □

Example 1 (Spatial Projection): To illustrate the application of the *spatial projection* mapping from Definition 1 for the characterization of spatially equivalent trajectories, consider the task of drawing a unit circle starting from the point A in the clockwise direction; see Fig. 1. Instead of using a dynamical model which captures the task of drawing a circle, to simplify the explanation consider just a static mapping which captures a unit circle. In particular, each point on a unit circle can be defined as $p(t) := (\sin(t), \cos(t))$ with the property $\sin(t)^2 + \cos(t)^2 = 1$. To simplify the illustration even further, consider drawing a unit circle only *once*, in which case restricting t to $[0,T],\ T > 0$, enables the static mapping to capture all the necessary features. In Fig. 1 this corresponds to the green color. Let the corresponding unit circle be a *nominal* unit circle. The nominal unit circle is drawn with a corresponding *rate* and it takes T amount of time to finish drawing it. Now, let $(\epsilon', \epsilon'') \in \mathbb{R}^2_{>0}$ be given and let $\epsilon' > 1 > \epsilon''$. It follows[6],

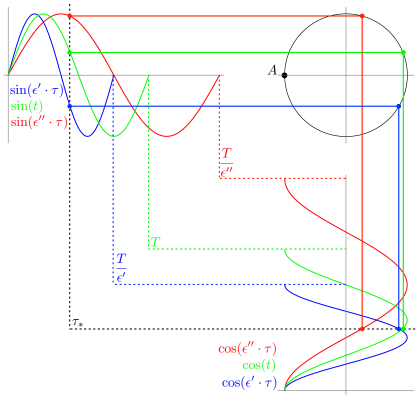
$\sin(\epsilon' \cdot \tau)$
$\sin(t)$
$\sin(\epsilon'' \cdot \tau)$
A
$\frac{T}{\epsilon''}$
T
$\frac{T}{\epsilon'}$
τ_*
$\cos(\epsilon'' \cdot \tau)$
$\cos(t)$
$\cos(\epsilon' \cdot \tau)$

Fig. 1. Preserving spatial information.

$$\sigma_{\epsilon'}(p(t)) = p(\epsilon' \cdot \tau),\ \tau \in \left[0, \frac{T}{\epsilon'}\right], \tag{9a}$$

$$\sigma_{\epsilon''}(p(t)) = p(\epsilon'' \cdot \tau),\ \tau \in \left[0, \frac{T}{\epsilon''}\right]. \tag{9b}$$

The unit circle that corresponds to (9a) is associated with blue color while the unit circle that corresponds to (9b) is associated with red color. As illustrated, in all three cases the *spatial* information is *preserved*; e.g., the unit circle which is in the black color. However, *temporally*, they to differ. In particular, the difference is the rate at which each circle is drawn and time it took to complete each circle; e.g., it is possible to *temporally* characterize them. In particular, because $\epsilon'' < 1 < \epsilon'$, the circle associated with the green color is drawn at a slower rate than the circle associated with the blue color and at the faster rate than the circle associated with the red color. To illustrate this even further, at the ("universal") time instant τ_*, the corresponding point on the unit circle associated with the blue color is ahead of the the corresponding point on the unit circle associated with the green color which is ahead of the corresponding point on the unit circle associated with the red color. Correspondingly circle completion times associated with the blue, green and red color, respectively, satisfy $T/\epsilon' < T < T/\epsilon''$. △

Now, consider,

$$\begin{aligned} &\dot{x}\big(i, \epsilon(i)\cdot t\big) = \epsilon(i) \cdot f\big(\epsilon(i)\cdot t, x(i,\epsilon(i)\cdot t)\big) \\ &+ \epsilon(i) \cdot B\big(\epsilon(i)\cdot t, x(i,\epsilon(i)\cdot t)\big)\, \mathrm{sat}\left(u(i,\epsilon(i)\cdot t),\bar{u}\right), \end{aligned} \tag{10}$$

where,

$$\big(i,t\big) \in \mathbb{N}_0 \times [0, \mathsf{T}_i], \tag{11}$$

[6] Note that in both cases it holds that $\sin(\cdot)^2 + \cos(\cdot)^2 = 1$.

is an element of a two dimensional temporal space with,

$$\mathsf{T}_i := \frac{\mathsf{T}}{\epsilon(i)}, \ \mathsf{T} \in \mathbb{R}_{>0}, \tag{12}$$

and,

$$\epsilon : \mathbb{N}_0 \to \mathbb{R}_{>0}, \tag{13}$$

is a mapping that captures at what rate the dynamics of (10) evolves at each iteration i. More details about this mapping is provided in the sequel but for now note that at each iteration i, $\epsilon(i)$ affects the duration of the time interval $[0, \mathsf{T}_i]$, see (12). With the concept of *spatial projection* (see Definition 1 and Example 1), the system (10) can be interpreted as the spatial projection of the system (1) with respect to the mapping ϵ (see (13)). To make the notation less verbose, the system (10) is succinctly represented as,

$$\begin{aligned}\dot{x}_{i|\epsilon_i} &= \epsilon_i \cdot f\left(\epsilon_i \cdot t, x_{i|\epsilon_i}\right) \\ &\quad + \epsilon_i \cdot B\left(\epsilon_i \cdot t, x_{i|\epsilon_i}\right) \operatorname{sat}\left(u_{i|\epsilon_i}, \bar{u}\right).\end{aligned} \tag{14}$$

Correspondingly, consider a model of a desired behavior,

$$\begin{aligned}\dot{x}_{\mathsf{d}}\left(\epsilon(i) \cdot t\right) &= \epsilon(i) \cdot f\left(\epsilon(i) \cdot t, x_{\mathsf{d}}(\epsilon(i) \cdot t)\right) \\ &\quad + \epsilon(i) \cdot B\left(\epsilon(i) \cdot t, x_{\mathsf{d}}(\epsilon(i) \cdot t)\right) u_{\mathsf{d}}(\epsilon(i) \cdot t),\end{aligned} \tag{15}$$

where $x_{\mathsf{d}} \in \mathbb{B}_{x_{\mathsf{d}}} \subseteq \mathbb{R}^n$ and $u_{\mathsf{d}} \in \mathbb{B}_{u_{\mathsf{d}}} \subseteq \mathbb{R}^m$, respectively, are the desired behavior state and input. Again, f and B in (15), are the same as in (10). Note that the desired behavior model (15) is also accordingly spatially projected. Now, similarly as for (10), the desired behavior model (15) is succinctly represented as

$$\dot{x}_{\mathsf{d}|\epsilon_i} = \epsilon_i \cdot \left(f(\epsilon_i \cdot t, x_{\mathsf{d}|\epsilon_i}) + B(\epsilon_i \cdot t, x_{\mathsf{d}|\epsilon_i}) u_{\mathsf{d}|\epsilon_i}\right), \tag{16}$$

The ILC objective remains the same. Namely, construct an ILC control law so that desired behavior is *learned* after sufficiently many iterations. However notice that the desired behavior, though spatially invariant, is affected by (13). Moreover, the rate (13) also directly affects (11), (14) and (16). Thus, before stating precisely the ILC objective it is necessary to elaborate more on it.

Variable $\epsilon(i) > 0$, captures at what rate the dynamics of (14) evolves at the iteration i. Correspondingly, it also directly affects the length of the corresponding time interval, see (12) and (11). Focusing on the applications such as rehabilitation process, mentioned above, it seems natural to assume that (13) converges to some value, as stated in the following assumptions.

Assumption 1 (Rate Convergence): Consider equation (13). There exists a tuple $(\epsilon_*, \eta) \in \mathbb{R}_{>0} \times (0, 1)$, such that,

$$\|\epsilon_{i+1} - \epsilon_*\|_1 \leq \eta \cdot \|\epsilon_i - \epsilon_*\|, \tag{17}$$

holds for each $i \in \mathbb{N}_0$. □

Remark 4 (Rate Convergence): For instance, the element ϵ_* can represent a *unique* rate of the execution of a given task for a (healthy) patient. Section IV explores different rate profiles which satisfy Assumption 1. □

Remark 5 (Rate Boundedness): Notice that because of Assumption 1 there exists $\bar{\epsilon} \in \mathbb{R}_{>0}$, such that $\epsilon_i \leq \bar{\epsilon}$ holds for each $i \in \mathbb{N}_0$. This means that the length of the corresponding time interval, see (12), is upper bounded as well. Also, note that the upper bounded on rate ϵ means that considered dynamics cannot evolve infinitely fast, which definitely applies to humans. □

Remark 6 (Rate Dynamics): Notice that the rate ϵ, as defined in (13), is a static mapping which is assumed to be known for each iteration $i \in \mathbb{N}_0$. Assumptions 1 seem reasonable, even without the full knowledge of (13). However, the conducted analysis does depend on it. Thus, the future work will address the design and analysis where (13) is not fully available but estimates are. In addition to that, a more complex description of the rate ϵ will be considered, e.g., it will be modeled as an output of a dynamical system which might be affected with data from the model (14) and/or (16). □

Now, the goal is defined as a construction of an ILC control law such that,

$$\lim_{i \to \infty} \Delta x_{i|\epsilon_i} = (0, \ldots, 0) \tag{18}$$

where

$$\Delta x_{i|\epsilon_i} := x_{\mathsf{d}|\epsilon_i} - x_{i|\epsilon_i}. \tag{19}$$

Remark 7 (State Error): Notice that because of Assumption 1, it follows that the state error limit (18), is equivalent to $\lim_{i\to\infty} \Delta x_{i|\epsilon_i} = \lim_{i\to\infty} \Delta x_{i|\epsilon_*} = 0$. □

One attempt in constructing such a law is to directly and correspondingly apply the reasoning behind (7). In particular, one might be tempted to apply the following ILC control law,

$$\begin{aligned}u_{i+1|\epsilon_{i+1}} &:= \operatorname{sat}\left(u_{i|\epsilon_i}, \bar{u}\right) \\ &\quad + \kappa \cdot B\left(\epsilon_{i+1} \cdot t, x_{i+1|\epsilon_{i+1}}\right)^\top \Delta x_{i+1|\epsilon_{i+1}}.\end{aligned} \tag{20}$$

However, unfortunately, when $\epsilon_i \neq c, \ c \in \mathbb{R}_{>0}, \ \forall i \in \mathbb{N}_0$, the length of $[0, \mathsf{T}_i]$ is not *fixed* (see (12)). Therefore, two consecutive time intervals are not necessarily the same length and using (20) to compute control values for the time interval $[0, \mathsf{T}_{i+1}]$ becomes problematic. Firstly, if $\mathsf{T}_{i+1} > \mathsf{T}_i$, then for the time interval $[\mathsf{T}_{i+1} - \mathsf{T}_i, \mathsf{T}_{i+1}]$ there are no control values from the time interval $[0, \mathsf{T}_i]$ that can be used; i.e., they are only defined for the time interval $[0, \mathsf{T}_i]$. On the other hand, if $\mathsf{T}_{i+1} < \mathsf{T}_i$, then the control values from the time interval $[\mathsf{T}_{i+1} - \mathsf{T}_i, \mathsf{T}_{i+1}]$ are not even used. To explain this more precisely, consider Example 1. For simplicity reasons, assume that the data associated to the blue color corresponds to the data from the time interval $[0, \mathsf{T}_i]$. Let $\mathsf{T}_{i+1} > \mathsf{T}_i$ and let the corresponding time interval $[0, \mathsf{T}_{i+1}]$ be associated to the green color; e.g., $\mathsf{T}_{i+1} = T$ while $\mathsf{T}_i = T/\epsilon'$. Now, using the "blue data" to compute the corresponding data for the time interval $[0, \mathsf{T}_{i+1}]$ becomes a problem because for the time interval $[\mathsf{T}_{i+1} - \mathsf{T}_i, \mathsf{T}_{i+1}]$ (in Example 1 this corresponds to $[T - T/\epsilon', T]$) the "blue data" is not defined. Conversely, now assume that the data associated to the green color corresponds to the data from the time interval $[0, \mathsf{T}_i]$. Let $\mathsf{T}_{i+1} < \mathsf{T}_i$ and let the corresponding time interval $[0, \mathsf{T}_{i+1}]$ be associated to the blue color; e.g., $\mathsf{T}_{i+1} = T/\epsilon'$ while $\mathsf{T}_i = T$. Now, using the "green data" to compute the corresponding data for the time interval $[0, \mathsf{T}_{i+1}]$ becomes also a problem. Namely, the "green data" that corresponds to the time interval $[\mathsf{T}_i - \mathsf{T}_{i+1}, \mathsf{T}_i]$ is not used. To address both issues the following ILC control law is proposed,

$$\begin{aligned} u_{i+1|\epsilon_{i+1}} := \sigma_{\epsilon_{i+1}}\Big(&\mathrm{sat}\left(u_{i|\epsilon_i}, \bar{u}\right) \\ &+ \kappa \cdot B\left(\epsilon_{i+1} \cdot t, x_{i+1|\epsilon_{i+1}}\right)^\top \Delta x_{i+1|\epsilon_{i+1}}\Big) \\ = \;&\mathrm{sat}\left(u_{i|\epsilon_{i+1}}, \bar{u}\right) \\ &+ \kappa \cdot B\left(\epsilon_{i+1} \cdot t, x_{i+1|\epsilon_{i+1}}\right)^\top \Delta x_{i+1|\epsilon_{i+1}}. \end{aligned} \tag{21}$$

The idea behind the ILC control law (21) is to preserve spatial reference. This appears to resemble human approach towards spatial tracking, i.e., when spatial reference is *fixed* and there are no time constraints. In the following section the proposed control law is analyzed in a more detail.

Remark 8 (Robotic Assistance): The proposed framework focuses on human learning without robotic assistance, e.g., cf. [16]. Thus, the control refers to human *voluntary* movement. In the future work the corresponding extensions will be addressed. □

III. Analysis

This section records the convergence analysis of the proposed SILC framework. Recall that the framework consists of models (14) and (16) and the control law (21). The analysis leverages existing results in the ILC literature which use an appropriate CEF function to show convergence, e.g., cf. [14]. In fact, the analysis documented in this section follows closely the corresponding analysis in [14]. Thus, full proofs are omitted and only some comments on key differences are provided.

Now, to claim (18) under the proposed control law (21), some additional sufficient assumptions are stated below. The first one is related to the initial states of (14) and (16). This assumption is often used in the ILC literature and it even has a reserved name, Identical Initialization Condition (IIC).

Assumption 2 (IIC): Consider (14) and (16). The following holds,

$$x(0, i) = x^\circ = x_{\mathsf{d}}(0), \tag{22}$$

for each $i \in \mathbb{N}_0$. □
Further, the mappings f and B from (14) and (16) are assumed to be Locally Lipschitz Continuous (LLC). In particular, the following is assumed.

Assumption 3 (LLC): Consider models (14) and (16). There exists $(\mathsf{L}, \mathsf{G}) \in \mathbb{R}^2_{>0}$ such that,

$$\|f(\epsilon \cdot t, \tilde{x}) - f(\epsilon \cdot t, \bar{x})\|_1 \leq \mathsf{L} \cdot \|\tilde{x} - \bar{x}\|_1, \tag{23a}$$

$$\|B(\epsilon \cdot t, \tilde{x}) - B(\epsilon \cdot t, \bar{x})\|_M \leq \mathsf{G} \cdot \|\tilde{x} - \bar{x}\|_1, \tag{23b}$$

holds $\forall\big((\epsilon \cdot t, \tilde{x}), (\epsilon \cdot t, \bar{x})\big) \in (\mathbb{R}_{\geq 0} \times \mathbb{B}_x)^2$ and $\forall \epsilon \in \mathbb{R}_{>0}$; $\|\cdot\|_M$ is induced matrix norm[7]. □

The last assumptions essentially excludes the *finite escape time* possibility by restricting attention to Bounded-Input-Bounded-Output (BIBO) stable systems.

Assumption 4 (BIBO Stability): Consider (14) and (16). The corresponding dynamics is BIBO stable on time interval $[0, \mathsf{T}/\epsilon_i]$, $\mathsf{T} > 0$ for all $\epsilon_i > 0$. □
Finally, the main theoretical result is stated next.

Theorem 1: Consider (14) and (16). Let the Assumptions 1 – 4 be satisfied. Then, the control law (21) achieves the limit (18). □

Proof: The proof follows similar lines as the proof of Theorem 2 in [14]. Thus, for space reasons it is omitted. However, the key differences is explained below. The used CEF is defined as, $E_{i|\epsilon_i} := \frac{1}{2} \cdot e^{-\lambda \cdot t} \cdot (\Delta x_{i|\epsilon_i})^2 + \frac{1}{2 \cdot \kappa} \cdot \int_0^{\mathsf{T}_i} e^{-\lambda \cdot \tau} \cdot (u_{\mathsf{d}|\epsilon_i} - u_{i|\epsilon_i})^2 d\tau$. One of the steps in demonstrating the convergence is to establish the corresponding dissipation inequality, e.g., step (2) in [14]. However, in the present case $[0, \mathsf{T}_i]$ is not *fixed*. Thus, directly emulating step (2) would result in trying to upper bound $\Delta E_{i|\epsilon_i} := E_{i+1|\epsilon_{i+1}} - E_{i|\epsilon_i}$ with a negative number. However, the same problems, as captured in the paragraph after (20), are encountered. To address this, similarly as for the control law (21), one considers $\Delta E_{i|\epsilon_i} := E_{i+1|\epsilon_{i+1}} - \sigma_{\epsilon_{i+1}}(E_{i|\epsilon_i}) = E_{i+1|\epsilon_{i+1}} - E_{i|\epsilon_{i+1}}$. The rest of the proof then follows similar lines as the proof of Theorem 2 in [14]. ■

IV. Simulations

A simple scalar example is used to illustrate that the limit (18) is achieved under the proposed ILC control law (21). The used example is borrowed from [14] and correspondingly modified. Indeed, the considered dynamics is simple, but still sufficient to demonstrate the convergence. First, the model and desired behavior dynamics are presented. Then, four different rate ϵ profiles, that satisfy Assumption 1, are depicted[8] and the corresponding results are presented.

So, consider the following dynamical system, $\dot{x}_{i|\epsilon_i} = \epsilon_i \cdot 3 \cdot x_{i|\epsilon_i} \cdot \sin(\epsilon_i \cdot t) + \epsilon_i \cdot \mathrm{sat}\left(u_{i|\epsilon_i}, \bar{u}\right)$, where $x^\circ = 0$ while $t \in [0, (2 \cdot \pi)/\epsilon_i]$; in Fig. 2, four different rate profiles are provided. Following the narrative from [14], let the desired behavior be captured with, $x_{\mathsf{d}|\epsilon_i} = 1.5 \cdot \sin(\epsilon_i \cdot t)^3$. Using the latter two equations, it follows that $u_{\mathsf{d}|\epsilon_i} = \epsilon_i \cdot 4.5 \cdot (\sin(\epsilon_i \cdot t)^2 \cdot \cos(\epsilon_i \cdot t) - \sin(\epsilon_i \cdot t)^4)$. From Fig. 2 it is easy to determine lower and upper bound for each individual rate profile and use it to determine the lengths of the corresponding time intervals. However, for simplicity and space reasons this is omitted. It can be shown that $\|u_{\mathsf{d}|\epsilon_i}\|_1 \leq 8.8$ regardless of the used rate profile. Thus let $\bar{u} = 8.8$. The convergence results for each rate profile, obtained by using the proposed control law (21), are captured in Fig. 3. Before concluding this section, few remarks are provided regarding the implementation of the control law (21) in a computer environment. To perform simulations, MATLAB and Simulink are used. Implementation of the *spatial projection* of the control values

[7] For definitions and corresponding theory, consult [17, Chapter 9].

[8] Recall that in this manuscript, the full knowledge of (13) is assumed available. Future work will address cases where this is not the case.

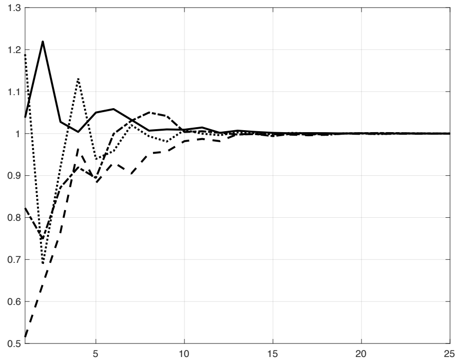

Fig. 2. Different line shapes correspond to different rate ϵ profiles. All four profiles converge to $\epsilon_* = 1$. Iteration time i is located on the horizontal axis while the vertical axis stores values.

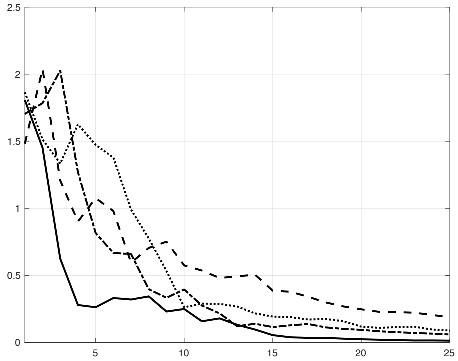

Fig. 3. Convergence of $\Delta x_{i|\epsilon_i}$ under (21) for rate profiles depicted in Fig. 2. The learning control gain in (21), $\kappa = 0.8$, while $u_{0|\epsilon_1} = 0$, for simplicity. Iteration time i is located on the horizontal axis while the vertical axis stores the values of $\|\Delta x_{i|\epsilon_i}\|_{\mathsf{s}}$.

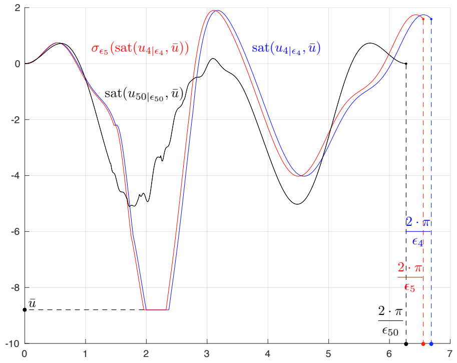

Fig. 4. Control signal $u_{i|\epsilon_i}$ for a dotted rate profile. The vertical axis stores control values while the horizontal axis stores the continuous time. The control signal for $i = 4$ is in blue color while its spatial projection to $i = 5$ is in red color. The control signal in black color corresponds to $i = 50$. At this stage, as shown in Fig. 2, the dotted rate ϵ has converged to $\epsilon_* = 1$. Thus, control signal and its *spatial projection* are the same.

form the previous iteration (see first part of the right hand side of (21)) involves sampling the corresponding control signal; 1500 samples are taken to be exact. Then, these values are equally spread along the time interval that corresponds to the iteration $i + 1$. Depending on the solver used in the Simulink, different interpolation methods are applied; in the case of the presented results, solver "ode23t" is used.

For a dotted rate profile, Fig. 4 shows the corresponding control signal for $i = 4$, its projection to $i = 5$ and the control signal for $i = 50$. Finally, as shown in Fig. 4 even with saturation, the proposed control eventually *learns* $u_{\mathsf{d}|\epsilon_i}$ which does not violate the saturation bound.

V. Conclusion

A SILC framework which utilizes the idea of *spatial projection* is proposed. The spatial tracking is achieved under the proposed SILC controller. Simulation results demonstrate this even with the imperfections introduced by the simulation environment. There are several future directions, including analysis and design where the full knowledge of the rate ϵ is not available.

References

[1] H.-S. Ahn, Y. Chen, and K. L. Moore, "Iterative learning control: brief survey and categorization," *IEEE Transactions on Systems Man and Cybernetics Part C Applications and Reviews*, vol. 37, no. 6, p. 1099, 2007.

[2] Y. Wang, F. Gao, and F. J. Doyle, "Survey on iterative learning control, repetitive control, and run-to-run control," *Journal of Process Control*, vol. 19, no. 10, pp. 1589–1600, 2009.

[3] J.-X. Xu, "A survey on iterative learning control for nonlinear systems," *International Journal of Control*, vol. 84, no. 7, pp. 1275–1294, 2011.

[4] J.-X. Xu and Y. Tan, *Linear and nonlinear iterative learning control*. Springer, 2003, vol. 291.

[5] Z. Bien and J.-X. Xu, *Iterative learning control: analysis, design, integration and applications*. Springer Science, 2012.

[6] K. L. Moore, *Iterative learning control for deterministic systems*. Springer Science, 2012.

[7] L. Shmuelof, J. W. Krakauer, and P. Mazzoni, "How is a motor skill learned? Change and invariance at the levels of task success and trajectory control," *Journal of neurophysiology*, vol. 108, no. 2, pp. 578–594, 2012.

[8] S. Sahoo, S. Panda, and J. Xu, "Application of spatial iterative learning control for direct torque control of switched reluctance motor drive," *Power Engineering Society General Meeting, IEEE*, pp. 1–7, 2007.

[9] Y.-H. Yang and C.-L. Chen, "Spatial-based adaptive iterative learning control of nonlinear rotary systems with spatially periodic parametric variation," *Asian Control Conference, IEEE*, pp. 698–703, 2009.

[10] D. J. Hoelzle and K. L. Barton, "A new spatial iterative learning control approach for improved micro-additive manufacturing," *American Control Conference, IEEE*, pp. 1805–1810, 2014.

[11] J.-X. Xu, "Direct learning of control efforts for trajectories with different time scales," *IEEE Transactions on Automatic Control*, vol. 43, no. 7, pp. 1027–1030, 1998.

[12] P. Janssens, W. Van Loock, G. Pipeleers, F. Debrouwere, and J. Swevers, "Iterative learning control for optimal path following problems," *Conference on Decision and Control, IEEE*, pp. 6670–6675, 2013.

[13] K. L. Moore, M. Ghosh, and Y. Q. Chen, "Spatial-based iterative learning control for motion control applications," *Meccanica*, vol. 42, no. 2, pp. 167–175, 2007.

[14] J.-X. Xu, Y. Tan, and T.-H. Lee, "Iterative learning control design based on composite energy function with input saturation," *Automatica*, vol. 40, no. 8, pp. 1371–1377, 2004.

[15] S. Kawamura and N. Sakagami, "Analysis on dynamics of underwater robot manipulators based on iterative learning control and time-scale transformation," *International Conference on Robotics and Automation, IEEE*, vol. 2, pp. 1088–1094, 2002.

[16] C. Freeman, A.-M. Hughes, J. Burridge, P. Chappell, P. Lewin, and E. Rogers, "Iterative learning control of FES applied to the upper extremity for rehabilitation," *Control Engineering Practice*, vol. 17, no. 3, pp. 368–381, 2009.

[17] W. Rudin, *Principles of mathematical analysis*. McGraw-Hill New York, 1976, vol. 3.