Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 50-1 (2017) 1977–1982

Spatial Iterative Learning Control: Output Tracking

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

* *University of Melbourne, Australia, (e-mail: x@unimelb.edu.au, $x \in \{$merid.ljesnjanin, yingt, doetomo$\}$).*
** *University of Southampton, UK, (e-mail: cf@ecs.soton.ac.uk)*

Abstract: This paper focuses on tracking *spatially repeatable* tasks. In addition, these tasks are *not* necessarily *temporally repeatable* in the sense that the *finite* length of the corresponding time interval may *change* with each repetition. Because of that, the standard Iterative Learning Control (ILC) framework is not directly applicable. Namely, the standing assumption that the *finite* length of the time interval is *fixed* with each repetition, is violated. Motivated by *human motor learning*, this paper proposes a Spatial ILC (SILC) framework which leverages the spatial repeatability. In particular, the concept of *spatial projection*, closely related to *temporal rescaling*, is proposed. This allows to *spatially* relate the relevant information from the past repetition to the present repetition. To demonstrate the proposed framework, a class of nonlinear time-varying systems with relative degree zero is selected. In particular, using contraction mapping technique, it is shown that under appropriate assumptions, the corresponding tracking error converges under the proposed SILC control law. Finally, simulation results support the obtained result.

Keywords: Spatial Iterative Learning Control, Spatial Projection, Output Tracking.

1. INTRODUCTION

Iterative Learning Control (**ILC**) is a control paradigm which focuses on problems which involve tasks that *repeat*. One classical example is the problem of packaging assembly line which includes several repeatable tasks performed by a robot, such as placing products in a box and sticking labels on it. After first defining a *desirable* way of performing each task, the goal of ILC is then to improve the corresponding *transient* behavior and to achieve *perfect* tracking. This is done by exploiting task repeatability through application of an appropriate *learning* mechanism. *Knowledge* that is acquired via *learning* enables the construction of a control law which effectively and efficiently [1] accomplishes the transient and tracking goals. Vast literature associated with ILC exists and it spans both, the theory and the practice; e.g., cf.: survey papers Ahn et al. (2007); Wang et al. (2009); Xu (2011), books Xu and Tan (2003); Bien and Xu (2012); Moore (1993) and references therein.

Indeed, ILC has been successfully applied to various problems involving repeatable tasks, as documented in the literature above. However, it is important to notice that each repeatable task satisfies the ILC standing assumption. Namely, the *finite* length of the time interval over which the task evolves is *fixed* for each repetition. Unfortunately, there are many problems where repeatable tasks do not satisfy this assumption, rendering *standard* ILC inapplicable. To illustrate this consider a repeatable task which consists of tracking a straight line of finite length from a point A to a point B; in a two dimensional space. Indeed, there are (many) examples where this task can be performed within the *fixed* finite time interval at each repetition; e.g., many industrial robots are capable of performing such a task. However, for certain problems such as the rehabilitation process, this assumption cannot be satisfied; Shmuelof et al. (2012) demonstrates this observation in a similar setup even for healthy subjects. For instance, consider stroke rehabilitation. To recover, during each session, a patient performs the same task (movement) sufficiently many times. Now, imagine that the rehabilitation process includes tracking the straight line mentioned above *without* any robotic assistance. It is unreasonable to expect that at each repetition, the patient will execute this task (movement) in a *fixed* finite time interval. Depending on the injury related factors, such as pain, range of motion, motivation, focus, fatigue, and the stage of the rehabilitation, it is very likely that the duration of the time interval will vary with each repetition. Indeed, it is natural to expect that over sufficiently many repetitions the duration of the time interval will converge to some value. Hence, for *repeatable* tasks which have its *spatial* constituent *fixed* while its *temporal* constituent *variable*, the standing assumption needs to be *relaxed* in order to leverage the existing ILC results. A natural relaxation is to consider *bounded* instead of *fixed* time intervals; notice that *fixed* time intervals are always *bounded*. This relaxation enables one to formulate problems so that the focus is on the *spatial* component in the corresponding ILC analysis and design.

[1] Other control methodologies can be used to solve this problem. However, in most cases they do not fully exploit the repeatability feature, and thus result in a less efficient and effective control laws.

2405-8963 © 2017, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2017.08.166

Related industrial research on Spatial ILC (SILC) is largely driven with particular industrial problems. For instance, these include switched reluctance motors Sahoo et al. (2007), nonlinear rotary systems Yang and Chen (2009) and micro-additive manufacturing Hoelzle and Barton (2014). Some theoretical results can be found in Janssens et al. (2013) and Moore et al. (2007). The former reference deals with the problem of minimization of the total tracking error of the repeated tasks with output constraints. On the other hand, the latter reference explores the freedom of not specifying temporal information related to the spatial movement and shows some significant practical yields. Neither reference provides a general analysis and design framework though.

This paper aims to address this need for a class of nonlinear time-varying systems with relative degree zero. Towards that goal, first a framework that focuses on the spatial aspect of the corresponding task is proposed. At the core of the proposed framework is the concept of *spatial projection* which is closely 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 Kawamura and Sakagami (2002). Namely, in Kawamura and Sakagami (2002), 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 very specific and moreover, no general analysis and design framework is provided. In the present paper, after the introduction of the spatial projection, the corresponding SILC controller is proposed. The conducted convergence analysis exploits the contraction mapping idea, cf.: Xu and Tan (2003). It is shown that under the standard ILC assumptions (akin to those in (Xu and Tan, 2003, Chapter 2)) for the considered class of systems and the appropriate assumptions related to the *spatial projection*, spatial tracking is achieved. Finally, simulation results demonstrate the convergence, even under disturbances due to computer implementation.

The paper is organized as follows. In the sequel, first the mathematical preliminaries and notational conventions are provided which is followed with the formulation of the problem in Section 2. Then, in Section 3, the corresponding assumptions are provided and main results are stated. This is followed with Section 4, in which the simulations results demonstrate the claims of the main results. Finally, in Section 5, the concluding remarks are documented.

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 denoted with $\mathbb{D}_{\diamond a} := \{x \in \mathbb{D} : x \diamond a\}$. The set of natural numbers is then defined as $\mathbb{N} := \mathbb{Z}_{>0}$ 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 $x_j \in \mathbb{D}$, $\forall j \in \{1, \ldots, d\}$, $d \in \mathbb{N}$, and throughout the document, wherever applicable, this notation is used to denote column vectors. Several norms are used throughout the document. First, $\|x\|_q^q := \sum_{j=1}^n |x_j|^q$, $(q, n) \in [1, \infty) \times \mathbb{N}$, 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 $\lambda > 0$.

2. PROBLEM FORMULATION

For the sake of clear problem formulation, first, a generic nonlinear time-varying system with relative degree zero, within the context of a *traditional* ILC framework, is considered; e.g., cf. Xu and Tan (2003). The follow up discussion then illustrates why such approach is not necessarily applicable for *spatially* repeatable tasks. Then, a *spatial projection* mapping is defined and an example of how this concept might be useful in characterizing these tasks is provided. Finally, using the concept of the spatial projection, the problem formulation is provided.

So, consider the following system, evolving over a two dimensional temporal space [2]

$$\dot{x}(i,t) = f\big(t, x(i,t), u(i,t)\big), \tag{1a}$$

$$y(i,t) = h\big(t, x(i,t), u(i,t)\big), \tag{1b}$$

where,

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

is an element of a two dimensional temporal space. Variables $x \in \mathbb{B}_x \subseteq \mathbb{R}^n$, $u \in \mathbb{B}_u \subseteq \mathbb{R}^m$ and $y \in \mathbb{B}_y \subseteq \mathbb{R}^p$, respectively, are the system state, input and output, while $(n, m, p) \in \mathbb{N}^3$. Further, respectively, $f : \mathbb{R}_{\geq 0} \times \mathbb{B}_x \times \mathbb{B}_u \to \mathbb{B}_x$ and $h : \mathbb{R}_{\geq 0} \times \mathbb{B}_x \times \mathbb{B}_u \to \mathbb{B}_y$, are the system state and output mappings. Correspondingly, consider a model of a desired behavior, $\dot{x}_{\mathsf{d}}(t) = f\big(t, x_{\mathsf{d}}(t), u_{\mathsf{d}}(t)\big)$, $y_{\mathsf{d}}(t) = h\big(t, x_{\mathsf{d}}(t), u_{\mathsf{d}}(t)\big)$, where $x_{\mathsf{d}} \in \mathbb{B}_{x_{\mathsf{d}}} \subseteq \mathbb{R}^n$, $u_{\mathsf{d}} \in \mathbb{B}_{u_{\mathsf{d}}} \subseteq \mathbb{R}^m$ and $y_{\mathsf{d}} \in \mathbb{B}_{y_{\mathsf{d}}} \subseteq \mathbb{R}^p$, respectively, are the desired behavior state, input and output. Important to note is that *only* y_{d} is given and it is assumed that there *exists* u_{d}; e.g., cf. (Xu and Tan, 2003, Assumption 2.2, Chapter 2).

Now, a standard ILC objective is to construct an ILC control law so that desired behavior is *learned* after sufficiently many iterations. One way of explicitly capturing this is by achieving $\lim_{i \to \infty} \Delta y(i,t) = (0, \ldots, 0)$, where $\Delta y(i,t) = y_{\mathsf{d}}(t) - y(i,t)$. For a relative degree zero systems, that satisfy standard ILC assumptions and some mild smoothness conditions (cf. (Xu and Tan, 2003, Chapter 2)), the so-called P-type ILC control law that achieves this limit is given as

$$u(i+1, t) = u(i,t) + \kappa \cdot \Delta y(i,t), \tag{3}$$

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

Remark 1. (Relative Degree). Note that dynamic systems with arbitrary relative degree are addressed with different types of ILC control laws. For instance, the so-called D-type ILC control law can be used for dynamic systems with relative degree one; cf. Ahn et al. (2007); Wang et al. (2009); Xu (2011); Xu and Tan (2003); Bien and Xu (2012); Moore (1993) and references therein for different types of ILC control laws. □

Now, consider again the rehabilitation example mentioned in Section 1. Namely, consider the example of a recovering patient. As explained, during each treatment (session) a patient repeats sufficiently many times a given movement

[2] Notice that $\dot{x} := dx/dt$.