2025 IEEE 64th Conference on Decision and Control (CDC)
December 10-12, 2025. Rio de Janeiro, Brazil

Learning Sparse Rational Feedforward Controllers

Tjeerd Ickenroth[1] and Tom Oomen[1,2]

2025 IEEE 64th Conference on Decision and Control (CDC) | 979-8-3315-2627-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/CDC57313.2025.11312107

***Abstract*— Iterative Learning Control (ILC) with basis function techniques are capable of improving tracking performance and task flexibility. The aim of this paper is to design a systematic approach to enable automatic rational basis function selection for feedforward learning. A sparse optimization framework is proposed to identify the most relevant rational basis functions from a large candidate set. The ILC algorithm that employs sparse optimization is able to automatically select relevant rational basis functions and is validated on an example motion system.**

I. Introduction

Feedforward control requires both structure selection and parameter tuning for high performance and varying tasks. The structure of the feedforward controller is important to enable approximations of the inverse system's dynamics. The corresponding parameters must be tuned carefully such that the feedforward filter actually fits the inverse system's dynamics. If both are well executed, the system at hand is able to accurately track varying tasks.

Iterative Learning Control (ILC) has proven in the past that it enables outstanding tracking performance for systems that operate repetitively, but that performance is limited when extrapolating the task. Successful applications for a fixed task are presented in various industries, including robotics [1], and mechatronics [2] [3]. Despite excellent tracking accuracy reported, the learned control signal is optimal for a fixed reference trajectory only. Since extrapolation to different tasks can lead to significant performance degradations [4], it is essential to investigate extrapolation capabilities in ILC to achieve high performance for multiple tasks.

Since traditional ILC lacks task extrapolation, several directions have been explored to develop task flexibility in ILC. In [5] task flexibility is achieved for a specific predefined set of references. A more general approach for task flexibility is ILC with polynomial basis functions, which construct the basis functions based on the task at hand, is successfully implemented in [6] [7] [8]. For polynomial basis function ILC it is required to determine the basis functions beforehand, and is mostly based on physical model ideas via derivatives of the task, including snap [9] [10]. Recently, in the work of [11] the selection of polynomial basis functions is automated based on data. All have shown that task extrapolation is achieved in ILC, however, the use of polynomial basis functions limits the ILC performance as only the poles of the system can be approximated.

To overcome these limitations, ILC has been developed towards rational basis functions, which include both poles and zeros to approximate the system's inverse dynamics. ILC with rational basis functions for single-input single-output (SISO) systems has initially been investigated in [12]. Follow-up studies further validated the efficacy of rational basis function ILC, including stable inversion aspects [13]. In [14] an optimal and task-flexible ILC framework with rational basis functions is established. ILC with rational basis functions is first applied on an industrial flatbed printer in [15], where it is shown to outperform polynomial basis functions. However, a major trade-off is that introducing rational basis functions makes the ILC optimization problem nonconvex, excluding the simple solution available for polynomial basis functions [16]. In practice, dedicated iterative procedures, e.g., Steiglitz-McBride iterations [17], are required to tune the rational coefficients, as shown in [12]. Moreover, the choice of the rational basis functions has until now relied on expert insight where one must select the number and form of poles and zeros *a priori* based on system knowledge. If this particular basis is mismatched to the true system, performance will be suboptimal.

Although important developments have been made in task-flexible ILC and several successful application have been reported, the rational basis function parameterization for performance and task flexibility is an open question. This motivates an automated approach to rational basis function selection in ILC.

The contributions in this work are threefold.

C1) The proposed ILC algorithm enables automatic rational basis function selection using sparse optimization techniques. This is the main contribution of this work.
C2) The proposed solution is able to deal with general rational feedforward parameterizations allowing to parameterize a large library of rational basis functions from which the relevant subset is selected.
C3) The results are numerically demonstrated on an example motion system.

The key idea is to start with a large library of candidate rational basis functions and to formulate the ILC feedforward learning algorithm as a sparse regression problem. In addition, the approach directly extends to additional polynomial terms or even friction [8]. Sparse subset selection methods in dynamic systems have been investigated in the field of system identification [18] [19] where it is a common problem to estimate the dynamics of a real system with minimal

[1]Tjeerd Ickenroth (corresponding author) and Tom Oomen are with the Control Systems Technology Section, department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands. t.s.ickenroth@tue.nl

[2]Tom Oomen is with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands.

This project is co-financed by Holland High Tech, top sector High-Tech Systems and Materials, with a PPP innovation subsidy for public-private partnerships for research and development.

parameters. Sparse estimation techniques have been successfully used for model structure selection in linear regression problems. Previous research in sparse system identification show that methods like the least absolute shrinkage and selection operator (LASSO) [20] and least angle regression and shrinkage (LARS) [21] can automatically identify a small set of significant model terms from an over-parameterized model structure, as demonstrated for polynomial basis functions in [11]. In line with results from system identification of sparse rational systems [22], in this paper a LASSO regularized optimization is developed to automate rational basis functions for ILC. This yields automatic inclusion of the most relevant rational basis functions and exclusion of irrelevant ones. The result is a low-order feedforward controller that retains the performance benefits of a more complex rational model.

Notation: In this paper, systems are discrete-time, linear, time-invariant (LIT), single-input single-output (SISO). Furthermore, $\|x\|_W^2 := x^\top W x$ for $W = W^\top \succ 0$, and $\|x\|_1 := \sum_{i=1}^{n} |x_i|$ with $x = \begin{bmatrix} x_1 \dots x_n \end{bmatrix}^\top$. The support of vector x is denoted by $\mathrm{Supp}(x) := \{i \in \mathbb{I}_1^n \mid [x]_i \neq 0\}$. For a given $\mathcal{T} \subset \mathbb{I}_1^n$, $x_\mathcal{T}$ denotes the projection of x to the support $\mathcal{T}$, i.e., $[x_\mathcal{T}]_i := [x]_i$ if $i \in \mathcal{T}$ and 0 otherwise.
$\|x\|_{\ell_p}$ denotes the usual ℓ_p-norm, $p \in \mathbb{N}$. Also, $\|x\|_0 = \sum_i (x_i \neq 0)$, i.e., the cardinality of x. Note that $\|x\|_0$ is not a norm, since it does not satisfy the homogeneity property. It relates to the general ℓ_p-norm by considering the limit $p \to 0$ of $\|x\|_p$.
Systems are generally rational in complex indeterminate z and indicated with the argument z, for example $H(z)$. Let $N \in \mathbb{Z}^+$ denote the trial length, i.e., the number of samples per trial. Let $h(k)$ at time instant $k \in \mathbb{Z}$ be the impulse response coefficients of $H(z)$, with the infinite impulse response $y(k) = \sum_{\tau=-\infty}^{\infty} h(\tau) u(k-\tau)$. Let $u(k) = 0$ for $k < 0$ and $k \geq N$ to obtain the finite-time convolution

$$\underbrace{\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(N-1) \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} h(0) & h(-1) & \cdots & h(-N+1) \\ h(1) & h(0) & \cdots & h(-N+2) \\ \vdots & \vdots & \ddots & h(-1) \\ h(N-1) & h(N-2) & \cdots & h(0) \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N-1) \end{bmatrix}}_{u},$$

with $y,\ u \in \mathbb{R}^N$ and $H \in \mathbb{R}^{N \times N}$ is the finite-time convolution matrix corresponding to $H(z)$. In case systems are described by transfer functions in z it is always mentioned, otherwise systems are described by convolution matrices.
Given a transfer function with parametric coefficients

$$H(\theta, z) = \sum_{i=1}^{m} \xi_i(z) \theta[i],$$

with parameters $\theta \in \mathbb{R}^m$, and basis functions $\xi_i(z)$ with $i \in 1, \dots, m$. The finite-time response of $H(\theta, z)$ to input u is given by

$$y = \Psi_{Hu} \theta, \tag{1}$$

with $\Psi_{Hu} = \begin{bmatrix} \xi_1 u, & \xi_2 u, & \dots, & \xi_m u \end{bmatrix} \in \mathbb{R}^{N \times m}$, here $\xi_i \in \mathbb{R}^{N \times N}$ are the convolution matrices corresponding to $\xi_i(z)$. Note that (1) is equivalent to $y = H(\theta) u$, with $H(\theta)$ the convolution matrix of $H(\theta, z)$.

II. Problem Formulation

In this section, the problem addressed in this brief is defined in detail. First, in Section II-A, the general ILC setup is introduced. Second, in Section II-B ILC with rational basis functions is introduced followed by a definition of the problem that is addressed in this brief.

A. Control setup

Consider the closed-loop system shown in Fig. 1. The system and stabilizing feedback controller are denoted by P and C, respectively. Both are considered to be discrete-time, single-input single-output (SISO), and linear time-invariant (LTI) systems. The associated ILC system, in finite time matrix notation, is described by

$$e_j = S(r_j - P f_j) - S v_j, \tag{2}$$

where the error signal $e_j \in \mathbb{R}^N$ of trial (or experiment) $j \in \mathbb{N}$ of length $N \in \mathbb{N}$ is found as a function of the sensitivity $S = (I_N + PC)^{-1} \in \mathbb{R}^{N \times N}$, the — possibly trial-varying — reference signal $r_j \in \mathbb{R}^N$, the command signal $f_j \in \mathbb{R}^N$, and the trial-varying disturbances $v_j \in \mathbb{R}^N$, including measurement noise.
Similarly to (2), the error at trial $j+1$ is given by

$$e_{j+1} = S(r_{j+1} - P f_{j+1}) - S v_{j+1}, \tag{3}$$

and under the assumption that the reference is trial invariant, i.e., $r_{j+1} = r_j$, the error propagation is given by

$$e_{j+1} = e_j - J(f_{j+1} - f_j) - S(v_{j+1} - v_j), \tag{4}$$

where $J = PS \in \mathbb{R}^{N \times N}$ is the finite-time convolution matrix of a model of the process sensitivity.

B. Norm-optimal ILC with rational basis functions

Inspired by inverse model feedforward, task flexibility is introduced in ILC via basis functions as

$$f_j = F(\theta_j) r_j, \tag{5}$$

where $F(\theta_j) \in \mathbb{R}^{N \times N}$ denotes the matrix notation of the feedforward filter parameterized by the learning parameters $\theta_j \in \mathbb{R}^m$, as shown in Fig 1. Substituting (5) into (2) yields $e_j = 0,\ \forall r_j$ if $F(\theta_j) = P^{-1}$. Hence, through the correct selection of parametrization $F(\theta_j)$ and the subsequent learning of optimal parameter values θ_j, zero error may be achieved.

The ILC optimization criterion in the context of a basis function parameterization of the feedforward model is typically defined as follows.

Definition 1 (Norm-optimal ILC with basis functions): The optimization problem for norm-optimal ILC with basis

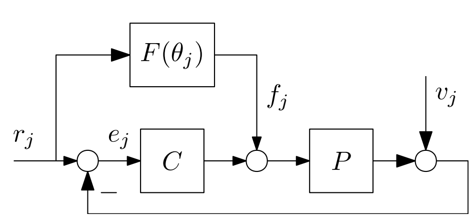

Fig. 1. Control setup.

functions is given by

$$\mathcal{J}(\theta_{j+1}) = \frac{1}{2}\|\hat{e}_{j+1}(\theta_{j+1})\|_{W_e}^2 + \frac{1}{2}\|F(\theta_{j+1})r\|_{W_f}^2 + \frac{1}{2}\|F(\theta_{j+1})r - f_j\|_{W_{\Delta f}}^2, \tag{6}$$

where $\hat{e}_{j+1}(\theta_{j+1}) = e_j - J(F(\theta_{j+1})r - f_j)$ is the predicted error propagation equivalent to (4), $\|x\|_W^2 = x^\top Wx$, and $W_e,\ W_f,\ W_{\Delta f} \succcurlyeq 0 \in \mathbb{R}^{N\times N}$ are symmetric positive (semi-)definite weighting matrices, and J is an approximate model of PS. Note that J must be sufficiently accurate, see [23] or [24] for convergence results.

Let the *rational basis function parametrization* of the feedforward update matrix be defined as

Definition 2 (Rational basis functions): The rational basis function parametrization of the matrix $F(\theta_j)$ (see (5)) in $\theta_j \in \mathbb{R}^m$ w.r.t. reference r_j as a basis, is defined through the equivalent transfer function

$$F(\theta_j, z) = \frac{A(\theta_j, z)}{B(\theta_j, z)}, \tag{7a}$$

for

$$A(\theta_j, z) = \xi_0^A(z) + \sum_{i=1}^{m} \xi_i^A(z)\theta_j[i],$$
$$B(\theta_j, z) = \xi_0^B(z) + \sum_{i=1}^{m} \xi_i^B(z)\theta_j[i], \tag{7b}$$

where $\xi_i^A(z), \xi_i^B(z) \in \mathbb{R}[z]$, for $i \in \{1, 2, \ldots, m\}$, are polynomials in z, and $A(\theta_j, z)$ and $B(\theta_j, z)$ are linear in the parameters θ.

Clearly, the substitution of (7a) into the criterion in (6) reveals that the resulting problem is nonlinear in the optimization variable θ_{j+1}, which does not allow one to directly apply sparse optimization techniques, such as LASSO, to control the number of zeros in the solution. The number of nonzero elements in a vector are measured by the ℓ_0 norm, i.e., $\|\theta_{j+1}\|_0$. The addition of the $\|\theta_{j+1}\|_0$ norm as a constraint to (6) would lead to a nonconvex optimization problem, which is difficult to solve. Then, the problem addressed in this paper is formulated as

Problem 1: Consider the ILC criterion in Definition 1 together with the rational feedforward filter structure in (7a). The problem is automatic selection of rational basis functions in ILC by controlling the number of nonzero elements in θ_{j+1}, i.e., by controlling $\|\theta_{j+1}\|_0$, given the objective that is nonlinear in the parameters θ_{j+1}.

III. Approach

Let the parameters that minimize the performance criterion (6) when subject to a rational feedforward parameterization be denoted as

$$\theta_{j+1}^* = \arg\min_{\theta_{j+1}}\ \mathcal{J}(\theta_{j+1}). \tag{8}$$

The use of rational basis functions instead of polynomial basis functions yields a far more complex form of problem (8).

Theorem 1 (No analytical solution): The minimizer defined in (8), if the feedforward is parameterized as in (2), cannot be computed analytically.

Proof: See [12]. ■

In the following, we first introduce a generalization of the convex solution method of [12] which is required to enforce linearity in the optimization variable θ_{j+1}. Thereafter, our main contribution is presented that enables automatic rational basis function selection in ILC via LASSO, and is summarized in an algorithm.

A. Convex solution to ILC with rational basis functions

This section first provides a generalization of the ILC with rational basis functions solution from [12]. The main idea is to solve a sequence of least-squares problems and to consider the nonlinear terms as *a priori* unknown weighting functions. Let us rewrite (6) as

$$\bar{\mathcal{J}}(\theta_{j+1}) = \frac{1}{2}\left\|B(\theta_{j+1})^{-1}\left[B(\theta_{j+1})\,\hat{e}_{j+1}\right]\right\|_{W_e}^2 + \frac{1}{2}\left\|B(\theta_{j+1})^{-1}\left[B(\theta_{j+1})\,f_{j+1}\right]\right\|_{W_f}^2 + \frac{1}{2}\left\|B(\theta_{j+1})^{-1}\left[B(\theta_{j+1})\,f_{j+1}\right] - f_j\right\|_{W_{\Delta f}}^2, \tag{9}$$

where $B(\cdot)$ is the matrix (lifted) equivalent of the $B(\cdot, z)$ transfer function (7). Clearly, (9) is still nonlinear due to the $B(\theta_{j+1})^{-1}$-term. Yet, all other terms ($B(\theta_{j+1})\hat{e}_{j+1}$ and $B(\theta_{j+1})f_{j+1}$) now do depend linearly on θ_{j+1}.

We then introduce an auxiliary index $k \in \mathbb{N}$, describing the approximate solution iteration number as $\theta_{j+1}^{\langle k\rangle}$, which we use to create

$$\bar{\mathcal{J}}_k\left(\theta_{j+1}^{\langle k\rangle}\right) = \frac{1}{2}\left\|B\left(\theta_{j+1}^{\langle k-1\rangle}\right)^{-1}\left[B\left(\theta_{j+1}^{\langle k\rangle}\right)\hat{e}_{j+1}^{\langle k\rangle}\right]\right\|_{W_e}^2 + \frac{1}{2}\left\|B\left(\theta_{j+1}^{\langle k-1\rangle}\right)^{-1}\left[B\left(\theta_{j+1}^{\langle k\rangle}\right)f_{j+1}^{\langle k\rangle}\right]\right\|_{W_f}^2 + \frac{1}{2}\left\|B\left(\theta_{j+1}^{\langle k-1\rangle}\right)^{-1}\left[B\left(\theta_{j+1}^{\langle k\rangle}\right)f_{j+1}^{\langle k\rangle}\right] - f_j\right\|_{W_{\Delta f}}^2, \tag{10}$$

where

$$\hat{e}_{j+1}^{\langle k\rangle} = e_j + Jf_j - B(\theta_{j+1}^{\langle k\rangle})^{-1}A(\theta_{j+1}^{\langle k\rangle})Jr_j.$$

Note that we recover (9) upon convergence, that is, for $\theta_{j+1} = \theta_{j+1}^{\langle k\rangle} = \theta_{j+1}^{\langle k-1\rangle}$.

For known $\theta_{j+1}^{\langle k-1\rangle}$, the criterion in (10) is quadratic in the optimization variables $\theta_{j+1}^{\langle k\rangle}$, and hence can be calculated analytically. The nonlinear $B(\theta_{j+1}^{\langle k-1\rangle})^{-1}$ is thus fixed at each iteration k, and interpreted as an iteratively adjusted weighting function. By iterating over $\theta_{j+1}^{\langle k\rangle}$, it is aimed that the *a priori* unknown weighting by $B(\theta_{j+1}^{\langle k-1\rangle})^{-1}$ is effectively compensated after convergence of the iterative procedure [12]. As, given $\theta_{j+1}^{\langle k-1\rangle}$, the least squares solution clearly is able to solve criterion (10), the following theorem manages to provide the analytic solution to finding the values $\theta_{j+1}^{\langle k\rangle}$ that minimize in (10).

Theorem 2: Given $\theta_{j+1}^{\langle k-1\rangle}$, f_j, r_j, e_j, and the rational feedforward parameterization of Definition 2. The minimizer

$$\theta_{j+1}^{\langle k\rangle *} = L^{\langle k\rangle} e_j + Q^{\langle k\rangle} f_j + R^{\langle k\rangle} r_j, \tag{11}$$

with

$$\begin{aligned} L^{\langle k\rangle} &= \Psi_0^{-1}\Psi_1^\top W_e B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\xi_0^B, \\ Q^{\langle k\rangle} &= \Psi_0^{-1}\left(\Psi_2^\top W_{\Delta f} - \Psi_1^\top W_e B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\xi_0^B J\right), \\ R^{\langle k\rangle} &= -\Psi_0^{-1}\Big(\Psi_1^\top W_e B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\xi_0^A J \\ &\qquad + \Psi_2^\top (W_f + W_{\Delta f}) B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\xi_0^A\Big), \end{aligned} \tag{12}$$

where

$$\begin{aligned} \Psi_0 &= \Psi_1^\top W_e \Psi_1 + \Psi_2^\top (W_f + W_{\Delta f})\Psi_2, \\ \Psi_1 &= B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\left(\Psi_{Jr_j}^A - \Psi_{e_j}^B - \Psi_{Jf_j}^B\right), \\ \Psi_2 &= B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\Psi_{r_j}^A, \end{aligned}$$

with the basis functions in lifted notation given by

$$\begin{aligned} \Psi_{Jr_j}^A &= \left[\xi_1^A J r_j \quad \xi_2^A J r_j \quad \cdots \quad \xi_m^A J r_j\right], && \in \mathbb{R}^{N\times m}, \\ \Psi_{r_j}^A &= \left[\xi_1^A r_j, \quad \xi_2^A r_j, \quad \ldots, \quad \xi_m^A r_j\right], && \in \mathbb{R}^{N\times m}, \\ \Psi_{Jf_j}^B &= \left[\xi_1^B J f_j, \quad \xi_2^B J f_j, \quad \ldots, \quad \xi_m^B J f_j\right], && \in \mathbb{R}^{N\times m}, \\ \Psi_{e_j}^B &= \left[\xi_1^B e_j, \quad \xi_2^B e_j, \quad \ldots, \quad \xi_m^B e_j\right], && \in \mathbb{R}^{N\times m}, \end{aligned}$$

finds a global minimum of (10).

Proof: Let

$$\begin{aligned} A(\theta_{j+1}^{\langle k\rangle}) J r_j &= \Psi_{Jr_j}^A \theta_{j+1}^{\langle k\rangle} + \xi_0^A J r_j, \\ A(\theta_{j+1}^{\langle k\rangle}) r_j &= \Psi_{r_j}^A \theta_{j+1}^{\langle k\rangle} + \xi_0^A r_j, \\ B(\theta_{j+1}^{\langle k\rangle}) J f_j &= \Psi_{Jf_j}^B \theta_{j+1}^{\langle k\rangle} + \xi_0^B J f_j, \\ B(\theta_{j+1}^{\langle k\rangle}) e_j &= \Psi_{e_j}^B \theta_{j+1}^{\langle k\rangle} + \xi_0^B e_j. \end{aligned} \tag{13}$$

Substituting the above expressions into (10), we note that (10) is quadratic in $\theta_{j+1}^{\langle k\rangle}$. A necessary condition for optimality is $\partial\bar{\mathcal{J}}_k/\partial\theta_{j+1}^{\langle k\rangle} = 0$. Solving for $\theta_{j+1}^{\langle k\rangle}$ through the use of identity

$$\frac{d\left(\frac{1}{2}\|Ax+b\|_W^2\right)}{dx} = (Ax+b)^\top W A,$$

which holds for $x,\ b \in \mathbb{R}^N,\ A \in \mathbb{R}^{N\times N}$, and $W = W^\top \in \mathbb{R}^{N\times N}$, completes the proof. ■

It is important to note that the parameter vector which minimizes the weighted cost function in (10) is a stationary point of that function, but is not guaranteed to be a minimum of the original, unweighted problem in (6). This is because the method's weighting alters the objective function's error landscape, a key difference from methods that apply weighting at the gradient level, as established in [14].

Remark 1: Note that Theorem 2 in [12] is recovered as a special case of our Theorem 2 when $\xi_0^A = 0_{N\times N}$ and $\xi_0^B = I_N$. This Theorem concludes contribution C2.

The specific choice of basis functions $\xi_i^A(z)$ and $\xi_i^B(z)$ for $i \in \{1, 2, \ldots, m\}$, determines the feedforward filter structure. Thus, this choice governs how well the feedforward filter $F(\theta, z)$ will be able to fit the model class of the inverse system P^{-1}, i.e., whether our parameterization is *rich enough*. For systems with complex dynamics, this selection of basis functions is non-trivial. Even for simple dynamic systems, if the set of basis functions is not rich enough to match the model class of P^{-1}, ILC performance can be significantly limited as a result. The next section provides our approach to automating the process of basis function selection, presenting the main contribution of the paper.

B. Automated rational basis function selection via ℓ_1 regularization

This work connects the convex solution presented in Section III-A with the sparsity promoting technique referred as ℓ_1-regularization. Automatic basis function selection is achieved by promoting sparse solutions to optimization criterion (10). Enforcing exact zeros in the solution vector $\theta_{j+1}^{\langle k^*\rangle}$ after k^* iterations, results in effectively canceling non-relevant basis functions. Sparse solutions are obtained by constraining the objective function in (10) as follows

$$\min_{\theta_{j+1}} \quad \bar{\mathcal{J}}_{k^*}(\theta_{j+1}^{\langle k^*\rangle}) \quad \text{s.t.} \quad \|\theta_{j+1}^{\langle k^*\rangle}\|_1 \le s, \tag{14}$$

where $s \in \Re_{>0}$ is an upper bound on the ℓ_1 norm of the parameters, i.e., a small s leads to highly sparse results. The optimization problem in (14) is generally referred as ℓ_1 regularization, or LASSO, which is the convex relaxation of the nonconvex $\|\theta_{j+1}\|_0$ constraint.

The convex optimization problem in (14) is solved along the regularization path via LARS, enabling exact sparsity in the parameters controlled by a hyperparameter n_θ, instead of manually tuning the value for s in (14).

In order to solve (14) let us rewrite the objective in the least-squares form, that is,

$$\frac{1}{2}\|Y_j - X_j\theta_{j+1}^{\langle k^*\rangle}\|_2^2 \quad \text{s.t.} \quad \|\theta_{j+1}^{\langle k^*\rangle}\|_1 \le s, \tag{15}$$

where Y_j the response vector and X_j the predictor matrix. Substitute

$$\hat{e}_{j+1}^{\langle k\rangle} = e_j + J f_j - B(\theta_{j+1}^{\langle k\rangle})^{-1} A(\theta_{j+1}^{\langle k\rangle}) J r_j$$

and the expressions defined in (13) in (10), and use that $\|\cdot\|_W^2 = \|W^{\frac{1}{2}}\cdot\|_2^2$ to obtain the following expressions:

$$Y_j = \begin{bmatrix} W_e^{\frac{1}{2}} B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\left(\xi_0^B(e_j + J f_j) - \xi_0^A J r_j\right) \\ W_f^{\frac{1}{2}} B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\xi_0^A r_j \\ W_{\Delta f}^{\frac{1}{2}} B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\xi_0^A r_j - W_{\Delta f}^{\frac{1}{2}} f_j \end{bmatrix} \in \mathbb{R}^{3N},$$

and

$$X_j = \begin{bmatrix} W_e^{\frac{1}{2}} B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\left(\Psi_{Jr_j}^A - \Psi_{e_j}^B - \Psi_{Jf_j}^B\right) \\ W_f^{\frac{1}{2}} B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\Psi_{r_j}^A \\ W_{\Delta f}^{\frac{1}{2}} B(\theta_{j+1}^{\langle k-1\rangle})^{-1}\Psi_{r_j}^A \end{bmatrix} \in \mathbb{R}^{3N\times m}. \tag{16}$$

The sparse rational basis function (SRBF-ILC) algorithm is presented in Algorithm 1, which is the main contribution C1 of this work. The algorithm enables efficient automatic rational basis function selection via sparse subset selection. This allows a large set of m initial basis functions to be parameterized from which the algorithm selects the n_θ

Algorithm 1 Sparse Rational Basis Function ILC

Require: a defined set of m rational basis functions according to Definition 2.

1: Select desired number of feedforward parameters n_θ.
2: Initialize weighting matrices $W_e,\ W_f,\ W_{\Delta f} \succcurlyeq 0$.
3: Initialize $\theta_{j\in 1,\ldots,N_{\text{trials}}} = \mathbf{0}^{m\times 1}$.
4: **for** $j = 1$ to N_{trials} **do**
5: Perform a trial with $f_j = F(\theta_j^*)r_j$ and measure e_j.
6: Set $k = 0$, and let $\theta_{j+1}^{\langle 1\rangle} = \theta_j$.
repeat
7: $k \leftarrow k + 1$ and determine $L^{\langle k\rangle}$, $Q^{\langle k\rangle}$, and $R^{\langle k\rangle}$ via (12).
8: Determine $\theta_{j+1}^{\langle k\rangle}$ via (11).
until $\theta_{j+1}^{\langle k\rangle}$ has converged or the maximum number of iterations is reached, resulting in $\theta_{j+1}^{\langle k=k^*\rangle}$, proceed.
9: Determine Y_j and X_j via (16) for $k = k^*$.
10: Obtain sparse estimate $\theta_{j+1}^{\mathcal{S}} = \text{LARS}(X_j, Y_j, n_\theta)$, i.e., solve (14).
11: Select n_θ non-sparse indices $\mathcal{T} \subseteq \text{Supp}(\theta_{j+1}^{\mathcal{S}})$.
12: Construct active set $X_{j,\mathcal{T}} \subseteq X_j$.
13: Determine debiased learning parameters $\theta_{j+1,\mathcal{T}}^* = (X_{j,\mathcal{T}}^\top X_{j,\mathcal{T}})^{-1} X_{j,\mathcal{T}}^\top Y_j$.
14: Set $\theta_{j+1}^* = \theta_{j+1,\mathcal{T}}^*$.
15: **end for**

most relevant that achieve high tracking performance. The corresponding parameters θ_j are optimally learned in the least-squares sense at each iteration j of the learning process.

Remark 2: Note that Algorithm 1 in [11] is recovered as a special case of our Algorithm (1) when $B(\theta_j, z) = 1$, i.e., for polynomial basis functions.

Remark 3: See [25] for a LARS implementation.

C. Implementation aspects

The sparse subset selection in Algorithm 1 is obtained by solving the LASSO problem in (14) via LARS which results in biased parameters, as is generally known for LASSO regression. An additional debiasing step is added to the algorithm to obtain the true learning parameter values by solving an additional least squares problem for the selected subset (see lines 11 till 13 in Algorithm 1.

Notice that there is no guarantee that the resulting feedforward filter $F(\theta_j)$ is a stable filter. If the plant P contains non-minimumphase zeros, then it is directly seen from (2) that the optimal $F(\theta_j^*, z)$ is an unstable filter. If $F(\theta_j, z)$ is unstable, the filtering operations cannot be performed in the usual manner, since forward time-domain computation leads to unbounded results. To obtain stable feedforward signals via an exact inverse of the system stable inversion is adopted, as shown in [15] and originates from [26]. The same approach is used in case $B(\theta_{j+1}^{\langle k-1\rangle})^{-1}$ in (12) is unstable.

IV. Numerical Example

In this section, a simulation example demonstrates that the proposed algorithm 1 is able to automatically select the relevant set of basis functions that describes the true inverse

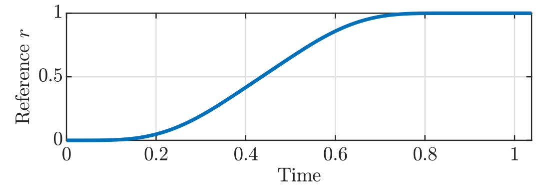

Fig. 2. Reference signal.

of the plant. It is shown that the sparse solution consistently achieves high tracking performance confirming contribution C3. An open-loop system is considered (i.e., $C = 0$) with the system defined as

$$P(\theta, z) = \frac{z^5 - 4.8z^4 + 9.4z^3 - 9.1z^2 + 4.4z - 0.8}{z^5 - 4.9z^4 + 9.7z^3 - 9.6z^2 + 4.7z - 0.9}, \quad (17)$$

with five poles including one integrator and five zeros, and is plotted in Fig. 3.

The reference is plotted in Fig. 2, with trial length $N = 1039$. The weighting filters in Definition 1 are set to $W_e = I$ and $W_f = W_{\Delta f} = 0$ in order to only weigh the error. Zero mean normally distributed measurement noise $v_j \sim \mathcal{N}(0, \sigma^2)$ with $\sigma = 10^{-5}$ is added to the simulation, that is trial varying, having an SNR of 96 dB.

The feedforward filter is parameterized as in Definition 2

$$F(\theta_j, z) = \frac{A(\theta_j, z)}{B(\theta_j, z)},\ \theta_j \in \mathbb{R}^{14}, \quad (18)$$

with $A(\theta_j, z) = \sum_{i=1}^{7} \xi_i^A(z)\theta_j[i]$, and $B(\theta_j, z) = 1 + \sum_{i=1}^{7} \xi_i^B(z)\theta_j[i]$, where $\xi_i^A(z) = \xi_i^B(z) = (\frac{z-1}{z})^i$ the discrete time derivative operator, and $\xi_0^A(z) = 0$ and $\xi_0^B(z) = 1$ to ensure a DC gain equal to 0 to fit the plant's integrator. Note that the feedforward filter has $m = 14$ learning parameters. Algorithm 1 has been executed for 6 ILC trials and $k = 10$ iterations. The desired number of sparse feedforward filter parameters $n_\theta = 10$, equal to the plant's number of parameters. Fig. 3 shows the resulting feedforward filters after 3 ILC trials for $F(\theta, z)$ with $\theta \in \mathbb{R}^{14}$ and sparse filter $F(\theta^{\mathcal{S}}, z)$ with $\|\theta^{\mathcal{S}}\|_0 = 10$. It is shown that the sparse solution approximates the inverse systems dynamics almost perfectly, whereas the overparameterized solution overfits the system's dynamics. Fig 4 (left) shows that the sparse solution consistently achieves high tracking performance, whereas the overparameterized solution is vulnerable to overfitting. Especially in the high frequency range as is also clear from the time domain error plot in Fig. 4 (right).

V. Conclusion

In this paper, a sparse rational basis function iterative learning control (SRBF-ILC) framework has been developed, incorporating automatic selection of rational basis functions to enhance tracking performance and robustness to varying reference trajectories. The proposed approach leverages a LASSO-based optimization to identify a minimal yet effective subset of rational basis functions from a comprehensive candidate set, ensuring accurate system inversion while mitigating overparameterization. This selection process identifies the rational basis functions that contribute most

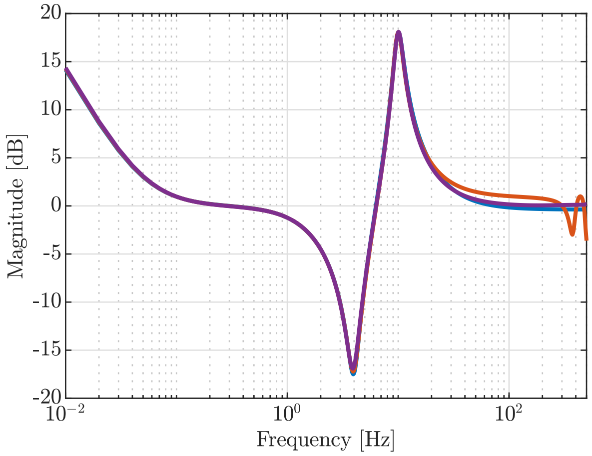

Fig. 3. Bode magnitude plots for the plant $P(\theta, z)$ with $\theta \in \mathbb{R}^{10}$ in (—), and the inverse feedforward filters $F(\theta, z)$ with $\theta \in \mathbb{R}^{14}$ in (—), and $F(\theta^{S}, z)$ with $\|\theta^{S}\|_0 = 10$ in (—) after 3 ILC trials. The sparse filter exhibits significantly lower sensitivity to overfitting and obtains almost a true inverse of the plant dynamics.

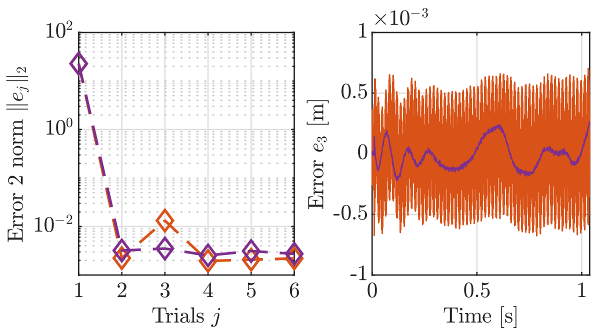

Fig. 4. (Left) Error 2 norm for 6 ILC trials for the non-sparse solution having 14 parameters in (-◇-), and for the proposed sparse solution with 10 parameters in (-◇-). (Right) Time domain error plot at trial $j = 3$ for the non-sparse solution in (—), and for the sparse solution in (—). The sparse solution consistently obtains high performance as the true plant inverse is obtained.

to minimizing the tracking error, with the level of sparsity directly controlled by the user. The SRBF-ILC algorithm is validated on a numerical motion system example demonstrating that it is robust to overparameterization and consistently achieves high ILC tracking performance while being robust to task variations. Future research will focus on applying the weighting at gradient level to enhance better convergence.

References

[1] J. W. Axehill, I. Dressler, S. Gunnarsson, A. Robertsson, and M. Norrlöf, "Estimation-based ILC applied to a parallel kinematic robot," *Control Engineering Practice*, vol. 33, pp. 1–9, 2014.

[2] J. Bolder, S. Kleinendorst, and T. Oomen, "Data-driven multivariable ILC: enhanced performance by eliminating L and Q filters," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 12, pp. 3728–3751, 2018.

[3] D. Huang, J. X. Xu, V. Venkataramanan, and C. T. Huynh, "High-performance tracking of piezoelectric positioning stage using current-cycle iterative learning control with gain scheduling," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 2, pp. 1085–1098, 2014.

[4] F. Boeren, A. Bareja, T. Kok, and T. Oomen, "Frequency-Domain ILC Approach for Repeating and Varying Tasks: With Application to Semiconductor Bonding Equipment," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 6, pp. 2716–2727, 2016.

[5] D. J. Hoelzle, A. G. Alleyne, and A. J. W. Johnson, "Basis Task Approach to Iterative Learning Control With Applications to Micro-Robotic Deposition," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 5, pp. 1138–1148, 2011.

[6] S. H. Van Der Meulen, R. L. Tousain, and O. H. Bosgra, "Fixed structure feedforward controller design exploiting iterative trials: Application to a wafer stage and a desktop printer," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 130, no. 5, pp. 0510061–05100616, 2008.

[7] J. Van De Wijdeven and O. H. Bosgra, "Using basis functions in iterative learning control: Analysis and design theory," *International Journal of Control*, vol. 83, no. 4, pp. 661–675, 2010.

[8] 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," 2014.

[9] M. Poot, M. van Haren, D. Kostic, J. Portegies, and T. Oomen, "Position-Dependent Motion Feedforward via Gaussian Processes: Applied to Snap and Force Ripple in Semiconductor Equipment," *IEEE Transactions on Control Systems Technology*, vol. 0, no. 0, pp. 1–16, 2024.

[10] P. Lambrechts, M. Boerlage, and M. Steinbuch, "Trajectory planning and feedforward design for electromechanical motion systems," *Control Engineering Practice*, vol. 13, no. 2, pp. 145–157, 2005.

[11] T. Ickenroth, M. V. Haren, J. Kon, M. V. Meer, J. V. Hulst, and T. Oomen, "Automatic Basis Function Selection in Iterative Learning Control : A Sparsity-Promoting Approach Applied to an Industrial Printer," No. March, pp. 1–6, 2025.

[12] J. Bolder and T. Oomen, "Rational basis functions in iterative learning control - With experimental verification on a motion system," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 722–729, 2015.

[13] L. Blanken, F. Boeren, D. Bruijnen, and T. Oomen, "Batch-To-Batch Rational Feedforward Control: From Iterative Learning to Identification Approaches, with Application to a Wafer Stage," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 2, pp. 826–837, 2017.

[14] 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.

[15] J. Bolder, J. Van Zundert, S. Koekebakker, and T. Oomen, "Enhancing flatbed printer accuracy and throughput: Optimal rational feedforward controller tuning via iterative learning control," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4207–4216, 2017.

[16] M. Poot, J. Portegies, D. Kostić, and T. Oomen, "Rational Basis Functions in Iterative Learning Control for Multivariable Systems," *Proceedings of the IEEE Conference on Decision and Control*, vol. 2023, pp. 4644–4649, 2023.

[17] K. Steiglitz and L. E. McBride, "A Technique for the identification of linear systems," *IEEE Transactions on Automatic Control*, vol. 10, no. 4, pp. 461–464, 1965.

[18] C. R. Rojas and H. Hjalmarsson, "Sparse estimation based on a validation criterion," *Proceedings of the IEEE Conference on Decision and Control*, pp. 2825–2830, 2011.

[19] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Sparse Identification of Nonlinear Dynamics with Control (SINDYc)," in *IFAC-PapersOnLine*, vol. 49, pp. 710–715, 2016.

[20] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *J. R. Statist. Soc. B*, no. 1, pp. 267–288, 1996.

[21] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least Angle Regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[22] C. R. Rojas, R. Tóth, and H. Hjalmarsson, "Sparse Estimation of Polynomial and Rational Dynamical Models," *IEEE Transactions on Automatic Control*, vol. 59, no. 11, 2014.

[23] D. Bristow, M. Tharayil, and A. Alleyne, "Survey of iterative learning control: a learning-based method for high-performance tracking control," *IEEE control systems magazine*, vol. 26, no. 3, pp. 96–114, 2006.

[24] J. van de Wijdeven, T. Donkers, and O. Bosgra, "Iterative Learning Control for uncertain systems: Robust monotonic convergence analysis," *Automatica*, vol. 45, no. 10, pp. 2383–2391, 2009.

[25] K. Sjöstrand, L. H. Clemmensen, G. Einarsson, R. Larsen, and B. Ersboll, "SpaSM: a MATLAB Toolbox for Sparse Statistical Modeling," *Journal of Statistical Software*, vol. 84, no. 10, 2018.

[26] S. Devasia, D. Chen, and B. Paden, "Nonlinear inversion-based output tracking," *IEEE Transactions on Automatic Control*, vol. 41, no. 7, pp. 930–942, 1996.