Random Learning Leads to Faster Convergence in 'Model-Free' ILC With Application to MIMO Feedforward in Industrial Printing

Aarnoudse, Leontine; Oomen, Tom

DOI
10.1002/acs.3903

Publication date
2024

Document Version
Final published version

Published in
International Journal of Adaptive Control and Signal Processing

International Journal of Adaptive Control and Signal Processing

WILEY

RESEARCH ARTICLE

Random Learning Leads to Faster Convergence in 'Model-Free' ILC: With Application to MIMO Feedforward in Industrial Printing

Leontine Aarnoudse[1] | Tom Oomen[1,2]

[1]Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands | [2]Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands

Correspondence: Leontine Aarnoudse (l.i.m.aarnoudse@tue.nl)

Received: 8 March 2024 | **Revised:** 13 August 2024 | **Accepted:** 15 August 2024

Funding: Nederlandse Organisatie voor Wetenschappelijk Onderzoek, Grant/Award Number: 15698

Keywords: feedforward control | gradient estimation | iterative learning control | MIMO systems | optimization

ABSTRACT

Model-free iterative learning control (ILC) can lead to high performance by attenuating repeating disturbances completely, using dedicated experiments on the real system to replace the traditional model. The aim of this paper is to develop a fast data-driven method for MIMO ILC that uses random learning in the form of efficient unbiased gradient estimates. This is achieved by developing a stochastic conjugate gradient algorithm, in which the search direction and optimal step size are generated using dedicated experiments. The approach is applied to MIMO automated feedforward tuning. Simulation and experimental results show that the method is superior to earlier stochastic and deterministic methods.

1 | Introduction

Iterative learning control (ILC) leads to high performance in many applications that perform repeating tasks, by attenuating repeating disturbances completely. Through iterative updating, a compensating input signal is learned. Typically, this is achieved by combining approximate models with measured data. Examples of ILC frameworks include frequency-domain ILC [1] and optimization-based approaches such as lifted norm-optimal ILC [2] and gradient-descent ILC [3].

In contrast to model-based ILC approaches, direct data-driven approaches do not suffer from performance limitations caused by model uncertainties, and in addition they avoid the costly process of modeling and identification [4–6]. Robustness is typically provided by Q-filters in model-based frequency-domain ILC, which also reduce the attenuation of repeating disturbances at certain frequencies [7], by regularization in norm-optimal ILC, which restricts the input signal, or through robust design [3, 8]. This paper considers 'model-free' ILC, where the term model-free is used to convey that these approaches use purely data-based gradients to minimize a criterion. These approaches of course assume several system properties, such as that the system is an $n_i \times n_o$ linear time-invariant MIMO system, and essentially replace filtering operations through a model by experiments on the real system. In the remainder of the paper, the term 'model-free' is omitted. Most ILC methods, in particular direct data-driven methods, are developed for SISO systems. In [9], a MIMO ILC algorithm is introduced, which uses gradient-descent ILC to minimize a cost criterion, using gradients that are generated through experiments on the adjoint system [10, 11]. The approach is related to approaches such as extremum-seeking based ILC [12]. In [9] and comparable experiment-based iterative methods for MIMO systems, for example, iterative feedback

tuning [13] and H_∞-norm estimation [14], the gradient for an $n_i \times n_o$ MIMO system is generated through $n_i n_o$ experiments. Therefore, the method does not scale well for massive MIMO systems: a 10×10 system already requires 100 experiments per iteration.

To improve the feasibility of MIMO ILC, it is required to reduce the number of experiments per iteration, as deterministic gradients are too experimentally expensive, and the number of iterations, as standard gradient descent is typically slow. In [15], ILC for MIMO systems is further developed by replacing the experimentally expensive deterministic gradient from [9] with an unbiased estimate generated by a single experiment. The approach relates to simultaneous perturbation stochastic approximation [16] and employs the gradient estimates in a stochastic gradient descent optimization scheme. While this stochastic approximation adjoint ILC (SAAILC) reduces the number of experiments significantly compared to [9], the number of iterations is still high because a first-order optimization algorithm is used.

Methods to increase the convergence speed of first-order optimization algorithms are typically focused on deterministic optimization, and cannot be applied directly to stochastic MIMO ILC. Regarding stochastic optimization, several stochastic quasi-Newton methods have been developed that consider mini-batching, an approach in the field of machine learning that is non-deterministic, but which nevertheless can use multiple gradient evaluations of one mini-batch to obtain locally deterministic Hessian estimates, see, for example, [17, 18]. The consistency assumption employed in these methods cannot be satisfied for the stochastic gradients used in ILC. While there exist some recent results that do not depend on consistency, see, for example, [19] which uses Gaussian processes to model the inverse Hessian, conjugate gradients turn out to be more suitable for the quadratic objectives used in ILC [20, sect. 8.3]. Existing stochastic conjugate gradient methods apply just a few conjugate gradient steps within a mini-batch, assuming exact gradients of the subset of data contained in that batch [21, 22]. To determine the step size, [21] uses an inexact Wolfe line search, while [22] uses an exact line search that employs Hessian-vector products [23] that can be obtained efficiently and automatically using gradient computations. This approach does not extend to the measured gradient estimates in stochastic MIMO ILC.

Although significant steps have been taken towards efficient ILC for MIMO systems, a fast and accurate data-driven approach is underdeveloped. The aim of this paper is the development of a method that enables fast and accurate ILC for massive MIMO systems using only experimental data. To that end, a stochastic conjugate gradient descent algorithm is developed that uses randomized experiments to determine experimentally cheap unbiased gradient approximations, conjugate search directions and optimal step sizes. In addition, the approach is extended to the automated tuning of feedforward parameters for MIMO systems in a case study. Since the number of parameters is limited, this results in a straightforward approach that is feasible in practice, as illustrated by the implementation on an industrial flatbed printer. The main contribution of this paper is the introduction of a framework for efficient data-driven MIMO ILC using stochastic conjugate gradient descent with optimal step sizes. This contribution consists of the following elements.

- A framework for efficient data-driven MIMO ILC using stochastic conjugate gradient descent with optimal step sizes is introduced that encompasses norm-optimal ILC and automated feedforward control.
- Complete proofs are provided, including an analysis of the convergence of the proposed algorithm.
- The proposed approach is compared to alternative deterministic and stochastic methods, including Broyden–Fletcher–Goldfarb–Shanno (BFGS) and stochastic gradient descent with optimal step sizes, both analytically and in simulation.
- Experimental results on an industrial flatbed printer are provided.

Preliminary results appear in [24], which presents an early version of the stochastic conjugate gradient ILC approach, and [25], which applies stochastic approximation adjoint ILC with optimal step sizes to automated feedforward control in simulation. In the current paper, new results are developed that replace these original ideas, leading to a single stochastic conjugate gradient descent ILC framework that encompasses both norm-optimal ILC and automated feedforward control for MIMO systems. In addition, the paper includes complete proofs, a convergence analysis, comparisons to alternative deterministic and stochastic methods including BFGS, extensive simulations, and experimental results.

This paper is structured as follows. In Section 2, the problem is introduced. In Section 3, a framework for ILC with unbiased gradient estimates is provided that encompasses both standard ILC and parameterized feedforward tuning, and stochastic conjugate gradient descent with optimal step sizes is introduced. In Section 4, implementation aspects are considered and the proposed method is compared to alternative stochastic and deterministic approaches. A case study on MIMO feedforward tuning is provided in Section 5. Simulation and experimental results are given in respectively Sections 6 and 7. Lastly, conclusions are given in Section 8.

2 | Problem Formulation

In this section, the problem considered in this article is introduced. The aim of finding an input signal f that minimizes the error e of a control system is expressed by the criterion

$$\mathcal{J}(\theta) = \|e\|_{W_e}^2 + \|f(\theta)\|_{W_f}^2 \tag{1}$$

with cost $\mathcal{J}$, $\|x\|_W^2 = x^\top W x$ and weighting matrices $W_e \succ 0$, $W_f \succcurlyeq 0$ (see, e.g., [2] on how to select these weights). The input $f(\theta)$ is directly parameterized by the decision variable θ, and the error e depends on θ through application of f to a system J. The MIMO linear time-invariant (LTI) system J with n_i inputs and n_o outputs can represent both open-loop and closed-loop systems and is given in lifted form by

$$\underbrace{\begin{bmatrix} e^1 \\ \vdots \\ e^{n_o} \end{bmatrix}}_{e} = \underbrace{\begin{bmatrix} r^1 \\ \vdots \\ r^{n_o} \end{bmatrix}}_{r} - \underbrace{\begin{bmatrix} J^{11} & \dots & J^{1n_i} \\ \vdots & & \vdots \\ J^{n_o 1} & \dots & J^{n_o n_i} \end{bmatrix}}_{J} \underbrace{\begin{bmatrix} f^1(\theta) \\ \vdots \\ f^{n_i}(\theta) \end{bmatrix}}_{f(\theta)} \tag{2}$$

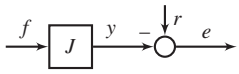

FIGURE 1 | System J with disturbance r, input f, output y and error $e = r - Jf$ according to (2). The $n_i \times n_o$ system J can represent both open-loop and closed-loop systems.

see also Figure 1. The dimensions of the input $f(\theta)$, error e, unknown exogenous disturbance r and output $y = Jf$ are given by $y^l, e^l, r^l, f^m(\theta) \in \mathbb{R}^{N\times 1}$ for $l = 1, \dots, n_o$, $m = 1, \dots, n_i$. Each SISO, LTI subsystem $J^{lm} \in \mathbb{R}^{N\times N}$ for finite signal length $N \in \mathbb{Z}^+$ is denoted in lifted form as

$$J^{lm} = \begin{bmatrix} h_0^{lm} & 0 & \dots & 0 \\ h_1^{lm} & h_0^{lm} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h_{N-1}^{lm} & h_{N-2}^{lm} & \dots & h_0^{lm} \end{bmatrix} \tag{3}$$

with h_i^{lm}, $i = 0, 1, \dots, N-1$ the Markov parameters of subsystem J^{lm}. Throughout the paper, it is assumed that no model of J is available. Instead, signals f and r can be applied experimentally to the system to measure e.

Input signal $f(\theta)$ is parameterized as $f = \psi(y_d)^\top\theta$ with $\theta \in \mathbb{R}^{N_\theta\times 1}$, $\psi(y_d) \in \mathbb{R}^{N_\theta\times Nn_i}$ and $\text{rank}(\psi(y_d)) = N_\theta$. Here, y_d is a known reference to which disturbance r typically relates, as illustrated in Section 5.3 for a closed-loop system. The parameterization $f = \psi(y_d)^\top\theta$ encompasses both standard norm-optimal ILC, by taking $\psi = I$, and ILC with basis functions. While standard norm-optimal ILC can attenuate a specific repeating disturbance completely, ILC with basis functions can achieve good performance for disturbances caused by varying references by choosing $\psi(y_d)$ as a function of the reference y_d. Further details regarding the choice of basis functions for MIMO systems are given in Section 5.

The aim of this paper is to develop an efficient data-driven approach to minimize (1). To this end, randomized experiments are used to generate an unbiased estimate of the gradient $\frac{\partial J}{\partial\theta}$. Since the estimate is unbiased, it can be employed in a stochastic conjugate gradient-based optimization scheme. The approach is introduced in the following section.

3 | Stochastic Conjugate Gradient Descent for MIMO ILC

In this section, ILC using stochastic conjugate gradient descent is introduced. Since criterion $\mathcal{J}(\theta)$ in (1) is quadratic and strictly convex, provided that J is non-singular, N_θ is sufficiently small or $W_f \succ 0$, gradient-based optimization is suitable. In particular, unbiased gradient estimates are used that follow from a single experiment on the unknown system. A conjugated gradient direction is considered to increase the convergence speed compared to stochastic gradient descent.

The aim is to find the optimal parameters $\theta^* = \arg\min_\theta \mathcal{J}(\theta)$ with $\mathcal{J}(\theta)$ according to (1), in which $f(\theta) = \psi(y_d)^\top\theta$ and

$$e = r - Jf(\theta) = r - J\psi(y_d)^\top\theta \tag{4}$$

according to (2). Using (4), the gradient of (1) is given by

$$g(\theta) = -2\psi(y_d)J^\top W_e e + 2\psi(y_d)W_f\psi(y_d)^\top\theta \tag{5}$$

The transpose $J^\top$ of the system J in (5) is unknown in this setting. However, an unbiased estimate of $g(\theta_j)$ at iteration j, denoted by $\hat{g}_j$, can be generated through a single experiment, regardless of the size of the MIMO system. This gradient estimate is used to determine a search direction p_j, which is then used to update the parameters iteratively according to

$$\theta_{j+1} = \theta_j + \varepsilon_j p_j \tag{6}$$

Here, ε_j denotes the optimal step size in search direction p_j. In the remainder of this section, it is first shown how $\hat{g}_j$ is obtained from a single experiment on J. Then, experiments to obtain conjugate search directions p_j and optimal step sizes ε_j are introduced, and the convergence is analyzed.

3.1 | Unbiased Gradient Estimates From a Single Experiment

The adjoint operator $J^\top$ of J is defined as the operator that satisfies $\langle f, Jg\rangle = \langle J^\top f, g\rangle\ \forall f, g \in \mathbb{R}^{N\times 1}$, where $\langle f, g\rangle$ denotes the inner product of two signals $f, g \in \mathbb{R}^{N\times 1}$. For a SISO, causal LTI system, matrix J^{lm} in (3) has a lower-triangular Toeplitz structure, and the adjoint $J^\top$ relates to J through a time reversal by noting that $\left(J^{lm}\right)^\top = \mathcal{T}J^{lm}\mathcal{T}$. The involutory permutation matrix

$$\mathcal{T} = \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} \in \mathbb{R}^{N\times N}$$

has the interpretation of a time-reversal operator. This well-known property is used in, for example, ILC [10] and L2-gain estimation [11]. For SISO systems, this enables direct experiments on $J^\top$ through one experiment on J with two time reversals. This does not hold for non-symmetric MIMO systems, for which the adjoint is given by [9]

$$J^\top = \begin{bmatrix} (J^{11})^\top & \dots & (J^{n_o 1})^\top \\ \vdots & & \vdots \\ (J^{1n_i})^\top & \dots & (J^{n_o n_i})^\top \end{bmatrix} = \underbrace{\begin{bmatrix} \mathcal{T} & & 0 \\ & \ddots & \\ 0 & & \mathcal{T} \end{bmatrix}}_{\mathcal{T}^{n_i}} \underbrace{\begin{bmatrix} J^{11} & \dots & J^{n_o 1} \\ \vdots & & \vdots \\ J^{1n_i} & \dots & J^{n_o n_i} \end{bmatrix}}_{\tilde{J}} \underbrace{\begin{bmatrix} \mathcal{T} & & 0 \\ & \ddots & \\ 0 & & \mathcal{T} \end{bmatrix}}_{\mathcal{T}^{n_o}} \tag{7}$$

For non-symmetric MIMO systems, $\tilde{J} \neq J$ and the term $J^\top W_e e$ in (5) cannot be determined from a single experiment on J. Instead, finding the exact gradient requires $n_i n_o$ experiments as shown in Section 4.4. However, an unbiased gradient estimate follows from a single experiment according to the following theorem. This auxiliary result extends the earlier result [15, theorem 1] with a projection on $\psi(y_d)$, to enable both standard ILC and parameterized feedforward.

Theorem 1. *An unbiased estimate $\hat{g}_j$ of (5) is given by*

$$\hat{g}_j = 2\psi(y_d)\left(W_f\psi(y_d)^\top\theta - \mathcal{T}^{n_i}A_j J A_j \mathcal{T}^{n_o} W_e e_j\right) \tag{8}$$

where the matrix $A_j \in \mathbb{R}^{(Nn_i)\times(Nn_o)}$ *is given by*

$$A_j = \begin{bmatrix} a_j^{11} & \dots & a_j^{1n_o} \\ \vdots & \ddots & \vdots \\ a_j^{n_i 1} & \dots & a_j^{n_i n_o} \end{bmatrix} \otimes I_N \tag{9}$$

with identity matrix $I_N \in \mathbb{R}^{N\times N}$. *The entries* $a_j^{lm} \in \{-1, 1\}$ *are samples from a symmetric Bernoulli* ± 1 *distribution with probability* $P(a_j^{lm} = 1) = P(a_j^{lm} = -1) = \frac{1}{2}$.

Proof. Since the term $\psi(y_d)W_f\psi(y_d)^\top\theta$ is known, only the experimental term $\psi(y_d)\mathcal{T}^{n_i}A_jJA_j\mathcal{T}^{n_o}W_e e_j$ is considered. Denote $\tilde{e}_j = W_e e_j$. It holds that

$$\mathcal{T}^{n_i}A_jJA_j\mathcal{T}^{n_o}\tilde{e}_j = \begin{bmatrix} \sum_{k=1}^{n_o}\left(\sum_{l=1}^{n_o} a_j^{1l}\sum_{m=1}^{n_i}(J^{lm})^\top a_j^{mk}\right)\tilde{e}_j^k \\ \vdots \\ \sum_{k=1}^{n_o}\left(\sum_{l=1}^{n_o} a_j^{n_i l}\sum_{m=1}^{n_i}(J^{lm})^\top a_j^{mk}\right)\tilde{e}_j^k \end{bmatrix} \tag{10}$$

Let $a_j^{\alpha\beta}$ and $a_j^{\gamma\delta}$, with $\alpha,\ \gamma = 1, 2, \dots, n_i$ and $\beta,\ \delta = 1, 2, \dots, n_o$, denote two entries of A_j. Since $a_j^{\alpha\beta}a_j^{\gamma\delta} = 1$ if $\alpha = \gamma, \beta = \delta$ and $\mathbb{E}\{a_j^{\alpha\beta}a_j^{\gamma\delta}\} = 0$ otherwise, (10) is equal to $\tilde{y}_j + \eta_j$, with η_j all terms for which $\alpha \neq \gamma$ or $\beta \neq \delta$ and

$$\tilde{y}_j = \begin{bmatrix} \sum_{k=1}^{n_o}(J^{k1})^\top\tilde{e}_j^k \\ \vdots \\ \sum_{k=1}^{n_o}(J^{kn_i})^\top\tilde{e}_j^k \end{bmatrix} \tag{11}$$

Note that $\tilde{y}_j = J^\top W_e e_j$ in (5). Since $\mathbb{E}\{a_j^{\alpha\beta}a_j^{\gamma\delta}\} = 0$ if $\alpha \neq \gamma$ or $\beta \neq \delta$, it holds that $\mathbb{E}\{\eta_j\} = 0$ and therefore

$$\mathbb{E}\{\mathcal{T}^{n_i}A_jJA_j\mathcal{T}^{n_o}W_e e_j\} = \tilde{y}_j = J^\top W_e e_j \tag{12}$$

By linearity of the expected value operator, it follows that

$$\mathbb{E}\{\hat{g}_j\} = 2\psi(y_d)\big(W_f\psi(y_d)^\top\theta - J^\top W_e e_j\big) = g_j \tag{13}$$

which concludes the proof. □

3.2 | Conjugate Search Directions

Next, the efficient unbiased gradient estimates are used in a stochastic conjugate gradient descent algorithm. The minimum of (1) can be found by setting $g(\theta) = 0$. Therefore, minimizing (1) is equivalent to solving a problem of the form $Ax = b$, which follows from rewriting (5) to

$$\underbrace{\psi(y_d)\big(J^\top W_e J + W_f\big)\psi(y_d)^\top}_{A_\mathcal{J}}\theta = \underbrace{\psi(y_d)J^\top W_e r}_{b_\mathcal{J}} \tag{14}$$

In the following the notations $A_\mathcal{J}$ and $b_\mathcal{J}$ are used for brevity. Suitable choices of W_f or N_θ ensure that $A_\mathcal{J} \succ 0$, and therefore an exact conjugate gradient step is at least as good as an exact gradient descent step [26, theorem 11.3.3]. Conjugate gradient descent minimizes a criterion along a sequence of conjugated gradient directions to achieve fast convergence. Conjugate vectors are defined as follows.

Definition 1. Two vectors x and y are A-conjugate if $x^\top Ay = 0$.

For deterministic gradients, (1) is minimized through conjugate search directions by taking $p_1 = g_1$ as initial search direction and choosing all subsequent gradient-based search directions such that $i \neq j \ \Rightarrow\ p_i^\top A_\mathcal{J} p_j = 0$ for $1 \leq i < j \leq N_\theta$, that is, p_i and p_j are $A_\mathcal{J}$-conjugate. This generates a sequence of Krylov subspaces [26, sect. 11.3], given by

$$S_j = \operatorname{span}\big\{g_1, A_\mathcal{J}g_1, (A_\mathcal{J})^2 g_1, \dots, (A_\mathcal{J})^{j-1}g_1\big\} \tag{15}$$

The conjugate search directions enable taking θ_{j+1} as the minimizer in a Krylov subspace $\tilde{S}_j$ that includes both θ_j and g_j, in contrast to standard gradient descent which uses a one-dimensional search in the direction of g_j. Each iteration, an exact line search (Section 3.3) is used to ensure that

$$\theta_j = \min_{\theta\in\tilde{S}_j}\mathcal{J}(\theta), \text{ with} \tag{16}$$

$$\tilde{S}_j = \operatorname{span}\big\{\theta_1, g_1, A_\mathcal{J}g_1, \dots, (A_\mathcal{J})^{j-1}g_1\big\} \tag{17}$$

Instead of a deterministic gradient, an unbiased gradient estimate $\hat{g}_j$ is available. Therefore, $p_1 = \hat{g}_1$ is chosen as initial search direction. Subsequent search directions are given by

$$p_{j+1} = \hat{g}_{j+1} + \tau_j p_j \tag{18}$$

where the scalar τ_j ensures that p_{j+1} and p_j are $A_\mathcal{J}$-conjugate. Since $\hat{g}_j \neq g_j$ in general, standard conjugate gradient expressions cannot be applied, see Section 4. For the stochastic conjugate gradient approach using estimate $\hat{g}_j$, the expression for τ_j is given in the following theorem.

Theorem 2. *The search directions* p_{j+1} *and* p_j *are* $A_\mathcal{J}$*-conjugate if*

$$\tau_j = -\frac{(J\psi(y_d)^\top p_j)^\top W_e(J\psi(y_d)^\top\hat{g}_{j+1}) + p_j^\top\psi(y_d)W_f\psi(y_d)^\top\hat{g}_{j+1}}{(J\psi(y_d)^\top p_j)^\top W_e(J\psi(y_d)^\top p_j) + (p_j^\top\psi(y_d)W_f\psi(y_d))^\top p_j} \tag{19}$$

Proof. If p_{j+1} and p_j are $A_\mathcal{J}$-conjugate, then

$$p_j^\top A_\mathcal{J} p_{j+1} = 0 \tag{20}$$

Substituting (18) in (20) gives

$$p_j^\top A_\mathcal{J} p_{j+1} = p_j^\top A_\mathcal{J}\hat{g}_{j+1} + \tau_j p_j^\top A_\mathcal{J} p_j = 0 \tag{21}$$

$$\tau_j = -\frac{p_j^\top A_\mathcal{J}\hat{g}_{j+1}}{p_j^\top A_\mathcal{J} p_j} \tag{22}$$

Substituting $A_\mathcal{J}$ in (22) and rearranging gives

$$\tau_j = -\frac{(J\psi(y_d)^\top p_j)^\top W_e(J\psi(y_d)^\top\hat{g}_{j+1}) + p_j^\top\psi(y_d)W_f\psi(y_d)^\top\hat{g}_{j+1}}{(J\psi(y_d)^\top p_j)^\top W_e(J\psi(y_d)^\top p_j) + (p_j^\top\psi(y_d)W_f\psi(y_d))^\top p_j} \tag{23}$$

which concludes the proof. □

Although J in (19) is unknown, $J\psi(y_d)^\top p_j$ and $J\psi(y_d)^\top \hat{g}_{j+1}$ can be evaluated through system experiments, leading to τ_j.

3.3 | Unbiased Optimal Step Sizes

Suitable step sizes are essential for the performance and stability of (stochastic) gradient-based optimization algorithms [27]. Here, the optimal step size $\varepsilon_j = \arg\min_\varepsilon \mathcal{J}(\theta_{j+1})$ for parameter update (6) follows from an exact line search, because the minimum of criterion (1) along a search direction can be computed exactly as shown in the following theorem. An exact line search leads to faster convergence, compared to inexact line searches that typically involve evaluating multiple candidate values for ε_j [28, chap. 3].

Theorem 3. *The optimal step size $\varepsilon_j = \arg\min_\varepsilon \mathcal{J}(\theta_{j+1})$ for a given search direction p_j is given by*

$$\varepsilon_j = \frac{e_j^\top W_e J\psi(y_d)^\top p_j - \theta_j^\top \psi(y_d) W_f \psi(y_d)^\top p_j}{(J\psi(y_d)^\top p_j)^\top W_e (J\psi(y_d)^\top p_j) + p_j^\top \psi(y_d) W_f \psi(y_d)^\top p_j} \tag{24}$$

Proof. Substitution of (6) in criterion $\mathcal{J}(\theta_{j+1})$ gives

$$\varepsilon_j = \arg\min_\varepsilon \|r - J\psi(y_d)^\top(\theta_j + \varepsilon p_j)\|^2_{W_e} + \|\psi(y_d)^\top(\theta_j + \varepsilon p_j)\|^2_{W_f} \tag{25}$$

Taking the derivative to ε gives

$$\begin{aligned}\frac{\partial}{\partial \varepsilon}\Big(\|r - J\psi(y_d)^\top(\theta_j + \varepsilon p_j)\|^2_{W_e} + \|\psi(y_d)^\top(\theta_j + \varepsilon p_j)\|^2_{W_f}\Big) \\ = \theta_j^\top A_\mathcal{J} p_j + p_j^\top A_\mathcal{J} \theta_j + 2\varepsilon p_j^\top A_\mathcal{J} p_j - p_j^\top B_\mathcal{J} - B_\mathcal{J}^\top p_j\end{aligned} \tag{26}$$

For the minimum it holds that $\frac{\partial \mathcal{J}(\theta_{j+1})}{\partial \varepsilon} = 0$, and therefore

$$\varepsilon_j = \frac{p_j^\top (A_\mathcal{J}\theta_j - B_\mathcal{J}) + (\theta_j^\top A_\mathcal{J} - B_\mathcal{J}^\top) p_j}{2 p_j^\top A_\mathcal{J} p_j} \tag{27}$$

Substitution of $A_\mathcal{J}$, $B_\mathcal{J}$, (4) and rewriting leads to

$$\varepsilon_j = \frac{e_j^\top W_e J\psi(y_d)^\top p_j - \theta_j^\top \psi(y_d) W_f \psi(y_d)^\top p_j}{(J\psi(y_d)^\top p_j)^\top W_e (J\psi(y_d)^\top p_j) + p_j^\top \psi(y_d) W_f \psi(y_d)^\top p_j} \tag{28}$$

which concludes the proof. □

Step size ε_j is determined by evaluating the term $J\psi(y_d)^\top p_j$ through an experiment on the system. Regardless of the size of the MIMO system, only three experiments are required per iteration, in addition to the experiment to determine the error e_j: a measurement of $\hat{g}(\theta_j)$ according to (8), a measurement of $J\psi(y_d)^\top \hat{g}_j$ to determine τ_{j-1} in (19) which leads to p_j (see 18), and a measurement of $J\psi(y_d)^\top p_j$ to determine ε_j in (24). Note that at iteration j, the term $J\psi(y_d)^\top p_{j-1}$ which is needed to determine τ_{j-1} is already known from the earlier experiment to determine ε_{j-1}.

3.4 | Convergence of Stochastic Conjugate Gradient Descent

The following lemma holds for the convergence of the presented stochastic conjugate gradient ILC approach.

Lemma 1. *Stochastic conjugate gradient ILC with update (6), optimal step sizes according to Theorem 3, and search direction (18) with τ_j according to Theorem 2 and unbiased gradient estimates according to Theorem 1, converges to the minimum θ^* of (1) in at most N_θ steps.*

Proof. Criterion (1) is quadratic. The directions p_j, $j = 1, 2, \ldots, N_\theta$ are $A_\mathcal{J}$-conjugate, and since $A_\mathcal{J}$ is positive-definite the N_θ directions are linearly independent. Step size (24) ensures that θ_j is optimal in $\tilde{S}_j$. After N_θ iterations, $\tilde{S}_j$ spans the whole space $\mathbb{R}^{N_\theta}$ such that $\theta_{N_\theta} = \theta^*$. This is a well-known result for conjugate direction methods and quadratic criteria, see, for example, [26, sect. 11]. □

In practice, good performance is often achieved in fewer than N_θ iterations, as shown in the results in Sections 6 and 7. If the evaluations of system J are noisy, which is often the case in practice, Lemma 1 cannot guarantee convergence. The influence of noise on the gradient estimates is limited, as these are already assumed to be stochastic and remain unbiased. However, the expressions for the search direction and the step size, that were previously assumed to be deterministic, become stochastic when evaluations of J are noisy. As a result, it is not possible to maintain conjugacy of search directions over multiple iterations. In this case, the search direction can be reset to the gradient estimate after a number of iterations, which is common practice for Krylov subspace methods, see, for example, [29], yet which does not guarantee convergence. It is also possible to revert to stochastic gradient descent by taking $\tau_j = 0$, such that $p_j = \hat{g}_j$, see Section 4.3. For decreasing step sizes, this leads to almost sure convergence by [15, theorem 2]. In Section 6, the presented approach is simulated for noisy system evaluations.

4 | Implementation and Relation to Other Methods

In this section, several implementation aspects are considered. First, an overview of the approach is given. Secondly, scaling for the gradient experiments is investigated. Lastly, the method is compared to alternative methods: stochastic gradient descent, conjugate gradient descent with exact gradients, and a quasi-Newton method presented by [9].

4.1 | Overview of the Approach

The unbiased gradient estimate in Section 3 is combined with the conjugate search direction in (18) and the optimal step size in (24) to obtain parameter update θ_{j+1} according to (6). In addition to the experiment to determine the error $e_j(\theta_j)$ of iteration j, only three experiments are required to determine the gradient estimate, search direction, and optimal step size, regardless of the size of the MIMO system. The complete approach is summarized in Algorithm 1.

ALGORITHM 1 | Efficient MIMO iterative learning control

1: **for** $j = 1 : n_{\text{iteration}}$
2: Apply input $f_j = \psi(y_d)^\top \theta_j$ to measure $e_j(\theta_j) = r - J\psi(y_d)^\top \theta_j$.
3: Find unbiased estimate $\hat{g}(\theta_j)$ using one experiment according to (8) in Section 3.
4: **if** j=1
5: Set $p_1 = \hat{g}_1$.
6: **else**
7: Apply input $\hat{g}_j$ to measure $J\psi(y_d)^\top \hat{g}_j$, and use $J\psi(y_d)^\top p_{j-1}$ to determine τ_{j-1} in (19).
8: Take direction $p_j = \hat{g}_j + \tau_{j-1} p_{j-1}$.
9: **end**
10: Apply input p_j to measure $J\psi(y_d)^\top p_j$ to determine step size ε_j in (24).
11: Update $\theta_{j+1} = \theta_j - \varepsilon_j p_j$.
12: **end**

4.2 | Scaling Gradient Experiments

The approach in Algorithm 1 involves three dedicated experiments: the gradient experiment with input $A_J \mathcal{T}^{n_o} W_e e_j$, the step size experiment with input $\psi(y_d)^\top p_j$ and the search direction experiment with input $\psi(y_d)^\top \hat{g}_j$. It may be necessary to scale these inputs before applying them to the system, for example when the signals are too large, such that implementing them could damage the system, or when their magnitude is too small to overcome the stick friction of a motion system. Therefore, a scaling factor $\alpha \in \mathbb{R}$ is introduced, that is, for example, implemented as

$$J\psi(y_d)^\top \hat{g}(\theta_j) = \frac{1}{\alpha} J\alpha\psi(y_d)^\top \hat{g}(\theta_j) \tag{29}$$

The implementation for the two other types of experiments is similar. The scaling factor α is chosen separately for each experiment, and it is assumed to be chosen such that the system is outside of the stick regime during the experiment, such that the system behaves linearly.

4.3 | Recovering Stochastic Gradient Descent

Stochastic gradient descent ILC differs from the conjugate gradient ILC algorithm in that it uses the gradient estimate as search direction, rather than a conjugate search direction derived from this gradient estimate. Stochastic gradient descent ILC can be recovered from the conjugate gradient ILC algorithm by taking $\tau_j = 0$, such that $p_j = \hat{g}_j$ and the parameter update is given by

$$\theta_{j+1} = \theta_j - \varepsilon_j \hat{g}_j \tag{30}$$

The optimal step size follows from substituting $p_j = \hat{g}_j$ in (24). For exact conjugate gradient (CG) and gradient descent (GD), it holds that a CG step is at least as good as a GD step since $\psi(y_d)(J^\top W_e J + W_f)\psi(y_d)^\top \succ 0$ [26, theorem 11.3.3, 30]. In [15], the gradient estimates $\hat{g}_j$ are used in a Robbins–Monro type stochastic gradient descent algorithm [31] and ε_j is chosen such that criteria of almost sure convergence for a Robbins–Monro algorithm are met. As shown in Section 6, this results in slow convergence compared to the proposed stochastic conjugate gradient approach.

4.4 | Conjugate Gradient Descent With Exact Gradients

The approach developed in this article uses experimentally efficient approximate gradients that are obtained through a single experiment. However, the expressions for the conjugated search directions and optimal step sizes can also be used with exact gradient expressions, which are experimentally expensive. Exact gradient expressions can be generated through $n_i \times n_o$ experiments according to [9]

$$g_j = -2\psi(y_d)\mathcal{T}^{n_i}\left(\sum_{l=1}^{n_i}\sum_{m=1}^{n_o} E^{lm} J E^{lm}\right)\mathcal{T}^{n_o} W_e e_j + 2\psi(y_d) W_f \psi(y_d)^\top \theta \tag{31}$$

where E^{lm} consists of zeros, with a one on the lm^{th} entry. This approach does not scale well for large MIMO systems because generating the gradient requires $n_i \times n_o$ experiments. In addition, obtaining exact gradients requires noise-free system evaluations. In the theoretical case with noise-free evaluations and gradients, the expressions for the search direction and step size can be simplified to the well-known standard expressions for conjugate gradient descent. If $\hat{g}_j = g_j \; \forall \; j$, expression (19) for τ_j reduces to

$$\tau_j = \frac{g_{j+1}^\top g_{j+1}}{g_j^\top g_j} \tag{32}$$

see for example, [26, sect. 11.3] for a derivation. This expression for τ_j does not require any experiments. In addition, expression (24) for ε_j reduces to

$$\varepsilon_j = \frac{-g_j^\top g_j}{p_j^\top A_J p_j} = \frac{-g_j^\top g_j}{(J\psi(y_d)^\top p_j)^\top W_e (J\psi(y_d)^\top p_j) + p_j^\top \psi(y_d) W_f \psi(y_d)^\top p_j} \tag{33}$$

These standard expressions for conjugate gradient descent depend on the assumption that $\hat{g}_j = g_j \; \forall \; j$ and cannot be used in case of stochastic gradients or noisy system evaluations.

4.5 | Quasi-Newton Methods

In [9], deterministic gradients are used in a quasi-Newton adjoint ILC approach. This approach uses the exact gradient, modified by an estimate of the inverse Hessian, as search direction, in contrast to the method in this article that uses conjugated search directions based on gradient estimates. The search direction is chosen as

$$p_j = -B_j g_j \tag{34}$$

where the BFGS inverse Hessian estimate B_j is given by

$$B_j = B_{j-1} - \frac{s_j \gamma_j^\top B_{j-1} + B_{j-1}\gamma_j s_j^\top}{s_j^\top \gamma_j} + \left(1 + \frac{\gamma_j^\top B_{j-1}\gamma_j}{s_j^\top \gamma_j}\right)\frac{s_j s_j^\top}{s_j^\top \gamma_j} \tag{35}$$

with $s_j = \theta_j - \theta_{j-1}$ and $\gamma_j = g_j - g_{j-1}$. The optimal step size, that follows from an exact line search, is given by

$$\varepsilon_j = \frac{g_j^\top B_j g_j}{(J\psi(y_d)^\top B_j g_j)^\top W_e J\psi(y_d)^\top B_j g_j} + \frac{g_j^\top B_j g_j}{g_j^\top B_j^\top \psi(y_d) W_f \psi(y_d)^\top B_j g_j} \tag{36}$$

which is based on the assumption that deterministic gradients are available. However, if only $\hat{g}_j$ is available, (36) does not give the optimal step size and the step size of Theorem 3 with $p_j = -B_j\hat{g}_j$ should be used instead. Yet even in that case, the inverse Hessian estimate B_j is biased, as shown in the following lemma.

Lemma 2. *When γ_j in (35) is replaced by the unbiased estimate $\hat{\gamma}_j = \hat{g}_j - \hat{g}_{j-1}$, the resulting $\hat{B}_j(\hat{\gamma}_j)$ is a biased estimator of $B_j(\gamma_j)$, that is, $\mathbb{E}\{\hat{B}_j\} \neq B_j$. In addition, while $\hat{B}_j\hat{\gamma}_j = s_j$, this secant condition is not satisfied for γ_j in expectation, that is, $\mathbb{E}\{\hat{B}_j\gamma_j\} \neq s_j$.*

Proof. Denote $\hat{\gamma}_j = \gamma_j + \omega_j$. First, to show that $\mathbb{E}\{\hat{B}_j\} \neq B_j$, consider

$$\hat{B}_j = \hat{B}_{j-1} - \frac{s_j\hat{\gamma}_j^\top\hat{B}_{j-1} + \hat{B}_{j-1}\hat{\gamma}_j s_j^\top}{s_j^\top\hat{\gamma}_j} + \left(1 + \frac{\hat{\gamma}_j^\top\hat{B}_{j-1}\hat{\gamma}_j}{s_j^\top\hat{\gamma}_j}\right)\frac{s_j s_j^\top}{s_j^\top\hat{\gamma}_j} \tag{37}$$

Here, $\mathbb{E}\left\{\frac{\hat{\gamma}_j^\top\hat{B}_{j-1}\hat{\gamma}_j}{s_j^\top\hat{\gamma}_j s_j^\top\hat{\gamma}_j}\right\} \neq \frac{\gamma_j^\top\hat{B}_{j-1}\gamma_j}{s_j^\top\gamma_j s_j^\top\gamma_j}$, that is, this scalar term is biased and as a result $\mathbb{E}\{\hat{B}_j\} \neq B_j$. Secondly, rewriting (35), see [32], shows that $\mathbb{E}\{\hat{B}_j\gamma_j\} \neq s_j$. In particular, $\gamma_j = \hat{\gamma}_j - \omega_j$ leads to

$$\begin{aligned}\hat{B}_j\gamma_j &= \left(\left(I - \frac{\hat{\gamma}_j s_j^\top}{s_j^\top\hat{\gamma}_j}\right)^\top\hat{B}_{j-1}\left(I - \frac{\hat{\gamma}_j s_j^\top}{s_j^\top\hat{\gamma}_j}\right) + \frac{s_j s_j^\top}{s_j^\top\hat{\gamma}_j}\right)\gamma_j \\ &= \left(I - \frac{\hat{\gamma}_j s_j^\top}{s_j^\top\hat{\gamma}_j}\right)^\top\hat{B}_{j-1}\left(-\omega_j + \frac{\hat{\gamma}_j s_j^\top\omega_j}{s_j^\top\hat{\gamma}_j}\right) \\ &\quad + s_j - \frac{s_j s_j^\top\omega_j}{s_j^\top\hat{\gamma}_j}\end{aligned} \tag{38}$$

Taking the expected value gives

$$\mathbb{E}\{\hat{B}_j\gamma_j\} = s_j + \mathbb{E}\left\{\left(I - \frac{\hat{\gamma}_j s_j^\top}{s_j^\top\hat{\gamma}_j}\right)^\top\hat{B}_{j-1}\left(-\omega_j + \frac{\hat{\gamma}_j s_j^\top\omega_j}{s_j^\top\hat{\gamma}_j}\right) - \frac{s_j s_j^\top\omega_j}{s_j^\top\hat{\gamma}_j}\right\} \tag{39}$$

where it is clear that the second part of the expression has $\mathbb{E}\{\cdot\} \neq 0$ such that $\mathbb{E}\{\hat{B}_j\gamma_j\} \neq s_j$. □

Lemma 2 illustrates that using the BFGS update (35) with unbiased gradient estimates, either generated through (8) or through (31) with noisy system evaluations, results in biased inverse Hessian estimates that do not meet the secant condition in expectation. As such, (35) is not suitable for acceleration of stochastic gradient descent. This is further illustrated through simulations in Section 6.

5 | Specialization to MIMO Basis Functions

In the previous sections, an efficient approach to MIMO iterative learning control is introduced that optimizes a parameterized input signal of the form

$$f(y_d, \theta) = \psi(y_d)^\top\theta \tag{40}$$

In standard iterative learning control, $\psi(y_d) = I$ such that $f = \theta$ is an input signal that attenuates one specific repeating disturbance completely. However, many applications require varying references that result in varying disturbances. In this case parameterized feedforward control can achieve both high accuracy and flexibility by choosing $\psi(y_d)$ as a set of reference-dependent basis functions, see, for example, [33].

In this section, an example of MIMO mass feedforward is given which is then extended into a general framework for MIMO basis functions, and suitable basis functions for MIMO motion systems are presented. In Section 7, the MIMO basis function feedforward framework is applied to an industrial flatbed printer.

5.1 | Example: MIMO Mass Feedforward

To illustrate a suitable structure for basis function feedforward for MIMO motion control systems with interaction, consider the example of mass feedforward f_m for a 2×2 system with reference $y_d = \begin{bmatrix} y_d^1 & y_d^2 \end{bmatrix}^\top$. Matrix $\psi(y_d)$ contains the second derivative of the reference, such that

$$\begin{bmatrix} f_m^1(y_d, \theta) \\ f_m^2(y_d, \theta) \end{bmatrix} = \underbrace{\begin{bmatrix} \ddot{y}_d^{\,1} & \ddot{y}_d^{\,2} & 0 & 0 \\ 0 & 0 & \ddot{y}_d^{\,1} & \ddot{y}_d^{\,2} \end{bmatrix}}_{\psi(y_d)^\top}\begin{bmatrix}\theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4\end{bmatrix} \tag{41}$$

The four transfer functions in the MIMO system, that is, two diagonal and two off-diagonal interaction terms, are each approximated by a mass line. ILC can be used to find the mass estimates, that is, the parameters θ, that minimize (1).

5.2 | General Structure for MIMO Parameterized Feedforward

A general structure for basis functions for MIMO systems is given by

$$\psi(y_d)^\top = \begin{bmatrix} \psi_1(y_d) & \psi_2(y_d) & \dots & \psi_{n_b}(y_d) & \dots & 0 & 0 & \dots & 0 \\ & & \vdots & & & & \vdots & & \\ 0 & 0 & \dots & 0 & \dots & \psi_1(y_d) & \psi_2(y_d) & \dots & \psi_{n_b}(y_d) \end{bmatrix} \tag{42}$$

with the n_b basis functions $\psi_n(y_d),\ n = 1, 2, \dots, n_b$ given by

$$\psi_n(y_d) = \begin{bmatrix} \psi_n^1(y_d^1) & \psi_n^2(y_d^2) & \dots & \psi_n^{n_o}(y_d^{n_o}) \end{bmatrix} \tag{43}$$

Here $\psi_n^1(y_d^1) \in \mathbb{R}^{N\times 1}$ such that $\psi(y_d)^\top \in \mathbb{R}^{n_i N \times n_b n_o n_i}$. The parameter vector is structured as

$$\theta = \begin{bmatrix} \theta_1 & \dots & \theta_{n_b n_o n_i} \end{bmatrix}^\top \in \mathbb{R}^{n_b n_o n_i \times 1} \tag{44}$$

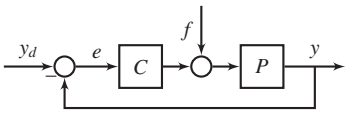

FIGURE 2 | Closed-loop system with parameterized feedforward f. Here $J = P(I + CP)^{-1}$ is the process sensitivity describing the transfer from f to $-e$. Reference y_d leads to disturbance $r = (I + PC)^{-1}y_d$.

Thus, $\psi(y_d)^\top$ consists of a set of n_b basis function matrices, applied to each of the n_i input directions. Each of the basis function matrices $\psi_n(y_d)$ consists of a specific basis function applied to each of the n_o output references. The parameter vector θ contains a separate parameter for each column in each of the $n_i \times n_b$ matrices $\psi_n(y_d)$. This results in n_i feedforward signals of the form

$$f^n(y_d, \theta) = \sum_{k=1}^{n_o} \sum_{l=1}^{n_b} \psi_l^k(y_d^k)\theta_{(n-1)n_o n_b + (l-1)n_o + k} \tag{45}$$

Note that for each input direction, the same set of n_b basis functions is used and only the parameters are different. To take into account couplings in the MIMO system, the input in each direction contains basis functions based on the reference in all output directions. The framework allows for any type of basis functions that is linear in the parameters, including finite impulse response (FIR) bases or, for example, non-causal rational basis functions with fixed poles.

5.3 | Basis Functions

Parameterized feedforward control aims to minimize the error for any reference and is typically implemented as shown in Figure 2. From (2) and Figure 2, it follows that zero tracking error requires

$$f = J^{-1}r = (P(I + CP)^{-1})^{-1}(I + PC)^{-1}y_d = P^{-1}y_d \tag{46}$$

It follows that $\psi(y_d)^\top\theta$ should approximate $P^{-1}y_d$, that is, the basis functions should approximate the inverse plant. Therefore, interpretable basis functions with parameters that approximate system parameters are preferred over generic parametrizations such as radial basis functions or Gaussian processes. For typical motion systems, suitable basis functions are the position reference and its derivatives: velocity, acceleration and snap terms compensate respectively viscous friction, mass dynamics and the compliance of the flexible dynamics. This leads to the following basis functions.

$$\psi(y_d)^\top = \begin{bmatrix} y_{d,\tau} & y'_{d,\tau} & \dots & y^{(4)}_{d,\tau} & \dots & 0 & 0 & \dots & 0 \\ & \vdots & & & & & \vdots & & \\ 0 & 0 & \dots & 0 & \dots & y_{d,\tau} & y'_{d,\tau} & \dots & y^{(4)}_{d,\tau} \end{bmatrix} \tag{47}$$

in which

$$y_{d,\tau} = \begin{bmatrix} y_d^1 & y_d^2 & \dots & y_d^{n_o} \end{bmatrix} \tag{48}$$

This parameterization relates each of the output references to each input direction, modeling both the diagonal and off-diagonal terms of the MIMO system. Therefore, it is suitable for MIMO systems with strong interaction.

6 | Application: Simulation of a Massive MIMO System

In this section, the proposed approach with $\psi = I$ is applied to a random 21×21 MIMO system, illustrated in Figure 3, in simulations. Several approaches are compared for cases with noise-free and noisy system evaluations.

1. Stochastic conjugate gradient ILC (this article) using $\hat{g}$ according to (8), conjugate search directions and optimal step sizes.
2. Stochastic approximation adjoint ILC [15] using $\hat{g}$ according to (8) with a fixed step size.
3. Deterministic adjoint ILC [9] using g according to (31) with a fixed step size.
4. Deterministic conjugate gradient ILC using g according to (31) with conjugated search directions and optimal step sizes according to Section 4.4.
5. (Noisy evaluations) Deterministic conjugate gradient ILC using g according to (31) with conjugate search directions and optimal step sizes according to Section 3.
6. (Noisy evaluations) Stochastic approximation adjoint ILC using $\hat{g}$ according to (8) with optimal step sizes.
7. (Noisy evaluations) BFGS with $\hat{g}$ and g using either the non-optimal step size of (36) or the optimal step size of Theorem 3.

First, it is shown that the proposed approach outperforms existing deterministic and stochastic approaches in case of noise-free system evaluations. Secondly, it is shown that deterministic methods diverge when the system evaluations are noisy, while the proposed stochastic conjugate gradient ILC (SCGILC) still converges. Thirdly, it is shown that BFGS is not suitable for stochastic settings. In all simulations the weights are given by $W_e = I,\ W_f = 0$.

6.1 | Stochastic Versus Deterministic Conjugate Gradient

Stochastic conjugate gradient ILC (SCGILC) is compared to stochastic approximation adjoint ILC [15], to deterministic adjoint ILC [9] and to a deterministic conjugate gradient method (Section 4.4), which employs deterministic gradients generated by (31).

In Figure 4, it is shown that the proposed stochastic conjugate gradient algorithm requires far fewer experiments to reach the same cost than the deterministic conjugate gradient algorithm. The conjugate gradient algorithms outperform both stochastic and deterministic gradient descent algorithms. In addition, the smoothness of stochastic ILC is greatly improved by the line searches of the conjugate gradient method.

6.2 | Noisy System Evaluations

In typical control applications, evaluations of the system J are noisy. As a result, gradient estimates generated through (31) are

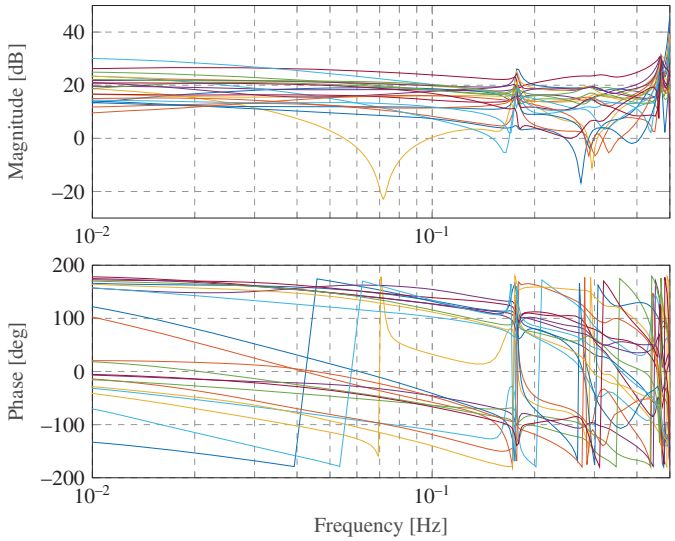

FIGURE 3 | Bode magnitude and phase plots of the last column of 21 subsystems of the random non-symmetric 21×21 MIMO system used for the simulations results.

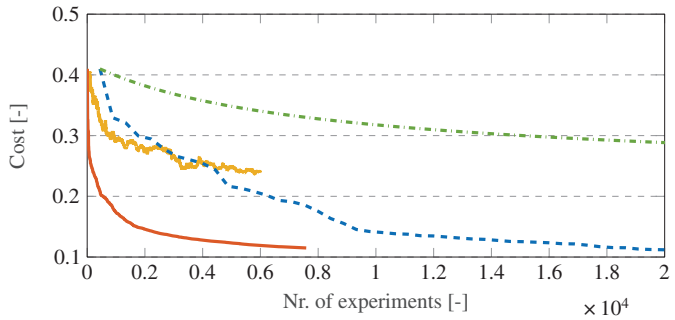

FIGURE 4 | The cost as a function of the number of experiments for a non-symmetric 21×21 system. Four approaches are shown: the proposed stochastic conjugate gradient ILC (SCGILC) method (—), deterministic conjugate gradient ILC (– –), stochastic gradient descent ILC (—) and deterministic gradient descent ILC (-.-). SCGILC (—) requires far fewer experiments than other approaches to reach the same cost.

not deterministic, although the variance of these gradient estimates is typically smaller than that of the unbiased estimates generated by (8). For the case of noisy system evaluations, SCGILC is compared to a conjugate gradient method designed for deterministic gradients (Section 4.4).

In Figure 5, it is shown that deterministic conjugate gradient descent diverges when system evaluations are noisy. The stochastic conjugate gradient ILC algorithm is implemented with gradient estimates generated by respectively (8) and (31). It is shown that while algorithm converges for both gradient estimates, using gradient estimates obtained from a single experiment results in much faster convergence in terms of the required number of experiments. The SCGILC method is also shown to converge slightly faster than stochastic gradient descent with optimal step sizes, which does not require the additional experiment to determine a conjugate direction.

6.3 | BFGS in a Non-Deterministic Setting

To illustrate the effect of non-optimal step sizes and biased inverse Hessian estimates in BFGS, which follow from applying (35) in case of noisy system evaluations, four different approaches are simulated. In Figure 6, it is shown that applying the non-optimal step size of (36) leads to divergence in a stochastic setting, even when the full gradient expressions are used. Using the optimal step size of Theorem 3 in combination with a biased estimate of the inverse Hessian also leads to a diverging or non-decreasing cost.

7 | Experimental Implementation on an Industrial Flatbed Printer

The presented SCGILC approach with the basis functions from Section 5 is applied to a 3×3 MIMO industrial Arizona flatbed printer, shown in Figure 7. The system has inputs $F = \begin{bmatrix} F_y & F_x & F_\varphi \end{bmatrix}$ and the outputs are the translation of the carriage y, and the translation x and rotation φ of the gantry. The reference consists of a translation in x- and y-direction. The weights are $W_e = I$ and $W_f = 10^{-7} I$ and the basis functions are the reference, velocity and acceleration in each direction. In Figure 8, it is shown that SCGILC converges to a low cost in a small number of iterations, with each iteration requiring four

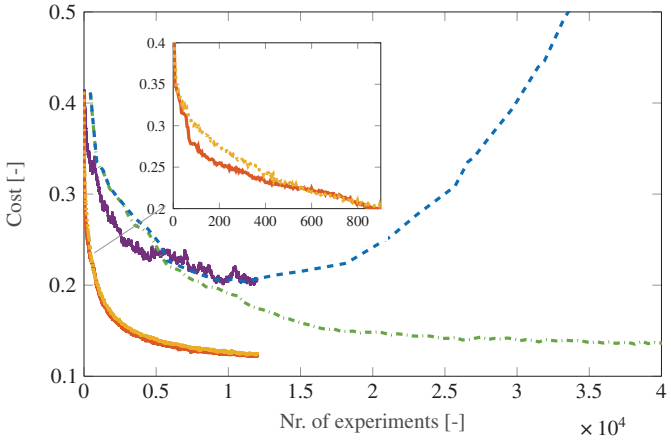

FIGURE 5 | The cost as a function of the number of experiments for a noisy non-symmetric 21 × 21 system. Stochastic conjugate gradient ILC (SCGILC) based on (8) (—) requires far fewer experiments than the implementation using (31) (-·-). Implementing deterministic conjugate gradient (- -) in case of noisy system evaluations and gradient estimates leads to a diverging cost. SCGILC (—) is slightly faster than stochastic gradient descent with optimal step sizes (·····), especially in the first iterations (zoom plot), and much faster and smoother than stochastic gradient descent with a fixed step size (—).

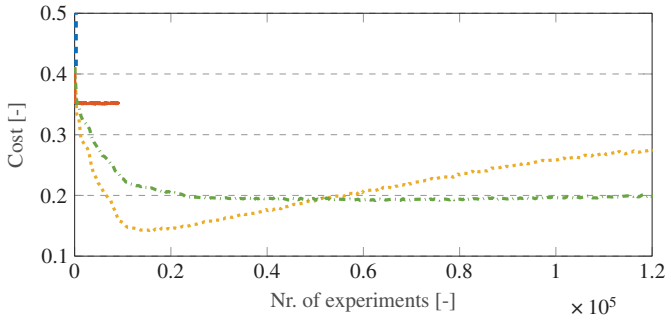

FIGURE 6 | The cost as a function of the number of experiments for a noisy non-symmetric 21 × 21 system with different BFGS implementations. For full gradients, BFGS leads to a diverging cost for both non-optimal (·····) and optimal (-·-) stepsizes. For stochastic gradients, BFGS leads to a diverging cost for non-optimal step sizes (- -) and a non-decreasing cost for optimal step sizes (—).

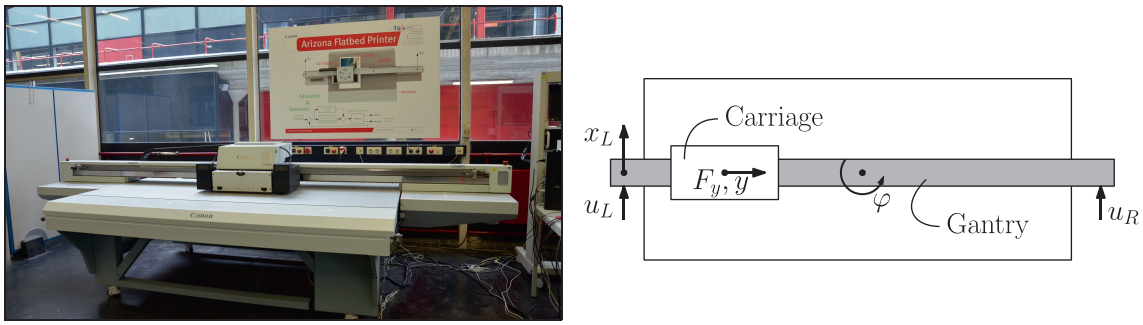

FIGURE 7 | Photograph (left) and schematic overview (right) of the industrial Arizona flatbed printer that is used for the experimental results in Section 7.

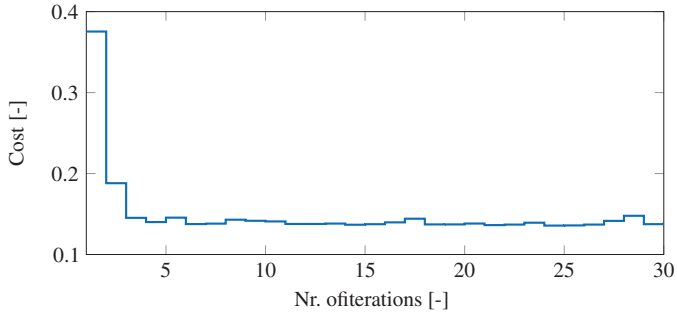

FIGURE 8 | Cost over iterations for stochastic conjugate gradient descent parallel ILC with basis functions (position, velocity and acceleration). Experimental results show convergence to small errors after only three iterations (12 experiments).

experiments to determine the error, gradient estimate, conjugate search direction and optimal step size.

8 | Conclusion

In this article, an efficient method for MIMO iterative learning control is developed that achieves fast convergence through random learning. The method employs a stochastic conjugate gradient descent algorithm based on unbiased gradient estimates that are generated through system experiments. Each iteration requires only three dedicated experiments to determine an unbiased gradient estimate, a conjugate search direction and an optimal step size. The proposed method converges faster than existing methods that use only unbiased gradient estimates or deterministic gradients that require $n_i n_o$ experiments per iteration, and it retains this fast convergence property in case of noisy system evaluations, as illustrated in simulations. The method is applied to the practical case of automated feedforward tuning for MIMO systems. Suitable basis functions for MIMO motion systems with interaction are proposed, and the method is applied to an industrial flatbed printer, leading to fast convergence and small errors.

Author Contributions

Conceptualization, Writing – review & editing, Software, Validation, Investigation, Writing – original draft: Leontine Aarnoudse. Conceptualization, Writing – review & editing, Supervision: Tom Oomen.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A Survey of Iterative Learning Control," *IEEE Control Systems* 26, no. 3 (2006): 96–114, https://doi.org/10.1109/MCS.2006.1636313.

2. S. Gunnarsson and M. Norrlöf, "On the Design of ILC Algorithms Using Optimization," *Automatica* 37, no. 12 (2001): 2011–2016, https://doi.org/10.1016/S0005-1098(01)00154-6.

3. D. H. Owens, J. J. Hatonen, and S. Daley, "Robust Monotone Gradient-Based Discrete-Time Iterative Learning Control," *International Journal of Robust and Nonlinear Control* 19 (2009): 634–661, https://doi.org/10.1002/rnc.1338.

4. M. Gevers, "A Decade of Progress in Iterative Process Control Design: From Theory to Practice," *Journal of Process Control* 12, no. 4 (2002): 519–531, https://doi.org/10.1016/S0959-1524(01)00018-X.

5. H. Hjalmarsson, "From Experiment Design to Closed-Loop Control," *Automatica* 41, no. 3 (2005): 393–438, https://doi.org/10.1016/j.automatica.2004.11.021.

6. F. Dorfler, J. Coulson, and I. Markovsky, "Bridging Direct and Indirect Data-Driven Control Formulations via Regularizations and Relaxations," *IEEE Transactions on Automatic Control* 68, no. 2 (2023): 883–897, https://doi.org/10.1109/TAC.2022.3148374.

7. M. Butcher, A. Karimi, and R. Longchamp, "A Statistical Analysis of Certain Iterative Learning Control Algorithms," *International Journal of Control* 81, no. 1 (2008): 156–166, https://doi.org/10.1080/00207170701484851.

8. T. D. Son, G. Pipeleers, and J. Swevers, "Robust Monotonic Convergent Iterative Learning Control," *IEEE Transactions on Automatic Control* 61, no. 4 (2016): 1063–1068, https://doi.org/10.1109/TAC.2015.2457785.

9. 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* 28, no. 12 (2018): 3728–3751, https://doi.org/10.1002/rnc.3611.

10. C. Freeman, P. L. Lewin, and E. Rogers, "Further Results on the Experimental Evaluation of Iterative Learning Control Algorithms for Non-Minimum Phase Plants," *International Journal of Control* 80, no. 4 (2007): 569–582, https://doi.org/10.1080/00207170601136726.

11. B. Wahlberg, M. B. Syberg, and H. Hjalmarsson, "Non-Parametric Methods for L2-Gain Estimation Using Iterative Experiments," *Automatica* 46, no. 8 (2010): 1376–1381, https://doi.org/10.1016/j.automatica.2010.05.012.

12. S. Z. Khong, D. Nešić, and M. Krstić, "Iterative Learning Control Based on Extremum Seeking," *Automatica* 66 (2016): 238–245, https://doi.org/10.1016/j.automatica.2015.12.019.

13. H. Hjalmarsson, "Efficient Tuning of Linear Multivariable Controllers Using Iterative Feedback Tuning," *International Journal of Adaptive Control and Signal Processing* 13 (1999): 553–572, https://doi.org/10.1002/(SICI)1099-1115(199911)13:7<553::AID-ACS572>3.0.CO;2-B.

14. T. Oomen, R. van der Mass, C. R. Rojas, and H. Hjalmarsson, "Iterative Data-Driven H-Infinity Norm Estimation of Multivariable Systems With Application to Robust Active Vibration Isolation," *IEEE Transactions on Control Systems Technology* 22, no. 6 (2014): 2247–2260, https://doi.org/10.1109/TCST.2014.2303047.

15. L. Aarnoudse and T. Oomen, "Model-Free Learning for Massive MIMO Systems: Stochastic Approximation Adjoint Iterative Learning Control," *IEEE Control Systems Letters* 5, no. 6 (2021): 1946–1951, https://doi.org/10.1109/LCSYS.2020.3046169.

16. J. C. Spall, "Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation," *IEEE Transactions on Automatic Control* 37, no. 3 (1992): 332–341, https://doi.org/10.1109/9.119632.

17. N. N. Schraudolph and G. Simon, "A Stochastic Quasi-Newton Method for Online Convex Optimization," in *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics* (San Juan, Puerto Rico: PMLR, 2007), 436–443.

18. A. Mokhtari and A. Ribeiro, "Stochastic Quasi-Newton Methods," *Proceedings of the IEEE* 108 (2020): 11, https://doi.org/10.1109/JPROC.2020.3023660.

19. A. G. Wills and T. B. Schön, "Stochastic Quasi-Newton With Line-Search Regularisation," *Automatica* 127 (2021): 109503, https://doi.org/10.1016/j.automatica.2021.109503.

20. K. P. Murphy, *Machine Learning: A Probabilistic Perspective* (Cambridge, MA: MIT Press, 2012).

21. X. B. Jin, X. Y. Zhang, K. Huang, and G. G. Geng, "Stochastic Conjugate Gradient Algorithm With Variance Reduction," *IEEE Transactions on Neural Networks and Learning Systems* 30, no. 5 (2019): 1360–1369, https://doi.org/10.1109/TNNLS.2018.2868835.

22. N. N. Schraudolph, T. Graepel, "Conjugate Directions for Stochastic Gradient Descent," in *International Conference on Artificial Neural Networks—ICANN*, (Berlin, Heidelberg, Madrid, Spain: Springer, 2002), 1351–1356.

23. B. A. Pearlmutter, "Fast Exact Multiplication by the Hessian," *Neural Computation* 6, no. 1 (1994): 147–160, https://doi.org/10.1162/neco.1994.6.1.147.

24. L. Aarnoudse and T. Oomen, "Conjugate Gradient MIMO Iterative Learning Control Using Data-Driven Stochastic Gradients," in *60th IEEE Conference Decision Control* (Austin, Texas, USA: IEEE, 2021), 3749–3754.

25. L. Aarnoudse and T. Oomen, "Automated MIMO Motion Feedforward Control: Efficient Learning Through Data-Driven Gradients via Adjoint Experiments and Stochastic Approximation," *IFAC-PapersOnLine* 55, no. 37 (2022): 125–130, https://doi.org/10.1016/j.ifacol.2022.11.172.

26. G. H. Golub and C. F. Van Loan, *Matrix Computations* (Baltimore, London: John Hopkins University Press, 2013).

27. M. Mahsereci and P. Hennig, "Probabilistic Line Searches for Stochastic Optimization," *Journal of Machine Learning Research* 18 (2017): 1–59.

28. J. Nocedal and S. J. Wright, *Numerical Optimization* (New York, NY: Springer, 2006).

29. A. Baker, E. Jessup, and T. Kolev, "A Simple Strategy for Varying the Restart Parameter in GMRES(m)," *Journal of Computational and Applied Mathematics* 230, no. 2 (2009): 751–761, https://doi.org/10.1016/j.cam.2009.01.009.

30. J. C. Allwright, "Conjugate Gradient Versus Steepest Descent," *Journal of Optimization Theory and Applications* 20, no. 1 (1976): 129–134, https://doi.org/10.1007/BF00933351.

31. H. Robbins and S. Monro, "A Stochastic Approximation Method," *Annals of Mathematical Statistics* 22, no. 3 (1951): 400–407.

32. R. Fletcher, "A New Approach to Variable Metric Algorithms," *Computer Journal* 13, no. 3 (1970): 317–322.

33. 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* 22, no. 2 (2017): 826–837, https://doi.org/10.1109/TMECH.2016.2625309.