Optimization-Based Constrained Iterative Learning Control

Sandipan Mishra, Ufuk Topcu, and Masayoshi Tomizuka

Abstract—**We consider the problem of synthesis of iterative learning control (ILC) schemes for constrained linear systems executing a repetitive task. The ILC problem with affine constraints and quadratic objective functions is formulated as a convex quadratic program, for which there exist computationally efficient solvers. The key difference between standard convex optimization and the corresponding constrained ILC problem is that each iteration in the latter requires an** *experiment run*. **We implement an interior-point-type method to reduce the number of iterations (and hence the number of experiment runs). We discuss the system-theoretic interpretations of the resulting optimization problem that lead to reductions in computational complexity and compare the performance of the implementation based on the interior-point method to another approach based on the active set method on a simulation example. We demonstrate the technique on a prototype wafer stage system with actuator saturation constraints and ℓ_2 norm of the tracking error as the objective function. The key contribution of this paper is the systematic use of numerical tools from constrained convex optimization in the ILC design.**

Index Terms—**Constrained optimization, iterative learning control (ILC), optimization.**

I. Introduction

ITERATIVE learning control (ILC) is a feedforward control design technique for repetitive processes. ILC algorithms use information from earlier trials of the repetitive process to improve the performance in the current trial. The key design issue in the ILC is the efficient utilization of this information to maximize performance of the closed-loop system with as few trials as possible. Since the initial rigorous formulations of the ILC problem by Uchiyama [1] and Arimoto [2], research in ILC has focused on improving performance and robustness guarantees of the ILC algorithms as well as extensions to nonlinear systems. Due to its simple design, analysis, and implementation, ILC has been employed in many applications including rehabilitation and industrial robotics, rapid thermal processing, and wafer stage positioning systems [3]. See [4] and [5] for detailed surveys of the research on and applications of ILC. It is critical to note that ILC algorithms are designed with the twofold goal of removing the effect of repeating disturbances such as cogging and compensating for plant-model mismatch.

Analogously to iterative optimization schemes, ILC algorithms use experimental data collected during the trials of the underlying repetitive process to minimize an objective function such as tracking error norm. There exist interesting similarities between the ILC design and iterative optimization algorithms. These similarities have already been explored by Hatőnen [6], Owens *et al.* [7], and Norrlof [8]. In more recent work [9], robustness and monotonicity of optimization-based ILC schemes have also been investigated.

While the design of ILC algorithms for unconstrained systems has seen substantial research, there still remain open questions in the design for systems with constraints such as saturation bounds, state bounds, and rate limits. Saturation bounds exist for actuation in most motion control systems [10] and process control systems, e.g., those with valve actuators [11]. Since ILC is essentially an integrator-type algorithm, accumulation of control effort may result from saturation [12]. In addition, [12] and [13] showed that, in the presence of saturation limits, ILC algorithms no longer guarantee convergence to the minimum. The work in [14] illustrates an interesting example of the same phenomenon, with application to traffic flow metering.

In [13], Driessen *et al.* proposed a learning scheme for multi-input multi-output square systems with bounded input constraints. Xu *et al.* [15] used a composite energy-function-based ILC algorithm without the global Lipschitz condition required by [13]. However, both of these approaches require that the optimum of the unconstrained ILC problem be optimal for the corresponding constrained problem. This condition is restrictive and may not be always satisfied, especially in applications where the system has been under-designed; for example, a saturating motor current in a motion control system. Similar hard constraints on control effort are common in process control [12]. Furthermore, there is no guarantee that the unconstrained optimal feedforward control effort satisfies the constraints. Therefore, there is a need to develop ILC algorithms that explicitly account for physical hard constraints on the process.

In this paper, we propose an ILC algorithm for linear systems with linear constraints on the feedforward control effort while removing the requirement that the global optimum of the unconstrained problem should lie in the constraint set. Using a lifted system representation, we formulate the ILC problem as a quadratic program (an optimization problem with convex quadratic objective and affine constraints [16]). This formulation captures many ILC design problems including those with truncated ℓ_2- and ℓ_∞-norm objective functions and input saturation, rate, and state constraints. An important feature of this framework is the availability of efficient computational tools [16], [17] as well as the possibility of using system-theoretic interpretations of the underlying optimization problem to reduce its computational complexity. We use the barrier method (a type of interior-point algorithm) [16], [18]–[20] to solve the

Manuscript received October 23, 2009; revised May 16, 2010; accepted July 05, 2010. Manuscript received in final form September 27, 2010. Date of publication November 11, 2010; date of current version September 16, 2011. Recommended by Associate Editor S. S. Saab.

S. Mishra is with the Mechanical, Aerospace, and Nuclear Engineering Department, Rensselaer Polytechnic Institute, Troy, NY 12180 USA.

U. Topcu is with the Control and Dynamical Systems, California Institute of Technology, Pasadena, CA 91125 USA.

M. Tomizuka is with the Department of Mechanical Engineering, University of California, Berkeley, CA 94720 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2010.2083663

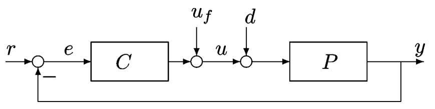


Fig. 1. Block diagram of the (unconstrained) closed-loop system.

proposed ILC problem. This implementation utilizes experimentally collected data for computing the current search direction which enhances robustness of the algorithm against modeling uncertainties. We demonstrate the method on a prototype single-degree-of-freedom wafer stage with actuator saturation bounds. Typically, the primary sources of tracking error for the wafer stage systems are cogging and plant-model mismatch.

II. Constrained ILC

A. System Description

Consider the stable closed-loop system in Fig. 1 that executes a repetitive process with N time samples starting at rest condition at the beginning of each trial. P is a discrete time linear time-invariant (LTI) plant controlled by an LTI feedback controller C. The repetitive nature of the process results in a 2-D system [21] with the time evolution of the within-trial system and the trial-to-trial evolution

$$\begin{aligned} y_k(j) &= G_r(q^{-1})r(j) + G_u(q^{-1})u_{f,k}(j) + G_d(q^{-1})d(j) \\ u_k(j) &= C(q^{-1})e_k(j) + u_{f,k}(j) \\ e_k(j) &= r(j) - y_k(j) \end{aligned} \tag{1}$$

where G_r, G_u, and G_d are transfer functions from r, u_f, and d to y, respectively, and q^{-1} denotes the unit delay. Note that u_k is the *feedback* control effort, while $u_{f,k}$ is the *feedforward* control effort. In (1), k is the trial index and $j = 0, \ldots, N-1$ is the time index (within trial) and we assume that d is bounded by a known finite number. In order to simplify the notation as well as the analysis, we use a *lifted system description* [8], [22]

$$\begin{aligned} \mathbf{y}_k &= \mathbf{G}_r\mathbf{r} + \mathbf{G}_u\mathbf{u}_{f,k} + \mathbf{G}_d\mathbf{d} \\ \mathbf{e}_k &= \mathbf{r} - \mathbf{y}_k \end{aligned} \tag{2}$$

where boldface vector notation ($\mathbf{y}_k$, $\mathbf{r}$, $\mathbf{u}_{f,k}$, $\mathbf{d}$, and $\mathbf{e}_k$) is used for the lifted versions of the corresponding vectors, e.g., for $u_k : \{0, \ldots, N-1\} \to \mathcal{R}^{m_u}$, $\mathbf{u}_k := \left[u_k(0)^T, u_k(1)^T, \ldots, u_k(N-1)^T\right]^T$, while, for the error signal, $e_k : \{p, \ldots, N-1+p\} \to \mathcal{R}^{m_e}$, $\mathbf{e}_k := \left[e_k(p)^T, e_k(p+1)^T, \ldots, e_k(N-1+p)^T\right]^T$, where the relative degree of the system is p. $\mathbf{G}_r$, $\mathbf{G}_u$, and $\mathbf{G}_d$ are determined from the impulse responses of loop transfer functions. For causal linear time-invariant P and C, the matrices $\mathbf{G}_r$, $\mathbf{G}_u$, and $\mathbf{G}_d$ are block lower triangular and block Toeplitz. Henceforth, we make the assumption that the overall closed-loop system is of relative degree 1. However, this assumption may be relaxed by shifting the error signal forward in time by p steps for systems with relative degree p. A detailed development of this may be found in [23].

While the lifted formulation is attractive from an analysis point of view, the fact that it assumes *finite trial length* may result in undesirable end-of-trial effects. Moreover, the initial and final rest conditions must be satisfied for the lifted system description to be valid. The interested reader is referred to [24], [25] for details regarding the connections between frequency-domain and lifted-domain analysis of ILC schemes.

B. Iterative Learning Control

ILC aims at choosing the *optimal* control effort u_f through an iterative procedure $\mathbf{u}_{f,k+1} = F\left(\mathbf{u}_{f,k}, \mathbf{e}_k\right)$, where $F : \mathcal{U}_f \times \mathcal{E} \to \mathcal{U}_f, \mathcal{U}_f$ is the set of admissible feedforward control effort and $\mathcal{E}$ is the set of measured error e, The learning update law F maps the current feedforward control law $u_{f,k}$ and measured error e_k to $u_{f,k+1}$ based on the intent of the designer while respecting the physical constraints on u_f (captured by $\mathcal{U}_f$). Usually, the intent of the designer is captured by maximization or minimization of an objective function, e.g., minimization of the ℓ_2-norm of the tracking error. Therefore, the ILC problem can be formulated as a constrained optimization problem and the recursion in $\mathbf{u}_{f,k+1} = F\left(\mathbf{u}_{f,k}, \mathbf{e}_k\right)$ can be considered as an iteration of the iterative procedure that solves this optimization problem. In the following, we focus on ILC problems that can be formulated in a specific form of constrained optimization problems, namely (convex) quadratic programs (QPs) [16] and how well-known iterative solution techniques for QPs can be used to establish iterative learning schemes.

C. Constrained ILC Design as a QP

Consider the convex optimization problem

$$\min_{\mathbf{u}_f \in \mathcal{R}^\ell} \mathbf{u}_f^T\mathbf{Q}\mathbf{u}_f + \mathbf{q}^T\mathbf{u}_f + c \quad \text{subject to} \quad \mathbf{A}\mathbf{u}_f \preceq \mathbf{b} \tag{3}$$

where $\mathbf{Q} \in \mathcal{R}^{\ell \times \ell}$ is positive semidefinite, $\mathbf{q} \in \mathcal{R}^\ell$, $\mathbf{A} \in \mathcal{R}^{p \times \ell}$, $\mathbf{b} \in \mathcal{R}^p$, $c \in \mathcal{R}$, and, for an n-dimensional vector, $x \succeq 0$ means that $x_k \geq 0$ for $k = 1, \ldots, n$. We now demonstrate how certain design objectives and constraints that are widely encountered in ILC design can be expressed as in (3).

1) Minimization of the Sum of Weighted Squared ℓ_2-Norms of Tracking Error and Feedforward Control Effort: The norm optimal ILC problem [8], [26] minimizes the sum of the weighted ℓ_2-norms of the tracking error and the feedforward control effort $\min_{\mathbf{u}_f \in \mathcal{R}^N} \mathbf{e}^T\mathbf{E}\mathbf{e} + \mathbf{u}_f^T\mathbf{U}\mathbf{u}_f$, where $\mathbf{E} \in \mathcal{R}^{N_e \times N_e}$ and $\mathbf{U} \in \mathcal{R}^{N_u \times N_u}$ are positive (semi-) definite. Note from (2) that $\mathbf{e} = \mathbf{w} - \mathbf{G}_u\mathbf{u}_f$, where $\mathbf{w} = (\mathbf{I} - \mathbf{G}_r)\mathbf{r} - \mathbf{G}_d\mathbf{d}$ and the norm optimal ILC problem is in the form of (3) by the choice of $\mathbf{Q} = \mathbf{U} + \mathbf{G}_u^T\mathbf{E}\mathbf{G}_u$, $\mathbf{q} = -2\mathbf{w}^T\mathbf{G}_u$, $c = \mathbf{w}^T\mathbf{w}$, $\mathbf{A} = 0$, and $\mathbf{b} = 0$ (i.e., unconstrained optimization problem). The iterative solution to this unconstrained problem has been studied in ILC literature [8], [27] and is given by $\mathbf{u}_{f,k+1} = \left(\mathbf{G}_u^T\mathbf{E}\mathbf{G}_u + \mathbf{U}\right)^{-1}\left[\mathbf{G}_u^T\mathbf{E}\mathbf{G}_u\mathbf{u}_{f,k} + \mathbf{G}_u^T\mathbf{E}\mathbf{e}_k\right]$.

2) Minimization of the Sum of Weighted ℓ_∞-Norms of Tracking Error and Feedforward Control Effort: Let $\mathbf{E} \in \mathcal{R}^{N_e \times N}$ and $\mathbf{U} \in \mathcal{R}^{N_u \times N}$ and consider the optimization problem $\min_{\mathbf{u}_f \in \mathcal{R}^N} \|\mathbf{E}\mathbf{e}\|_\infty + \|\mathbf{U}\mathbf{u}\|_\infty$ which minimizes the sum of the weighted ℓ_∞-norms of the tracking error and the feedforward control effort. This problem can

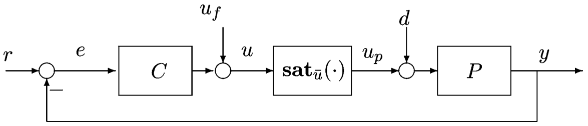


Fig. 2. Block diagram of the closed-loop system with actuator saturation at the plant input.

be rewritten in the form of (3) by setting $\mathbf{Q} = 0$, $\mathbf{q} = 0$, $c = \tau_1 + \tau_2$, $\mathbf{A} = \left[-(\mathbf{EG}_u)^T, (\mathbf{EG}_u)^T, -\mathbf{U}^T, \mathbf{U}^T\right]^T$, and $\mathbf{b} = \left[-(\mathbf{Ew} - \tau_1 \mathbf{1}_{N_e})^T, (\mathbf{Ew} + \tau_1 \mathbf{1}_{N_e})^T, \tau_2 \mathbf{1}_{N_u}^T, \tau_2 \mathbf{1}_{N_u}^T\right]^T$, where $\mathbf{1}_n$ denotes the vector of 1's of length n. Although there is no analytical, gradient-based ILC scheme for this problem, it can be reformulated as a QP that can be solved using the techniques discussed in Section III.

Furthermore, minimization of the sum of the weighted ℓ_1 norm as well as any nonnegative combination of the ℓ_1-, ℓ_2-, and ℓ_∞-norms of the tracking error and feedforward control effort can be formulated in the form of (3). In addition to the design objectives considered above, a series of physical constraints can be translated into affine constraints on $\mathbf{u}_f$ and accounted for in the QP formalism.

3) Input Saturation Constraints: Consider the closed-loop system in Fig. 2, where saturation (for the signal u defined as $\mathbf{sat}_{\bar{u}}(u)^i := \mathrm{sign}(u^i) \min\{|u|^i, \bar{u}^i\}$, u^i denotes the ith component of u while $\bar{u}^i$ denotes the ith component of the saturation limit vector) imposes constraints on the feedforward control effort for $j = 0, \ldots, N-1$ as $\bar{u} \succeq |u(j)| = \left|(I+CP)^{-1}Cr(j) + (I+CP)^{-1}u_f(j) - (I+CP)^{-1}CPd(j)\right|$. These constraints can be written in the form $\mathbf{Au}_f \preceq \mathbf{b}$, where $\mathbf{A} = \left[\tilde{\mathbf{A}}^T, -\tilde{\mathbf{A}}^T\right]^T$, $\mathbf{b} = \left[\mathbf{b}_1^T, \mathbf{b}_2^T\right]^T$, $\tilde{\mathbf{A}}$ is the lifted matrix form of the transfer function $(I + CP)^{-1}$, and $\mathbf{b}_1$ and $\mathbf{b}_2$ are lifted vectors of the signals $b_1(j) := \bar{u} - w(j)$ and $b_2(j) := \bar{u} + w(j)$. $w(j)$ is determined from the first run of the process as $w(j) = u_0(j) - (I + CP)^{-1}u_{f,0}(j)$. It is important to note that although b_1 and b_2 depend on d, they can be expressed in terms of the signals measured in the first run of the process and therefore determination of these terms *does not* require the knowledge of d.

4) Rate Constraints: Let $\mathbf{z} \in \mathcal{R}^N$ be the lifted version of the discrete time signal z. Then, the rate of change in z is $\Delta z(t) := z(t+1) - z(t)$ (up to a scaling constant related to the sampling time) and more compactly $\Delta \mathbf{z} = \mathbf{Dz}$, where $\Delta \mathbf{z} \in \mathcal{R}^{N-1}$, $\mathbf{D} \in \mathcal{R}^{(N-1)\times N}$, $\Delta \mathbf{z}$ is the lifted version of Δz, and $\mathbf{D}$ is a matrix of zeros except for $\mathbf{D}_{i,i} = -1$ and $\mathbf{D}_{i,i+1} = 1$ for $i = 1, \ldots, N$. Consequently, bounds on the rate of change of u_f, y, and e can be expressed as affine inequality constraints, e.g., an upper bound γ on Δu_f can be imposed by the constraint $\mathbf{Du}_f \preceq \gamma \mathbf{1}_{N-1}$.

III. Quadratic Programming

The optimization problem in (3) is a QP, an optimization problem with convex quadratic objective function and affine inequality constraints [16]. QPs constitute a class of problems in the family of convex conic programming problems (with linear programming and semidefinite programming at the "extremes" of its complexity spectrum [17]) for which there exist computationally efficient solvers. To name a few, the active set methods, the gradient projection methods, and interior-point (penalty and barrier) methods are suitable for QPs [16], [18]–[20]. We next discuss a variant of the barrier method following [16, Sec.11.3], and some implementation details for a specific ILC design problem.

A. Barrier Method

The barrier method, which is a type of interior-point method [16], [28], is an iterative algorithm for solving inequality constrained convex optimization problems. For notational simplicity, let us rewrite (3) as

$$\min_{\mathbf{x}} \; \mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{q}^T\mathbf{x} + c \quad \text{subject to} \quad \mathbf{Ax} \preceq \mathbf{b}. \tag{4}$$

Applied to the QP in (4), the barrier method replaces the cost function by

$$\min_{\mathbf{x}} \; \mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{q}^T\mathbf{x} + c + \kappa\phi(\mathbf{x}) \tag{5}$$

where $\kappa > 0$, ϕ is the *logarithmic barrier* function $\phi(\mathbf{x}) := -\sum_{i=1}^m \log(\mathbf{b}_i - \mathbf{a}_i^T\mathbf{x})$ with $\phi(\mathbf{x}) = \infty$ if $\mathbf{Ax} \npreceq \mathbf{b}$, and $\mathbf{b}_i$ and $\mathbf{a}_i^T$ are the ith row of $\mathbf{b}$ and $\mathbf{A}$, respectively. The barrier method transforms the inequality constrained problem in (4) to an unconstrained problem in (5) by penalizing constraint violations through the variable κ. The larger is the value of κ, the larger is the penalty on constraint violation. Subsequently, a sequence of problems of the form in (5) is solved using an unconstrained optimization solver, e.g., Newton's method, starting from the previous problem's optimal point for a decreasing sequence of κ.[1] In a typical application of the barrier method, κ is decreased by a constant factor $\mu > 1$ (typically 10) until a sufficient accuracy is reached (a sufficiently small suboptimality level). In fact, it can be shown that the solution of (5) is no more than κm suboptimal with respect to the solution of (3), and this bound provides a stopping criterion for the barrier method. For the examples in this paper, we adapted the following barrier method from [16].

```
*Barrier method*: **Given** $x$ strictly feasible for (3) and $\kappa^{(0)} > 0$, $\mu > 1$ and $\epsilon_{IP} > 0$; set $\kappa \leftarrow \kappa^{(0)}$.
**Repeat** until $\kappa m < \epsilon_{IP}$
Starting at $x$, compute $\mathbf{x}^* := \min_x x^TQx + q^Tx + c + \kappa\phi(x)$; set $x \leftarrow x^*$ and $\kappa \leftarrow \kappa/u$.
```

In the above implementation, $\mathbf{x}^*$ can be computed using any solver for unconstrained optimization problems. We use the Newton's method primarily because of its fast convergence (in fact, the Newton's method converges quadratically despite the linear convergence rate of computationally less demanding first-order methods, see [18]).

```
*Newton's method*: **Given** $x$ in the domain of $\phi$ and tolerance $\epsilon_N > 0$
**Repeat**
    — Compute the Newton step $\Delta x_{nt}$ by solving $\nabla^2 f(x)\Delta x_{nt} = -\nabla f(x)$ for $\Delta x_{nt}$ and
```

[1] The barrier method may be applied starting either with feasible or infeasible initial points [16]. Nevertheless, we focus on feasible starts since infeasible feedforward control efforts would require trials (experiments) that are not physically realizable. See Section IV-B for details.

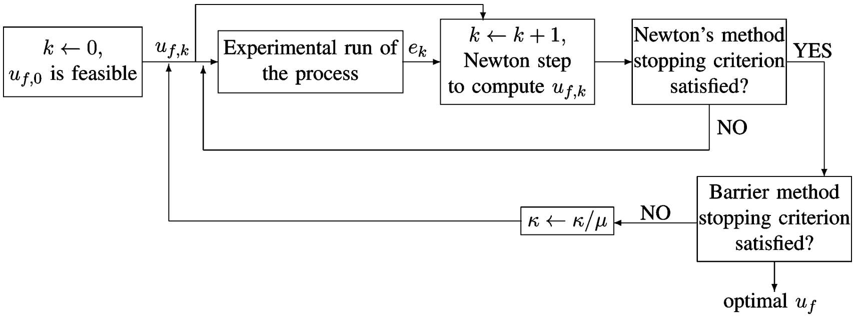

$k \leftarrow 0$,
$u_{f,0}$ is feasible
$u_{f,k}$
Experimental run of the process
e_k
$k \leftarrow k+1$, Newton step to compute $u_{f,k}$
Newton's method stopping criterion satisfied?
YES
NO
Barrier method stopping criterion satisfied?
NO
$\kappa \leftarrow \kappa/\mu$
optimal u_f

Fig. 3. Summary of the iterative procedure.

$\lambda = (\Delta x_{nt}^T \nabla^2 f(x) \Delta x_{nt})^{1/2}$, where $f(x) = x^T Q x + q^T x + c + \kappa\phi(x)$.

— **Quit** if $\lambda^2/2 \leq \epsilon_N$.

— Choose the step size τ by line search on $f(x+\tau\Delta x_{nt})$; set $x \leftarrow x + \tau\Delta_{nt}$.

Remark 3. 1: For fixed κ, the problem in (5) is guaranteed convergence to the solution $\mathbf{x}_\kappa^*$ when the Newton's method described above is used as the unconstrained solver. Furthermore, for a sequence of decreasing κ, we obtain a sequence of $\mathbf{x}_\kappa^*$ that is no more than κm suboptimal for the original constrained QP (see [18] and [16]).

B. Implementation Details

In order to discuss the implementation details of the constrained ILC design using an interior-point method, define $\mathbf{w} := (\mathbf{I} - \mathbf{G}_r)\mathbf{r} - \mathbf{G}_d\mathbf{d}$ and consider the problem

$$\min_{\mathbf{u}_f} \frac{1}{2}\mathbf{e}^T\mathbf{e} \quad \text{subject to } \mathbf{A}\mathbf{u}_f \preceq \mathbf{b},\ \mathbf{e} = \mathbf{w} - \mathbf{G}_u\mathbf{u}_f. \tag{6}$$

The design objective here is to determine the feedforward control effort which minimizes the ℓ_2-norm of the tracking error respecting the affine inequality constraints $\mathbf{A}\mathbf{u}_f \preceq \mathbf{b}$ (such as the saturation constraints). By eliminating the equality constraints, the problem in (6) is equivalent to the QP[2]

$$\min_{\mathbf{u}_f} \frac{1}{2}\mathbf{u}_f^T\mathbf{G}_u^T\mathbf{G}_u\mathbf{u}_f - \mathbf{w}^T\mathbf{G}_u\mathbf{u}_f \quad \text{subject to } \mathbf{A}\mathbf{u}_f \preceq \mathbf{b}. \tag{7}$$

Fig. 3 provides a pictorial summary of the ILC design strategy where each step involves an execution of the process (and recording $\mathbf{e}$ for given $\mathbf{u_f}$) and an iteration of the optimization scheme (namely Newton step that computes a new feedforward control effort for the current feedforward control effort and the corresponding tracking error). Each Newton step requires solving the set of linear equations (with given $\mathbf{G}_u$, κ, $\mathbf{A}$, $\mathbf{u}_f$, and θ)

$$\begin{aligned}(\mathbf{G}_u^T\mathbf{G}_u + \kappa\mathbf{A}^T\mathbf{diag}(\theta_i)\mathbf{A})\Delta\mathbf{u}_{f,nt}& \\ = -(\mathbf{G}_u^T\mathbf{G}_u\mathbf{u}_f - \mathbf{G}_u^T\mathbf{w} + \kappa\mathbf{A}^T\theta)&\end{aligned} \tag{8}$$

for $\Delta\mathbf{u}_{f,nt}$, where $\theta_i = (\mathbf{b}_i - \mathbf{a}_i^T\mathbf{u}_f)^{-2}$, $\mathbf{diag}(\theta_i)$ denotes a diagonal matrix with the ith diagonal entry θ_i, and θ is a vector with the ith entry θ_i. However, determining the term $\mathbf{w}$ requires the knowledge of $\mathbf{d}$ that is not available and cannot be determined experimentally. On the other hand, the right-hand side of (8) can be rewritten as $-(\mathbf{G}_u^T\mathbf{G}_u\mathbf{u}_f - \mathbf{G}_u^T\mathbf{w} + \kappa\mathbf{A}^T\theta) = -\mathbf{G}_u^T(\mathbf{G}_u\mathbf{u}_f - \mathbf{w}) - \kappa\mathbf{A}^T\theta = \mathbf{G}_u^T\mathbf{e} - \kappa\mathbf{A}^T\theta$. As explained in Section IV, the tracking error $\mathbf{e}$ can be measured at each iteration although $\mathbf{w}$ is not measurable. Therefore, the right-hand side of (8), $\mathbf{G}_u^T\mathbf{e} - \kappa\mathbf{A}^T\theta$, can be determined at each iteration and the ILC design scheme based on the interior-point method can be applied for the problem in (6).

In summary, we emphasize the fact that the applicability of the barrier method (with Newton's method as the unconstrained optimization solver at each iteration) is contingent on the availability (either in closed form or as measured) of the Hessian and the gradient of the augmented objective function and may depend on the formulation of the optimization problem as well as the physical constraints.

C. Computational Complexity and Exploiting the Problem Structure

The computational cost of the barrier method with the Newton's method as the unconstrained optimization solver is dominated by the solution of the linear equality system $\nabla^2 f(\mathbf{x})\Delta\mathbf{x}_{nt} = -\nabla f(\mathbf{x})$ for $\Delta\mathbf{x}_{nt}$ with given $\nabla^2 f(\mathbf{x})$ and $\nabla f(\mathbf{x})$. This linear equality system can be solved using a procedure consisting of Cholesky factorization [29] and forward and backward substitutions. In general, this takes $O(N^3)$ floating-point operations [16, App. C.1.1], where N is the length of the vector $\Delta\mathbf{x}_{nt}$. On the other hand, the structure of $\nabla^2 f(\mathbf{x})$ offers several sources for computational savings.

1) Bandedness of $\nabla^2 F(\mathbf{X})$*:* If $\nabla^2 f(\mathbf{x})$ is banded with a bandwidth[3] K, the computational complexity reduces to $O(K^2 N)$. For example, for the case discussed in Section III-B, $\nabla^2 f(\mathbf{x}) = \mathbf{G}_u^T\mathbf{G}_u + \kappa\mathbf{A}^T\mathbf{diag}(\theta_i)\mathbf{A}$. Recall from Section II-A that $\mathbf{G}_u$ is lower triangular and Toeplitz. Furthermore, it can be well-approximated by a banded matrix because of the exponential stability of the system. Therefore, $\mathbf{G}_u^T\mathbf{G}_u$ is banded. Furthermore, for the saturation constraints as discussed in Section III-B, $\mathbf{A}$ is block-diagonal, and therefore $\kappa\mathbf{A}^T\mathbf{diag}(\theta_i)\mathbf{A}$ is banded.

2) Sparsity: The sparsity of $\nabla^2 f(\mathbf{x})$ offers computational advantages in computing the Cholesky factorization. Although the computational cost of the Cholesky factorization of sparse

[2] Optimal values of $\mathbf{u}$ for (6) and (7) are the same since $\mathbf{d}$ is assumed to be constant.

[3] A matrix $M \in \mathcal{R}^{N\times N}$ is said to have a *bandwidth* of K if $M_{ij} = 0$ for $|i-j| > K$.

matrices depends in a complicated way on the size of the matrix, the level and pattern of sparsity, and the algorithm used, it is significantly smaller than that for dense matrices [16], and this fact has been exploited in the implementation of the barrier method coupled with the Newton's method discussed in this paper.

In summary, exploiting the structure and the system theoretic interpretations of the underlying optimization problem in the ILC design result in significant reductions in computationally complexity of the techniques for QP based on the barrier method.

IV. Simulation and Experimental Evaluation

ILC has been used for wafer stage control by several researchers, notably in [3], [30], and [31]. These investigations include analysis and design of ILC algorithms using H_∞ techniques, input signal shaping for vibration minimization, and norm-optimal ILC algorithm design. In Section V, we use a prototype wafer stage system with actuator saturation to demonstrate the application of the proposed constrained ILC design procedure.

A. Wafer Stage System Description

The prototype single-degree-of-freedom setup consists of a wafer stage and a counter-mass driven by linear motors. The wafer stage position is measured by a laser interferometer (with a resolution of 5 nm). A detailed description of the experimental setup may be found in [32]. For the purpose of this paper, we will focus on the *design of the feedforward signal* for the wafer stage system. The wafer stage is modeled by $P(s) = 11.79/(5.3s^2 + 7.2s)$. Since the stage is mounted on air bearings, Coulomb friction can be neglected. The peak motor voltage 2 V is determined by the maximum amplifier output specifications. The signals that can be measured in this setup are actuator input u_k, tracking error e_k, and plant output y_k (see Fig. 2). The external signals are of two types: the reference signal r is known, while the disturbance d is unknown but remains fixed from one trial to another. The feedback controller PID C was designed based on a modified Ziegler-Nichols tuning method and achieved a closed-loop bandwidth of 100 Hz and stability phase margin of $\pi/6$; $C(q^{-1}) = 30000q^{-1}(1 + 2(T_s/1 - q^{-1}) + 0.012(1 - q^{-1}/T_s))$, where the sampling rate $T_s = 400\ \mu$s. The discretized closed-loop model (without actuator saturation) is $y_k(j) = G_r(q^{-1})r(j) + \tilde{G}(q^{-1})\left(u_{f,k}(j) + d(j)\right)$ with $G_r(q^{-1}) = \left(0.2783q^{-1} - 0.2694\right)/(q^{-2} - 1.717q^{-1} + 0.7261)$ and $\tilde{G}(q^{-1}) = (1.6 \times 10^{-7}q^{-2} - 1.625 \times 10^{-8}q^{-1} - 1.44 \times 10^{-7})/(q^{-3} - 2.716q^{-2} + 2.442q^{-1} - 0.7255)$. The reference trajectory (position and velocity) to be tracked is a typical forward and reverse scan trajectory, as shown in Fig. 4. The goal of the ILC design is to determine the feedforward control effort that minimizes the ℓ_2 norm of the error profile over the scan, while guaranteeing that the saturation bounds on control input to the plant are not violated. Consequently, the corresponding optimization problem is of the form of (7) where $\mathbf{A}$ and $\mathbf{b}$ are constructed as discussed in Section II-C for input saturation constraints.

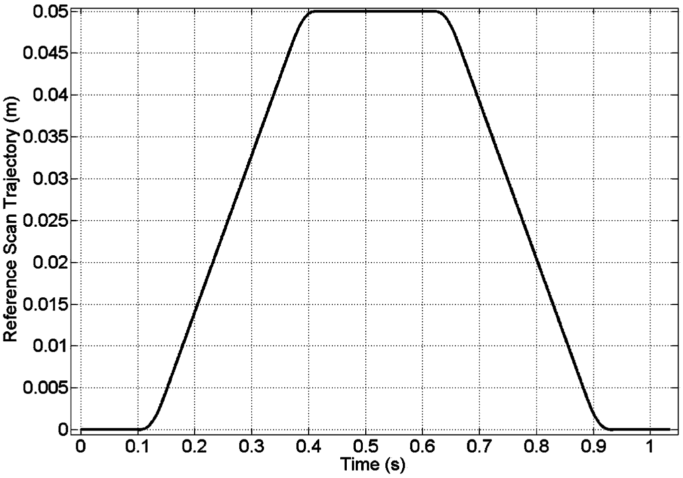


Fig. 4. Plot of the reference position $r(t)$ versus time t.

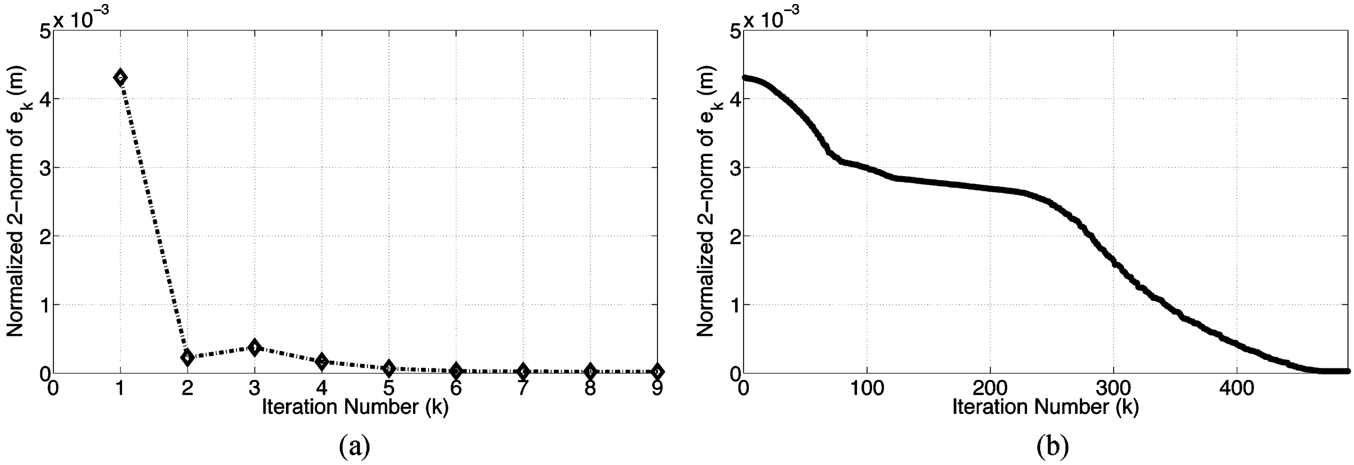
B. Constructing an Initial Feasible Feedforward Control Effort

The iterative scheme discussed in Section III-A relies on experimental measurements (e.g., the signal e) and therefore requires an initial (strictly) feasible feedforward control effort, i.e., $\mathbf{u}_f$ satisfying $\mathbf{A}\mathbf{u}_f \prec \mathbf{b}$. Although constructing such a signal may, in general, be challenging, we use an ad hoc procedure for the results reported hereafter:

Simulate the system with no saturation and feedforward control effort and define $u_{sim}(j) := C\left(I + PC\right)^{-1} r(j),\ j = 0, \ldots, N-1$. The worst case effect of the (unknown) disturbance on the control effort is $u_{wd} := \|(1 + PC)^{-1}P\|_1 \|d\|_\infty \geq \max_j |(1 + PC)^{-1}Pd(j)|$. Let u_{des} be such that $u_{des} + u_{wd} < \bar{u}$. Then, set $u_{f,0}(j) = (I + PC)(\mathbf{sat}_{u_{des}}(u_{sim}(j)) - u_{sim}(j)), j = 0, \ldots, N-1$, i.e, the feedforward control signal cancels the excess control effort over saturation. The norm $\|d\|_\infty$ of d may be estimated by analyzing the nature of expected disturbances in the system. In the absence of such information, a sequence of experiments with increasing magnitude of u_{wd} may be performed until the saturation bounds are not violated during the experimental run.

C. Evaluation of Interior-Point Algorithm and Active Set Algorithm

We now present a comparison of two alternative optimization techniques in simulation: one using an active set method algorithm and the other using an interior-point algorithm in order to choose an efficient algorithm (with fewer necessary iterations) in experimental implementation. The iterative scheme is applied as discussed in Section III-A (summarized in Fig. 3) by *simulating* the repetitive process. Fig. 5 shows the normalized ℓ_2 norm $1/N\left(\mathbf{e}_k^T\mathbf{e}_k\right)^{1/2}$ of the tracking error $\mathbf{e}_k$ versus the iteration index k for the barrier method (left figure) and the active set method (right figure) based implementations. Note that each Newton step in the barrier method is computationally more demanding compared to a single iteration of the active set method roughly because the active set method uses only first order information (gradients) whereas the Newton's method requires second order information (gradients and Hessians). On the other



(a)

(b)

Fig. 5. Normalized ℓ_2-norm of the trajectory following error $\mathbf{e}_k$ versus number of trials k for the barrier method (left) and the active set method (right).

hand, the barrier method converges with a smaller number of total iterations (counted as the total number of Newton steps) compared to the active set method (9 iterations in the barrier method versus 490 iterations in the active set method). Considering the superiority of the interior-point method in terms of the number of iterations required for convergence, we choose the interior-point method for experimental implementation.

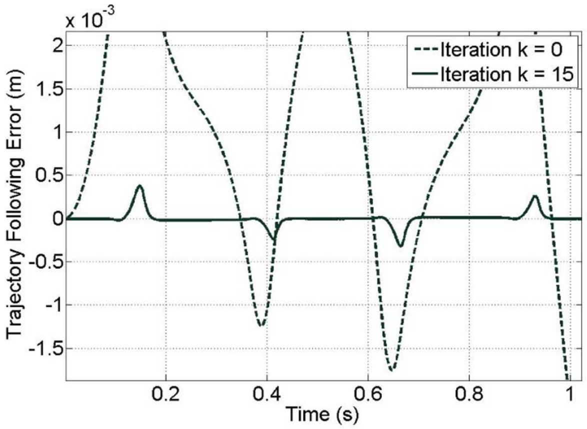


Fig. 6. Tajectory following error e_k versus the time for $k = 0, 15$.

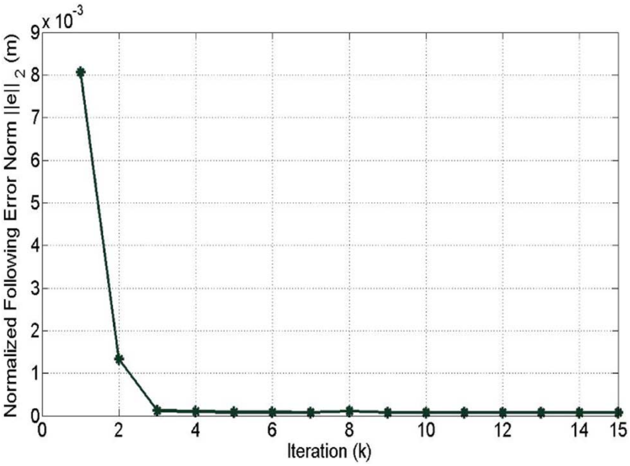


Fig. 7. Normalized ℓ_2-norm of e_k versus the trial number k for the box-type ILC design in (9).

D. Experimental Results and Discussion

We now present experimental evaluation of the performance of the ILC scheme based on the interior-point optimization algorithm. We demonstrate the implementation and performance of the proposed constrained ILC algorithm in comparison to a box-type ILC algorithm given by

$$\mathbf{u}_{f,k+1} = sat_{\bar{u}}(\mathbf{Q}\left(\mathbf{u}_{f,k} + \mathbf{L}\mathbf{e}_k\right)) \tag{9}$$

where $\mathbf{Q}$ is low-pass filter and $\mathbf{L}$ is the learning gain. The box-type ILC scheme essentially clips the feedforward control output generated by the ILC scheme at each iteration. This is an intuitive solution for a saturation constrained ILC system (and is next demonstrated to be suboptimal).

1) Box-Type ILC: The learning gain $\mathbf{L}$ is picked as the approximate inverse of the closed-loop system model and the filter $\mathbf{Q}$ is a low pass filter with bandwidth of 100 Hz. In order to allow some variation in disturbances, $\bar{u}$ is set to 1.9 V. Fig. 7 shows the normalized ℓ_2-norm of e_k versus the iteration number k. Fig. 6 shows the initial and final [obtained at the convergence of the scheme in (9)] error profiles. After the algorithm converges, the peak of the trajectory following error is 370 μm (Fig. 6), while the normalized ℓ_2-norm is 79 μm.

2) Constrained ILC Based on the Interior-Point Method: The ILC scheme based on the interior-point method is implemented as described in Section III. As for the box type ILC, $\bar{u}$ is set to 1.9 V. Fig. 8 shows the trajectory following error plots for iterations 1, 5, 15, and 20. Fig. 10 shows the decay of the normalized ℓ_2-norm of the trajectory following error versus the iteration number. For $k = 20$, the peak of the trajectory following error becomes less than 85 μm (Fig. 9), while the normalized ℓ_2-norm is 6 μm. Fig. 12 shows the *optimal* (i.e., for $k = 20$) feedforward control effort. Note that the feedforward effort slows down the system at the beginning of the acceleration phase so that the saturation bounds are not violated and then increases the speed of the system midway through the acceleration phase when saturation is no longer an issue. During the constant velocity phase the feedforward signal compensates for the viscous friction in

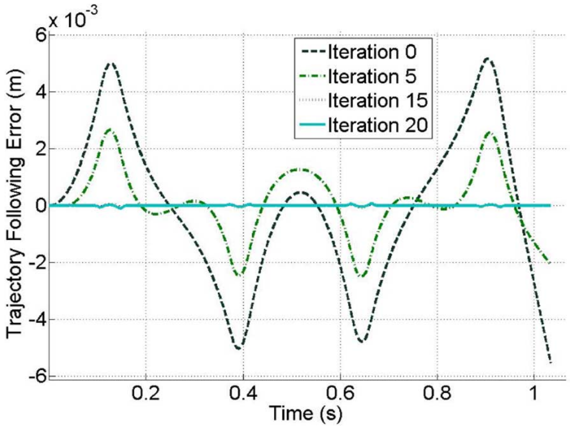


Fig. 8. Trajectory following errors e_k versus the time for trials $k = 0, 5, 15$, and 20.

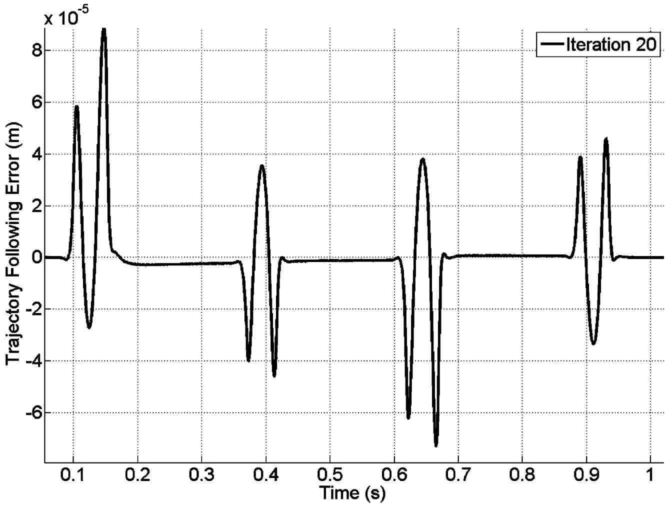


Fig. 9. Trajectory following error for trial $k = 20$.

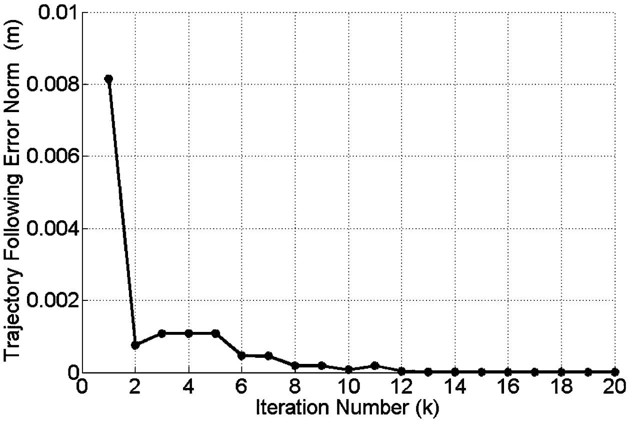


Fig. 10. Normalized ℓ_2-norm of $\mathbf{e}_k$ versus trial number k.

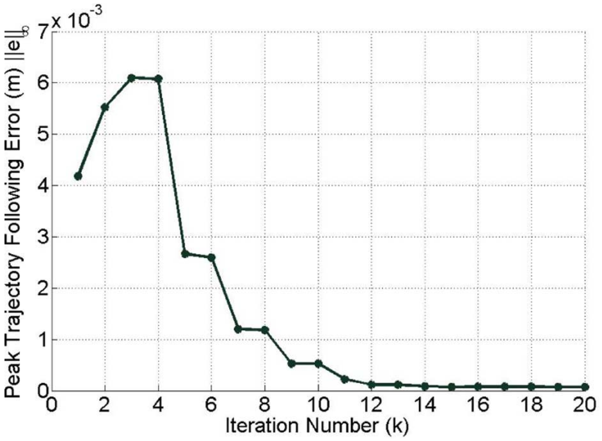


Fig. 11. Peak following error $\|\mathbf{e}_k\|_\infty$ versus trial number k.

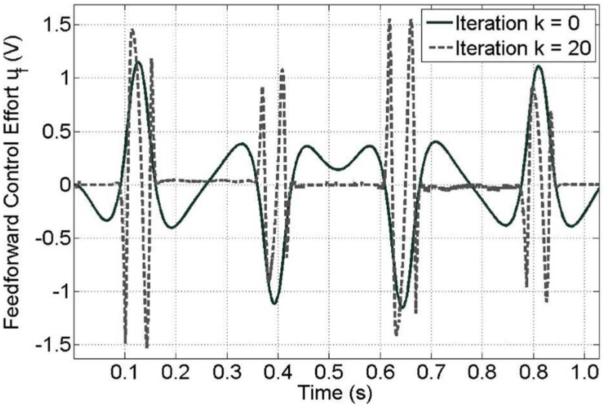


Fig. 12. Feedforward ILC effort $u_{f,k}$ for iterations 0 and 20.

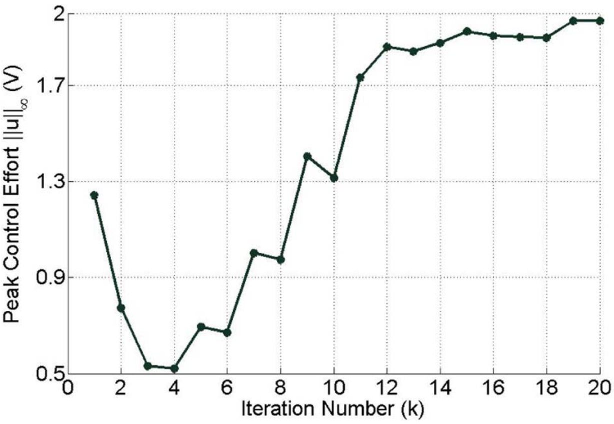


Fig. 13. Maximum *total* control effort versus trial number k. Note that the control effort stays within the saturation bounds (±2 V).

the bearings. The ILC effort is small during the constant velocity phase, primarily because the feedback controller has sufficient bandwidth to remove any disturbances and ensure good tracking. Fig. 13 illustrates that the total control effort is always within the saturation bounds for all iterations. Furthermore, the control effort remains conservatively small for the first few iterations since the weight on staying feasible is large. However, in later iterations, the control effort peak approaches the saturation bounds in order to extract higher performance. Contrary to the control effort, the error normalized ℓ_2-norm and infinity norm (shown in Figs. 10 and 11) are larger for earlier iterations, but as we extract more performance and are less conservative on the saturation bounds, the error norm decreases. This highlights

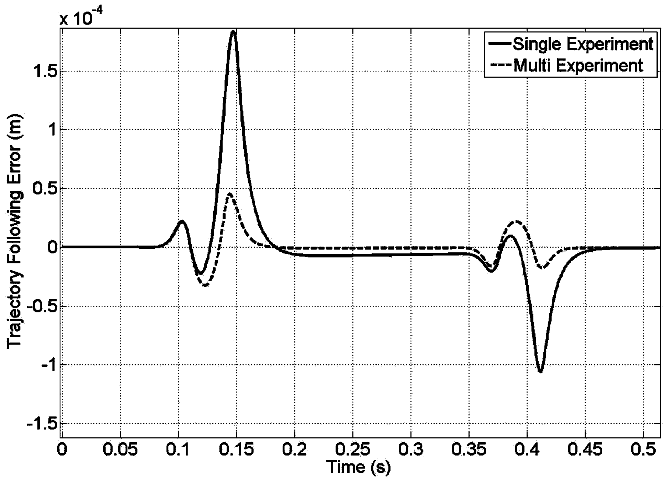


Fig. 14. Trajectory following error after convergence for the single-experiment ILC and multiple experiment ILC cases, in the presence of model uncertainty.

the (possible) nonmonotonicity of the proposed algorithm. Note that the normalized ℓ_2-norm of the trajectory following error for the box-type scheme in (9) is 79 μm whereas it is 6 μm for the scheme based on the interior-point method. Therefore, although the box-type ILC scheme converges, the converged signal may be suboptimal.

E. ILC and Offline Optimization

Looking at the problem in (6), one may suggest that even though the signal d (i.e., the supervector $\mathbf{d}$) is unknown, with one single run of the experiment, we can determine $\mathbf{w}$ (as $\mathbf{e}_0 - \mathbf{G}_u \mathbf{u}_{f,0}$) and compute the optimal feedforward control effort through an *offline* optimization. However, this computed signal would be guaranteed optimality only if $\mathbf{G}_u$ was exact. In fact, in the presence of modeling uncertainties this procedure leads to a suboptimal solution. This suboptimality is demonstrated in Fig. 14 for the plant given in Section IV-A with a hypothetical model uncertainty, where the tracking errors along a cycle are shown for the single experiment and multiple experiment cases (one experiment at each iteration, with a total of nine iterations). The resulting error profiles from a single experiment and multiple experiments illustrate that the iterative nature of the learning process yields better robustness to model uncertainty. Therefore, integrating offline computation and experimental data is essential for ensuring the robust performance of the optimization-based ILC algorithm presented here. The trial-to-trial robustness of ILC has been one of its most attractive features (the interested reader is referred to [33]).

V. Conclusion

This paper developed optimization-based ILC schemes for constrained linear systems executing a repetitive task. Casting the constrained ILC problem as an optimization problem enables the use of numerical tools from optimization theory. In particular, the ILC problem for constrained linear systems with quadratic objective functions and linear constraints is equivalent to a convex quadratic program (QP), for which there exist computationally efficient solvers. We provided QP formulations for linear systems executing repetitive processes with truncated ℓ_2- and ℓ_∞-norm objective functions and input saturation and rate constraints.

The key difference between standard convex optimization and the corresponding constrained ILC problem is that each iteration in the ILC design problem requires an *experiment run*. Therefore, we used an interior-point method (specifically the barrier method) to reduce the number of iterations (hence the number of experiment runs). The performance of the implementation based on the interior-point method was compared to another approach based on the active set method on a simulation example where the interior-point method required much smaller number of iterations compared to the active set method. We implemented ILC scheme based on the interior-point method on a prototype wafer stage system with actuator saturation constraints and ℓ_2-norm of the tracking error as the objective function. The convergence of the algorithm took less than 20 iterations, justifying the use of the optimization-based scheme.

The primary contribution of this paper is the synergistic use of numerical tools for constrained convex optimization in the ILC design framework with several relevant and interesting extensions. State constrained ILC problems can be included in the general class of systems considered in the paper with simple modifications to the proposed method. The robustness to modeling uncertainties has been one of the most attractive features of conventional ILC schemes. Hence, understanding the robustness properties of the optimization-based ILC algorithms will be critical to determining their practical relevance. Furthermore, approaching the constrained ILC problem in a formal optimization framework may lead to interesting extensions in the intersection of the ILC and model predictive control in cooperation of both the experimental data (typically used in ILC) and the predictions (typically used in model predictive control). This extension is subject to current research.

References

[1] M. Uchiyama, “Formulation of high-speed motion pattern of a mechanical arm by trial,” *Trans. Soc. Instrum. Control Eng.*, vol. 14, no. 6, pp. 706–712, 1978.

[2] S. Arimoto, S. Kawamura, and F. Miyazaki, “Bettering operation of robots by learning,” *J. Robot. Syst.*, vol. 1, no. 2, pp. 123–140, 1984.

[3] D. D. Roover and O. Bosgra, “Synthesis of robust multivariable iterative learning controllers with application to a wafer stage motion system,” *Int. J. Control*, vol. 73, no. 10, pp. 968–979, 2000.

[4] H.-S. Ahn, Y. Chen, and K. Moore, “Iterative learning control: Brief survey and categorization,” *IEEE Trans. Syst., Man, Cybern. C*, vol. 37, no. 6, pp. 1099–1121, Nov. 2007.

[5] D. Bristow, M. Tharayil, and A. Alleyne, “A survey of iterative learning control,” *IEEE Control Syst. Mag.*, vol. 26, no. 3, pp. 96–114, Jun. 2006.

[6] J. Hatőnen, *Issues of Algebra and Optimality in Iterative Learning Control*. Oulu, Finland: Univ. of Oulu, 2004.

[7] D. H. Owens and J. Hatonen, “Iterative learning control—An optimization paradigm,” *Annu. Rev. Control*, vol. 29, no. 1, pp. 57–70, 2005.

[8] S. Gunnarsson and M. Norrlof, “On the design of ilc algorithms using optimization,” *Automatica*, vol. 37, no. 1, pp. 2011–2016, 2001.

[9] D. Owens and S. Daley, “Robust gradient iterative learning control: Time and frequency domain conditions,” *Int. J. Modelling, Identification Control*, vol. 4, pp. 315–322, 2008.

[10] Y. Hong and B. Yao, “A globally stable high-performance adaptive robust control algorithm with input saturation for precision motion control of linear motor drive systems,” *IEEE/ASME Trans. Mechatron.*, vol. 12, no. 2, pp. 198–207, Apr. 2007.

[11] P.-C. Chen and J. S. Shamma, “Gain-scheduled l1-optimal control for boiler-turbine dynamics with actuator saturation,” *J. Process Control*, vol. 14, no. 3, pp. 263–277, 2004.

[12] C.-T. Chen and S.-T. Peng, "Learning control of process systems with hard input constraints," *J. Process Control*, vol. 9, no. 2, pp. 151–160, 1999.

[13] B. J. Driessen, N. Sadegh, and K. S. Kwok, "Multi-input square iterative learning control with input rate limits and bounds," *IEEE Trans. Syst., Man, Cybern. B*, vol. 32, no. 4, pp. 545–550, Aug. 2002.

[14] J.-X. X. Z. Houa and J. Yana, "An iterative learning approach for density control of freeway traffic flow via ramp metering," *Transport. Res. C: Emerging Technol.*, vol. 16, no. 1, pp. 71–97, 2008.

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

[16] S. Boyd and L. Vandenberghe*, Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[17] A. Ben-Tal and A. Nemirovski*, Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. Philadelphia, PA: MPS-SIAM Series on Optimization, 2001.

[18] D. G. Luenberger*, Linear and Nonlinear Programming*. New York: Springer, 2003.

[19] A. Bjorck*, Numerical Methods for Least Squares Problems*. Philadelphia, PA: SIAM, 1996.

[20] R. Fletcher*, Practical Methods of Optimization*, 2nd ed. New York: Wiley, 1991.

[21] D. Owens, N. Amann, E. Rogers, and M. French, "Analysis of linear iterative learning control schemes—A 2-D systems/repetitive processes approach," *Multidimens. Syst. Signal Process.*, vol. 11, pp. 125–177, 2000.

[22] P. Khargonekar, K. Poolla, and A. Tannenbaum, "Robust control of linear time-invariant plants using periodic compensation," *IEEE Trans. Autom. Control*, vol. AC-30, no. 11, pp. 1088–1096, Nov. 1985.

[23] S. Saab, "Optimal selection of the forgetting matrix into an iterative learning control algorithm," *IEEE Trans. Autom. Control*, vol. 50, no. 12, pp. 2039–2043, Dec. 2005.

[24] M. Norrlof and S. Gunnarsson, "Time and frequency domain convergence properties in iterative learning control," *Int. J. Control*, vol. 75, pp. 1114–1126, 2002.

[25] J. J. Hatonen, D. H. Owens, and K. L. Moore, "An algebraic approach to iterative learning control," *Int. J. Control*, vol. 77, pp. 45–54, 2004.

[26] N. Amann, D. H. Owens, and E. Rogers, "Iterative learning control using optimal feedback and feedforward actions," *Int. J. Control*, vol. 65, no. 2, pp. 277–293, 1996.

[27] N. Amann, H. Owens, and E. Rogers, "Iterative learning control for discrete-time systems with exponential rate of convergence," *Proc. Inst. Elect. Eng.—Control Theory Applic.*, vol. 143, no. 2, pp. 217–224, 1996.

[28] Y. Nesterov and A. Nemirovskii*, Interior Point Polynomial Methods in Convex Programming: Theory and Applications*. Philadelphia, PA: SIAM, 1994.

[29] G. H. Golub and C. F. V. Loan*, Matrix Computations*, 3rd ed. Baltimore, MD: The Johns Hopkins Univ. Press, 1996.

[30] B. Dijkstra*, Iterative Learning Control With Application to a Wafer Stage*. Delft, The Netherlands: Delft Univ. Press, 2003.

[31] 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," *J. Dynam. Syst., Meas., Control*, vol. 130, no. 5, pp. 1–16, 2008.

[32] S. Mishra and M. Tomizuka, "Precision positioning of wafer scanners: An application of segmented iterative learning control," *IEEE Control Syst. Mag.*, vol. 27, no. 4, pp. 20–25, 2007.

[33] H.-S. Ahn, K. L. Moore, and Y. Chen*, Iterative Learning Control: Robustness and Monotonic Convergence for Interval Systems*. New York: Springer, 2007.