2015 American Control Conference
Palmer House Hilton
July 1-3, 2015. Chicago, IL, USA

Constrained Optimal Iterative Learning Control for Mixed-Norm Cost Functions

Yijie Guo[1] and Sandipan Mishra[2]

Abstract— **Iterative learning control (ILC) is a technique for determining feedforward signals for systems that execute a task repeatedly. One approach towards designing ILC algorithms is to pose it as an optimization problem. Traditionally, norm optimal iterative learning control (NOILC) algorithms use ℓ_2-norm-type cost functions. However, many applications require optimizing non-smooth cost functions, e.g., in trajectory tracking where it is desirable to minimize the peak tracking error, i.e., its ℓ_∞-norm. In this paper, we explore the performance of a class of non-smooth cost functions along with constraints which can be recast into the constrained optimal ILC (COILC) framework. For linear systems with constraints (linear in the feedforward input) and certain cost functions (such as ℓ_2, ℓ_∞ norms of tracking error and control effort), this optimization problem can be formulated as a quadratic program (QP) or a linear program (LP). These COILC problems can then be solved with a modified interior-point-type method. In this manuscript, we derive ILC algorithms for linear systems (and linear constraints) with (1) a pure ℓ_∞ norm cost, (2) a mixed $\ell_2 - \ell_\infty$ norm cost. We compare the results to the traditional ℓ_2 norm (NOILC) in simulation and experiment to illustrate the effect of the choice of the cost function on the design of the optimized feedforward control effort and hence the optimal error profile.**

I. INTRODUCTION

Iterative learning control is a methodology to determine feedforward control inputs for processes executed over a finite time interval repeatedly by incorporating error information from prior iteration(s). ILC has been successfully used in many applications such as industrial robots, wafer stage motion systems, rapid thermal processing and assistive rehabilitation.

Since ILC aims to refine the feedforward control input from iteration to iteration with error measurements from earlier iterations, the ILC problem can be recast as an optimization problem being solved in an iterative manner. One of the earliest optimization-based ILC algorithms was proposed in [1]. Since then, several norm optimal ILC (NOILC) schemes, which design ILC update laws by minimizing a next-iteration cost function, have been explored [2], [3], [4], and the robustness and monotonicity properties of this family of algorithms have also been investigated [5]. Cross-coupled ILC [6] uses a modified NOILC objective function that minimizes contour tracking error. In more recent work, Lim et al. [7] presented the generalized NOILC problem as a pareto optimization-based ILC to address multiple objectives, e.g., error at certain points and energy cost.

Traditionally, ILC design did not account explicitly for constraints in the system. The integration of constraints into formulation of the typical NOILC, i.e., COILC, presents the potential to remove the requirement that the optimum of the unconstrained problem be in the constraint set. Mishra et al. [8], [9] used a modified interior-point-type method that combined experimental data and model information in solving the optimization-based ILC problem for minimizing tracking error with input saturation. Freeman et al. [10] used a similar approach to address the mixed constraints in point-to-point tracking problem, the requirement of point-to-point tracking is embedded as an equality constraint. Volckaert et al. [11] added a model correction step and then used a sparse implementation of interior point method to solve the optimal ILC problem for nonlinear system with constraints.

These algorithms have all been successfully demonstrated for solving the constrained ILC problem with an ℓ_2-norm type cost function. This norm is widely used as the cost function because of its smoothness (i.e., the existence of unique and bounded derivative). However, many applications require choice of non-smooth cost functions. For example, in trajectory tracking applications, we want to reduce the peak tracking error, which is the ℓ_∞-norm of error. Therefore, a generalized framework for addressing a larger set of cost functions including non-smooth type cost functions can be very beneficial for many applications.

In this paper, we construct a framework for COILC that can be extended to the use of ℓ_∞ norm and mixed $\ell_2 - \ell_\infty$ norm as cost functions to solve the constrained optimization based ILC problem. In this framework, one can use different norms or combining them together as cost function according to different applications. Using lifted form representation, we formulate the ILC problem as a quadratic program (QP) or a linear program (LP). We then use the modified interior-point-type method presented in [8] to solve the optimization problems iteratively. The main contribution of this paper is the formulation and verification of COILC problem with non-smooth type cost functions (specifically ℓ_∞-norm cost and mixed $\ell_2 - \ell_\infty$-norm cost function) and the comparison of their error performances, which are not addressed in [8], [10].

The paper is organized in following manner: Sec. II describes the system and the general form of optimization problems considered as QPs/LPs, then an interior-point-type ILC update law is introduced to solve this class of problems. Sec. III presents the formulation of COILC with non-smooth cost functions as a constrained QP or LP. Finally, the results

[1]Yijie Guo is with the Mechanical, Aerospace and Nuclear Engineering Department, Rensselaer Polytechnic Institute, Troy, NY 12180 USA guoy7@rpi.edu

[2]Sandipan Mishra is with faculty of the Mechanical, Aerospace and Nuclear Engineering Department, Rensselaer Polytechnic Institute, Troy, NY 12180 USA mishrs2@rpi.edu

of implementing the constructed optimization problems in simulation and experiment will be presented in Sect. IV. Finally, conclusions and open issues that must be addressed in the future are presented in Sec. V.

II. Constrained Optimal ILC

In this section, we will formulate the COILC problem for a linear (stable) system and present a modified interior-point-type method as the learning law. While we introduce the problem formulation with actuator saturation as the constraint, it is straightforward to extend the idea to an arbitrary linear constraint on the design variable (in this case, the feedforward control effort). Moreover, while for clarity of notation the following is presented for a single-input-single-output (SISO) system, these algorithms are applicable for multiple-input-multiple-output (MIMO) systems as well.

A. System Description

Consider the stable closed-loop system with input saturation shown in Fig. 1. A repetitive process (i.e., a trajectory) is executed by this system with a finite length of N time samples starting at rest condition for each iteration. Let $P(z)$ represent the discrete time linear time-invariant (LTI) plant, which is stabilized by an discrete time LTI feedback controller $C(z)$. Let k be the iteration number and j be the time index, then y_k, $u_{f,k}$, u_k and e_k respectively denote the output (position), the feedforward control effort, the total control effort and the tracking error at the kth iteration. Moreover, r and d indicate the output reference trajectory and repetitive disturbance respectively, while $\mathbf{sat}_{\bar{u}}(\cdot)$ is the saturation function, defined as $\mathbf{sat}_{\bar{u}} := \text{sign}(u_k) \cdot \min(|u_k|, \bar{u})$, where $\bar{u}$ is the maximum admissible value of total control effort. The kth iteration system output and error, when neglecting the input saturation are:

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

where G_r, G_u, and G_d are transfer functions from r, u_f, and d to y respectively and z^{-1} denotes the unit delay. The repetitive nature of the process makes this system a two-dimensional system; with evolution along an iteration (time) and from iteration to iteration [12].

In order to construct the ILC design as an optimization problem, in this manuscript we use the lifted system description of this system to transform it into a one-dimensional system along iteration only. Lifting the system [13] yields:

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

where $\mathbf{y}_k$, $\mathbf{u}_{f,k}$, $\mathbf{u}_k$, $\mathbf{e}_k$ are the (super)vectors that contain all the corresponding signals in the kth iteration, $\mathbf{u}_{f,k} = [u_{f,k}(0)\ u_{f,k}(1)\ ...\ u_{f,k}(N-1)]^T$, for example. $\mathbf{G}_r, \mathbf{G}_u, \mathbf{G}_d \in \mathcal{R}^{N\times N}$ are determined from the impulse response of the corresponding transfer function and then banded into $N \times N$ matrices.

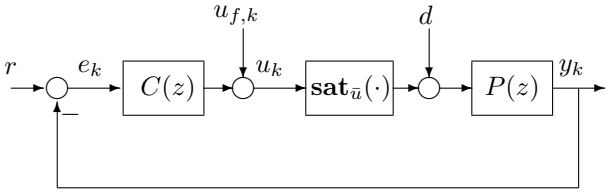

Fig. 1. Block diagram of the closed-loop system with actuator saturation at the plant input. The feedforward input $u_{f,k}$ is to be designed through ILC.

B. Constrained ILC Design as an Optimization Problem

Designing an ILC algorithm implies determining a learning function that incorporates information of the previous iteration's feedforward control effort and the corresponding error into the generation of the feedforward control effort for next iteration. Furthermore, this learning function should generate control efforts converged to the optimal control effort (for a prescribed cost function) as the iteration number increases. We can express this as $\mathbf{u}_{f,k+1} = F(\mathbf{u}_{f,k}, \mathbf{e}_k)$, where $F : \mathcal{U}_f \times \mathcal{E} \to \mathcal{U}_f$, $\mathcal{U}_f$ is the space of feasible input ($\mathbf{u}_{f,k}$) and $\mathcal{E}$ is the space of measured error ($\mathbf{e}_k$). F is the learning function.

In this manuscript, we consider a constrained optimal ILC (COILC) problem for systems with linear constraints for non-smooth type cost functions. The specific form of the constructed problem varies with the choice of the cost function, but the general form can be shown to be a constrained Quadratic Program (QP) of the form:

$$\begin{aligned} &\min_{\mathbf{u}_f \in \mathcal{R}^N} \mathbf{u}_f^T\mathbf{T}\mathbf{u}_f + \mathbf{q}^T\mathbf{u}_f + c \\ &\text{subject to} \quad \mathbf{A}\mathbf{u}_f \preceq \mathbf{b}, \end{aligned} \tag{3}$$

or a Linear Program (LP) of the form

$$\begin{aligned} &\min_{\mathbf{u}_f \in \mathcal{R}^N} \mathbf{q}^T\mathbf{u}_f + c \\ &\text{subject to} \quad \mathbf{A}\mathbf{u}_f \preceq \mathbf{b}, \end{aligned} \tag{4}$$

where $\mathbf{T} \in \mathcal{R}^{N\times N}$ is positive semidefinite, $\mathbf{q} \in \mathcal{R}^N$, $\mathbf{A} \in \mathcal{R}^{p\times N}$, $\mathbf{b} \in \mathcal{R}^p$, and $c \in \mathcal{R}$. $\mathbf{A}\mathbf{u}_f \preceq \mathbf{b}$ captures the constraints for $\mathbf{u}_f$.

Since there is no exact analytical solution for the constrained QP or LP, we cannot directly obtain the learning function F from the optimization problem posed above in (3), (4). However, such constrained optimization QP/LP problems may be solved efficiently by computational algorithms such as interior-point methods and active set methods. These algorithms are essentially iterative search schemes, which implies that we can adapt them to be iterative learning algorithms by incorporating experimentally measured data into each search step, as shown in [8]. We will next present this modified interior-point method that can solve the COILC iteratively with experimental data.

C. Interior-point-type ILC update law

A modified interior-point-type method [8] is used to solve the constrained optimization problem (3) iteratively.

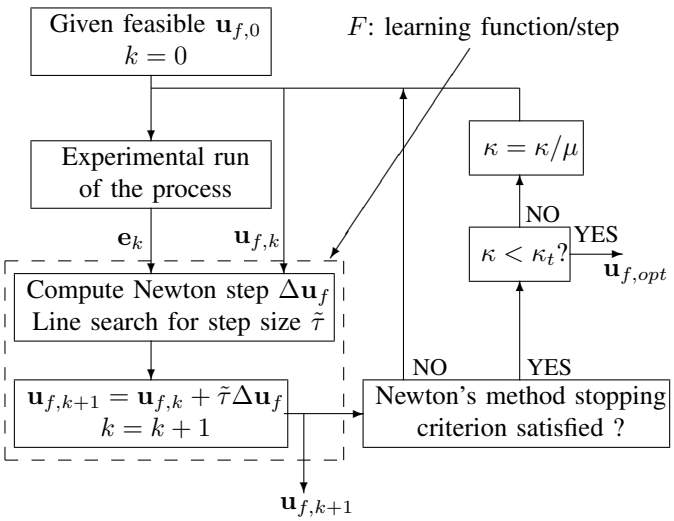

Fig. 2. Optimization process of the modified interior-point method. The Newton step in the dashed box can be seen as the learning step, which takes in $\mathbf{e}_k$ and $\mathbf{u}_{f,k}$ and generates $\mathbf{u}_{f,k+1}$.

This method uses experimental data in every iteration (i.e., every gradient search step) of the optimization process. This property makes it suitable as an ILC algorithm for a repetitive process. We here briefly introduce this method using the QP example (3), and refer the reader to [8] for details.

The typical barrier method algorithm replaces the cost function (3) subject to constraints with:

$$\min_{\mathbf{u}_f} \quad \mathbf{u}_f^T\mathbf{T}\mathbf{u}_f + \mathbf{q}^T\mathbf{u}_f + c + \kappa\phi(\mathbf{u}_f), \tag{5}$$

where $\kappa \in \mathcal{R}^+$, while ϕ is the logarithmic barrier function

$$\phi(\mathbf{x}) := -\sum_{i=1}^{m} \log(\mathbf{b}_i - \mathbf{a}_i^T\mathbf{u}_f),$$

where $\mathbf{b}_i$ and $\mathbf{a}_i^T$ are the i^{th} row of $\mathbf{b}$ and $\mathbf{A}$, respectively. By using this logarithmic barrier function, the barrier method transforms the inequality constrained problem to an unconstrained problem (5). Then, a sequence of problems as in (5) with a decreasing κ is solved using Newton's method, with initial guess of each successive optimization problem being the previous problems optimal solution. Fig. 2 shows the optimization process.

The Newton's method in the dashed block in Fig. 2 takes $\mathbf{e}_k$ and $\mathbf{u}_{f,k}$ as inputs, and generates $\mathbf{u}_{f,k+1}$ as outputs, which is similar to $F(\cdot,\cdot)$ for a typical ILC algorithm, i.e.:

$$\mathbf{u}_{f,k+1} = \mathbf{u}_{f,k} + \tilde{\tau}\Delta\mathbf{u}_{f,k},$$

where it can be seen as the learning function F. The Newton step $\Delta\mathbf{u}_{f,k}$ is determined by

$$\nabla^2 f(\mathbf{u}_{f,k})\Delta\mathbf{u}_{f,k} = -\nabla f(\mathbf{u}_{f,k}),$$

where $f(\mathbf{u}_{f,k}) = \mathbf{u}_{f,k}^T\mathbf{T}\mathbf{u}_{f,k} + \mathbf{q}^T\mathbf{u}_{f,k} + c + \kappa\phi(\mathbf{u}_{f,k})$, and:

$$\nabla f(\mathbf{u}_{f,k}) = 2\mathbf{T}\mathbf{u}_{f,k} + \mathbf{q} + \kappa\mathbf{A}^T\theta \tag{6}$$

$$\nabla^2 f(\mathbf{u}_{f,k}) = 2\mathbf{T} + \kappa\mathbf{A}^T\mathbf{diag}(\theta)^2\mathbf{A}, \tag{7}$$

where $\theta \in \mathcal{R}^P$ are given by $\theta_i = 1/(\mathbf{b}_i - \mathbf{a}_i^T\mathbf{u}_{f,k})$, while $\mathbf{diag}(\theta)$ is a diagonal matrix with ith entry θ_i. The measured error data $\mathbf{e_k}$ will be used in the calculation of $\nabla f(\mathbf{u}_{f,k})$, details will be introduced in Sec. III for different cost functions.

The step size is then chosen as:

$$\tilde{\tau} = \arg\min_{\bar{\tau}} \quad f(\mathbf{u}_{f,k} + \bar{\tau}\Delta\mathbf{u}_{f,k}).$$

With this modified barrier method, we are now able to solve the COILC problem iteratively. Next we will show the detailed construction of a family of COILC problems with non-smooth type cost functions.

III. COILC with Nonsmooth Cost Functions

Traditionally, the ℓ_2-norm of the error (and control effort) is chosen as the cost function to be minimized in NOILC problems. However, in applications where we want to reduce the peak (maximum) error, the ℓ_2-norm is not the best choice of cost function since it only captures the averaged square error characteristics of the trajectory. The peak error is essentially the ℓ_∞-norm of error. Therefore we want to explore the error performance when using ℓ_∞-norm type cost function. In many applications, both the average and peak errors have to be minimized. To address such problems, algorithm for a mixed $(\ell_2 - \ell_\infty)$-norm cost is also introduced.

A. ℓ_∞-norm with saturation constraint

The ℓ_∞-norm of the tracking error is the maximum error during the course of the entire trajectory. The ℓ_∞ error norm optimization problem is described by

$$\begin{array}{ll} \min\limits_{\mathbf{u}_f} & \|\mathbf{e}\|_\infty \\ \text{subject to} & \mathbf{e} = \mathbf{w} - \mathbf{G}_u\mathbf{u}_f \\ & \mathbf{A}\mathbf{u}_f \preceq \mathbf{b}, \end{array} \tag{8}$$

where $\mathbf{w} := (\mathbf{I} - \mathbf{G}_r)\mathbf{r} - \mathbf{G}_d\mathbf{d}$. Note that $\mathbf{w}$ can be obtained from the first iteration of the experiment. Using a feasible initial feedforward control effort $\mathbf{u}_{f,1}$, the error $\mathbf{e}_1$ can be measured. Then we can get $\mathbf{w}$ by $\mathbf{w} = \mathbf{e}_1 + \mathbf{G}_u\mathbf{u}_{f,1}$ [8]. $\mathbf{A}\mathbf{u}_f \preceq \mathbf{b}$ captures the input saturation constraint. With the lifted representation we can have $\bar{\mathbf{u}} \succeq |\mathbf{u}| = |(I+CP)^{-1}C\mathbf{r} + (I+CP)^{-1}\mathbf{u}_f - (I+CP)^{-1}CP\mathbf{d}|$, where $\bar{\mathbf{u}} = \bar{u}\mathbf{1}_N$, $\mathbf{1}_N$ is the vector of 1s of length N. So this constraint can be rewritten in the linear inequality constraint form $\mathbf{A}\mathbf{u}_f \preceq \mathbf{b}$, where $\mathbf{A} = \left[\tilde{\mathbf{A}}^T, -\tilde{\mathbf{A}}^T\right]^T$, $\mathbf{b} = \left[\mathbf{b}_1{}^T, \mathbf{b}_2{}^T\right]^T$ and $\tilde{\mathbf{A}} = (I+CP)^{-1}$, $\mathbf{b}_1 = \bar{\mathbf{u}} - \mathbf{w}_u$ and $\mathbf{b}_2 = \bar{\mathbf{u}} + \mathbf{w}_u$. $\mathbf{w}_u$ is also determined from the first iteration of the process, $\mathbf{w}_u = \mathbf{u}_1 - (I+CP)^{-1}\mathbf{u}_{f,1}$.

Let $\|\mathbf{e}\|_\infty = \tau$, We can reformulate (8) into an LP [14].

$$\begin{array}{ll} \min\limits_{\bar{\mathbf{u}}_f} & \left[\mathbf{0}_N^T\ \mathbf{1}\right]\bar{\mathbf{u}}_f \\ \text{subject to:} & \mathbf{A}_e\bar{\mathbf{u}}_f \preceq \mathbf{b}_e \\ & \mathbf{A}\mathbf{u}_f \preceq \mathbf{b} \end{array}, \tag{9}$$

where

$$A_e = \begin{bmatrix} -\mathbf{G}_u & -\mathbf{1}_N \\ \mathbf{G}_u & -\mathbf{1}_N \end{bmatrix}, \mathbf{b}_e = \begin{bmatrix} -\mathbf{w} \\ \mathbf{w} \end{bmatrix}, \bar{\mathbf{u}}_f = \begin{bmatrix} \mathbf{u}_f \\ \tau \end{bmatrix}.$$

Rewriting the inequality (saturation) constraint using the augmented vector $\bar{\mathbf{u}}_f$ instead of $\mathbf{u}_f$ and combining the two constraints, we can formulate the ILC design with ℓ_∞-norm objective function as a constrained LP.

$$\textbf{LP1} \quad \boxed{\begin{array}{ll} \min\limits_{\bar{\mathbf{u}}_f} & \left[\mathbf{0}_N^T \ \mathbf{1}\right] \bar{\mathbf{u}}_f \quad \text{subject to} \\ & \bar{\mathbf{A}}\bar{\mathbf{u}}_f \preceq \bar{\mathbf{b}}, \end{array}} \tag{10}$$

where $\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0}_{2N} \\ \mathbf{A}_e & \end{bmatrix}$, $\bar{\mathbf{b}} = \begin{bmatrix} \mathbf{b} \\ \mathbf{b}_e \end{bmatrix}$.

This is a specific case of the general form (4). Recalling (6),

$$\nabla f(\bar{\mathbf{u}}_{f,k}) = \left[\mathbf{0}_N^T \ \mathbf{1}\right]^T + \kappa \mathbf{A}^T \theta \tag{11}$$

Although there seems no $\mathbf{e}_k$ term in (6), θ is computed from $\bar{\mathbf{A}}$, $\bar{\mathbf{b}}$, where $\bar{\mathbf{b}}$ contains $\mathbf{b}_e$, which is constructed by $\mathbf{w}$. For example, for one of the constraints in $\bar{\mathbf{A}}\bar{\mathbf{u}}_f \preceq \bar{\mathbf{b}}$, i.e., $[-\mathbf{G}_u \ -\mathbf{1}_N] \begin{bmatrix} \mathbf{u}_f \\ \tau \end{bmatrix} \preceq [-\mathbf{w}]$, θ_i's expression can be showed to include the tracking error $\mathbf{e}_k(i)$, i.e.,

$$\theta_i = 1/(-\mathbf{w}_i + \mathbf{G}_{u,i}\mathbf{u}_f + \tau_i) = 1/(\tau_i - \mathbf{e}_k(i)). \tag{12}$$

Thus we can combine measured error $\mathbf{e}_k$ in the computation of $\nabla f(\bar{\mathbf{u}}_{f,k})$ in (6).

B. Mixed $\ell_2 - \ell_\infty$-norm cost with saturation constraint

Since the ℓ_∞-norm of error only accounts for the *peak* error in the optimization process, the *average* error may be large. To reduce both the average and the peak error, both the ℓ_∞ and ℓ_2 norms must be minimized simultaneously.

The most common method to accomplish this is by using the weighted form of these two norms. Recall (8), $\frac{1}{2}\mathbf{e}^T\mathbf{e} = \frac{1}{2}\mathbf{u}_f^T\mathbf{G}_u^T\mathbf{G}_u\mathbf{u}_f - \mathbf{w}^T\mathbf{G}_u\mathbf{u}_f + \mathbf{w}^T\mathbf{w}$. The optimization problem can be then written as

$$\begin{array}{ll} \min\limits_{\mathbf{u}_f} & \frac{1}{2}\mathbf{u}_f^T\mathbf{G}_u^T\mathbf{G}_u\mathbf{u}_f - \mathbf{w}^T\mathbf{G}_u\mathbf{u}_f + \alpha\left[\mathbf{0}_N^T \ \mathbf{1}\right]\bar{\mathbf{u}}_f \\ \text{subject to} & \bar{\mathbf{A}}\bar{\mathbf{u}}_f \preceq \bar{\mathbf{b}}. \end{array} \tag{13}$$

Letting $\bar{\mathbf{G}}_u = [\mathbf{G}_u \ \mathbf{0}_N]$, we can rewrite (13) as a constrained QP

$$\textbf{QP2} \quad \boxed{\begin{array}{ll} \min\limits_{\bar{\mathbf{u}}_f} & \frac{1}{2}\bar{\mathbf{u}}_f^T\bar{\mathbf{G}}_u^T\bar{\mathbf{G}}_u\bar{\mathbf{u}}_f - \mathbf{w}^T\bar{\mathbf{G}}_u\bar{\mathbf{u}}_f \\ & +\alpha\left[\mathbf{0}_N^T \ \mathbf{1}\right]\bar{\mathbf{u}}_f \\ \text{subject to} & \bar{\mathbf{A}}\bar{\mathbf{u}}_f \preceq \bar{\mathbf{b}}. \end{array}} \tag{14}$$

Comparing the part before plus sign in (14) with the general form (3), $\mathbf{T} = \frac{1}{2}\bar{\mathbf{G}}_u^T\bar{\mathbf{G}}_u$, $\mathbf{q} = -\bar{\mathbf{G}}_u^T\mathbf{w}$, $c = 0$, $\mathbf{A} = \bar{\mathbf{A}}$, $\mathbf{b} = \bar{\mathbf{b}}$.

Recalling (6),

$$\begin{aligned} \nabla f(\bar{\mathbf{u}}_{f,k}) &= \bar{\mathbf{G}}_u^T\bar{\mathbf{G}}_u\bar{\mathbf{u}}_{f,k} - \bar{\mathbf{G}}_u^T\mathbf{w} + \kappa\bar{\mathbf{A}}^T\theta \\ &= \bar{\mathbf{G}}_u^T(\bar{\mathbf{G}}_u\bar{\mathbf{u}}_{f,k} - \mathbf{w}) + \kappa\bar{\mathbf{A}}^T\theta \\ &= -\bar{\mathbf{G}}_u^T\mathbf{e}_k + \kappa\bar{\mathbf{A}}^T\theta. \end{aligned} \tag{15}$$

So the first part's contribution to descent direction can be expressed by the measured error data $\mathbf{e}_k$ [8], the second part's contribution to the descent direction also includes $\mathbf{e}_k$ (12). Hence, we obtain the ILC-like update law.

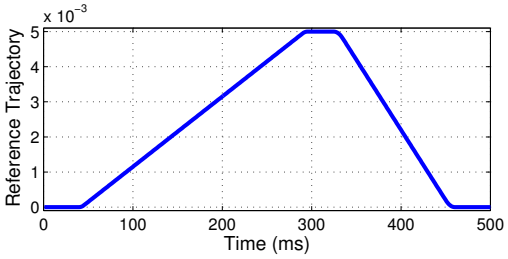

Fig. 3. Plot of the reference position $r(t)$ versus time t.

IV. Results

We here show the results of implementation of the COILC algorithms derived above on a precision motion control system. The plant $\mathbf{P}$, which is the model of the motor stage, and the controller $\mathbf{C}$ are characterized by the transfer functions:

$$P(s) = \frac{3.194}{s^2 + 16.63s}, C(s) = 1000 \cdot \frac{31s^2 + 1010s + 1000}{s^2 + 1000s},$$

the input is a current (Amp) while the system output is a position (m). The plant and controller are sampled at time step $T_s = 0.001$s. The input saturation is set at 1 Amp. The reference trajectory to be followed is shown in Fig. 3.

A. COILC using ℓ_∞-norm and saturation

This section presents the simulation results from implementing COILC using ℓ_∞-norm and saturation (as in Sec. III-A). The first plot in Fig. 7 shows that the maximum error (ℓ_∞-norm of error) when using ℓ_∞-norm as cost function converges in 25 iterations. The solid line in Fig. 4 shows the converged error. The converged error when using ℓ_2-norm as cost function is also showed as the dashed line for comparison. We observe that ℓ_∞-norm does indeed decrease the peak error, which is reduced from $6.433 \cdot 10^{-5}m$ to $4.883 \cdot 10^{-5}m$, but the RMS error is larger than that of using ℓ_2-norm as cost function. The RMS error when using ℓ_∞-norm as cost function is $1.492 \cdot 10^{-5}$, while, for ℓ_2-norm case, the RMS error is $9.965 \cdot 10^{-6}m$. Therefore, we want to combine ℓ_∞-norm and ℓ_2-norm together to reduce both RMS value and maximum value of error.

B. COILC using mixed $\ell_2 - \ell_\infty$-norm and saturation

This section shows the simulation results from implementing COILC using the weight mixed $\ell_2 - \ell_\infty$-norm and saturation (as in Sec. III-B). Fig. 6 shows the comparison of results using different α (weighting coefficient on ℓ_∞-norm) (14) in the mixed $\ell_2 - \ell_\infty$-norm. We observe that as α increases, the ℓ_∞-norm moves the peak downward to decrease the peak error, while the RMS error will increase as showed in Tab. I.

The second plot in Fig. 7 shows the convergence of weighted cost when using mixed $\ell_2 - \ell_\infty$-norm as cost function (α is chosen as 0.04). We observe that the weighted

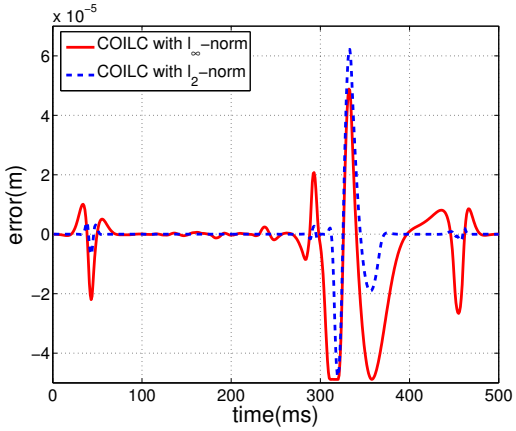

Fig. 4. Plot of the converged error versus time using ℓ_2-norm and ℓ_∞-norm. ℓ_∞-norm do decrease the peak error, but other parts are not as good as the ℓ_2-norm.

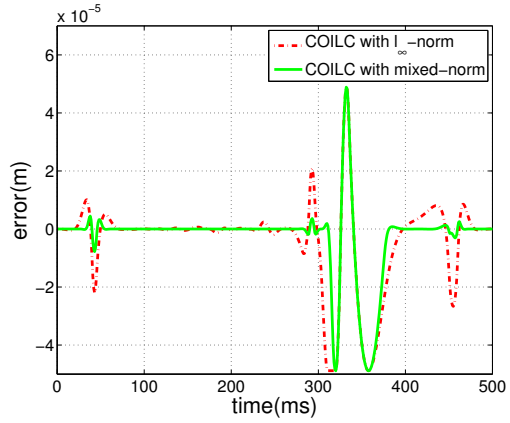

Fig. 5. Plot of the converged error versus time using ℓ_∞-norm and mixed $\ell_2 - \ell_\infty$-norm. The mixed norm in (III-B) takes cares of both the RMS and the maximum value of error, We can see that the peak error is as small as the results only using ℓ_∞ norm. And at other parts, the error performance are also good.

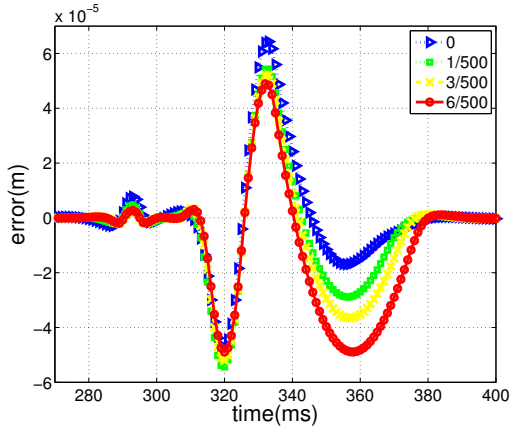

Fig. 6. Plot of the converged error versus time (from 270ms to 400ms) using mixed $\ell_2 - \ell_\infty$-norm with different coefficient $\alpha = \{0, 0.002, 0.006, 0.012\}$. We can see that as α increases, the ℓ_∞-norm moves the peak downward to decrease the peak error, while sacrificing ℓ_2-norm performance.

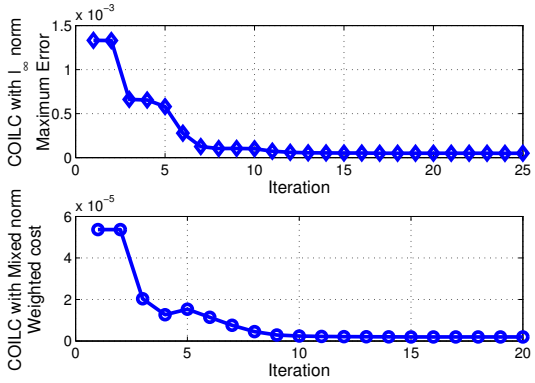

Fig. 7. Plots of the maximum error (ℓ_∞-norm of error) versus iteration when using ℓ_∞-norm cost and the weighted cost versus iteration when using mixed $\ell_2 - \ell_\infty$-norm cost.

Norm	RMS error (m)	Peak error (m)
ℓ_2-norm	$9.965 \cdot 10^{-6}$	$6.433 \cdot 10^{-5}$
ℓ_∞-norm	$1.492 \cdot 10^{-5}$	$4.883 \cdot 10^{-5}$
mixed norm (0.002)	$1.041 \cdot 10^{-5}$	$5.438 \cdot 10^{-5}$
mixed norm (0.006)	$1.140 \cdot 10^{-5}$	$5.103 \cdot 10^{-5}$
mixed norm (0.012)	$1.251 \cdot 10^{-5}$	$4.894 \cdot 10^{-5}$
mixed norm (0.04)	$1.263 \cdot 10^{-5}$	$4.888 \cdot 10^{-5}$

TABLE I

SUMMARY OF ERROR NORMS FOR DIFFERENT COST FUNCTIONS IN THE CO-ILC PROBLEM FORMULATION.

error converges in 20 iterations. Fig. 5 shows the comparison of the converged error when using mixed-norm as cost function and ℓ_∞-norm as cost function. The mixed-norm reduces both the RMS error and the peak error, We observe that the peak error is as small as when only using ℓ_∞ norm as cost function, the peak error is $4.888 \cdot 10^{-5}m$. The average error is also reduced compared with the ℓ_∞-norm case, the RMS error is $1.263 \cdot 10^{-5}m$. Thus, mixed-norm cost functions can generate good error performance in terms of both RMS error and peak error.

For the calculation cost, taking the mixed $\ell_2 - \ell_\infty$-norm for example, the calculation time of each iteration takes 0.564 to 4.451 seconds, and the average time is 2.184 seconds.

C. Experiment Results

The COILC using ℓ_∞-norm and saturation (as in Sec. III-A) is also implemented on the motion control system experimentally and compared with the case using ℓ_2-norm and saturation. Fig. 8 shows that ℓ_∞-norm does indeed decrease the peak error, which is reduced from $2.478 \cdot 10^{-4}m$ to $1.943 \cdot 10^{-4}m$. The RMS error is increased, comparing with that of using ℓ_2-norm as cost function, from $6.078 \cdot 10^{-5}m$ to $6.930 \cdot 10^{-5}m$. We can observe that the experiment verifies our simulation result.

D. Robustness Evaluation

Since the performance of CO-ILC algorithms is reliant on a good model, we now present an evaluation of their

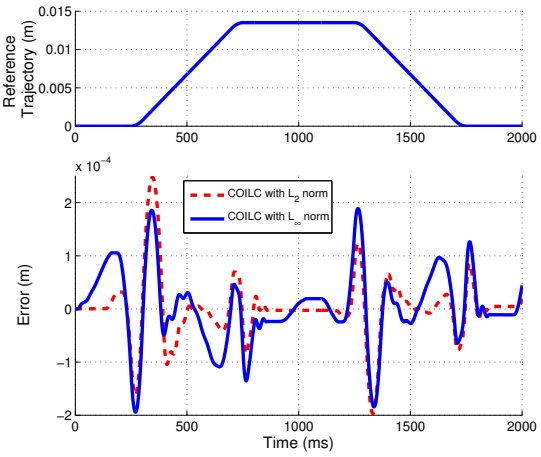

Fig. 8. Reference trajectory used in experiment and the error profile comparison of COILC using ℓ_∞-norm and ℓ_2-norm. ℓ_∞-norm do decrease the peak error, but other parts are not as good as the ℓ_2-norm.

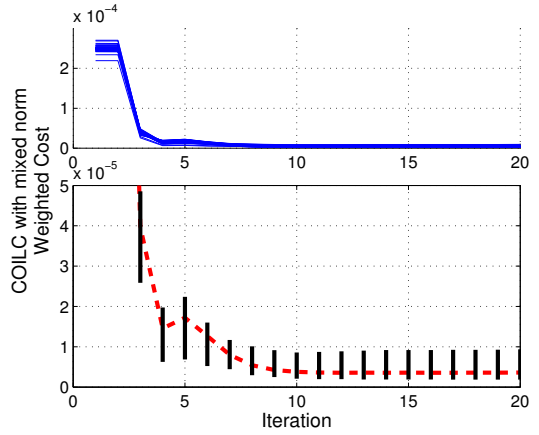

Fig. 9. Plot of the weighted cost versus iteration number for 50 randomly changed models (parametric model mismatch up to 60%) using mixed $\ell_2 - \ell_\infty$-norm. The second plot is the zoom-in plot of the first plot, the dashed line is the average evolution of 50 simulation runs. The horizontal line shows the range of the cost function value for different models at each iteration.

robustness to modeling uncertainty. Using the optimization process of mixed $\ell_2 - \ell_\infty$-norm in III-B as example. We add parametric model uncertainty to the plant $P(s) = C(sI - A)^{-1}B + D$ by randomly changing each element of matrices A, B, C, D within 60 percentage every simulation run and run 50 times. Fig. 9 shows the evolution and convergence of the cost function over 20 iterations for these model mismatch cases. It is observed that with upto 60% parametric model mismatch, the cost function converges. Since the control input saturation's influence on the error profile is different with different models, the converged cost function values are different (though similar). This demonstrates that the proposed method has good robustness to parametric uncertainty, specifically for the motion control system under study. In order to present analytical guarantees of robustness, rigorous robustness proofs for the COILC algorithms are currently being investigated by the authors.

V. CONCLUSIONS

This paper introduces constrained optimal ILC with non-smooth cost functions. Specifically, we derive algorithms for (a) a pure ℓ_∞-norm cost, (b) a mixed $\ell_2 - \ell_\infty$-norm cost; and compare the results to the traditional ℓ_2-norm (NOILC). It was found that using mixed weighted norm of $\ell_2 - \ell_\infty$ resulted in better overall trajectory tracking; with multiple objectives such as peak error and RMS error minimization addressed simultaneously. With this COILC framework, future work will include experimental validation for the mixed $\ell_2 - \ell_\infty$-norm case and a rigorous robustness analysis of the proposed algorithms. The use of other optimization algorithms and computationally effective techniques instead of the interior-point-method will also be investigated.

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation Career Award grant CMMI-1254313 and in part by the Center for Automation Technologies and Systems (CATS) under a block grant from the New York State Empire State Development Division of Science, Technology and Innovation (NYSTAR).

REFERENCES

[1] M. Togai and O. Yamano, "Analysis and design of an optimal learning control scheme for industrial robots: A discrete system approach," in *Decision and Control, 1985 24th IEEE Conference on*, vol. 24. IEEE, 1985, pp. 1399–1404.

[2] N. Amann and D. H. Owens, "Iterative learning control for discrete time systems using optimal feedback and feedforward actions," in *Proceedings of the 34th Conference on Decision and Control*, New Orleans, LA, July 1995.

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

[4] D. H. Owens and J. Hatonen, "Iterative learning control – an optimization paradigm," *Annual Reviews in Control*, vol. 29(1), pp. 57–70, 2005.

[5] D. Owens, J. Hatonen, and S. Daley, "Robust monotone gradient-based discrete-time iterative learning control," *International Journal of Robust and Nonlinear Control*, vol. 19, no. 6, pp. 634–661, 2009.

[6] K. L. Barton and A. G. Alleyne, "A cross-coupled iterative learning control design for precision motion control," *Control Systems Technology, IEEE Transactions on*, vol. 16, no. 6, pp. 1218–1231, 2008.

[7] I. Lim and K. L. Barton, "Pareto iterative learning control: Optimized control for multiple performance objectives," *Control Engineering Practice*, vol. 26, pp. 125–135, 2014.

[8] S. Mishra, U. Topcu, and M. Tomizuka, "Optimization-based constrained iterative learning control," *Control Systems Technology, IEEE Transactions on*, vol. 19, no. 6, pp. 1613–1621, 2011.

[9] ——, "Iterative learning control with saturation constraints," in *American Control Conference, 2009. ACC'09.* IEEE, 2009, pp. 943–948.

[10] C. T. Freeman and Y. Tan, "Iterative learning control with mixed constraints for point-to-point tracking," *Control Systems Technology, IEEE Transactions on*, vol. 21, no. 3, pp. 604–616, 2013.

[11] M. Volckaert, M. Diehl, and J. Swevers, "Generalization of norm optimal ilc for nonlinear systems with constraints," *Mechanical Systems and Signal Processing*, vol. 39, no. 1, pp. 280–296, 2013.

[12] E. Rogers and D. H. Owens, *Stability Analysis for Linear Repetitive Processes*. Berlin: Springer-Verlag, 1992.

[13] D. Bristow, M. Tharayil, and A. Alleyne, "A survey of iterative learning control," *Control Systems Magazine, IEEE*, vol. 26, no. 3, pp. 96–114, June 2006.

[14] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge Univ. Press, 2004.