Model-free Gradient Iterative Learning Control for Non-linear Systems

B. Huo * **C. T. Freeman** **

* *School of Electrical Engineering, Zhengzhou University, 100 Science Avenue, Zhengzhou, Henan, 450001 China (e-mail: B.Huo@soton.ac.uk).*
** *Electronics and Computer Science, University of Southampton, Southampton, UK, SO17 1BJ (e-mail: cf@ecs.soton.ac.uk)*

Abstract: Iterative learning control (ILC) is a well-established approach to precision tracking for systems that perform a repeated task. Gradient-based update laws are amongst the most widely applied in practice due to their attractive robustness properties. However, they are limited by requiring a model of the system dynamics to be identified. This paper shows how gradient ILC can be extended for use with a general class of nonlinear systems, and additionally how the update can be generated using an extra experiment conducted between trials. This 'model-free' approach extends previous work for the linear systems, and is illustrated in a rehabilitation application requiring accurate control of human upper-limb movement.

Keywords: iterative learning control, non-linear systems, stroke rehabilitation, optimisation

1. INTRODUCTION

Iterative learning control (ILC) is an approach used to facilitate accurate tracking of tasks that are executed repeatedly over a finite time interval (Bristow et al., 2006). ILC uses the tracking error over past attempts to modify the control signal for the next attempt, with the aim of ultimately converging to zero error. ILC has found rich application to industrial tasks such as robotic systems (Adloo et al., 2009; Barton and Alleyne, 2011a), chemical batch processing (Mezghani et al., 2002), servo systems (Shaojuan et al., 2008; Zheng et al., 2009) and rehabilitation (Freeman, 2016). See (Bristow et al., 2006; Ahn et al., 2007) for a summary and categorisation of algorithms.

A well established algorithmic framework has emerged for the class of gradient based algorithms whose convergence and robustness properties have been extensively studied by many groups, including Bristow (2008); Ratcliffe et al. (2006); van de Wijdeven et al. (2009); Wang et al. (2010); Barton et al. (2008); Mishra and Tomizuka (2005); Freeman et al. (2012); Butcher et al. (2008b). A prominent member of this class is Norm Optimal ILC (NOILC) which has received considerable attention in the ILC community due to its mature theoretic basis Buchheit et al. (1994); Amann et al. (1996). The framework has been applied to a range of systems including gantry robots Ratcliffe et al. (2006), multi-axis robotic testbeds Barton and Alleyne (2011b), rehabilitation platforms Rogers et al. (2010), lasers Rogers et al. (2010) and pneumatic muscle actuators Schindele and Aschemann (2011). Extensions have been proposed using a predictive mechanism Bristow and Alleyne (2006), constraints Chu and Owens (2010), projections Chu and Owens (2009), and accelerated learning Owens and Chu (2009). Another well established algorithm is 'gradient ILC', also termed 'adjoint ILC' which has been studied by many groups, including Owens et al. (2009); Owens and Feng (2003); Jian-Xu and Ji (1998); Furuta et al. (1991).

The majority of ILC methods require models of the plant dynamics, such as frequency-domain ILC in Norrlof (2000) and Freeman et al. (2009b), inverse model approach in Harte et al. (2005), gradient approach in Hätönen et al. (2004) and Hätönen et al. (2006), norm optimal approach in Dinh et al. (2013). The convergence conditions and performance of these ILC methods are built on the nominal model and uncertain properties of the controlled system. The nominal model plays a key role in the design of model-based ILC methods. However, for some cases, nominal models are too difficult or/and expensive to obtain, such as multivariate systems and musculoskeletal system which is nonlinear and lack of sensing methods. What's more, the robustness will lead to a poor performance when the uncertainty is too large.

To remove the restrictions of model and uncertainty, a 'model-free' concept is first proposed

⋆ Funded by the Zhengzhou University-Southampton Collaborative Research Project under Grant 16306/01.

in Freeman (2004), in which an additional, non-standard experiment at each trial is applied to the real system in order to estimate the gradient. Butcher et al. (2008a) figured out that though the additional experiment will have its own disturbance, an unbiased estimate of the gradient can still be found. The model-free gradient ILC (MFILC) approach will feed the time reversed error signal into the real system as its input and use the reversed output as the gradient to generate the control signal for the next trial. This time reverse operation requires the lifted matrix of the system to be symmetric as shown in Bolder et al. (2018) which extended MFILC to multivariate system and presented a rigorous analysis of this method. The convergence conditions and connections with common pre-existing ILC algorithms were given. However, the symmetric lifted system matrix indicates that the system should be linear or weak nonlinear which has little effect. To extend this method to nonlinear systems, this paper proposes a MFILC method which discrete the system at a fixed working point by introducing another experiment. Along with the algorithm convergence, the input signal will converge, thus the experiment for discretization only need to be executed few times in a trial.

The main contributions of this paper are as following. A model-free gradient ILC for nonlinear system is proposed for the first time. A scale parameter is introduced to enhance the precision. The convergence conditions are given rigorously. The proposed method is verified via Simulations on the real data from human arm musculoskeletal system, the comparisons between our method and linear MFILC and conventional gradient ILC method are carried out.

The paper is arranged as following. The nonlinear gradient ILC algorithm is presented in the second part. Then the model-free gradient ILC method is set up on the concept of time inverse. We give the convergence conditions of the proposed method at part 4. Our method is supported by simulation results on the two-muscle elbow systems, which is presented in the section 5. At the last, the conclusions and ongoing researches are given.

2. GRADIENT ILC FOR NONLINEAR SYSTEMS

In this section, we construct the gradient iterative learning control algorithm for nonlinear systems.

Consider the following general nonlinear discrete-time system

$$\begin{aligned} x(i+1) &= f\left(x(i), u(i)\right), \quad x(0) = x_0, \\ y(i) &= h\left(x(i), u(i)\right), \quad i = 0, 1, 2, \ldots, N \end{aligned} \tag{1}$$

with state vector $x \in \mathbb{R}^n$. Here $f(\cdot)$ and $h(\cdot)$ are continuously differentiable with respect to their arguments. The relationship between the input and output time-series can be expressed by the following algebraic functions

$$\begin{aligned} y(0) &= h(x(0), u(0)) = g_0(x(0), u(0)) \\ y(1) &= h(x(1), u(1)) = h(f(x(0), u(0)), u(1)) \\ &:= g_1(x(0), u(0), u(1)) \\ &\vdots \\ y(N) &= h(x(N), u(N)) \\ &= h(f(x(N-1), u(N-1)), u(N)) \\ &:= g_N(x(0), u(0), u(1), \ldots, u(N)) \end{aligned} \tag{2}$$

so that the system (1) can be represented by the vector function

$$\boldsymbol{y} = \begin{bmatrix} g_0(x(0), u(0)) \\ g_1(x(0), u(0), u(1)) \\ g_2(x(0), u(0), u(1), u(2)) \\ \vdots \\ g_N(x(0), u(0), u(1), \ldots, u(N)) \end{bmatrix}^\top := \boldsymbol{g}(\boldsymbol{u})$$

where the supervectors,

$$\boldsymbol{u} = [u(0), u(1), \ldots, u(N)]^\top \quad \in \mathbb{R}^{mN} \tag{3}$$

$$\boldsymbol{y} = [y(0), y(1), \ldots, y(N)]^\top \quad \in \mathbb{R}^{pN} \tag{4}$$

and the reference

$$\boldsymbol{r} = [r(0), r(1), \ldots, r(N)]^\top \quad \in \mathbb{R}^{pN} \tag{5}$$

then we can write

$$\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{u}) \tag{6}$$

The term $\bigtriangledown \boldsymbol{g}(\bar{\boldsymbol{u}})$ is the system linearization around an operating point $\bar{\boldsymbol{u}}$, where

$$\bigtriangledown \boldsymbol{g}(\boldsymbol{u}) = \begin{bmatrix} \frac{\partial g_0}{\partial u(0)} & \frac{\partial g_0}{\partial u(1)} & \cdots & \frac{\partial g_0}{\partial u(N)} \\ \frac{\partial g_1}{\partial u(0)} & \frac{\partial g_1}{\partial u(1)} & \cdots & \frac{\partial g_1}{\partial u(N)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_N}{\partial u(0)} & \frac{\partial g_N}{\partial u(1)} & \cdots & \frac{\partial g_N}{\partial u(N)} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\partial g_0}{\partial u(0)} & 0 & 0 & \ldots & 0 \\ \frac{\partial g_1}{\partial u(0)} & \frac{\partial g_1}{\partial u(1)} & 0 & \ldots & 0 \\ \frac{\partial g_2}{\partial u(0)} & \frac{\partial g_2}{\partial u(1)} & \frac{\partial g_2}{\partial u(2)} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_N}{\partial u(0)} & \frac{\partial g_N}{\partial u(1)} & \frac{\partial g_N}{\partial u(2)} & \cdots & \frac{\partial g_N}{\partial u(N)} \end{bmatrix} \tag{7}$$

The linearized system dynamics then equates to the linear time-varying system

$$\begin{aligned} x(i+1) &= A(i)x(i) + B(i)\tilde{u}(i), \quad x(0) = x_0, \\ \tilde{y}(i) &= C(i)x(i) + D(i)\tilde{u}(i), \quad i = 0, 1, 2, \ldots, N \end{aligned} \tag{8}$$

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$, $D \in \mathbb{R}^{p \times m}$, $\tilde{u}(i) \in \mathbb{R}^m$ and $\tilde{y}(i) \in \mathbb{R}^p$. In particular

$$A(i) = \left.\frac{\partial f}{\partial x}\right|_{\substack{u(i)=\bar{u}(i)\\ x(i)=\bar{x}(i)}} \quad B(i) = \left.\frac{\partial f}{\partial u}\right|_{\substack{u(i)=\bar{u}(i)\\ x(i)=\bar{x}(i)}},$$
$$C(i) = \left.\frac{\partial h}{\partial x}\right|_{\substack{u(i)=\bar{u}(i)\\ x(i)=\bar{x}(i)}}, \quad D(i) = \left.\frac{\partial h}{\partial u}\right|_{\substack{u(i)=\bar{u}(i)\\ x(i)=\bar{x}(i)}} \tag{9}$$

in which $\bar{\boldsymbol{x}}$ is generated by running $\bar{\boldsymbol{u}}$ through system (1).

Now apply the gradient method to minimise $J(\boldsymbol{u}) := \|\boldsymbol{e}\|^2, \boldsymbol{e} = \boldsymbol{r} - \boldsymbol{g}(\boldsymbol{u})$, subject to dynamics (1). Here $\nabla J(\boldsymbol{u}) = -\bigtriangledown \boldsymbol{g}(\boldsymbol{u})^\top \boldsymbol{e}$ and the update is

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \beta \bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k)^\top \boldsymbol{e}_k \tag{10}$$

where β is a positive definite scalar step length. Thus, we get the gradient ILC update law for nonlinear systems.

3. MODEL-FREE GRADIENT ILC FOR NONLINEAR SYSTEMS

Precise model is always not easy or too expensive to obtain, especially for nonlinear systems. In the following, we will present how to remove the model by experiments for nonlinear systems, which will naturally lead to a model-free algorithm.

To apply our model-free approach, we must exchange (10) for

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \beta \bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)^\top \boldsymbol{e}_k \tag{11}$$

where $\hat{\boldsymbol{u}}_k$ is a fixed operating point chosen to be 'near' $\boldsymbol{u}_k$. Hence the optimal choice is

$$\hat{\boldsymbol{u}}_k(i) = \hat{u}_k, \qquad \hat{u}_k := \min_{\hat{u}_k \in \mathbb{R}} \|\boldsymbol{u}_k - \hat{\boldsymbol{u}}_k\|^2$$

with solution

$$\hat{u}_k = \frac{1}{N}\sum u_k(i) \tag{12}$$

This produces a steady-state value (to ensure $\hat{\boldsymbol{x}}$ has identical elements, $\hat{\boldsymbol{x}}_k(i) = \hat{\boldsymbol{u}}_k \ \forall i$). Hence

$$A(i) = \left.\frac{\partial f}{\partial x}\right|_{\substack{u(i)=\hat{u}\\ x(i)=\hat{\boldsymbol{u}}_k}} := \hat{A}_k, \ B(i) = \left.\frac{\partial f}{\partial u}\right|_{\substack{u(i)=\hat{u},\\ x(i)=\hat{\boldsymbol{u}}_k}} := \hat{B}_k,$$
$$C(i) = \left.\frac{\partial h}{\partial x}\right|_{\substack{u(i)=\hat{u}\\ x(i)=\hat{\boldsymbol{u}}_k}} := \hat{C}_k, \ D(i) = \left.\frac{\partial h}{\partial u}\right|_{\substack{u(i)=\hat{u}\\ x(i)=\hat{\boldsymbol{u}}_k}} := \hat{D}_k \tag{13}$$

and it follows that $\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k) =$

$$\begin{bmatrix} \hat{D}_k & 0 & 0 & \cdots & 0 \\ \hat{C}_k\hat{B}_k & \hat{D}_k & 0 & \cdots & 0 \\ C\hat{A}_k\hat{B}_k & \hat{C}_k\hat{B}_k & \hat{D}_k & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hat{C}_k\hat{A}_k^{N-1}\hat{B}_k & \hat{C}_k\hat{A}_k^{N-2}\hat{B}_k & \hat{C}_k\hat{A}_k^{N-3}\hat{B}_k & \cdots & \hat{D}_k \end{bmatrix} \tag{14}$$

$\in \mathbb{R}^{(p\times m)N}$. To compute $\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)^\top \boldsymbol{e}_k$ in (11), write the Taylor expansion of $\boldsymbol{g}(\boldsymbol{u})$ around $\hat{\boldsymbol{u}}_k$ as

$$\boldsymbol{g}(\boldsymbol{u}) = \boldsymbol{g}(\hat{\boldsymbol{u}}_k) + \bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)(\boldsymbol{u} - \hat{\boldsymbol{u}}_k) \tag{15}$$

Now select $\boldsymbol{u} = \hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k$. For a sufficiently small $\alpha > 0$, we can then approximate with arbitrary precision as

$$\boldsymbol{g}(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k) \approx \boldsymbol{g}(\hat{\boldsymbol{u}}_k) + \bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k - \hat{\boldsymbol{u}}_k) \tag{16}$$

so that

$$\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)\alpha\tilde{\boldsymbol{e}}_k \approx \boldsymbol{g}(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k) - \boldsymbol{g}(\hat{\boldsymbol{u}}_k) \tag{17}$$

$$\Rightarrow \bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)\tilde{\boldsymbol{e}}_k \approx \frac{1}{\alpha}\left(\boldsymbol{g}(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k) - \boldsymbol{g}(\hat{\boldsymbol{u}}_k)\right) \tag{18}$$

Now let $\tilde{\boldsymbol{s}}$ denote the time reversal of vector $\boldsymbol{s}$, i.e.

$$\boldsymbol{s} = [s(0), s(1), \ldots, s(N)]^\top \quad \in \mathbb{R}^{aN}, \tag{19}$$

$$\tilde{\boldsymbol{s}} = [s(N), s(N-1), \ldots, s(0)]^\top \quad \in \mathbb{R}^{aN}, \tag{20}$$

$s(i) \in \mathbb{R}^a$. We can then write $\overbrace{\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)\tilde{\boldsymbol{e}}_k}$ =

$$\overbrace{\begin{bmatrix} \hat{D}_k & 0 & \cdots & 0 \\ \hat{C}_k\hat{B}_k & \hat{D}_k & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{C}_k\hat{A}_k^{N-1}\hat{B}_k & \hat{C}_k\hat{A}_k^{N-2}B & \cdots & \hat{D}_k \end{bmatrix} \begin{bmatrix} e_k(N) \\ e_k(N-1) \\ \vdots \\ e_k(0) \end{bmatrix}}$$

$$= \overbrace{\begin{bmatrix} \hat{D}_k e_k(N) \\ \hat{D}_k e_k(N-1) + \hat{C}_k\hat{B}_k e_k(N) \\ \vdots \\ \hat{D}_k e_k(1) + \cdots + \hat{C}_k\hat{A}_k^{N-2}\hat{B}_k e_k(N) \\ \hat{D}_k e_k(0) + \cdots + \hat{C}_k\hat{A}_k^{N-1}\hat{B}_k e_k(N) \end{bmatrix}}$$

$$= \begin{bmatrix} \hat{D}_k e_k(0) + \cdots + \hat{C}_k A^{N-1}\hat{B}_k e_k(N) \\ \hat{D}_k e_k(1) + \cdots + \hat{C}_k\hat{A}_k^{N-2}\hat{B}_k e_k(N) \\ \vdots \\ \hat{D}_k e_k(N-1) + \hat{C}_k\hat{B}_k e_k(N) \\ \hat{D}_k e_k(N) \end{bmatrix} \tag{21}$$

$$= \begin{bmatrix} \hat{D}_k & \hat{C}_k\hat{B}_k & \cdots & \hat{C}_k\hat{A}_k^{N-1}\hat{B}_k \\ 0 & \hat{D}_k & \cdots & \hat{C}_k\hat{A}_k^{N-2}\hat{B}_k \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \hat{D}_k \end{bmatrix} \begin{bmatrix} e_k(0) \\ e_k(1) \\ \vdots \\ e_k(N) \end{bmatrix} \tag{22}$$

$= \bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)^\top \boldsymbol{e}_k$. Combine with (17) to produce

$$\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)^\top \boldsymbol{e}_k = \overbrace{\frac{1}{\alpha}\left(\boldsymbol{g}(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k) - \boldsymbol{g}(\hat{\boldsymbol{u}}_k)\right)} \tag{23}$$

$$= \frac{1}{\alpha}\overbrace{\left(\boldsymbol{g}(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k) - \boldsymbol{g}(\hat{\boldsymbol{u}}_k)\right)} \tag{24}$$

$$= \frac{1}{\alpha}\left(\overbrace{\boldsymbol{g}(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k)} - \overbrace{\boldsymbol{g}(\hat{\boldsymbol{u}}_k)}\right) \tag{25}$$

where we select α such that the approximation (17) is accurate. This produces the nonlinear model-free ILC update as:

(1) Apply input $\boldsymbol{u}_k$ to system, and compute $\boldsymbol{e}_k$
(2) Compute $\hat{\boldsymbol{u}}_k$, the mean of input $\boldsymbol{u}_k$, and apply it to the system, measure the output $\boldsymbol{g}(\hat{\boldsymbol{u}}_k)$
(3) Time-reverse the signal $\boldsymbol{e}_k$, scale it by α, add it to the mean of $\boldsymbol{u}_k$, apply it to the system and measure the output $\boldsymbol{g}(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k)$
(4) Update the input according to equations 11 and 25.

4. CONVERGENCE ANALYSIS

Theorem 1. Let $\boldsymbol{g}$, J be Lipschitz continuous with constants M, L respectively. Suppose the ILC up-

date (11) is applied with the operating point deviation bounded by

$$\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\| \leq \frac{\| \bigtriangledown J(\boldsymbol{u}_k)\|}{M\|\boldsymbol{e}_k\|} \tag{26}$$

and the step size satisfying

$$\beta < \frac{2}{L}\left(\frac{1-c}{1+2c}\right) \tag{27}$$

with $c = \frac{\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|\|\boldsymbol{e}_k\|M}{\|\bigtriangledown J(\boldsymbol{u}_k)\|}$. Then the tracking error norm J strictly decreases with each iteration of gradient descent until it reaches the optimal value $J(\boldsymbol{u}_k) = J(\boldsymbol{u}^*)$.

Proof. First note that Lipschitz condition

$$\| \bigtriangledown J(\boldsymbol{x}) - \bigtriangledown J(\boldsymbol{y})\| \leq L\|\boldsymbol{x} - \boldsymbol{y}\| \quad \text{for any } \boldsymbol{x}, \boldsymbol{y}$$

leads to

$$\bigtriangledown^2 J(\boldsymbol{x}) \preceq LI \tag{28}$$

and similarly for M. Quadratic expansion of $J()$ around $J(\boldsymbol{x})$ gives

$$J(\boldsymbol{y}) \leq f(\boldsymbol{x}) + \bigtriangledown J(\boldsymbol{x})^\top(\boldsymbol{y} - \boldsymbol{x}) + \frac{1}{2}\bigtriangledown^2 f(\boldsymbol{x})\|\boldsymbol{y} - \boldsymbol{x}\|^2 \tag{29}$$

$$\leq f(\boldsymbol{x}) + \bigtriangledown J(\boldsymbol{x})^\top(\boldsymbol{y} - \boldsymbol{x}) + \frac{1}{2}L\|\boldsymbol{y} - \boldsymbol{x}\|^2 \tag{30}$$

Assuming the update form (11) given by

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \beta(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top \boldsymbol{e}_k \tag{31}$$

where $\hat{\boldsymbol{u}}_k$ may deviate from $\boldsymbol{u}_k$, we set $\boldsymbol{x} = \boldsymbol{u}_k$, $\boldsymbol{y} = \boldsymbol{u}_{k+1}$ to get

$$\begin{aligned} &J(\boldsymbol{u}_{k+1}) \leq J(\boldsymbol{u}_k) \\ &+ \bigtriangledown J(\boldsymbol{u}_k)^\top(\boldsymbol{u}_k + \beta(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top \boldsymbol{e}_k - \boldsymbol{u}_k) \\ &+ \frac{1}{2}L\|\boldsymbol{u}_k + \beta(\bigtriangledown g(\hat{\boldsymbol{u}}_k))^\top \boldsymbol{e}_k - \boldsymbol{u}_k\|^2 \\ &= J(\boldsymbol{u}_k) + \beta \bigtriangledown J(\boldsymbol{u}_k)^\top(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top \boldsymbol{e}_k \\ &+ \frac{L\beta^2}{2}\|(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top \boldsymbol{e}_k\|^2 \end{aligned} \tag{32}$$

Then use expansion $\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k) \leq \bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k) + M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|$ where

$$\bigtriangledown^2 \boldsymbol{g}(\boldsymbol{u}_k) \preceq MI \tag{33}$$

to give

$$\begin{aligned} &J(\boldsymbol{u}_{k+1}) \leq J(\boldsymbol{u}_k) + \beta \bigtriangledown J(\boldsymbol{u}_k)^\top(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top \boldsymbol{e}_k \\ &+ \frac{L\beta^2}{2}\|(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top \boldsymbol{e}_k\|^2 \\ &\leq J(\boldsymbol{u}_k) + \beta \bigtriangledown J(\boldsymbol{u}_k)^\top(\bigtriangledown g(\boldsymbol{u}_k))^\top \boldsymbol{e}_k \\ &+ \frac{L\beta^2}{2}\|(\bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k))^\top \boldsymbol{e}_k\|^2 \\ &+ t \bigtriangledown J(\boldsymbol{u}_k)^\top M\boldsymbol{e}_k\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\| \\ &+ \frac{L\beta^2}{2}\boldsymbol{e}_k^\top \bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k)M\boldsymbol{e}_k\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\| \\ &+ \frac{L\beta^2}{2}\boldsymbol{e}_k^\top M(\bigtriangledown g(\boldsymbol{u}_k))^\top \boldsymbol{e}_k\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\| \end{aligned} \tag{34}$$

$$\begin{aligned} &= J(\boldsymbol{u}_k) - \beta\Big(\| \bigtriangledown J(\boldsymbol{u}_k)\|^2 \\ &- \| \bigtriangledown J(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|\Big) \\ &+ \beta^2\Big(\frac{1}{2}L\| \bigtriangledown J(\boldsymbol{u}_k)\|^2 \\ &+ L\| \bigtriangledown J(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|\Big) \end{aligned} \tag{35}$$

We first require

$$0 < \| \bigtriangledown J(\boldsymbol{u}_k)\|^2 - \| \bigtriangledown J(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|$$

so

$$\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\| < \frac{\| \bigtriangledown J(\boldsymbol{u}_k)\|}{\|\boldsymbol{e}_k\|M} \tag{36}$$

It then follows that we require

$$\beta^2 L\Big(\frac{1}{2}\| \bigtriangledown J(\boldsymbol{u}_k)\|^2 + \| \bigtriangledown g(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|^2 M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|\Big) < \beta\Big(\| \bigtriangledown J(\boldsymbol{u}_k)\|^2 - \| \bigtriangledown J(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|\Big)$$

so that

$$\beta < \frac{2}{L}\frac{\| \bigtriangledown J(\boldsymbol{u}_k)\|^2 - \| \bigtriangledown J(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|}{\| \bigtriangledown J(\boldsymbol{u}_k)\|^2 + 2\| \bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|^2 M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|} \tag{37}$$

Lemma 2. If $\boldsymbol{g}()$ is linear (i.e. $\bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k) = G \ \forall \ k$) then Theorem 1 holds with $M = 0$ and $L = \|G\|^2$.

Proof. Note that L captures the nonlinearity as

$$\bigtriangledown^2 J(\boldsymbol{u}_k) = \bigtriangledown\left(- \bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k)\boldsymbol{e}_k\right) \tag{38}$$

$$= -\left(\bigtriangledown^2 \boldsymbol{g}(\boldsymbol{u}_k)\right)\boldsymbol{e}_k - \bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k)(\bigtriangledown g(\boldsymbol{u}_k)^\top) \tag{39}$$

so if linear, $\bigtriangledown^2 g(\boldsymbol{u}_k) = 0$, and we get

$$\bigtriangledown^2 J(\boldsymbol{u}_k) = - \bigtriangledown g(\boldsymbol{u}_k)(\bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k)^\top) \tag{40}$$

so L is not zero, but can be bounded as

$$L = L_{\text{lin}} = \| \bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k)(\bigtriangledown \boldsymbol{g}(\boldsymbol{u}_k)^\top)\| \tag{41}$$

Lemma 3. Theorem 1 hols with

$$L = \|G\|^2 + M\|\boldsymbol{e}_0\| \tag{42}$$

Proof. Follows from (39) and noting that $\|\boldsymbol{e}_0\|$ is the maximum value of $\|\boldsymbol{e}_k\|$.

Lemma 4. If either $\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\| = 0$ or $M = 0$, then bound (27) becomes

$$\beta < \frac{2}{L} \tag{43}$$

This is the bound if (11) is replaced by the 'exact' operating point, i.e. (12).

Theorem 5. Under the conditions of Theorem 1, selecting

$$\beta = \frac{1}{L}\left(\frac{1-c}{1+2c}\right) \tag{44}$$

in the update (11) bounds the error norm as

$$\|\boldsymbol{e}_{k+1}\|^2 \leq \prod_{i=0}^{k}\Big(1-\frac{\beta}{2}\underline{\sigma}(1-c)\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_i)(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_i))^\top\Big)\|\boldsymbol{e}_0\|^2 \frac{\beta\underline{\sigma}}{2}(1-c)\big(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top\big)\Big)\|\boldsymbol{e}_k\|^2 \tag{57}$$

Proof. First recall (35)

$$\begin{aligned} J(\boldsymbol{u}_{k+1}) &\leq J(\boldsymbol{u}_k) - \beta\Big(\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \\ &- \|\bigtriangledown J(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|\Big) \\ &+ \beta^2 L\Big(\frac{1}{2}\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \\ &+ \|\bigtriangledown J(x)\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|\Big) \end{aligned} \tag{45}$$

and from (36) there is a $0 \leq c < 1$ such that

$$\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\| = c\frac{\|\bigtriangledown J(\boldsymbol{u}_k)\|}{\|\boldsymbol{e}_k\|M} \tag{46}$$

so that

$$\begin{aligned} &J(\boldsymbol{u}_{k+1}) \leq J(\boldsymbol{u}_k) - \beta\Big(\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 - c\|\bigtriangledown J(\boldsymbol{u}_k)\|^2\Big) \\ &+ \beta^2 L\Big(\frac{1}{2}\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 + c\|\bigtriangledown J(\boldsymbol{u}_k)\|^2\Big) \end{aligned} \tag{47}$$

$$\begin{aligned} &= J(\boldsymbol{u}_k) - \beta\Big(1-c\Big)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \\ &+ \beta^2 L\Big(\frac{1}{2}+c\Big)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \end{aligned} \tag{48}$$

Therefore (37) becomes

$$\beta < \frac{2}{L}\frac{\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 - \|\bigtriangledown J(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|}{\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 + 2\|\bigtriangledown J(\boldsymbol{u}_k)\|\|\boldsymbol{e}_k\|M\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k\|}$$

using (46) gives

$$\beta < \frac{2}{L}\left(\frac{1-c}{1+2c}\right) \tag{49}$$

For satisfactory convergence, we can choose, for example, half this maximum value

$$\beta = \frac{1}{L}\left(\frac{1-c}{1+2c}\right) \tag{50}$$

so that (48) becomes

$$\begin{aligned} J(\boldsymbol{u}_{k+1}) &\leq J(\boldsymbol{u}_k) - \beta\Big(1-c\Big)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \\ &+ \beta^2 L\Big(\frac{1}{2}+c\Big)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \end{aligned} \tag{51}$$

$$\begin{aligned} &= J(\boldsymbol{u}_k) - \beta\Big(1-c\Big)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \\ &+ 2\beta\left(\frac{1-c}{1+2c}\right)\left(\frac{1}{2}+c\right)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \end{aligned} \tag{52}$$

$$\begin{aligned} &= J(\boldsymbol{u}_k) - \beta\Big(1-c\Big)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \\ &+ 2\beta\left(\frac{1-c}{2}\right)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \end{aligned} \tag{53}$$

$$= J(\boldsymbol{u}_k) - \frac{\beta}{2}\Big(1-c\Big)\|\bigtriangledown J(\boldsymbol{u}_k)\|^2 \tag{54}$$

Writing

$$\begin{aligned} \|\bigtriangledown J(\boldsymbol{u}_k)\|^2 &= \|(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top \boldsymbol{e}_k\|^2 \\ &\geq \underline{\sigma}\big(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top\big)\|\boldsymbol{e}_k\|^2 \end{aligned} \tag{55}$$

so that (54) becomes

$$\begin{aligned} &J(\boldsymbol{u}_{k+1}) \leq J(\boldsymbol{u}_k) \\ &- \frac{\beta\underline{\sigma}}{2}\Big(1-c\Big)\big(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top\big)\|\boldsymbol{e}_k\|^2 \\ &\Rightarrow \|\boldsymbol{r} - \boldsymbol{g}(\boldsymbol{u}_{k+1})\|^2 \leq \Big(1- \end{aligned} \tag{56}$$

from which we have the convergence property

$$\begin{aligned} &\|\boldsymbol{r} - \boldsymbol{g}(\boldsymbol{u}_{k+1})\|^2 \leq \prod_{i=0}^{k}\Big(1- \\ &\frac{\beta\underline{\sigma}}{2}(1-c)\big(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_i)(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_i))^\top\big)\|\boldsymbol{r} - \boldsymbol{g}(\boldsymbol{u}_0)\|^2 \end{aligned}$$

where from (50), and since $0 \leq c < 1$ and $\underline{\sigma}\big(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_i)(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_i))^\top \leq L$

$$\frac{1}{2} \leq \Big(1 - \frac{\beta\underline{\sigma}}{2}(1-c)\big(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k)(\bigtriangledown \boldsymbol{g}(\hat{\boldsymbol{u}}_k))^\top\big)\Big) < 1$$

5. EXPERIMENTAL SETUP AND RESULTS

In this section, an upper limb rehabilitation system based on functional electrical stimulation (FES) is modeled and is used to verified the proposed algorithm. Every year, millions people suffer a stroke worldwide and are left permanently disabled, and only 5% of survivors with severe paralysis regain upper limb function Barrreca et al. (2003). FES is an effective method to reduce impairment post-stroke. Due to the difficulties in obtaining an accurate model since the indistinct principle of muscle and lacking of sensing methods, few approaches have transferred into clinical practice, though a wide variety of FES upper limb control technologies having been applied in laboratory.

In this paper, we focus on the elbow angle tracking control which is a key role in the reaching-and-touching rehabilitation. The main muscles that drive the movement of elbow are triceps and biceps, which are antagonizing muscle pair. In the following, we first setup the elbow model based on elbow dynamics and hammerstein model which is popular in the literature, e.g Le et al. (2010) and has confirmed accuracy, structural simplicity and correspondence with biophysics. Then, simulations are carried out on it whose parameters are measured from health candidate. For the purpose of comparison, a linear SISO nominal model is built up via simplification and feedback linearization.

5.1 Modeling of Elbow Musculoskeletal System

The hammerstein structure is employed to model the triceps and biceps, which is consisted with a nonlinear isometric recruitment curve (IRC) and linear activation dynamics (LAD). IRC captures the relation between the applied FES and the steady output torque when the muscle length is fixed. While, LAD models the muscle contraction dynamics. IRC and LAD of triceps and biceps are donated by $h_{IRC,TR}(u), H_{LDA,TR}(s)$ and

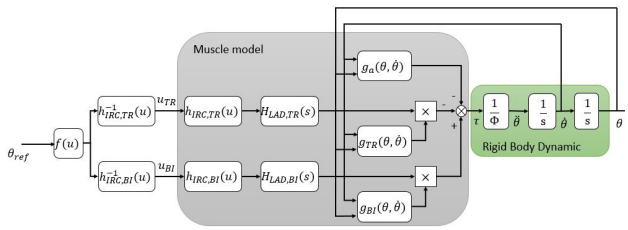

Fig. 1. Elbow musculoskeletal system

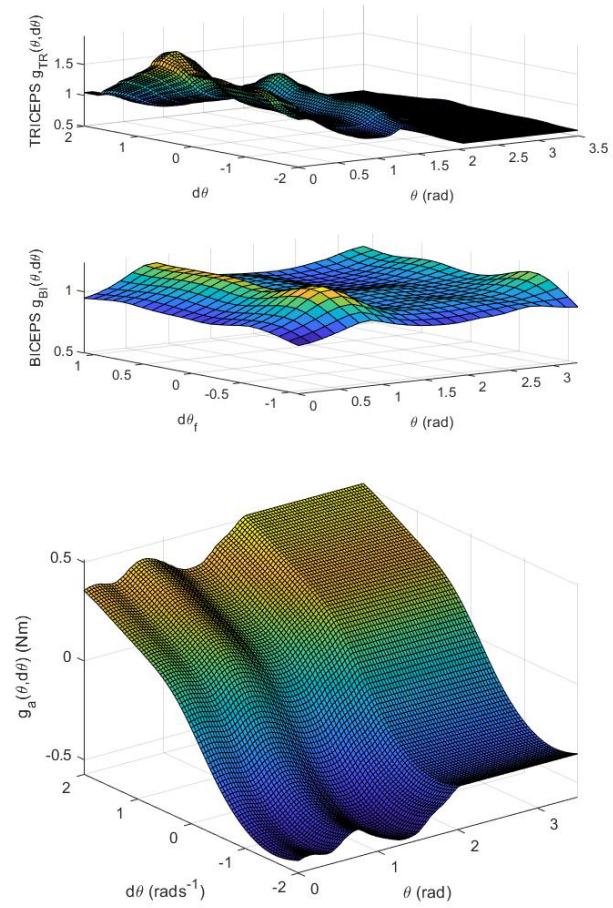

Fig. 2. Effect of elbow angle and speed.

$h_{IRC,BI}(u), H_{LDA,BI}(s)$ respectively, as shown in Fig. 1. The muscle outputs will be affected by the angle θ and angular velocity $\dot{\theta}$ of elbow. Those effects are represented by nonlinear functions $g_{TR}(\theta,\dot{\theta})$ and $g_{BI}(\theta,\dot{\theta})$. Thus, the output τ of the biceps-triceps pair is

$$\tau = g_{BI}(\theta,\dot{\theta})h_{LDA,BI}(t) \times h_{IRC,BI}(u_{BI}) \quad (58)$$
$$- g_{TR}(\theta,\dot{\theta})h_{LDA,TR}(t) \times h_{IRC,TR}(u_{TR}) \quad (59)$$
$$- g_a(\theta,\dot{\theta}) \quad (60)$$

Here, u_{BI}, u_{TR} are FES input to the biceps and triceps, $g_a(\theta,\dot{\theta})$ is caused by the gravity and joint friction, $h_{LDA,\cdot}(t)$ is the inverse Laplace transform of $H_{LDA,\cdot}(s)$. The nonlinear functions g_{TR}, g_{BI} and g_a are measured by experiments, as shown in Fig 2.

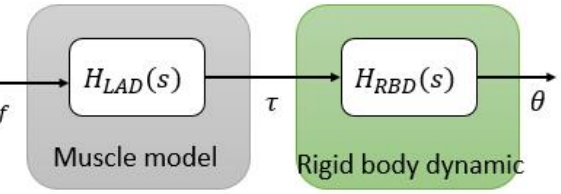

Fig. 3. The structure of nominal model

The simplified bode rigid dynamic is described as a second-order integral system as shown in Fig. 1. Φ is the forearm inertia. The mapping from τ to elbow angle θ is

$$G_{RBD} = \frac{1}{\Phi s^2}. \quad (61)$$

To remove the nonlinearity caused by IRC, the feedback linearization is employed. The FES inputs will go through the inverse of IRC firstly and then applied to the muscle. As the triceps and biceps are antagonistic, the input signals u_{TR} and u_{BI} couldn't be nonzero at the same time. Thus, we can transform the two-input one-output model to single-input single-output (SISO) model by combining the two input signals into a signal input signal u. This could be achieved by

$$u_{TR} = \begin{cases} h^{-1}_{IRC,TR}(-u) & u \le 0 \\ 0 & u > 0 \end{cases} \quad (62)$$

$$u_{BI} = \begin{cases} h^{-1}_{IRC,BI}(u) & u > 0 \\ 0 & u \le 0 \end{cases}. \quad (63)$$

For comparative purposes, we simplify this system and provide a nominal model which is used in the model-based simulation experiments, as shown in Fig. 3. The nonlinearity of IRC is eliminated by the inverse block. The dynamics of triceps and biceps are approximated to two-order linear systems with the same parameters. Thus, we could merge the two-input system into a single-input system. To simplify the nonlinearities caused by the elbow angle and angular velocity, the nonlinear functions are simplified to be linear mapping functions and are absorbed by the rigid body dynamics. Thus, the muscle model dynamics is

$$H_{LAD} = \frac{\omega_n^2}{s^2 + 2\omega_n s + \omega_n^2}. \quad (64)$$

And the rigid body dynamics is

$$H_{RBD} = \frac{a_2}{s^2 + a_1 s + a_2}. \quad (65)$$

The simplified nominal model is

$$Gn = H_{LAD}H_{RBD}. \quad (66)$$

In the next section, we conduct simulation experiments on above complete model and nominal model, and compare the simulation results. The used parameters are from Freeman et al. (2009a).

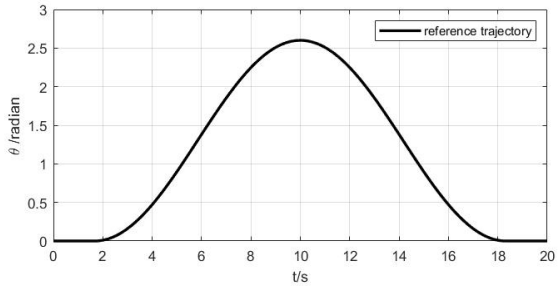

Fig. 4. Reference trajectory.

5.2 Algorithm settings

In this paper, we conduct simulation experiments to compare the conventional gradient ILC method (gILC), proposed nonlinear model-free gradient ILC method (NMF) and data-driven ILC method (DILC) proposed by Bolder et al. (2018). In the following, the simulation parameters are given.

The performance is defined as $J(\boldsymbol{u}) = \|\boldsymbol{e}\|^2$. In this case, the weighted matrixes for DILC are $\boldsymbol{W}_e = \boldsymbol{I}, \boldsymbol{W}_f = 0, \boldsymbol{W}_{\Delta f} = 0$. The learning rate β for all the methods are set to be the same, that is 1. In conventional gradient ILC method, the input update law is

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \beta \boldsymbol{G}^* \boldsymbol{e}_k \tag{67}$$

Here, G is the 'lifted' model matrix generated from the nominal model 66.

The experimental process of proposed algorithm is shown in section 3. The scale rate α is 0.1. To update the next input, two extra trials are needed. The first trial is to generate the output $\mathbf{g}(\hat{\mathbf{u}}_\mathbf{k})$, $\hat{\mathbf{u}}_k$ is the fixed operating point. And the second trial is to generate the output of $g(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k)$, $\tilde{\boldsymbol{e}}_k$ is the time inverse of tracking error. Then, the input is updated as

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \frac{\beta}{\alpha}\left(\overbrace{\boldsymbol{g}(\hat{\boldsymbol{u}}_k + \alpha\tilde{\boldsymbol{e}}_k)} - \overbrace{\boldsymbol{g}(\hat{\boldsymbol{u}}_k)}\right) \tag{68}$$

The enhancing convergence rate method in Bolder et al. (2018) could be used to improve our algorithm, which will not introduced in this paper in order to compare the kernel algorithms. Thus, the data-driven gradient-descent algorithm is compared with our algorithm. The input update law is

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \beta \boldsymbol{G}^* \boldsymbol{e}_k \tag{69}$$

which is the same as conventional gradient ILC, except that $\boldsymbol{G}^*\boldsymbol{e}_k$ is obtained by applying the time inverse of tracking error $\boldsymbol{e}_k$.

The reference trajectory is presented in figure 4. The elbow angle moves to 2.6 radian and then backs to original position smoothly. Each trial lasts 20s and the sampled frequency is 100Hz. Every simulation experiment includes 100 trials. In the following section, the main simulation results are presented.

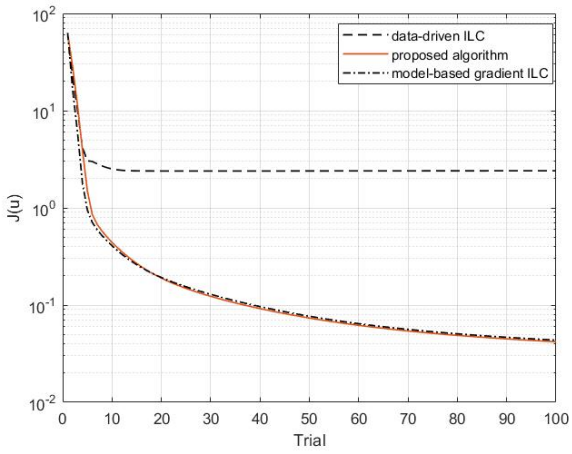

Fig. 5. Convergence comparison of the three algorithms.

5.3 Simulation Results

Under the settings presented above, simulations are carried out. The main results are the algorithms performance defined as $J(\boldsymbol{u}) = \|\boldsymbol{e}\|^2$. The convergence processes of algorithm performance along trials are presented in figure 5. The results demonstrate that performance of the proposed method is almost the same as the model-based gradient ILC, and is better than data-driven ILC for the nonlinear elbow musculoskeletal system. The following observations are made.

The convergence rates of proposed method and conventional gradient ILC are almost the same, which could prove the ability of our nonlinear model-free gradient ILC algorithm. While the data-driven ILC has a poor performance. At the first several trials, the tree algorithms almost have the same convergence rate. After 5 trials, the performance value of data-driven ILC algorithm remains. The possible reason is the nonlinearity of the isometric recruitment curve (IRC). As shown in figure 6, the property of the curve when the input u is small is similar to dead region. The data-driven ILC algorithm applies the time inverse of tracking error $\boldsymbol{e}_k$ to the system to update the next input. After several trials, the tracking error is small enough, and when applied to the system, the input, that is the time inverse of tracking error, will fall into the 'dead region'. That will generate a output around zero, as shown in figure 7. Thus, the input will not be updated and the performance will not be optimized. While, the proposed algorithm adds a fixed working point $\hat{\boldsymbol{u}}_k$ to the tracking error, which could be regarded as a discretization of the nonlinear system at $\hat{\boldsymbol{u}}_k$. Thus, the proposed algorithm could reduce the tracking error significantly.

6. CONCLUSION

In this paper, a model-free gradient ILC algorithm for nonlinear system is proposed. The key idea is employing a fixed working point $\hat{\boldsymbol{u}}_k$, which could

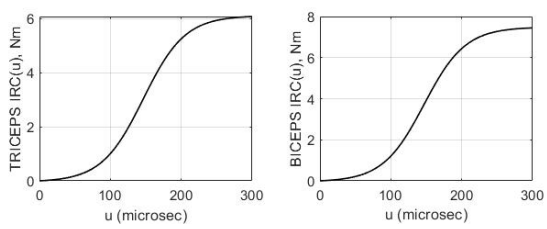

Fig. 6. Nonlinear isometric recruitment curves of triceps and biceps.

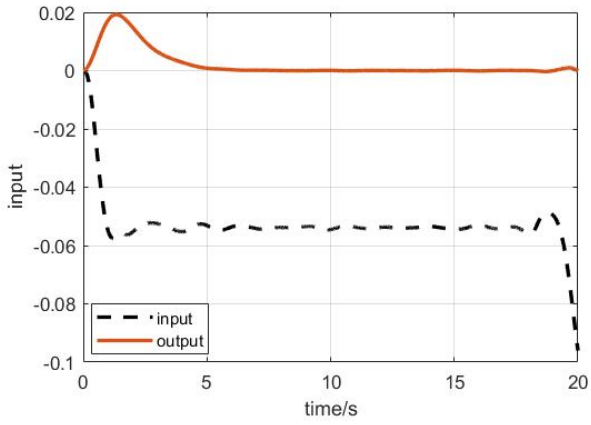

Fig. 7. The input and output of data-driven ILC at trial 50.

be regarded as a discretization of the nonlinear system. The proposed algorithm is compared with conventional model-based gradient ILC and data-driven ILC on the elbow musculoskeletal system via simulations. The experimental results prove that the proposed algorithm could achieve the same performance as model-based gradient ILC, while avoiding the process of model identification by adding more trials, and a better performance than data-driven ILC which is more suitable for linear systems or weak-nonlinear systems. The convergence conditions are given and proved in section 4, which gives a guidance of the selection of learning rate and scale factor.

The proposed algorithm is only verified by simulations in this paper. Ongoing research is applying this model-free gradient ILC algorithm to healthy participants. Another research is how to reduce the additional trials.

ACKNOWLEDGEMENTS

Place acknowledgments here. This research is supported by the China Postdoctoral Science Foundation (NO.2018M632801).

REFERENCES

Adloo, H., Deghat, M., and Karimaghaee, P. (2009). Iterative state feedback control and its application to robot control. *IEEE International Conference on Mechatronics*, 1–6.

Ahn, H.S., Chen, Y., and Moore, K.L. (2007). Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(6), 1099 –1121. doi: 10.1109/TSMCC.2007.905759.

Amann, N., Owens, D.H., and Rogers, E. (1996). Iterative learning control using optimal feedback and feedforward actions. *International Journal of Control*, 65(2), 277–293.

Barrreca, S., Wolf, S.L., Fasoli, S., and Bohannon, R. (2003). Treatment interventions for the paretic upper limb of stroke survivors: A critical review. *Neurorehabilitation and Neural Repair*, 17(4), 220–226.

Barton, K.L. and Alleyne, A.G. (2011a). A norm optimal approach to time-varying ILC with application to a multi-axis robotic testbed. *IEEE Transactions on Control Systems Technology*, 19(1), 166–180.

Barton, K.L. and Alleyne, A.G. (2011b). A norm optimal approach to time-varying ILC with application to a multi-axis robotic testbed. *IEEE Transactions on Control System Technology*, 19(1), 166–180.

Barton, K.L., van de Wijdeven, J., Alleyne, A., Bosgra, O., and Steinbuch, M. (2008). Norm optimal cross-coupled iterative learning control. *47th IEEE Conference on Decision and Control*, 3020–3025.

Bolder, J., Kleinendors, S., and Oomen, T. (2018). Data-driven multivariable ILC: enhanced performance byeliminating L and Q filters. *International Journal of Robust and Nonlinear Control*, 28, 3728–3751.

Bristow, D.A. (2008). Weighting matrix design for robust monotonic convergence in norm optimal iterative learning control. In *American Control Conference*, 4554–4560.

Bristow, D.A. and Alleyne, A.G. (2006). A high precision motion control system with application to microscale robotic deposition. *IEEE Transactions on Control Systems Technology*, 14(6), 1008–1020.

Bristow, D.A., Tharayil, M., and Alleyne, A.G. (2006). A survey of iterative learning control a learning-based method for high-performance tracking control. *IEEE Control Systems Magazine*, 26(3), 96–114.

Buchheit, K., Pandit, M., and Befort, M. (1994). Optimal iterative learning control of an extrusion plant. In *International Conference on Control*, volume 1, 652–657.

Butcher, M., Karimi, A., and Longchamp, R. (2008a). Iterative learning control based on stochastic approximation. In *IFAC Triennial World Congress*, volume 17, 1478–1483.

Butcher, M., Karimi, A., and Longchamp, R. (2008b). A statistical analysis of certain iterative learning control algorithms. *International Journal of Control*, 81(1), 156–166.

Chu, B. and Owens, D.H. (2009). Accelerated norm-optimal iterative learning control algo-

rithms using successive projection. *International Journal of Control*, 82(8), 1469–1484.

Chu, B. and Owens, D.H. (2010). Iterative learning control for constrained linear systems. *International Journal of Control*, 83(7), 1397–1413.

Dinh, T.V., Freeman, C.T., and Lewin, P.L. (2013). Evaluation of norm optimal iterative learning control for multivariable systems. *Journal of Dynamic Systems Measurement and Control*.

Freeman, C.T. (2004). *Experimental Evaluation of Iterative Learning Control on a Non-minimum Phase Plant*. Ph.D. thesis, School of Electronics and Computer Science, University of Southampton.

Freeman, C.T. (2016). *Control System Design for Electrical Stimulation in Upper Limb Rehabilitation*. Springer International Publishing. Springer International Publishing.

Freeman, C.T., Hughes, A.M., Burridge, J.H., Chappell, P.H., Lewin, P.L., and Rogers, E. (2009a). A model of the upper extremity using FES for stroke rehabilitation. *ASME Journal of Biomechanical Engineering*, 131(3), 031006–1–031006–10.

Freeman, C.T., Lewin, P.L., Rogers, E., Owens, D.H., and Hatonen, J.J. (2009b). Discrete fourier transform based iterative learning control design for linear plants with experimental verification. *Journal of Dynamic Systems, Measurement, and Control*, 131(3), 1006–1016.

Freeman, C.T., Rogers, E., Hughes, A.M., Burridge, J.H., and Meadmore, K.L. (2012). Iterative learning control in health care: Electrical stimulation and robotic-assisted upper-limb stroke rehabilitation. *IEEE Control Systems Magazine*, 32, 18–43.

Furuta, K., Yamakita, M., and Kobayashi, S. (1991). Swing up control of inverted pendulum. In *International Conference on Industrial Electronics, Control and Instrumentation*, volume 3, 2193–2198.

Harte, T., Hätönen, J.J., and Owens, D.H. (2005). Discrete-time inverse model-based iterative learning control: stability, monotonicity and robustness. *International Journal of Systems Science*, 78(8), 557–586.

Hätönen, J.J., Freeman, C.T., Owens, D.H., Lewin, P.L., and Rogers, E. (2004). Robustness analysis of a gradient-based repetitive algorithm for discrete-time systems. In *Proceedings of the 43th Conference on Decision and Control*, 1301–1306. Paradise Island, Bahamas.

Hätönen, J.J., Freeman, C.T., Owens, D.H., Lewin, P.L., and Rogers, E. (2006). A gradient-based repetitive control algorithm combining ILC and pole placement. *European Journal of Control*, 12(3), 278–292.

Jian-Xu, X. and Ji, Q. (1998). New ilc algorithms with improved convergence for a class of non-affine functions. In *Proceedings of the 37th IEEE Decision and Control*, volume 1, 660–665.

Le, F., Markovsky, I., Freeman, C.T., and Rogers, E. (2010). Identification of electrically stimulated muscle models of stroke patients. *Control Engineering Practice*, 18(4), 396–407.

Mezghani, M., Roux, G., Cabassud, M., Lann, M.L., Dahhou, B., and Casamatta, G. (2002). Application of iterative learning control to an exothermic semibatch chemical reactor. *IEEE Transactions on Control Systems Technology*, 10(6), 822–834.

Mishra, S. and Tomizuka, M. (2005). An optimization-based approach for design of iterative learning controllers with accelerated rates of convergence. In *44th IEEE Conference in Decision and Control*, 2427–2432.

Norrlof, M. (2000). Comparative study on first and second order ilc- frequency domain analysis and experiments. In *Proceedings of the 39th Conference on Decision and Control*. CDROM.

Owens, D.H. and Chu, B. (2009). Accelerated norm-optimal iterative learning control algorithms using successive projection. *International Journal of Control*, 82, 1469–1484.

Owens, D.H. and Feng, K. (2003). Parameter optimization in iterative learning control. *International Journal of Control*, 76(11), 1059–1069.

Owens, D.H., Hätönen, J.J., and Daley, S. (2009). Robust monotone gradient-based discrete-time iterative learning control. *International Journal of Robust and Nonlinear Control*, 19(6), 634–661.

Ratcliffe, J.D., Lewin, P.L., Rogers, E., Hatonen, J.J., and Owens, D. (2006). Norm-optimal iterative learning control applied to gantry robots for automation applications. *IEEE Transactions on Robotics*, 22(6), 1303–1307.

Rogers, E., Owens, D.H., Werner, H., Freeman, C.T., Lewin, P.L., Kichhoff, S., Chmidt, S., and Lichtenberg, G. (2010). Norm optimal iterative learning control with application to problems in accelerator based free electron lasers and rehabilitation robotics. *European Journal of Control*, 16(5), 497–524.

Schindele, D. and Aschemann, H. (2011). ILC for a fast linear axis driven by pneumatic muscle actuators. In *IEEE International Conference on Mechatronics*, 967–972.

Shaojuan, Y., Shibo, X., and Jingcai, B. (2008). Design of iterative learning controller combined with feedback control for electrohydraulic servo system. *Fourth International Natural Computation*, 2, 622–626.

van de Wijdeven, J., Donkers, T., and Bosgra, O. (2009). Iterative learning control for uncertain systems: Robust monotonic convergence analysis. *Automatica*, 45(10), 2383–2391.

Wang, Y., Dassau, E., and Doyle, F.J. (2010). Closed-loop control of artificial pancreatic β-Cell in type 1 diabetes mellitus using model predictive iterative learning control. *IEEE Transactions on Biomedical Engineering*, 57(2), 211–219.

Zheng, J., Zhao, S., and Wei, S. (2009). Adaptive fuzzy iterative learning control for SRM direct-drive volume control servo hydraulic press. *International Conference on Sustainable Power Generation and Supply*, 1–6.