

Repetitive Learning Control: A Lyapunov-Based Approach*

[†]W. E. Dixon, E. Zergeroglu, D. M. Dawson, and B. T. Costic

[†]Oak Ridge National Laboratory Department of Electrical & Computer Engineering
Bldg. 7601, Mailstop 6305 Clemson University
P.O. Box 2008 Clemson, SC 29634-0915
Oak Ridge, TN 37831
dixonwe@ornl.gov email: ezerger, ddawson, bcostic@ces.clemson.edu

Abstract

In this paper, a learning-based feedforward term is developed to solve a general control problem in the presence of unknown nonlinear dynamics with a known period. Since the learning-based feedforward term is generated from a straightforward Lyapunov-like stability analysis, the control designer can utilize other Lyapunov-based design techniques to develop hybrid control schemes that utilize learning-based feedforward terms to compensate for periodic dynamics and other Lyapunov-based approaches (e.g., adaptive-based feedforward terms) to compensate for non-periodic dynamics. To illustrate this point, a hybrid adaptive/learning control scheme is utilized to achieve global asymptotic link position tracking for a robot manipulator.

1 Introduction

Many industrial applications require robots to perform repetitious tasks including: assembly, manipulation, inspection, *etc.* Given the myriad of industrial applications that require a robot to move in repetitive manner, researchers have been motivated to investigate control methods that exploit the periodic nature of the robot dynamics, and hence, increase link position tracking performance. As a result of this work, many types of learning controllers have been developed to compensate for periodic disturbances. Some advantages of these controllers over other approaches include: i) the ability to compensate for disturbances without high frequency or high gain feedback terms, and ii) the ability to compensate for time-varying disturbances that can include time-varying parametric effects.

Some of the initial learning control research targeted the development of betterment learning controllers (see [2] and [3]). Unfortunately, one of the drawbacks of the betterment learning controllers is that the robot is required to return to the same desired initial configuration after each learning trial. Moreover, in [14], Heinzinger *et al.* provided several examples that illustrated the lack of robustness of the betterment learning controllers to variations in the initial conditions of

*This research was performed in part by a Eugene P. Wigner Fellow and staff member at the Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the U.S. Department of Energy under contract DE-AC05-00OR22725 and is supported in part by the Office of Naval Research under Interagency Agreement 1866-Q356-A1 with the U.S. DOE and with U.S. NSF Grants DMI-9457967, DMI-9813213, EPS-9630167, DOE Grant DE-FG07-96ER14728, a DOC Grant, an ARO Automotive Center Grant, and the Gebze Institute for Advanced Technology, Gebze Turkey.

the robot. To address these robustness issues, Arimoto [1] incorporated a forgetting factor in the betterment learning algorithm given in [2] and [3]. Motivated by the results from the betterment learning research, several researchers investigated the use of repetitive learning controllers. One of the advantages of the repetitive learning scheme is that the requirement that the robot return to the exact same initial condition after each learning trial, is replaced by the less restrictive requirement that the desired trajectory of the robot be periodic. Some of the initial repetitive learning control research was performed by [13], [28], and [29]; however, the asymptotic convergence of these basic repetitive control schemes can only be guaranteed under restrictive conditions on the plant dynamics that might not be satisfied. To enhance the robustness of these repetitive control schemes, researchers in [13] and [28] modified the repetitive update rule to include the so-called Q-filter. Unfortunately, the use of the Q-filter eliminates the ability of the tracking errors to converge to zero. In the search for new learning control algorithms, researchers in [15] and [20] proposed an entirely new scheme that exploited the use of kernel and influence functions in the repetitive update rule; however, this class of controllers tends to be fairly complicated in comparison to the control schemes that utilize a standard repetitive update rule.

In [26] and [30], iterative learning controllers (ILCs) were developed that do not require differentiation of the update rule, so that the algorithm can be applied to sampled data without introducing differentiation noise. In [4], [5], [6], and [31], ILCs were developed to address the motion and force control problem for constrained robot manipulators. In [7], Cheah and Wang develop a model-reference learning control scheme for a class of nonlinear systems in which the performance of the learning system is specified by a reference model. In [32], Xu and Qu utilize a Lyapunov-based approach to illustrate how an ILC can be combined with a variable structure controller to handle a broad class of nonlinear systems. In [10], Ham *et al.* utilized Lyapunov-based techniques to develop an ILC that is combined with a robust control design to achieve global uniformly ultimately bounded link position tracking for robot manipulators. The applicability of this design was extended to a broader class of nonlinear systems by Ham *et al.* in [11]. Recently, several researchers (see [8], [12], [16], and [17]) have utilized a class of multiple-step “functional” iterative learning controllers to damp out steady-state oscillations. As stated in [17], the fundamental difference between the previous learning controllers and the controllers proposed in [8], [12], [16], and [17], are that the ILC is not updated continuously with time, rather, it is switched at iterations triggered by *steady-state oscillations*. Han *et al.* utilized this iterative update procedure in [12], to damp out steady-state oscillations in the velocity set-point problem for servo-motors. The work in [12] was extended by [8] to compensate for friction effects and applied in [16] to VCR servo-motors (see [21] and [22] for a comprehensive review and tutorial on ILC).

Upon examination of some of the aforementioned work, it seems that many of the recent ILC and repetitive control results exploit a standard repetitive update rule as the core part of the controller; however, to ensure that the stability analysis¹ validates the proposed results, the authors utilize many types of additional rules in conjunction with the standard repetitive update rule. As we show in this paper, these additional rules and additional complexity injected into the stability analysis, are not necessary for the development of learning controllers that utilize the standard repetitive update rule. We also conjecture that a statement concerning the boundedness of learning controllers made in [20] may have caused some researchers to: i) attempt a modification of the standard repetitive update rule with additional rules, or ii) abandon the use of the standard repetitive update rule entirely. To clarify the above statements, we present the following simple, closed-loop system

$$\dot{x} = -x + \varphi(t) - \hat{\varphi}(t) \tag{1}$$

¹We also note that the proofs of the stability associated with this work tend to be rather complex.

where $x(t) \in \mathbb{R}^1$ is a tracking error signal, $\varphi(t) \in \mathbb{R}^1$ is an unknown nonlinear function, and $\hat{\varphi}(t) \in \mathbb{R}^1$ is a learning-based estimate of $\varphi(t)$. It is assumed that the unknown nonlinear function $\varphi(t)$ is periodic with a known period T (i.e., $\varphi(t - T) = \varphi(t)$). For the system given by (1), the standard repetitive update rule is given by

$$\hat{\varphi}(t) = \hat{\varphi}(t - T) + x. \quad (2)$$

With regard to the error system given (1) and (2), Messner *et al.* [20] noted that the techniques used in [20] could not be used to show that $\hat{\varphi}(t)$ is bounded if $\hat{\varphi}(t)$ is generated by (2). To address the boundedness problem associated with the standard repetitive update rule, Sadegh *et al.* [25] proposed to saturate the entire right-hand side of (2) as follows

$$\hat{\varphi}(t) = \text{sat}(\hat{\varphi}(t - T) + x), \quad (3)$$

and hence, guarantee that $\hat{\varphi}(t)$ is bounded for all time (the function $\text{sat}(\cdot)$ is the standard linear piecewise bounded saturation function). Unfortunately, it was not exactly clear from the analysis given in [25] how the Lyapunov-based stability analysis accommodates the saturation of the standard repetitive update rule (e.g., it is well known how one can apply a projection algorithm to the adaptive estimates of a gradient adaptive update law and still accommodate the Lyapunov-based stability analysis).

In this paper, we attempt to address the above issues via a modification of the standard repetitive update rule. That is, as opposed to (3), we saturate the standard repetitive update rule as follows

$$\hat{\varphi}(t) = \text{sat}(\hat{\varphi}(t - T)) + x. \quad (4)$$

We then utilize a Lyapunov-based approach to: i) illustrate how the stability analysis accommodates the use of the saturation function in (4), ii) prove that $x(t)$ is forced asymptotically to zero, and iii) show that $\hat{\varphi}(t)$ remains bounded. To illustrate the generality of the learning-based update law given by (4), we apply the update law to force the origin of a general error system with a nonlinear disturbance with a known period to achieve global asymptotic tracking. To illustrate the fact that other Lyapunov-based techniques can be exploited to compensate for additional disturbances that are not periodic, we design a hybrid adaptive/repetitive learning scheme to achieve global asymptotic link position tracking for a robot manipulator. In comparison with the previous work of [8], [12], [16], and [17], we note that: i) the proposed learning-based controller utilizes standard Lyapunov-based techniques, and hence, one can easily fuse in other Lyapunov-based tools, ii) the stability analysis is straightforward, iii) the proposed learning-based controller utilizes a simple modification of the standard repetitive update rule as opposed to use of a multiple step process or menu, and iv) the proposed control scheme is updated continuously with time during the transient response (versus during the steady-state), and hence, an improved transient response is facilitated.

This paper is organized as follows. In Section 2, we present the error dynamics for a general problem, develop a learning-based algorithm, and utilize a Lyapunov-based stability analysis to prove a global asymptotic tracking result. In Section 3, we develop a hybrid adaptive/learning algorithm for robot manipulators that compensates for dynamics with periodic and non-periodic components. In Section 4, we demonstrate the effectiveness of the learning algorithm through experimental results obtained from a 2-link revolute, direct-drive robot manipulator. Concluding remarks are given in Section 5.

2 General Problem

To illustrate the generality of the proposed learning control scheme, we consider the following error dynamics examined in [20]

$$\dot{e} = f(t, e) + B(t, e)[w(t) - \hat{w}(t)] \quad (5)$$

where $e(t) \in \mathbb{R}^n$ is an error vector, $w(t) \in \mathbb{R}^m$ is an unknown nonlinear function, $\hat{w}(t) \in \mathbb{R}^m$ is a subsequently designed learning-based estimate of $w(t)$, and the auxiliary functions $f(t, e) \in \mathbb{R}^n$ and $B(t, e) \in \mathbb{R}^{n \times m}$ are bounded provided $e(t)$ is bounded. In a similar manner as [20], we assume that (5) satisfies the following assumptions.

Assumption 1: The origin of the error system $e(t) = 0$ is uniformly asymptotically stable for

$$\dot{e} = f(t, e); \quad (6)$$

furthermore, there exists a first-order differentiable, positive-definite function $V_1(e, t) \in \mathbb{R}^1$, a positive-definite, symmetric matrix $Q(t) \in \mathbb{R}^{n \times n}$, and a known matrix $R(t) \in \mathbb{R}^{n \times m}$ such that

$$\dot{V}_1 \leq -e^T Q e + e^T R [w - \hat{w}]. \quad (7)$$

Assumption 2: The unknown nonlinear function $w(t)$ is periodic with a known period T ; hence,

$$w(t - T) = w(t). \quad (8)$$

Furthermore, we assume that the unknown function $w(t)$ is bounded as follows

$$|w_i(t)| \leq \beta_i \quad \text{for } i = 1, 2, \dots, m \quad (9)$$

where $\beta = [\beta_1 \ \beta_2 \ \dots \ \beta_m] \in \mathbb{R}^m$ is a vector of known, positive bounding constants.

2.1 Control Objective

The control objective for the general problem given in (5) is to design a learning-based estimate $\hat{w}(t)$ such that

$$\lim_{t \rightarrow \infty} e(t) = 0 \quad (10)$$

for any bounded initial condition denoted by $e(0)$. To quantify the mismatch between the learning-based estimate and $w(t)$, we define an estimation error term, denoted by $\tilde{w}(t) \in \mathbb{R}^m$, as follows

$$\tilde{w}(t) = w(t) - \hat{w}(t). \quad (11)$$

2.2 Learning-Based Estimate Formulation

Based on the error system given in (5) and the subsequent stability analysis, we design the learning-based estimate $\hat{w}(t)$ as follows

$$\hat{w}(t) = \text{sat}_\beta(\hat{w}(t - T)) + k_e R^T e \quad (12)$$

where $k_e \in \mathbb{R}^1$ is a positive constant control gain, and $\text{sat}_\beta(\cdot) \in \mathbb{R}^m$ is a vector function whose elements are defined as follows

$$\text{sat}_{\beta_i}(\xi_i) = \begin{cases} \xi_i & \text{for } |\xi_i| \leq \beta_i \\ \text{sgn}(\xi_i) \beta_i & \text{for } |\xi_i| > \beta_i \end{cases} \quad \forall \xi_i \in \mathbb{R}^1, i = 1, 2, \dots, m \quad (13)$$

where β_i represent the elements of β defined in (9), and $\text{sgn}(\cdot)$ denotes the standard signum function. From the definition of $\text{sat}_\beta(\cdot)$ given in (13), we can prove that (see Appendix A)

$$(\xi_{1i} - \xi_{2i})^2 \geq (\text{sat}_{\beta_i}(\xi_{1i}) - \text{sat}_{\beta_i}(\xi_{2i}))^2 \quad \forall |\xi_{1i}| \leq \beta_i, \xi_{2i} \in \mathbb{R}^1, i = 1, 2, \dots, m. \quad (14)$$

To facilitate the subsequent stability analysis, we substitute (12) into (11) for $w(t)$ and $\hat{w}(t)$, respectively, to rewrite the expression for $\tilde{w}(t)$ as follows

$$\tilde{w} = \text{sat}_\beta(w(t-T)) - \text{sat}_\beta(\hat{w}(t-T)) - k_e R^T e. \quad (15)$$

where we utilized (8), (9), and the fact that

$$w(t) = \text{sat}_\beta(w(t)) = \text{sat}_\beta(w(t-T)). \quad (16)$$

2.3 Stability Analysis

Theorem 1 *The learning-based estimate defined in (12) ensures that*

$$\lim_{t \rightarrow \infty} e(t) = 0 \quad (17)$$

for any bounded initial condition denoted by $e(0)$.

Proof: To prove Theorem 1, we define a non-negative function $V_2(t, e, \tilde{w}) \in \mathbb{R}^1$ as follows

$$V_2 = V_1 + \frac{1}{2k_e} \int_{t-T}^t [\text{sat}_\beta(w(\tau)) - \text{sat}_\beta(\hat{w}(\tau))]^T [\text{sat}_\beta(w(\tau)) - \text{sat}_\beta(\hat{w}(\tau))] d\tau \quad (18)$$

where $V_1(t, e)$ was described in Assumption 1. After taking the time derivative of (18), we obtain the following expression

$$\begin{aligned} \dot{V}_2 &\leq -e^T Q e + e^T R \tilde{w} \\ &\quad + \frac{1}{2k_e} [\text{sat}_\beta(w(t)) - \text{sat}_\beta(\hat{w}(t))]^T [\text{sat}_\beta(w(t)) - \text{sat}_\beta(\hat{w}(t))] \\ &\quad - \frac{1}{2k_e} [\text{sat}_\beta(w(t-T)) - \text{sat}_\beta(\hat{w}(t-T))]^T [\text{sat}_\beta(w(t-T)) - \text{sat}_\beta(\hat{w}(t-T))] \end{aligned} \quad (19)$$

where (7) was utilized. After utilizing (15), we can rewrite (19) as follows

$$\begin{aligned} \dot{V}_2 &\leq -e^T Q e + e^T R \tilde{w} - \frac{1}{2k_e} (\tilde{w} + k_e R^T e)^T (\tilde{w} + k_e R^T e) \\ &\quad + \frac{1}{2k_e} [\text{sat}_\beta(w(t)) - \text{sat}_\beta(\hat{w}(t))]^T [\text{sat}_\beta(w(t)) - \text{sat}_\beta(\hat{w}(t))]. \end{aligned} \quad (20)$$

After performing some simple algebraic operations, we can further simplify (20) as follows

$$\begin{aligned} \dot{V}_2 &\leq -e^T \left(Q + \frac{k_e}{2} R R^T \right) e \\ &\quad - \frac{1}{2k_e} \left[\tilde{w}^T \tilde{w} - [\text{sat}_\beta(w(t)) - \text{sat}_\beta(\hat{w}(t))]^T [\text{sat}_\beta(w(t)) - \text{sat}_\beta(\hat{w}(t))] \right]. \end{aligned} \quad (21)$$

Finally, we can utilize (9), (11), and (14) to simplify (21) as shown below

$$\dot{V}_2 \leq -e^T Q e. \quad (22)$$

Based on (18), (22), and the fact that Q is a positive-definite symmetric matrix, it is clear that $e(t) \in \mathcal{L}_2 \cap \mathcal{L}_\infty$. Based on the fact that $e(t) \in \mathcal{L}_\infty$, it is clear from (5), (12), (13), and (15) that $\hat{w}(t), \tilde{w}(t), f(t, e), B(t, e) \in \mathcal{L}_\infty$. Given that $\hat{w}(t), \tilde{w}(t), f(t, e), B(t, e) \in \mathcal{L}_\infty$, it is clear from (5) that $\dot{e}(t) \in \mathcal{L}_\infty$, and hence, $e(t)$ is uniformly continuous. Since $e(t) \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and uniformly continuous, we can utilize Barbalat's Lemma [18] to prove (17).

Remark 1 *From the previous stability analysis, it is clear that we exploit the fact that the learning-based feedforward term given in (12) is composed of a saturation function. That is, it is easy to from the structure of (12), that if $e(t) \in \mathcal{L}_\infty$ then $\hat{w}(t) \in \mathcal{L}_\infty$.*

3 Hybrid Adaptive/Control Example

In the previous section, we exploited the fact that the unknown nonlinear dynamics, denoted by $w(t)$, were periodic with a known period T . Unfortunately, some physical systems may not adhere to the ideal assumption that all of the unknown nonlinear dynamics are entirely periodic. Since the learning-based feedforward term, developed in the previous section, is generated from a straightforward Lyapunov-like stability analysis, we can utilize other Lyapunov-based control design techniques to develop hybrid control schemes that utilize learning-based feedforward terms to compensate for periodic dynamics and other Lyapunov-based approaches (*e.g.*, adaptive-based feedforward terms) to compensate for non-periodic dynamics. To illustrate this point, we now develop a hybrid adaptive/learning control scheme for a n -rigid link, revolute, direct-drive robot manipulator in the following sections.

3.1 Dynamic Model

The dynamic model for a n -rigid link, revolute, direct-drive robot is assumed to have the following form [18]

$$M(q)\ddot{q} + V_m(q, \dot{q})\dot{q} + G(q) + F_d\dot{q} + F_s(\dot{q}) = \tau \quad (23)$$

where $q(t), \dot{q}(t), \ddot{q}(t) \in \mathbb{R}^n$ denote the link position, velocity, and acceleration vectors, respectively, $M(q) \in \mathbb{R}^{n \times n}$ represents the link inertia matrix, $V_m(q, \dot{q}) \in \mathbb{R}^{n \times n}$ represents centripetal-Coriolis matrix, $G(q) \in \mathbb{R}^n$ represents the gravity effects, $F_d \in \mathbb{R}^{n \times n}$ is the constant, diagonal, positive-definite, viscous friction coefficient matrix, $F_s \in \mathbb{R}^{n \times n}$ is a constant, diagonal, positive-definite, matrix composed of static friction coefficients, and $\tau(t) \in \mathbb{R}^n$ represents the torque input vector. With regard to dynamics given by (23), we make that the standard assumption that all of the terms on the left-hand side of (23) are bounded if $q(t), \dot{q}(t)$, and $\ddot{q}(t)$ are bounded.

The dynamic equation of (23) has the following properties [18] that will be used in the controller development and analysis.

Property 1: The inertia matrix $M(q)$ is symmetric, positive-definite, and satisfies the following inequalities

$$m_1 \|\xi\|^2 \leq \xi^T M(q) \xi \leq m_2 \|\xi\|^2 \quad \forall \xi \in \mathbb{R}^n \quad (24)$$

where m_1, m_2 are known positive constants, and $\|\cdot\|$ denotes the standard Euclidean norm.

Property 2: The inertia and centripetal-Coriolis matrices satisfy the following skew-symmetric relationship

$$\xi^T \left(\frac{1}{2} \dot{M}(q) - V_m(q, \dot{q}) \right) \xi = 0 \quad \forall \xi \in \mathbb{R}^n \quad (25)$$

where $\dot{M}(q)$ denotes the time derivative of the inertia matrix.

Property 3: The norm of the centripetal-Coriolis, gravity, and viscous friction terms of (23) can be upper bounded as follows

$$\|V_m(q, \dot{q})\|_{i_\infty} \leq \zeta_{c1} \|\dot{q}\|, \quad \|G(q)\| \leq \zeta_g, \quad \|F_d\|_{i_\infty} \leq \zeta_{fd} \quad (26)$$

where $\zeta_{c1}, \zeta_g, \zeta_{fd} \in \mathbb{R}^1$ denote known positive bounding constants, and $\|\cdot\|_{i_\infty}$ denotes the infinity-norm of a matrix.

In addition to the above properties, we will also make the following assumption with regard the static friction effects that are contained in (23).

Assumption 3: The static friction terms given in (23) can be linear parameterized as follows

$$Y_s(\dot{q})\theta_s = F_s(\dot{q}) \quad (27)$$

where $\theta_s \in \mathbb{R}^n$ contains the unknown, constant static friction coefficients, and the regression matrix $Y_s(\dot{q}) \in \mathbb{R}^{n \times n}$ contains known functions of the link velocity $\dot{q}(t) \in \mathbb{R}^n$.

3.2 Control Objective

The control objective is to design a global link position tracking controller despite parametric uncertainty in the dynamic model given in (23). To quantify this objective, we define the link position tracking error $e(t) \in \mathbb{R}^n$ as follows

$$e = q_d - q \quad (28)$$

where we assume that $q_d(t) \in \mathbb{R}^n$ and its first two time derivatives are assumed to be bounded, periodic functions of time with a known period T such that

$$q_d(t) = q_d(t - T) \quad \dot{q}_d(t) = \dot{q}_d(t - T) \quad \ddot{q}_d(t) = \ddot{q}_d(t - T). \quad (29)$$

In addition, we define the difference between the actual parameter vector and the estimated parameter vector as follows

$$\tilde{\theta}_s = \theta_s - \hat{\theta}_s \quad (30)$$

where $\tilde{\theta}_s(t) \in \mathbb{R}^n$ represents a parameter estimation error vector and $\hat{\theta}_s(t) \in \mathbb{R}^n$ denotes a subsequently designed estimate of θ_s .

3.3 Control Formulation

To facilitate the subsequent control development and stability analysis, we reduce the order of the dynamic expression given in (23) by defining a filtered tracking error-like variable $r(t) \in \mathbb{R}^n$ as follows

$$r = \dot{e} + \alpha e \quad (31)$$

where $\alpha \in \mathbb{R}^1$ is a positive constant control gain. After taking the time derivative of (31), pre-multiplying the resulting expression by $M(q)$, utilizing (23) and (28), and then performing some algebraic manipulation, we obtain the following expression

$$M\dot{r} = -V_m r + w_r + \chi + Y_s \theta_s - \tau \quad (32)$$

where the auxiliary expressions $w_r(t), \chi(t) \in \mathbb{R}^n$ are defined as follows

$$w_r = M(q_d)\ddot{q}_d + V_m(q_d, \dot{q}_d)\dot{q}_d + G(q_d) + F_d\dot{q}_d \quad (33)$$

$$\chi = M(q)(\ddot{q}_d + \alpha\dot{e}) + V_m(q, \dot{q})(\dot{q}_d + \alpha e) + G(q) + F_d\dot{q} - w_r. \quad (34)$$

By exploiting Properties 1 and 3 of the robot dynamics, and then using (28) and (31), we can utilize the results given in [24] to prove that

$$\|\chi\| \leq \rho(\|z\|)\|z\| \quad (35)$$

where the auxiliary signal $z(t) \in \mathbb{R}^{2n}$ is defined as

$$z(t) = [e(t) \quad r(t)]^T \quad (36)$$

and $\rho(\cdot) \in \mathbb{R}^1$ is a known, positive bounding function. Furthermore, based on the expression given in (33) and the boundedness assumptions with regard to the robot dynamics and the desired trajectory, it is clear that

$$|w_{ri}(t)| \leq \beta_{ri} \quad \text{for } i = 1, 2, \dots, n \quad (37)$$

where $\beta_r = [\beta_{r1}, \dots, \beta_{rn}] \in \mathbb{R}^n$ is a vector of known, positive bounding constants.

Given the open-loop error system in (32), we design the following control input

$$\tau = kr + k_n \rho^2(\|z\|)r + e + \hat{w}_r + Y_s \hat{\theta}_s \quad (38)$$

where $k, k_n \in \mathbb{R}^1$ are positive constant control gains, $\rho(\cdot)$ was defined in (35), $\hat{w}_r(t) \in \mathbb{R}^n$ is generated on-line according to the following learning-based algorithm

$$\hat{w}_r(t) = \text{sat}_{\beta_r}(\hat{w}_r(t - T)) + k_L r \quad (39)$$

$k_L \in \mathbb{R}^1$ is a positive, constant control gain, $\text{sat}_{\beta_r}(\cdot)$ is defined in the same manner as in (13), and the parameter estimate vector $\hat{\theta}_s(t) \in \mathbb{R}^n$ is generated on-line according to the following gradient-based adaptation algorithm

$$\dot{\hat{\theta}}_s(t) = \Gamma_s Y_s^T r \quad (40)$$

where $\Gamma_s \in \mathbb{R}^{n \times n}$ is a constant, diagonal, positive-definite, adaptation gain matrix.

To develop the closed-loop error system for $r(t)$, we substitute (38) into (32) to obtain the following expression

$$M\dot{r} = -V_m r - kr - e + Y_s \tilde{\theta}_s + \tilde{w}_r + \chi - k_n \rho^2(\|z\|)r \quad (41)$$

where $\tilde{\theta}_s(t)$ was defined in (30), and $\tilde{w}_r(t)$ is a learning estimation error signal defined as follows

$$\tilde{w}_r = w_r - \hat{w}_r. \quad (42)$$

After substituting (39) into (42) for $\hat{w}_r(t)$, utilizing the fact that $w_r(t)$ is periodic, and then utilizing (37) to construct the following equality

$$w_r(t) = \text{sat}_{\beta_r}(w_r(t)) = \text{sat}_{\beta_r}(w_r(t - T)), \quad (43)$$

we can rewrite (42) in the following form

$$\tilde{w}_r = \text{sat}_{\beta_r}(w_r(t - T)) - \text{sat}_{\beta_r}(\hat{w}_r(t - T)) - k_L r. \quad (44)$$

3.4 Stability Analysis

Theorem 2 *Given the robot dynamics of (23), the proposed hybrid adaptive/learning controller given in (38), (39), and (40), ensures global asymptotic link position tracking in the sense that*

$$\lim_{t \rightarrow \infty} e(t) = 0 \quad (45)$$

where the control gains α , k , k_n , and k_L introduced in (31), (38), and (39) must be selected to satisfy the following sufficient condition

$$\min \left(\alpha, k + \frac{k_L}{2} \right) > \frac{1}{4k_n}. \quad (46)$$

Proof: To prove Theorem 2, we define a non-negative function $V_3(t) \in \mathbb{R}^1$ as follows

$$\begin{aligned} V_3 &= \frac{1}{2}e^T e + \frac{1}{2}r^T M r + \frac{1}{2}\tilde{\theta}_s^T \Gamma_s^{-1} \tilde{\theta}_s \\ &\quad + \frac{1}{2k_L} \int_{t-T}^t [\text{sat}_{\beta_r}(w_r(\tau)) - \text{sat}_{\beta_r}(\hat{w}_r(\tau))]^T [\text{sat}_{\beta_r}(w_r(\tau)) - \text{sat}_{\beta_r}(\hat{w}_r(\tau))] d\tau. \end{aligned} \quad (47)$$

After taking the time derivative of (47), we obtain the following expression

$$\begin{aligned} \dot{V}_3 &= e^T (r - \alpha e) + r^T \left(-kr - e - k_n \rho^2 (\|z\|) r + Y_s \tilde{\theta}_s + \tilde{w}_r + \chi \right) - \tilde{\theta}_s^T Y_s^T r \\ &\quad + \frac{1}{2k_L} [\text{sat}_{\beta_r}(w_r(t)) - \text{sat}_{\beta_r}(\hat{w}_r(t))]^T [\text{sat}_{\beta_r}(w_r(t)) - \text{sat}_{\beta_r}(\hat{w}_r(t))] \\ &\quad - \frac{1}{2k_L} [\text{sat}_{\beta_r}(w_r(t-T)) - \text{sat}_{\beta_r}(\hat{w}_r(t-T))]^T [\text{sat}_{\beta_r}(w_r(t-T)) - \text{sat}_{\beta_r}(\hat{w}_r(t-T))] \end{aligned} \quad (48)$$

where (25), (31), (40), and (41) were utilized. After utilizing (35), (37), (44), and then simplifying the resulting expression, we can rewrite (48) as follows

$$\begin{aligned} \dot{V}_3 &\leq -\alpha e^T e - kr^T r + r^T \tilde{w}_r + [\rho(\|z\|) \|z\| \|r\| - k_n \rho^2(\|z\|) \|r\|^2] \\ &\quad - \frac{1}{2k_L} (\tilde{w}_r + k_L r)^T (\tilde{w}_r + k_L r) \\ &\quad + \frac{1}{2k_L} [\text{sat}_{\beta_r}(w_r(t)) - \text{sat}_{\beta_r}(\hat{w}_r(t))]^T [\text{sat}_{\beta_r}(w_r(t)) - \text{sat}_{\beta_r}(\hat{w}_r(t))]. \end{aligned} \quad (49)$$

After expanding the second line of (49) and then cancelling common terms, we obtain the following expression

$$\begin{aligned} \dot{V}_3 &\leq -\alpha e^T e - \left(k + \frac{k_L}{2} \right) r^T r + [\rho(\|z\|) \|z\| \|r\| - k_n \rho^2(\|z\|) \|r\|^2] \\ &\quad - \frac{1}{2k_L} \left[\tilde{w}_r^T \tilde{w}_r - [\text{sat}_{\beta_r}(w_r(t)) - \text{sat}_{\beta_r}(\hat{w}_r(t))]^T [\text{sat}_{\beta_r}(w_r(t)) - \text{sat}_{\beta_r}(\hat{w}_r(t))] \right]. \end{aligned} \quad (50)$$

By exploiting the property given in (14), completing the square on the bracketed term in the first line of (50) (or simply utilizing the nonlinear damping tool [19]), and then utilizing the definition of $z(t)$ given in (36), we can simplify the expression given in (50) as shown below

$$\dot{V}_3 \leq - \left(\min \left(\alpha, k + \frac{k_L}{2} \right) - \frac{1}{4k_n} \right) \|z\|^2. \quad (51)$$

Based on (36), (46), (47), and (51), it is clear that $e(t), r(t) \in \mathcal{L}_2 \cap \mathcal{L}_\infty$. Based on the fact that $r(t) \in \mathcal{L}_\infty$, it is clear from (31), (39), and (44) that $\hat{w}_r(t), \tilde{w}_r(t), \dot{e}(t) \in \mathcal{L}_\infty$, and hence, $e(t)$ is uniformly continuous. Since $e(t) \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and uniformly continuous, we can utilize Barbalat's Lemma [18] to prove (45). ■

Remark 2 *From the previous stability analysis, it is again clear that we exploit the fact that the learning-based feedforward term given in (39) is composed of a saturation function. That is, it is easy to see from the structure of (39), that if $r(t) \in \mathcal{L}_\infty$ then $\hat{w}_r(t) \in \mathcal{L}_\infty$.*

4 Experimental Results

To illustrate the effectiveness of the proposed learning-based controller, the following controller² was implemented on a two-link direct drive, planar robot manipulator manufactured by Integrated Motion Inc. [23]

$$\tau = kr + \hat{w}_r \quad (52)$$

where $r(t)$ was defined in (31) and $\hat{w}_r(t)$ is generated according to (39). The two-link robot is directly actuated by switched-reluctance motors. A Pentium 266 MHz PC running RT-Linux (real-time extension of Linux OS) hosted the control algorithm. The Matlab/Simulink environment with Real Time Linux Target [33] for RT-Linux was used to implement the controller. The Servo-To-Go I/O board provided for data transfer between the computer subsystem and the robot. The two-link IMI robot has the following dynamic model [23]

$$\begin{aligned} \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} &= \begin{bmatrix} p_1 + 2p_3c_2 & p_2 + p_3c_2 \\ p_2 + p_3c_2 & p_2 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} -p_3s_2\dot{q}_2 & -p_3s_2(\dot{q}_1 + \dot{q}_2) \\ p_3s_2\dot{q}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \\ &+ \begin{bmatrix} f_{d1} & 0 \\ 0 & f_{d2} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} f_{s1} & 0 \\ 0 & f_{s2} \end{bmatrix} \begin{bmatrix} \text{sgn}(\dot{q}_1) \\ \text{sgn}(\dot{q}_2) \end{bmatrix} \end{aligned} \quad (53)$$

where $p_1 = 3.473$ [kg-m²], $p_2 = 0.193$ [kg-m²], $p_3 = 0.242$ [kg-m²], $f_{d1} = 5.3$ [Nm-sec], $f_{d2} = 1.1$ [Nm-sec], $f_{s1} = 8.45$ [Nm], $f_{s2} = 2.35$ [Nm], $\text{sgn}(\cdot)$ denotes the standard signum function, $c_2 \triangleq \cos(q_2)$, and $s_2 \triangleq \sin(q_2)$. The experiment was performed using the following periodic desired position trajectory (see Figure 1)

$$\begin{bmatrix} q_{d1}(t) \\ q_{d2}(t) \end{bmatrix} = \begin{bmatrix} (0.8 + 0.2 \sin(0.5t)) \sin(0.5 \sin(0.5t)) (1 - \exp(-0.6t^3)) \\ (0.6 + 0.2 \sin(0.5t)) \sin(0.5 \sin(0.5t)) (1 - \exp(-0.6t^3)) \end{bmatrix} \text{ [rad]} \quad (54)$$

where the exponential term was included to provide a “smooth-start” to the system.

The experiment was performed at a control frequency of 1KHz. After a tuning process, the control gains were selected as follows

$$k = \text{diag}\{40, 12\}, \quad \alpha = \text{diag}\{20, 14\}, \quad k_L = \text{diag}\{30, 10\} \quad (55)$$

where $\text{diag}\{\cdot\}$ denotes the diagonal elements of a matrix. Note that in previous sections the control gains k , α , and k_L are defined as scalars for simplicity, whereas in (55), the control gains are selected as diagonal matrices to facilitate the “tuning” process. The link position tracking errors are depicted in Figure 2. Note that the tracking error reduces after each period of the desired trajectory. The control torque input for each link motor is shown in Figure 3 where the learning component of the controller is given in Figure 4.

²During experimental trials, we determined that the proposed learning-based controller did not require the non-linear damping term and the adaptive feedforward term utilized in (38) to provide good link position tracking performance.

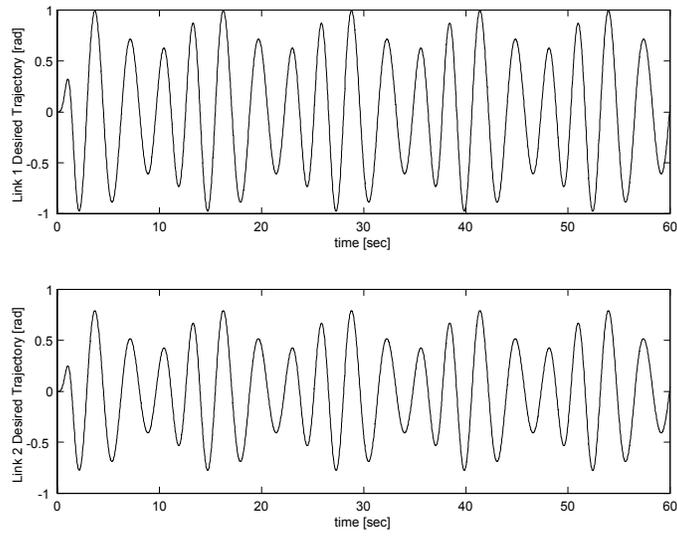


Figure 1: Desired Trajectory

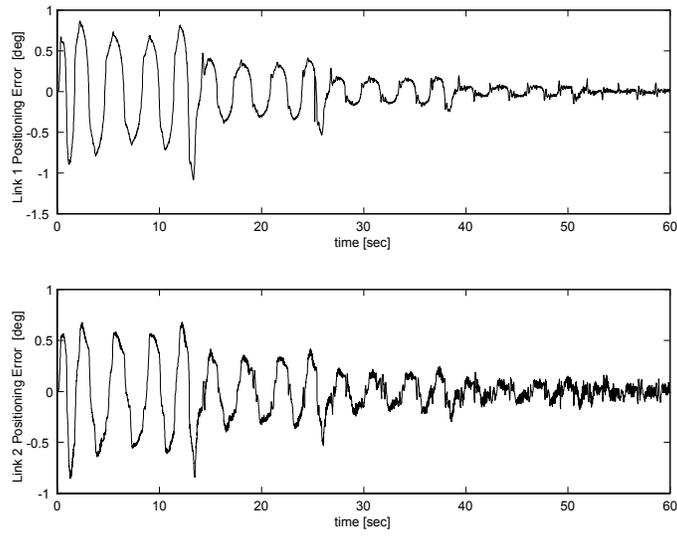


Figure 2: Link Position Tracking Error

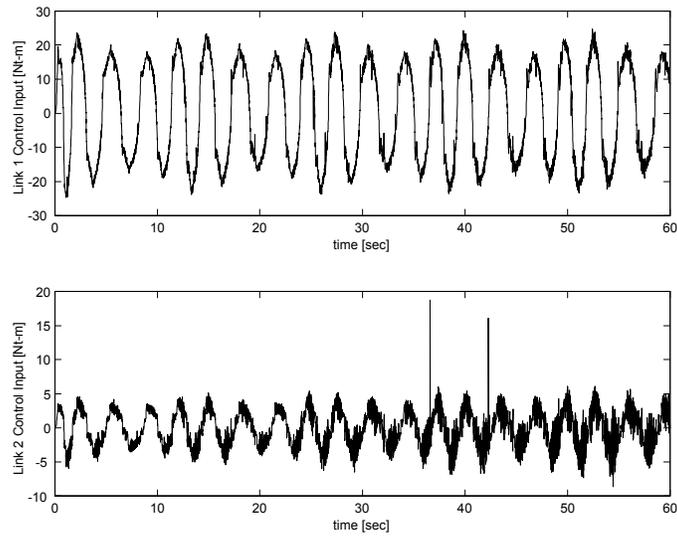


Figure 3: Control Torque Input

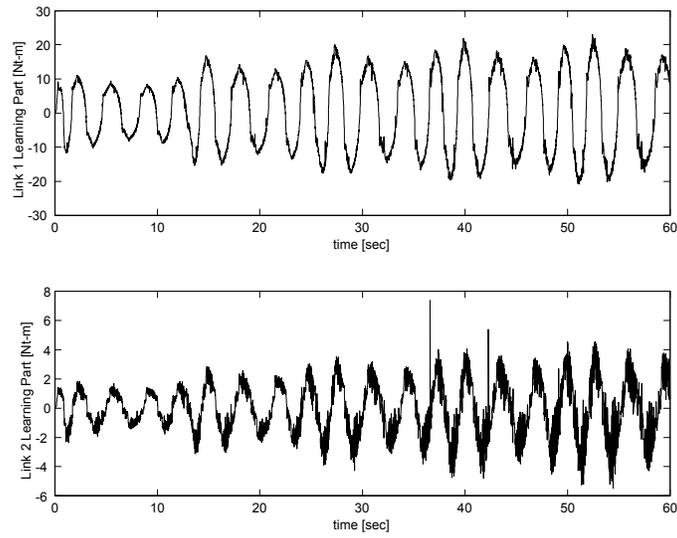


Figure 4: Learning-Based Feedforward Component of the Control Input

5 Conclusion

In this paper, we illustrate how a learning-based estimate can be used to achieve asymptotic tracking in the presence of a nonlinear disturbance. Based on the fact that the learning-based controller estimate is generated from a Lyapunov-based stability analysis, we also demonstrated how additional Lyapunov-based design techniques can be utilized to reject components of the unknown dynamics which are not periodic. Specifically, we designed a hybrid adaptive/learning controller for the robot manipulator dynamics. Experimental results illustrated that the link tracking performance of a 2-link robot manipulator improved at each period of the desired trajectory due to the mitigating action of the learning estimate.

References

- [1] S. Arimoto, “Robustness of Learning Control for Robot Manipulators”, *Proceedings of the IEEE International Conference on Robotics and Automation*, 1990.
- [2] S. Arimoto, S. Kawamura, and F. Miyazaki, “Bettering Operation of Robots by Learning”, *Journal of Robotic Systems*, Vol. 1, No. 2, pp. 123-140, 1984.
- [3] S. Arimoto, S. Kawamura, and F. Miyazaki, “Realization of Robot Motion Based on a Learning Method”, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 18, No. 1, pp. 126-134, 1988.
- [4] C. C. Cheah, D. Wang, and Y. C. Soh, “Convergence and Robustness of A Discrete-Time Learning Control Scheme for Constrained Manipulators”, *Journal of Robotic Systems*, Vol. 11, No. 3, pp. 223-238, 1994.
- [5] C. C. Cheah, D. Wang, and Y. C. Soh, “Learning Control of Motion and Force for Constrained Robotic Manipulators”, *International Journal of Robotics and Automation*, Vol. 10, No. 3, pp. 79-88, 1995.
- [6] C. C. Cheah, D. Wang, and Y. C. Soh, “Learning Control for a Class of Nonlinear Differential-Algebraic Systems with Application to Constrained Robots”, *Journal of Robotic Systems*, Vol. 13, No. 3, pp. 141-151, 1996.
- [7] C. C. Cheah and D. Wang, “A Model-Reference Learning Control Scheme for a Class of Nonlinear Systems”, *International Journal of Control*, Vol. 66, No. 2, pp. 271-287, 1997.
- [8] S.-I. Cho and I.-J. Ha, “A Learning Approach to Tracking in Mechanical Systems with Friction”, *IEEE Transactions on Automatic Control*, Vol. 45, No. 1, pp. 111-116, 2000.
- [9] B. Costic, M. S. de Queiroz, and D.M. Dawson, “A New Learning Control Approach to the Active Magnetic Bearing Benchmark System”, *Proc. of the American Control Conference*, Chicago, IL, pp. 2639-2643, 2000.
- [10] C. Ham, Z. Qu, and R. Johnson, “A Nonlinear Iterative Learning Control for Robot Manipulators in the Presence of Actuator Dynamics”, *International Journal of Robotics and Automation*, in press.

- [11] C. Ham, Z. Qu, and J. Kaloust, “Nonlinear Learning Control for a Class of Nonlinear Systems”, *Automatica*, to appear.
- [12] S. H. Han, Y. H. Kim, and I.-J. Ha, “Iterative Identification of State-Dependent Disturbance Torque for High Precision Velocity Control of Servo Motors”, *IEEE Transactions on Automatic Control*, Vol. 43, No. 5, pp. 724-729, 1998.
- [13] S. Hara, Y. Yamamoto, T. Omata, and M. Nakano, “Control Systems: A New Type Servo System for Periodic Exogenous Signals”, *IEEE Transactions on Automatic Control*, Vol. 33, No. 7, pp. 659-668, 1988.
- [14] D. Heinzinger, B. Fenwick, B. Paden, and F. Miyazaki, “Robot Learning Control”, *Proceedings of the IEEE Conference on Decision and Control*, 1989.
- [15] R. Horowitz, “Learning Control of Robot Manipulators”, *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 115, pp. 402-411, 1993.
- [16] Y.-H. Kim and I.-J. Ha, “A Learning Approach to Precision Speed Control of Servomotors and Its Application to a VCR”, *IEEE Transactions on Automatic Control*, Vol. 43, No. 5, pp. 724-729, 1998.
- [17] Y.-H. Kim and I.-J. Ha, “Asymptotic State Tracking in a Class of Nonlinear Systems via Learning-Based Inversion”, *IEEE Transactions on Automatic Control*, Vol. 45, No. 11, pp. 2011-2027, 2000.
- [18] F. Lewis, C. Abdallah, and D. Dawson, *Control of Robot Manipulators*, New York: MacMillan Publishing Co., 1993.
- [19] M. Krstic, I. Kanellakopoulos, P. Kokotovic, *Nonlinear and Adaptive Control Design*, New York: John Wiley and Sons, Inc., 1995.
- [20] W. Messner, R. Horowitz, W.-W. Kao, and M. Boals, “A New Adaptive Learning Rule”, *IEEE Transactions on Automatic Control*, Vol. 36, No. 2, pp. 188-197, 1991.
- [21] K. L. Moore “Iterative Learning Control – An Expository Overview”, *Applied and Computational Controls, Signal Processing, and Circuits*, Vol. 1, pp. 151-214, 1999.
- [22] K. L. Moore, *Iterative Learning Control for Deterministic Systems*, Springer-Verlag, Advances in Industrial Control, London, U.K., 1993.
- [23] *Direct Drive Manipulator Research and Development Package Operations Manual*, Integrated Motion Inc., Berkeley, CA, 1992.
- [24] N. Sadegh and R. Horowitz, “Stability and Robustness Analysis of a Class of Adaptive Controllers for Robotic Manipulators”, *Int. J. Robot. Res.*, Vol. 9, No. 9, pp. 74-92, June, 1990.
- [25] N. Sadegh, R. Horowitz, W. W. Kao, and M. Tomizuka, “A Unified Approach to the Design of Adaptive and Controllers for Robot Manipulators”, *ASME Journal on Dynamic Systems, Measurement, and Control*, Vol. 112, pp. 618-629, 1990.
- [26] M. Sun, D. Wang, “Sampled-Data Iterative Learning Control for Nonlinear Systems with Arbitrary Relative Degree”, *Automatica*, Vol. 37, pp. 283-289, 2001.

- [27] M. Tomizuka, “On the Design of Digital Tracking Controllers”, *ASME Journal on Dynamic Systems, Measurement, and Control*, 50th Anniversary Issues of the ASME Journal on Dynamic Systems, Measurement, and Control, 1992.
- [28] M. Tomizuka, T. Tsao, and K. Chew, “Discrete-Time Domain Analysis and Synthesis of Controllers”, *ASME Journal on Dynamic Systems, Measurement, and Control*, Vol. 111, pp. 353-358, 1989.
- [29] M. Tsai, G. Anwar, and M. Tomizuka, “Discrete-Time Control for Robot Manipulators”, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1341-1347, 1988.
- [30] D. Wang, “On D-type and P-type ILC Designs and Anticipatory Approach”, *International Journal of Control*, Vol. 23, No. 10, pp. 890-901, 2000.
- [31] D. Wang, Y. C. Soh, and C. C. Cheah, “Robust Motion and Force Control of Constrained Manipulators by Learning”, *Automatica*, Vol. 31, No. 2, pp. 257-262, 1995.
- [32] J.-X. Xu and Z. Qu, “Robust Iterative Learning Control for a Class of Nonlinear Systems”, *Automatica*, Vol. 34, No. 8, pp. 983-988, 1998.
- [33] Z. Yao, N. P. Costescu, S. P. Nagarkatti, and D. M. Dawson, “Real-Time Linux Target: A MATLAB-Based Graphical Control Environment”, *Proc. of the IEEE Conference on Control Applications*, Anchorage, AK, pp. 173-178, September 2000.

A Inequality Proof

To prove the inequality given in (14), we divide the proof into three possible cases as follows.

Case 1: $|\xi_{1i}| \leq \beta_i, |\xi_{2i}| \leq \beta_i$

From the definition of $\text{sat}_{\beta_i}(\cdot)$ given in (13), we can see that for this case

$$\text{sat}_{\beta_i}(\xi_{1i}) = \xi_{1i} \quad \text{sat}_{\beta_i}(\xi_{2i}) = \xi_{2i}. \quad (56)$$

After substituting (56) into (14), we obtain the following expression

$$(\xi_{1i} - \xi_{2i})^2 = (\text{sat}_{\beta_i}(\xi_{1i}) - \text{sat}_{\beta_i}(\xi_{2i}))^2 \quad \text{for } |\xi_{1i}| \leq \beta_i, |\xi_{2i}| \leq \beta_i \quad (57)$$

hence, the inequality given in (14) is true for Case 1.

Case 2a: $|\xi_{1i}| \leq \beta_i, \xi_{2i} > \beta_i$

From the definition of $\text{sat}_{\beta_i}(\cdot)$ given in (13), it is clear for this case that

$$(\xi_{2i} + \beta_i) \geq 2\xi_{1i} \quad \text{for } |\xi_{1i}| \leq \beta_i, \xi_{2i} > \beta_i. \quad (58)$$

After multiplying both sides of (58) by $(\xi_{2i} - \beta_i)$ and then simplifying the left-hand side of the inequality, we can rewrite (58) as follows

$$\xi_{2i}^2 - \beta_i^2 \geq 2(\xi_{2i} - \beta_i)\xi_{1i} \quad (59)$$

where we have utilized the fact that $\xi_{2i} - \beta_i > 0$ for this case. After adding the term ξ_{1i}^2 to both sides of (59) and then rearranging the resulting expression, we obtain the following expression

$$\xi_{1i}^2 - 2\xi_{1i}\xi_{2i} + \xi_{2i}^2 \geq \xi_{1i}^2 - 2\beta_i\xi_{1i} + \beta_i^2. \quad (60)$$

Based on the expression given in (60), we can utilize the facts that

$$\text{sat}_{\beta_i}(\xi_{1i}) = \xi_{1i} \quad \text{sat}_{\beta_i}(\xi_{2i}) = \beta_i \quad (61)$$

to prove that

$$(\xi_{1i} - \xi_{2i})^2 \geq (\text{sat}_{\beta_i}(\xi_{1i}) - \text{sat}_{\beta_i}(\xi_{2i}))^2 \quad \text{for } |\xi_{1i}| \leq \beta_i, \xi_{2i} > \beta_i. \quad (62)$$

Case 2b: $|\xi_{1i}| \leq \beta_i, \xi_{2i} < -\beta_i$

From the definition of $\text{sat}_{\beta_i}(\cdot)$ given in (13), it is clear for this case that

$$(\xi_{2i} - \beta_i) \leq 2\xi_{1i} \quad \text{for } |\xi_{1i}| \leq \beta_i, \xi_{2i} < -\beta_i. \quad (63)$$

After multiplying both sides of (63) by $(\xi_{2i} + \beta_i)$ and then simplifying the left-hand side of the inequality, we can rewrite (63) as follows

$$\xi_{2i}^2 - \beta_i^2 \geq 2(\xi_{2i} + \beta_i)\xi_{1i} \quad (64)$$

where we have utilized the fact that $\xi_{2i} + \beta_i < 0$ for this case. After adding the term ξ_{1i}^2 to both sides of (64) and then rearranging the resulting expression, we obtain the following expression

$$\xi_{1i}^2 - 2\xi_{1i}\xi_{2i} + \xi_{2i}^2 \geq \xi_{1i}^2 + 2\beta_i\xi_{1i} + \beta_i^2. \quad (65)$$

Based on the expression given in (65), we can utilize the facts that

$$\text{sat}_{\beta_i}(\xi_{1i}) = \xi_{1i} \quad \text{sat}_{\beta_i}(\xi_{2i}) = -\beta_i \quad (66)$$

to prove that

$$(\xi_{1i} - \xi_{2i})^2 \geq (\text{sat}_{\beta_i}(\xi_{1i}) - \text{sat}_{\beta_i}(\xi_{2i}))^2 \quad \text{for } |\xi_{1i}| \leq \beta_i, \xi_{2i} < -\beta_i; \quad (67)$$

hence, we have proven that (13) is true for all possible cases.