Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

A Note on Iterative Learning Control for Nonlinear Systems with Input Uncertainties

Y. Tan * and **J.X. Xu** **

* *The Department of Electrical & Electronic Engineering, The University of Melbourne, Parkview, VIC 3010, Australia. (e-mail: y.tan@ee.unimelb.edu.au).*
** *The Electrical and Computer Engineering Department, National University of Singapore, Singapore 117576. (e-mail: elexujx@nus.edu.sg)*

Abstract: The problem of designing an iterative learning controller in the presence of input uncertainties is of great importance in practical implementations. This paper addresses this important issue for a simple scalar nonlinear dynamic system with general input uncertainties. A dual iterative learning loop is applied to systems to "learn" both unknown dynamics and static input uncertainties respectively and can ensure that the output of the system converges to the desired trajectory. Two analytic examples show that the proposed dual learning control scheme can work well under input uncertainties such as saturation and dead zone.

1. INTRODUCTION

A special control problem is considered in this paper: the control task in a finite time interval $[0, T]$ repeats itself along iteration domain (see Xu and Tan (2003) and references herein). This kind of control problem is frequently encountered in many industrial processes such as assembly lines and chemical batch processes. Intuitively, the information obtained from last iteration would be used to improve the control performance of this iteration. Iterative learning control (ILC) is well-known for its ability to improve the control performance along the iteration domain as it takes advantage of the repeatable control environment.

Since Arimoto published his first paper on ILC (Arimoto et al. (1984)), ILC methodologies have become the focus of many researchers over the last three decades (see, for instance,Moore (1993), Saab (1994), Amann et al. (1996), Chen et al. (1998), Moore (1999), Longman (2000), Chin et al. (2004), Norrlof and Gunnarsson (2005), Saab (2005), Sugie and Sakai (2007) and references herein), leading to numerous practical implementations of ILC schemes, as well as better theoretical understanding.

The focus of this paper to design ILC schemes for nonlinear dynamic systems with input uncertainties. Input uncertainties are quite common phenomenon in engineering applications. Examples of input uncertainties include saturation, deadzone, hysteresis and so on. The existence of these input uncertainties may severely deteriorate the control performance or cause oscillations, even lead to system instability.

In order to deal with unknown dynamics as well as unknown input uncertainties, a dual iterative learning loop is proposed. Loop 1 is a normal ILC scheme, which can ensure the convergence of output of the dynamics without input uncertainties. Loop 2 is another ILC scheme to deal with static input uncertainties. The input signal of loop 1 becomes the "desired output" for the loop 2. ILC scheme in loop 2 drives the output of static mapping (input uncertainties) to this "desired output" obtained from loop 1. It should be noted that in numerical literature, many iterative numerical algorithms are available to drive the output of an unknown static mapping to the desired one (see Ortega and Rheinboldt (1970) and references herein). In this paper, those available numerical methods are used in loop 2 to deal with input uncertainties (unknown static mapping) and incorporating with an ILC scheme in loop 1 to ensure the desired tracking performance.

It is worthwhile to highlight, among numerous available numerical algorithms that can drive the output of unknown static mapping to the desired one, in order to ensure the convergence of the system (dynamics with input uncertainties), some numerical algorithms which have "nice" convergence properties are employed. Two scenarios are discussed here: numerical algorithms that converge either monotonically (Assumption 2) or non-monotonically (Assumption 3). Our first result (Theorem 1) shows that when the numerical algorithm converges monotonically, the proposed dual ILC loop can ensure the perfect tracking performance[1]. The second result (Theorem 2) states a sufficient condition that can ensure the perfect tracking performance with numerical algorithms that is not monotonically convergent, but satisfies Assumption 3.

In order to illustrate how to design the dual ILC loop, two analytical examples, in which the input uncertainty is chosen as saturation and deadzone respectively, are employed to show the effectiveness, therefore, providing some insight in the proposed scheme.

[1] Using the terminology in Xu and Tan (2003), the perfect tracking performance means that the tracking error converges to 0 pointwisely on the compact time interval $[0, T]$ when the number of iteration i approaches to infinity.

This paper is organized as follows. Problem formulation and preliminaries are provided in Section 2. Main results as well as two analytic examples are stated in Section 3 followed by a summary. Some proofs are presented in the Appendix.

2. PROBLEM FORMULATION AND PRELIMINARIES

In this paper, the set of integers is denoted as $\mathbb{N}$. i is the number of iteration and $i \in \mathbb{N}$. the set of real numbers is denoted as $\mathbb{R}$.

The following single-input-single-output nonlinear uncertain dynamic system is considered:

$$\dot{x} = \eta(x,t) + u, \tag{1}$$
$$u = f(v, \boldsymbol{\theta}), \tag{2}$$

where $\eta(x,t)$ is a lumped uncertainty, continuous in t and global Lipschitz continuous in x, i.e.

$$|\eta(x_1,t) - \eta(x_2,t)| \leq L_\eta |x_1 - x_2|,$$

where L_η is a Lipschitz constant. $v \in \mathbb{R}$ is a controller input where $u \in \mathbb{R}$ is the output of the controller (or the input of the system (1)). $f(v, \boldsymbol{\theta})$ is a static input uncertainty (the mapping between controller input and output) that is parameterized by an parameter vector $\boldsymbol{\theta}$, where $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^m$ and it is not completely known (see Example 2).

Remark 1. A very simple scalar dynamic system is considered to simplify the presentation. The same result can be extended to more general multi-input-multi-output systems with slight modification.

Remark 2. If the input uncertainty is a saturation function, $f(v, \boldsymbol{\theta})$ takes the following form

$$f(v,\boldsymbol{\theta}) = f_s(v, v_{max}) = \begin{cases} v_{max} & \text{if } \ v \geq v_{max} \\ v & \text{if } \ |v| \leq v_{max} \\ -v_{max} & \text{if } \ -v \leq -v_{max} \end{cases}, \tag{3}$$

with $\boldsymbol{\theta} = v_{max}$ is a positive constant that determines the saturation bound.

The dynamics (1) satisfies the following identical initialization condition

Assumption 1. $x_i(0) = x_1(0) = x^r(0)$, for all $i \in \mathbb{N}$.

Remark 3. Due to space limitation, robustness with respect to measurement errors as well initial condition errors are not addressed in this paper, though robustness is of great importance in ILC design.

Control objective is to find a desired control input $v^r(t)$ which realizes

$$\begin{aligned} \dot{x}^r(t) &= \eta(x^r, t) + u^r(t), \\ u^r(t) &= f(v^r(t), \boldsymbol{\theta}), \forall t \in [0,T]. \end{aligned} \tag{4}$$

Denote the tracking error at i^{th} iteration as $e_i = x^r - x_i$, the error dynamic of the system becomes

$$\dot{e}_i = [\eta(x^r,t) + f(v^r(t), \boldsymbol{\theta})] - [\eta(x_i,t) + u_i(t)]. \tag{5}$$

Aim of loop 1 ILC design: loop 1 ILC is to find a sequence of $u_i(t)$ to ensure the perfect tracking performance of (1). In other words, when there is no input uncertainties, i.e., $u = f(v, \boldsymbol{\theta}) = v$, loop 1 ILC iteratively updates $u_i(t)$ to achieve perfect tracking performance.

A very simple ILC scheme is used in loop 1 as follows

$$u_i(t) = u_{i-1}(t) + q e_i(t), \tag{6}$$

where $q > 0$ is the learning gain. The following proposition shows that the perfect tracking can be achieved.

Proposition 1. Assume that Assumption 1 holds true for the system (1), then the perfect tracking performance is achieved with updating laws (6).

Proof: see Appendix. ∘

The updating law (6) iteratively modifies $u_i(t)$ which is the output of the actuator, instead of the input of the actuator. The static mapping between the actuator's input and output is not known (input uncertainty). The next step is find an appropriate sequence of the actuator's input $v_i(t)$ so as to move the output of the actuator to the desired one $u_i(t)$.

Aim of loop 2 ILC design: loop 2 ILC is to find a sequence of input $v_i(t)$ such that the output of unknown static mapping will move to the desired output $u_i(t)$. In other words, we can re-formulate the aim of loop 2 ILC as:

> Given $u_i(t)$ obtained from loop 1 ILC (see, (6)), how to update the input of actuator v_i such that when $i \to \infty$, $f(v_i(t), \boldsymbol{\theta}) \to u_i(t) \to u^r(t)$.

In order to better understand the above problem, let $t = t_0 \in [0,t]$, $u_i(t_0) = y$, the above problem becomes to find "z" such that a nonlinear mapping $f(z, \boldsymbol{\theta}) = y$ holds. This problem has been well addressed in numerical literature (see Ortega and Rheinboldt (1970)) and there are many numerical algorithm available to solve this problem. In general, the updating law of the above problem takes the following form

$$z_i = h(z_{i-1}, \Delta_{i-1}), \tag{7}$$

where $h(\cdot,\cdot)$ is a linear/nonliear mapping, $\Delta_{i-1} = y - f(z_{i-1}, \boldsymbol{\theta})$. In this paper, without losing generality, it is assumed that the mapping $h(\cdot,\cdot)$ exists and satisfies some conditions. Later, the existence of such a mapping is illustrated by two illustrative examples.

Assumption 2. Let $y \in \mathbb{R}$. For a static mapping $f(z, \boldsymbol{\theta})$, there exists a mapping $h(\cdot,\cdot)$ such that there exists $\rho \in (0,1)$, such that

$$|\Delta_i| \leq \rho |\Delta_{i-1}|. \tag{8}$$

Remark 4. Assumption 2 implies that there exists a numerical algorithm (7) to ensure that $f(z_i, \boldsymbol{\theta})$ converges to y monotonically, i.e.,

$$|\Delta_i| \leq \rho^i |\Delta_0| \Rightarrow \lim_{i \to \infty} |\Delta_i| = 0. \tag{9}$$

Remark 5. This inequality (8) is widely used in numerical literature (Ortega and Rheinboldt (1970)). For example, if the static mapping $f(z, \boldsymbol{\theta})$ satisfies

$$\frac{df(z,\boldsymbol{\theta})}{dz} \in [a,b], \forall z \in \mathbb{R},$$

$b > a > 0$ or $b < a < 0$, it is easy to design a $h(\cdot,\cdot)$ satisfying Assumption 2.

Instead of requiring a monotonic convergence numerical algorithm as in Assumption 2, a weaker assumption is also used in this paper.

Assumption 3. Let $y \in \mathbb{R}$. For a static mapping $f(z, \boldsymbol{\theta})$, there exists a mapping $h(\cdot,\cdot)$ such that there exists $\rho \in (0,1)$ and a sequence $\{\sigma_i\}_{i\in\mathbb{N}}$ satisfying $\lim_{i\to\infty} |\sigma_i| = 0$, such that,

$$|\Delta_i| \le \rho|\Delta_{i-1}| + |\sigma_i|. \tag{10}$$

Remark 6. Since $|\sigma_i|$ is uniformly bounded and $\lim_{i\to\infty} |\sigma_i| = 0$, therefore, the limit of $|\Delta_i|$ exists. Assume that $\lim_{i\to\infty} |\Delta_i| = c$, where c is a positive constant. It is not hard to show that

$$\begin{aligned} c &= \lim_{i\to\infty} |\Delta u_i| \le \lim_{i\to\infty} \rho|\Delta u_{i-1}| + \lim_{i\to\infty} |\sigma_i| \\ &\Rightarrow (1-\rho)c \le 0 \Rightarrow c = 0, \end{aligned} \tag{11}$$

which ensures $\lim_{i\to\infty} |\Delta_i| = 0$.

Remark 7. Assumption 3 also ensures the convergence of Δ_i, but it is weaker than Assumption 2, since the convergence of $|\Delta_i|$ is not monotonic. However, not all convergent numerical algorithms satisfy Assumption 3, even Assumption 3 is weak. Both Assumption 2 and Assumption 3 require some "nice" convergent properties and they provide selection criteria among available numerical algorithms.

Remark 8. It should be noted that the updating law (7) is time-invariant. In order to "learn" $u_i(t)$ in (6), $h(\cdot,\cdot)$ is applied at each "t" in $[0, T]$ and each iteration "i". Or, at each time instant and each iteration i, a different desired output value $y = u_i(t)$ is given in (7) with the same updating format.

Therefore, a dual ILC loop is proposed: one is to "learn" unknown dynamics while the other is to "learn" unknown static mapping. Next it will show this dual ILC loop can ensure the perfect tracking performance.

3. MAIN RESULTS AND EXAMPLES

3.1 Main results

The learning law is thus constructed as follows

$$u_i^r(t) = u_{i-1}^r(t) + qe_i(t) \qquad u_0^r = 0 \tag{12}$$

$$\Delta^i u_i(t) = u_i^r - f(v_i, \boldsymbol{\theta}), \tag{13}$$

$$v_i(t) = h(v_{i-1}(t), \Delta^i u_{i-1}(t)), \qquad \forall t \in [0, T]. \tag{14}$$

The first result is stated as follows.

Theorem 1. Assume that Assumption 1 holds and $h(\cdot,\cdot)$ in (14) satisfies Assumption 2, then the perfect tracking is achieved with the learning control laws (12-14).

Proof: see Appendix. ◦

Remark 9. If $f(v_i(t), \boldsymbol{\theta}) = u_i^r(t)$ at each iteration, by applying Proposition 1, the perfect tracking performance can be guaranteed. Since $u_i^r(t)$ updates itself by (12), the desired output value y in (7) changes at each time instant as well as each iteration. In the sequel, $u_i(t) \to u^r(t)$ and $f(v_i(t), \boldsymbol{\theta}) \to u^r(t)$ as the number of iteration approaches to infinity.

Remark 10. In proposed dual ILC loop (12-14), there is a freedom in selecting loop 2 ILC scheme from numerical literature while loop 1 ILC is fixed. However, there should be a freedom in selecting loop 1 ILC scheme as well since (12) is not the unique ILC scheme to ensure the perfect tracking performance for the system (1) without input uncertainties. In general, the dual ILC loop should consists of one ILC scheme to deal with the system dynamics; another ILC scheme to deal with unknown static input uncertainties; and some "consistent" conditions between two ILC schemes to ensure the perfect tracking performance for overall systems (dynamics and static).

Intuitively, this "consistent" condition is related to some input-to-state (ISS) gain from inter-connected systems (1) and (2). The well-known small gain theorem (Khalil. , 2002, Chapter 5) may be a possible way to characterize this "consistent" condition and motivate a general "framework" for the ILC design of any dynamic system with input uncertainties. Example 1 also provides some insight on the choice of loop 1 ILC to make two loops "consistent" to ensure the perfect tracking performance. How to formulate this "framework" and find feasible solutions of this problem will be further investigated in future.

Theorem 1 shows that if at each time instant, the numerical updating $h(\cdot,\cdot)$ can make $f(\cdot, \boldsymbol{\theta})$ **monotonically** converge to the desired $u_i^r(t)$, the proposed dual ILC loop determined by (12-14) can ensure the perfect tracking performance. However, for a general unknown nonlinear mapping $f(\cdot,\cdot)$, it is hard to guaranteed that a numerical algorithm $h(\cdot,\cdot)$ that can ensure monotonic convergence as in Assumption 2 exists. Sometime, it may be much easier to find numerical algorithms that are convergent instead of monotonically convergent, for example, numerical algorithms satisfy Assumption 3. The second main result provides a sufficient condition to ensure the perfect tracking performance of the dual ILC loop when numerical algorithms satisfy Assumption 3.

Theorem 2. Assume that Assumption 1 holds and $h(\cdot,\cdot)$ in (14) satisfies Assumption 3. If there exists a positive constant M_C such that the sequence $\{\sigma_i\}_{i\in\mathbb{N}}$ in Assumption 3 satisfies

$$\sum_{i=0}^{\infty} |\sigma_i|^2 = M_c < \infty, \tag{15}$$

then the perfect tracking is achieved with the learning control laws (12-14).

Proof: See Appendix. ◦

Remark 11. Theorem 2 provides a sufficient condition (see (15)) that can guarantee the perfect tracking performance with a dual ILC scheme (12-14). As shown in Example 2, this condition (15) is satisfied when the input uncertainty is a deadzone.

Next, it will show that how to choose $h(\cdot,\cdot)$ for some common input uncertainties: saturation and deadzone and thus provide the insight on how to use main results of this paper.

3.2 Example 1: input saturation

When the input uncertainty is saturation function defined in (3), $h(\cdot, \cdot)$ is chosen as

$$v_i = h(v_{i-1}, \Delta u_{i-1}) = f_s(v_{i-1}, v_{max}) + q_s \Delta u_{i-1} \quad (16)$$

where $q_s \in (0, 1)$. (16) is a very common form in numerical literature.

In order to ensure that (16) can work, (16) has to satisfy either Assumption 2 or Assumption 3. It is observed that when the desire output value is within saturation bound, Assumption 2 is satisfied.

Lemma 1. Given a saturation function (3), for any $|y| \leq v_{max}$, inequality (8) holds true where $\rho = 1 - q_s$.

Sketch of the proof: Using the result (Xu et al. , 2003, Property1), the result holds after simple computations. ◦

Remark 12. In Lemma 1, $|y| \leq v_{max}$ is needed. It is a very natural requirement as it is impossible to drive $f_s(z, v_{max})$ to y, $y > v_{max}$. However this requirement for loop 2 is not "consistent" with loop 1 as $u_i^r(t)$ generated from loop 1 may be larger than the saturation bound for some $t \in [0, T]$ and some $i \in \mathbb{N}$. Therefore, if (12) is used in loop 1, there is no nonlinear mapping $h(\cdot, \cdot)$ such that the output of the static mapping can reach $u_i^r(t)$.

However, if following updating laws are used,

$$u_i^r = f_s(u_{i-1}^r, v_{max}) + q e_i \quad (17)$$

$$v_i(t) = f_s(v_{i-1}(t), v_{max}) + q_s \Delta^i u_{i-1}(t) \quad (18)$$

where $\Delta^i u_{i-1}(t)$ is defined in (13), the perfect tracking performance can be achieved.

Corollary 1. Assume that Assumption 1 holds for the system (1) and $\max_{t \in [0,T]} |u^r(t)| \leq v_{max}$. The learning control laws (17-18) guarantees the perfect tracking performance of systems (1-2).

Proof: see Appendix. ◦.

Remark 13. Example 1 also indicates that the ILC scheme (12) is **not** the only choice of loop 1 to ensure the perfect tracking performance. However, how to design (17) for more general input uncertainties is still open. As indicated in Remark 10, further work is needed.

3.3 Example 2: Input deadzone

Next input deadzone is considered with the following characteristics:

$$f(v, \boldsymbol{\theta}) = f_{dz}(v, \boldsymbol{\theta}) = \begin{cases} m_r(v - \eta_r) & v > \eta_r \\ 0 & \eta_l \leq v \leq \eta_r \\ m_l(v - \eta_l) & v < \eta_l \end{cases}, \quad (19)$$

where $\boldsymbol{\theta} = [\, m_r \; m_l \; \eta_r \; \eta_l \,]^T$. If the following learning law is used

$$v_i = v_{i-1} + \beta \Delta u_{i-1}, \quad (20)$$

where β is a positive constant satisfying

$$0 < 1 - \beta B_1 < 1, \quad (21)$$

where B_1 is the upper bound of m_l and m_r and it is assume to be known. In this example $\boldsymbol{\theta} \in \mathbb{R}^4$ is not completely known. The following proposition can be found in (Xu , 2003, Theorem 2.1).

Proposition 2. For the static mapping (19), if B_1 is known, the updating law (20) that satisfies the condition (21) can guarantee that Δu_i converges to zero as i approaches to infinity.

In the proof of Proposition 2 Xu (2003), it was shown that $|\sigma_i|$ becomes 0 after a finite iteration number p_k. This shows that $\lim_{i \to \infty} \sum_{k=0}^{i} |\sigma_k|^2 < \infty$. Condition (15) is thus satisfied. Therefore, by applying Theorem 2, the following result is obvious.

Corollary 2. Consider the system (1) with $f(\cdot, \boldsymbol{\theta})$ that is defined in (19). Assume that Assumption 1 holds. The learning control laws (12-13) with (20) where β satisfies (21) guarantees the perfect tracking performance.

The above examples show that how proposed dual ILC loop works. Taking advantage of rich literature in numerical analysis, the proposed method can be applied to deal with very general input uncertainties.

4. CONCLUSION

In this paper, a dual ILC scheme is proposed for nonlinear dynamic systems with input uncertainties. Loop 1 ILC deals with systems' dynamics while Loop 2 ILC uses numerical algorithms to learn unknown static input uncertainties. A simple ILC scheme is used in loop 1 while loop 2 uses available numerical algorithms. By incorporating two ILC schemes in the dual loop, it is shown that the proposed ILC scheme can ensure the perfect tracking performance. Two examples, which the input uncertainties are chosen to be saturation and deadzone, are used to show the effectiveness of the proposed method.

REFERENCES

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

S. Arimoto, S.Kawamura and F.Miyazaki Bettering operation of robots by learning. *Journal of Robotic Systems*, 1 (2),379–388, 1984.

Y.Q. Chen, Z.M. Gong and C.Y. Wen. Analysis of a high-order iterative learning control algorithm for uncertain nonlinear systems with state delays. *Automatica*, 34: 345–353, 1998.

S. Chin, J. Qin, K.S. Lee and M. Cho. A two-stage iterative learning control technique combined with real-time feedback for independent disturbance rejection. *Automatica*, 40:1913-1922, 2004.

H. K. Khalil. *Nonlinear Systems*. Upper Saddle River, New Jersey, Prentice Hall 2003.

R. Longman. Iterative learning control and repetitive control for engineering practice. *International Journal of Control*, 73:930–954, 2000.

K. L. Moore. *Iterative Learning Control For Deterministic Systems*. Advances in industrial control. Berlin: Springer, 1993.

K. L. Moore. Iterative learning control: An expository overview. *Applied and Computational Controls, Signal Processing, and Circuits*, 1:151–214, 1999.

M. Norrlof and S. Gunnarsson. Anote on causal and CITE iterative learning control algorithms. *Automatica*, 41: 345-350, 2005.

J.M.Ortega and W.C,Rheinboldt. *Iterative Solution Of Nonlinear Equations In Several Variables.* Academic Press, NewYork, 1970.

S. S. Saab On the P-type learning control. *IEEE Transactions on Automatic Control*, 39:2298-2302, 1994.

S. S. Saab Optimal selection of the forgetting matrix into an iterative learning control algorithm. *IEEE Transactions on Automatic Control*, 50:2039-2043, 2005.

T. Sugie and F. Sakai. Noise tolerant iterative learning control for a class of continuous-time systems. *Automatica*, 43:1766-1771, 2007.

J. Xu. *Development Of The Finite And Infinite Interval Learning Control Theory.* Ph.D Thesis. National University of Singapore, Singapore, 2003.

J.X. Xu and Y.Tan *Linear and Nonlinear Iterative Learning Control.* Lecture Notes in Control and Information Science. Berlin, Springer, 2003.

J.X. Xu, Y. Tan and T.H. Lee. Iterative learning control design based on composite energy function with input saturation. *Automatica*, 40:1371–1377, 2003.

APPENDIX

Sketch of proof of Proposition 1 To facilitate the derivation and analysis of the learning system, define the following time weighted composite energy function

$$E_i(t) = \frac{1}{2}e^{-\lambda t}e_i^2(t) + \frac{1}{2q}\int_0^t e^{-\lambda\tau}\left(u_i - f(v^r,\boldsymbol{\theta})\right)^2 d\tau \quad (22)$$

where $\lambda > 2L_\eta$ is a finite positive constant.

First, it is shown that the CEF (22) is non-increasing along the iteration domain. The difference of $E_i(t)$, $\Delta E_i(t)$, is defined as $\Delta E_i(t) \overset{\triangle}{=} E_i(t) - E_{i-1}(t)$. Denoting $\Delta_f(u_i) := u_i - u^r = u_i - f(v^r,\boldsymbol{\theta})$, a simple computation yields

$$\begin{aligned}\Delta E_i &= \frac{1}{2}e^{-\lambda t}\left[e_i^2(t) - e_{i-1}^2(t)\right]\\ &+\frac{1}{2q}\int_0^t e^{-\lambda\tau}\left[\Delta_f^2(u_i) - \Delta_f^2(u_{i-1})\right]d\tau \end{aligned}\quad (23)$$

We can compute each term on the right hand side of (23) separately. First, noting the error dynamics (5), simple computation leads to

$$\begin{aligned}\frac{1}{2}e^{-\lambda t}e_i^2(t) &= -\frac{\lambda}{2}\int_0^t e^{-\lambda\tau}(e_i^2(\tau))d\tau\\ &+\int_0^t e^{-\lambda\tau}e_i(\tau)\left[\eta(x^r,\tau) + f(v^r,\boldsymbol{\theta})\right]d\tau\\ &-\int_0^t e^{-\lambda\tau}e_i(\tau)\left[\eta(x_i,\tau) + u_i\right]d\tau\\ &\le -\frac{\lambda - 2L_\eta}{2}\int_0^t e^{-\lambda\tau}(e_i^2(\tau))d\tau\\ &+\int_0^t e^{-\lambda\tau}e_i(\tau)\left[f(v^r,\boldsymbol{\theta}) - u_i\right]d\tau \end{aligned}\quad (24)$$

Now looking into the third term on the right hand side of (23). Using updating law (6) leads to

$$\begin{aligned}&\frac{1}{2q}\int_0^t e^{-\lambda\tau}\left[\Delta_f^2(u_i) - \Delta_f^2(u_{i-1})\right]\\ &= \frac{1}{2}\int_0^t e^{-\lambda\tau}e_i(\tau)\left[2u_i - qe_i(\tau) - 2f(u^r,\boldsymbol{\theta})\right]\\ &\le \frac{1}{2}\int_0^t e^{-\lambda\tau}e_i(\tau)\left[2u_i - 2f(u^r,\boldsymbol{\theta})\right]. \end{aligned}\quad (25)$$

Substituting (24) and (25) into (23) yields

$$\begin{aligned}\Delta E_i &\le -\frac{\lambda - 2L_\eta}{2}\int_0^t e^{-\lambda\tau}(e_i^2(\tau))d\tau - \frac{1}{2}e^{-\lambda t}(e_{i-1}^2(t)),\\ &\le -\frac{1}{2}e^{-\lambda t}(e_{i-1}^2(t)) \end{aligned}\quad (26)$$

which shows that the energy function E_i is non-increasing along the iteration axis. The proof is completed by following the similar steps in Xu et al. (2003). ■

Proof of Theorem 1. The following time weighted composite energy function is employed.

$$\begin{aligned}E_i(t) &= \frac{1}{2}e^{-\lambda t}e_i^2(t) + \frac{1}{2q}\int_0^t e^{-\lambda\tau}\left(\Delta_f(u_i^r)\right)^2 d\tau\\ &+\frac{1}{2q}\int_0^t e^{-\lambda\tau}\left(\Delta^i u_i\right)^2 d\tau, \end{aligned}\quad (27)$$

where $\lambda > 2L_\eta + 2\frac{\rho^2}{1-\rho^2}$ is a finite positive constant, ρ is from (8) and u_i^r is from (12) and $\Delta^i u_i$ is defined in (13). First, differencing $E_i(t)$ yields.

$$\begin{aligned}\Delta E_i &= \frac{1}{2}e^{-\lambda t}(e_i^2(t)) - \frac{1}{2}e^{-\lambda t}(e_{i-1}^2(t))\\ &+\frac{1}{2q}\int_0^t e^{-\lambda\tau}\left[\Delta_f^2(u_i^r) - \Delta_f^2(u_{i-1}^r)\right]d\tau\\ &+\frac{1}{2q}\int_0^t e^{-\lambda\tau}\left(\Delta^i u_i\right)^2 - \left(\Delta^{i-1}u_{i-1}\right)^2 d\tau \end{aligned}\quad (28)$$

Each term on the right hand side of (28) is bounded separately. Using error dynamics (5), the bound the first term on the right hand side of (28) becomes.

$$\begin{aligned}\frac{1}{2}e^{-\lambda t}e_i^2(t) &\le -\frac{\lambda - 2L_\eta}{2}\int_0^t e^{-\lambda\tau}(e_i^2(\tau))d\tau\\ &+\int_0^t e^{-\lambda\tau}e_i(\tau)\left[f(v^r,\boldsymbol{\theta}) - u_i^r) + (u_i^r - u_{i-1}^r)\right]\\ &+\int_0^t e^{-\lambda\tau}e_i(\tau)(u_{i-1}^r - f(v_i,\boldsymbol{\theta}))d\tau\\ &\le -\frac{\lambda - 2L_\eta - 2q}{2}\int_0^t e^{-\lambda\tau}(e_i^2(\tau))d\tau\\ &+\int_0^t e^{-\lambda\tau}e_i(\tau)(u_{i-1}^r - f(v_i,\boldsymbol{\theta}))d\tau \end{aligned}\quad (29)$$

The third term on the right hand side of (28) can be bounded

$$\begin{aligned}&\frac{1}{2q}\int_0^t e^{-\lambda\tau}\left[\Delta_f^2(u_i^r) - \Delta_f^2(u_{i-1}^r)\right]\\ &= \frac{1}{2}\int_0^t e^{-\lambda\tau}e_i(\tau)\left[2u_i^r - 2f(v^r,\boldsymbol{\theta})\right] - \frac{q}{2}\int_0^t e^{-\lambda\tau}e_i^2(\tau). \end{aligned}\quad (30)$$

Now it is the last term in on the right hand side of (28). Using condition 8 in Assumption 2, the following fact is obtained.

$$\left(\Delta^{i-1}u_{i-1}\right)^2 = \left(u_{i-1}^r - f(v_{i-1},\boldsymbol{\theta})\right)^2 \geq \frac{1}{\rho^2}\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2,$$

leading to

$$\begin{aligned}&\left(\Delta^i u_i\right)^2 - \left(\Delta^{i-1}u_{i-1}\right)^2\\ &\leq [\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2 - \left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2\\ &\quad -\frac{1-\rho^2}{\rho^2}\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2\\ &= 2qe_i(u_i^r - f(v_i,\boldsymbol{\theta})) - q^2e_i^2\\ &\quad -\frac{1-\rho^2}{\rho^2}\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2. \end{aligned} \quad (31)$$

Consequently, it yields

$$\begin{aligned}&\frac{1}{2q}\int_0^t e^{-\lambda\tau}\left(\Delta^i u_i\right)^2 - \left(\Delta^{i-1}u_{i-1}\right)^2 d\tau\\ &\leq \int_0^t e^{-\lambda\tau}e_i(\tau)(u_i^r - f(v_i,\boldsymbol{\theta})) - \frac{q}{2}e_i^2(\tau)d\tau - \\ &\quad -\int_0^t e^{-\lambda\tau}\frac{1-\rho^2}{\rho^2}\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2 d\tau \end{aligned} \quad (32)$$

Substituting (29), (30) and (32) into (28) yields

$$\begin{aligned}\Delta E_i &\leq -\frac{\lambda - 2L_\eta}{2}\int_0^t e^{-\lambda\tau}(e_i^2(\tau))d\tau\\ &\quad +2\int_0^t e^{-\lambda\tau}e_i(\tau)(u_{i-1}^r - f(v_i,\boldsymbol{\theta})d\tau\\ &\quad -\int_0^t e^{-\lambda\tau}\frac{1-\rho^2}{\rho^2}\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2 d\tau\\ &\quad -\frac{1}{2}e^{-\lambda t}(e_{i-1}^2(t)),\\ &\leq -\frac{\lambda - 2L_\eta - \frac{2\rho^2}{1-\rho^2}}{2}\int_0^t e^{-\lambda\tau}(e_i^2(\tau))d\tau\\ &\quad -\frac{1}{2}e^{-\lambda t}(e_{i-1}^2(t))\\ &\leq -\frac{1}{2}e^{-\lambda t}(e_{i-1}^2(t)) \end{aligned} \quad (33)$$

which shows that the energy function E_i is non-increasing along the iteration axis. The proof is completed by following the similar steps in Xu et al. (2003). ■

Proof of Theorem 2. The time weighted composite energy function (27) is employed. The Difference of $E_i(t)$ is obtained in (28). In this proof, we just need to bound the last term on the right hand side of (28).

From (10) in Assumption 3, it follows

$$\rho^2|\Delta u_{i-1}| \geq |\Delta u_i|^2 - 2\rho|\Delta u_{i-1}|\cdot|\sigma_i| - |\sigma_i|^2. \quad (34)$$

In the sequel, it yields

$$\begin{aligned}&\left(\Delta^i u_i\right)^2 - \left(\Delta^{i-1}u_{i-1}\right)^2\\ &\leq [\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2 - \left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2\\ &\quad -\frac{1-\rho^2}{2\rho^2}\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2\\ &\quad -\frac{1-\rho^2}{2\rho^2}\left[\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right) - \frac{2\rho^3}{1-\rho^2}|\sigma_i|\right]^2\\ &\quad +\left(1+\frac{2\rho^4}{1-\rho^2}\right)|\sigma_i|^2\\ &= \left(u_i^r - u_{i-1}^r\right)\left(u_i^r + u_{i-1}^r - 2f(v_i,\boldsymbol{\theta})\right)\\ &\quad -\frac{1-\rho^2}{\rho^2}\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2\\ &= 2qe_i(u_i^r - f(v_i,\boldsymbol{\theta})) - q^2e_i^2\\ &\quad -\frac{1-\rho^2}{\rho^2}\left(u_{i-1}^r - f(v_i,\boldsymbol{\theta})\right)^2\\ &\quad +\left(1+\frac{2\rho^4}{1-\rho^2}\right)|\sigma_i|^2. \end{aligned} \quad (35)$$

By similar computation as in the proof of Theorem 1, we have

$$\Delta E_i(t) \leq -\frac{1}{2}e^{-\lambda t}(e_{i-1}^2(t)) + \left(1+\frac{2\rho^4}{1-\rho^2}\right)|\sigma_i|^2, \quad (36)$$

which implies that

$$\begin{aligned}\lim_{i\to\infty}E_i(t) &= E_0(t) + \lim_{i\to\infty}\sum_{k=1}^{i}\Delta E_k\\ &\leq E_0 - \lim_{i\to\infty} -\frac{1}{2}um_{k=1}^{i-1}e^{-\lambda t}(e_k^2(t))\\ &\quad +M_c\left(1+\frac{2\rho^4}{1-\rho^2}\right), \end{aligned} \quad (37)$$

where M_c is from (15). This implies that $\lim_{i\to\infty} e_i^2(t) = 0$ for any $t \in [0,T]$, concluding the proof. ■

Sketch of proof of Corollary 1. The following time-weighted CEF is employed.

$$\begin{aligned}E_i(t) &= \frac{1}{2}e^{-\lambda t}e_i^2(t) + \frac{1}{2q}\int_0^t e^{-\lambda\tau}\left(\bar{\Delta}_f(u_i^r)\right)^2 d\tau\\ &\quad +\frac{1}{2q}\int_0^t e^{-\lambda\tau}\left(\Delta^i u_i\right)^2 d\tau \end{aligned} \quad (38)$$

where $\lambda > 2L_\eta + 2\frac{\rho^2}{1-\rho^2} + 2q$ is a finite positive constant, ρ is from (8) and u_i^r is from (12), $\bar{\Delta}_f(u_i^r) = (sat(u_i^r) - f(v^r,\boldsymbol{\theta}))^2$ and $\Delta^i u_i$ is defined in (13). By using Property 1 and Property 2 in Xu et al. (2003), simple calculations lead to the result. ■