Nonlinear Discrete-time System Identification without Persistence of Excitation: A Finite-time Concurrent Learning

Farzaneh Tatari, Chiristos Panayiotou, and Marios Polycarpou,

Abstract—This paper deals with the problem of finite-time learning for unknown discrete-time nonlinear systems’ dynamics, without the requirement of the persistence of excitation. A finite-time concurrent learning approach is presented to approximate the uncertainties of the discrete-time nonlinear systems in an on-line fashion by employing current data along with recorded experienced data satisfying an easy-to-check rank condition on the richness of the recorded data which is less restrictive in comparison with persistence of excitation condition. Rigorous proofs guarantee the finite-time convergence of the estimated parameters to their optimal values based on a discrete-time Lyapunov analysis. Compared with the existing work in the literature, simulation results illustrate that the proposed method can timely and precisely approximate the uncertainties.

Index Terms—Discrete-time systems, Finite-time concurrent learning (FTCL), Nonlinear systems, Unknown dynamics.

I. INTRODUCTION

Learning a high-fidelity model of a nonlinear system via stream of data is of vital importance in many engineering applications since such systems are highly subjected to uncertainties that can degrade the performance of the system controllers. It is well known that many learning strategies, such as least-square and gradient descent [1], depend heavily on the persistency of excitation (PE) condition that permanently requires a complete span of the space over which the learning is performed, and failure to fulfill this condition will lead to poor learning results. However, the PE condition might be hard to achieve or even might not be feasible in some scenarios, especially in the context of on-line learning. Concurrent learning [2-6] has emerged as a promising paradigm in the direction that guarantees the exponential convergence of the estimated parameters to their optimal values with relaxing the strict assumption of the PE condition to some easy-to-check verifiable conditions on the richness of data. Concurrent learning technique benefits from recorded experienced data along with current data to replace the PE condition on the regressor with a rank condition on the memory stack of the regressor recorded data. Based on this rank condition, the matrix of the regressor recorded data must contain the same number of linearly independent elements as the dimension of the independent basis functions in the regressor.

In many practical situations, the system dynamics are employed in on-line monitoring and control applications; therefore, learning the system’s unknown dynamics over a finite-time interval is required. Finite-time learning is of more interest rather than learning with asymptotic or exponential convergence rate since it is more physically realizable than concerning infinite time. Moreover, such finite-time learning scheme is of utmost importance for learning-based controlled systems that demand fast and reliable actions. Although, finite-time control methods have been extensively employed for discrete [7-9] and continuous-time [10-12] systems, fewer attempts have been proposed by several researchers to tackle finite-time learning such as [13-19]. While these results have studied finite-time learning schemes, the majority of them are concerning with continuous-time systems. The results on finite-time learning of continuous-time systems are more fruitful in comparison with discrete-time ones due to the existence of abundant mathematical tools. In many practical applications, however, identifying precise discrete-time dynamics of the system is required due to the development of computer technology and the introduction of digital controllers and sensors. Hence, it is of great practical importance to investigate finite-time learning of discrete-time systems. There are only a few results discussing the identification of discrete-time systems including [19,20]. The work of [19], studied the discrete-time systems uncertainty identification in finite-time, where the proposed method required the on-line invertibility check of a regressor matrix and its inverse computation, along with interval excitation of the regressor. In [20], a concurrent learning-based method for discrete-time function approximation is presented that relaxed the PE condition and guaranteed the asymptotic convergence and boundedness of the parameters estimation errors, respectively, for function approximators with zero and non-zero residual approximation errors in case of optimal parameters.

According to the aforementioned discussions, it is favorable to employ finite-time learning schemes that can alleviate the restrictive condition of the PE. To partially bridge this gap, recently, the authors in [18] proposed a finite-time identification method for continuous-time systems based on concurrent learning. However, the counterpart results in the discrete-time domain are not available. Discrete-time systems are quite different with continuous-time systems; therefore, the tools applied to the continuous-time domain cannot be directly employed for the discrete-time domain. Moreover, different from finite-time stability analysis of continuous-time systems, which can draw support from many mathematical tools, the mathematical tools for finite-time stability analysis of discrete-time systems are not plenty. Therefore, the research on finite-time convergent concurrent learning identification method for discrete-time systems is more challenging and complex.

Motivated by the above-mentioned discussions, this work aims to propose a finite-time concurrent learning (FTCL) scheme for discrete-time systems that guarantees finite-time parameter convergence without the restrictive PE condition by employing a memory stack of data, satisfying a rank condition. To this end, the proposed method guarantees the finite-time convergence of the parameters estimation errors to the origin for adaptive approximators with zero minimum functional approximation errors (MFAEs) where MFAE is the residual approximation error in the case of optimal parameters. And for adaptive approximators with non-zero MFAE, the presented method ensures that the parameters estimation errors are uniformly ultimately bounded. The proposed adaptive estimation method in this work is simpler in comparison with the continuous finite-time estimation laws in [16,17], containing fractional functions, and it is able to approximate the unknown dynamics of higher-order discrete-time nonlinear systems. Contrary to [14,15,19], the proposed method does not need the inverse of the recorded regressor matrix or
approximating it using an auxiliary matrix.

Compared with previous results on finite-time learning and concurrent learning approaches, contributions of the current work are summarized as follows:

1) A novel FTCL scheme is presented for learning the unknown dynamics of nonlinear discrete-time systems.
2) Rigorous proofs ensure the finite-time convergence of the parameters estimation errors to the origin for adaptive approximators with zero MFAE using the proposed FTCL method and discrete-time Lyapunov analysis. It is also guaranteed that for adaptive approximators with non-zero MFAE, the parameter estimation errors are uniformly ultimately bounded.
3) The settling-time functions for the finite time of the parameters’ error convergence to zero and to a bound around zero are respectively given for the adaptive approximators with zero and non-zero MFAEs. In addition, based on the finite-time analysis, a condition on the learning rate is derived for finite-time convergence.

Notation: Throughout the paper, the following notation is adopted. \( \mathbb{R}, \mathbb{Z}, \) and \( \mathbb{N}^+ \) respectively show the set of real, integer and natural numbers without zero. \( \| \cdot \| \) is used to denote the Euclidean norm for vectors and induced 2-norm for matrices. Trace of a matrix is indicated with \( tr(\cdot) \). The minimum and maximum eigenvalues of matrix \( A \) are respectively denoted by \( \lambda_{\min}(A) \) and \( \lambda_{\max}(A) \). The matrix \( I \) is the identity matrix of appropriate dimensions. \( \lfloor : \mathbb{R} \rightarrow \mathbb{Z} \) is the floor function. In some cases, the time variable \( k \) is dropped from the state or input variables for the sake of brevity.

II. PROBLEM FORMULATION

Consider a discrete-time nonlinear system as follows,

\[
x(k + 1) = f(x(k)) + g(x(k))u(k),
\]

where \( x \in \mathcal{D}_x \subseteq \mathbb{R}^n \) is the state vector and \( u \in \mathcal{D}_u \subseteq \mathbb{R}^m \) is the control input vector, \( \mathcal{D}_x \) and \( \mathcal{D}_u \) are compact sets; \( f : \mathcal{D}_x \rightarrow \mathbb{R}^n \) and \( g : \mathcal{D}_x \rightarrow \mathbb{R}^{n \times m} \) are respectively the unknown nonlinear drift and input terms. The integers \( n \) and \( m \) denote the dimensions of the state and input vectors, respectively. This paper aims to learn the system unknown dynamics in (1), namely to approximate the uncertain functions \( f(x) \) and \( g(x) \) in a finite time using a concurrent learning technique.

Assumption 1: \( x(k) \) is a measurable state vector, and \( f(x(k)) \) and \( g(x(k)) \) are both locally Lipschitz in \( x(k) \).

In order to learn the uncertain functions \( f(x) \) and \( g(x) \) in the system dynamics, the first unknown nonlinear functions \( f(x) \) and \( g(x) \) are formulated in regressor forms. Here, linearly parameterized adaptive approximation models are used to, respectively, represent \( f(x(k)) \) and \( g(x(k)) \) as follows,

\[
f(x(k)) = \hat{f}(x(k), \Theta_f^\ast) + e_f(x(k)),
\]

\[
g(x(k)) = \hat{g}(x(k), \Theta_g^\ast) + e_g(x(k)),
\]

where

\[
\hat{f}(x(k), \Theta_f^\ast) = \Theta_f^T \varphi(x(k)),
\]

\[
\hat{g}(x(k), \Theta_g^\ast) = \Theta_g^T \chi(x(k)).
\]

The matrices \( \Theta_f \in \mathcal{D}_f \subseteq \mathbb{R}^{p \times n} \) and \( \Theta_g \in \mathcal{D}_g \subseteq \mathbb{R}^{q \times n} \) denote the unknown optimal parameters of the adaptive approximation models, defined as follows

\[
\Theta_f^\ast = \arg \min_{\Theta_f \in \mathcal{D}_f} \left\{ \sup_{x(k) \in \mathcal{D}_x} \| f(x(k)) - \hat{f}(x(k), \Theta_f) \| \right\},
\]

\[
\Theta_g^\ast = \arg \min_{\Theta_g \in \mathcal{D}_g} \left\{ \sup_{x(k) \in \mathcal{D}_x} \| g(x(k)) - \hat{g}(x(k), \Theta_g) \| \right\},
\]

where \( \mathcal{D}_f \) and \( \mathcal{D}_g \) are compact sets. The vectors \( \varphi : \mathcal{D}_x \rightarrow \mathbb{R}^p \) and \( \chi : \mathcal{D}_x \rightarrow \mathbb{R}^q \), denote the basis functions, whereas \( p \) and \( q \) are the number of linearly independent basis functions for approximating \( f(x(k)) \) and \( g(x(k)) \) respectively. The quantities \( e_f(x(k)) \) and \( e_g(x(k)) \) are the MFAEs for \( f(x(k)) \) and \( g(x(k)) \) respectively, representing the residual approximation error in the case of optimal parameters. In the special case that the unknown functions \( f(x(k)) \) and \( g(x(k)) \) can be approximated exactly by the adaptive approximation models \( \hat{f}(x(k), \Theta_f) \) and \( \hat{g}(x(k), \Theta_g) \) respectively, MFAEs are zero, i.e., \( e_f(x(k)) = e_g(x(k)) = 0 \).

By using (2)-(5), the system dynamics (1) can be rewritten as

\[
x(k + 1) = \Theta_f^T \varphi(x(k)) + \varepsilon(x(k), u(k)),
\]

where \( \Theta_f = [\Theta_f^T, \Theta_g^T] \in \mathbb{R}^{(p+q) \times n} \).
where $\Theta(k) = [\hat{\Theta}_f^T(k), \hat{\Theta}_g^T(k)]^T \in \mathbb{R}^{(p+q) \times n}$, $\hat{\Theta}_f(k)$ and $\hat{\Theta}_g(k)$ are respectively the estimation of parameters matrices $\Theta^*_f$, $\Theta^*_g$ and $\Theta^*_g$ at time $k$. Let the state estimation error for system (1) be defined as

$$
\varepsilon(k) = \hat{x}(k) - \bar{x}(k) = \hat{\Theta}(k)^T \bar{d}(k) - \varepsilon(k),
$$

(15)

where $\hat{\Theta}(k) := \hat{\Theta}(k) - \Theta^* := [\hat{\Theta}_f^T(k), \hat{\Theta}_g^T(k)]^T$ is the parameter estimation error with $\hat{\Theta}_f(k) := \hat{\Theta}_f(k) - \Theta^* F$, $\hat{\Theta}_g(k) := \hat{\Theta}_g(k) - \Theta^*_g$.

To fulfill the finite-time learning of the uncertainties $f(x)$ and $g(x)$ in system (1), the paper objective is to propose a finite-time concurrent learning-based estimation method that relaxes the PE estimation law, the past data is collected and stored in the memory

$$
\text{condition and satisfies the following criteria:}
$$

1) For adaptive approximators with zero MFAE, the parameters’ estimation error $\hat{\Theta}(k)$ converges to zero in finite time.
2) For adaptive approximators with non-zero MFAE, the parameters’ estimation error $\hat{\Theta}(k)$ are uniformly ultimately bounded.

### III. Finite-Time Concurrent Learning of the Unknown Dynamics

In this section, the concurrent learning-based parameter estimation method for approximating the unknown dynamics of system (1) in finite time is presented and the required preliminaries for the finite-time convergence analysis of the proposed estimation method are developed.

For using the idea of concurrent learning, which employs recorded experienced data along with current data in the identifier’s parameter estimation law, the past data is collected and stored in the memory stacks $M \in \mathbb{R}^{(p+q) \times P}$, $L \in \mathbb{R}^{n \times P}$ and $X \in \mathbb{R}^{n \times P}$, at time steps $\tau_1, ..., \tau_P$ as follows.

$$
M = [\tilde{d}(\tau_1), \tilde{d}(\tau_2), ..., \tilde{d}(\tau_P)], \quad L = [\bar{I}(\tau_1), \bar{I}(\tau_2), ..., \bar{I}(\tau_P)],
$$

$$
X = [\bar{x}(\tau_1), \bar{x}(\tau_2), ..., \bar{x}(\tau_P)],
$$

(16)

where $P$ is the number of data points stored in every stack. Note that $P$ is chosen so that $M$ contains at least as many linearly independent elements as the dimension of $d(k)$ (i.e., the total number of linearly independent basis functions for approximating $f(x)$ and $g(x)$), given in (10), that is called as $M$ rank condition and requires $P \geq p + q$.

Define the error $e_h(k)$ for the $h^{th}$ recorded sample as

$$
e_h(k) = \hat{x}_h(k) - \bar{x}(\tau_h),
$$

(17)

where

$$
\hat{x}_h(k) = \hat{\Theta}(k)^T \bar{d}(\tau_h) - \bar{I}(\tau_h) + c^h \bar{e}(0),
$$

(18)

is the state estimation at $0 \leq \tau_h < k$ time step, $h = 1, ..., P$, using the current estimated parameters matrix $\hat{\Theta}(k)$ and the recorded $\bar{d}(\tau_h)$ and $\bar{I}(\tau_h)$. Substituting $\bar{x}(\tau_h)$ into (17), one has

$$
e_h(k) = \hat{\Theta}(k)^T \bar{d}(\tau_h) - \varepsilon(\tau_h).
$$

(19)

Now, the proposed FTCL law for estimating the parameters of the system approximator is given as

$$
\dot{\hat{\Theta}}(k+1) = \hat{\Theta}(k) - \Gamma [\Xi_C \bar{d}(k) e^T(k) + \Xi_C (\sum_{h=1}^P \tilde{d}(\tau_h) e_h^T(k) + \sum_{h=1}^P \tilde{d}(\tau_h) \varepsilon_h(\tau_h))] + \frac{\sum_{h=1}^P \tilde{d}(\tau_h) e_h^T(k)}{\beta + \| \sum_{h=1}^P \tilde{d}(\tau_h) \varepsilon_h(\tau_h) \|},
$$

(20)

where all the matrices $\Gamma, \Xi_C, \Xi_G \in \mathbb{R}^{(p+q) \times (p+q)}$, $\Gamma = \gamma I$ is the learning rate matrix with positive constant $\gamma > 0$, $\beta$ is a design constant parameter satisfying $\beta > \| \sum_{h=1}^P \tilde{d}(\tau_h) \varepsilon_h(\tau_h) \| = P b$, $\Xi_C = \xi C I$ and $\Xi_G = \xi G I$ with positive constants $\xi_C > 0$ and $\xi_G > 0$. The above estimation law has two learning terms where the term $\Xi_C \hat{d}(k) e^T(k)$, containing the current state approximation error, is also widely used in gradient descent method and the term $\Xi_C (\sum_{h=1}^P \tilde{d}(\tau_h) e_h^T(k) + \sum_{h=1}^P \tilde{d}(\tau_h) \varepsilon_h(\tau_h) )$, containing the past experienced data, is called the concurrent learning term. The learning weights $\Xi_C$ and $\Xi_G$ do not need to be equal and by setting appropriate $\xi_C$ and $\xi_G$, respectively, one of the two learning terms of gradient descent and concurrent learning can be prioritized over the other.

Before proceeding further, the following definitions and lemma are needed. In the following, Lemma 2, inspired by Lemma 3.1 in [21], contains a fundamental result on Lyapunov representation of finite-time stability and convergence for discrete-time systems, not being reported in the literature. It is worth mentioning that in Lemma 2, the required conditions on the difference of the Lyapunov function candidate are completely different from the ones in [21] and the settling-time function is provided which is not given in [21].

**Definition 1**: [22] The bounded signal $d(k)$ is said to be persistently exciting if there exist positive scalars $\mu_1, \mu_2$ and $T \in \mathbb{N}^+$ such that $\forall \tau \in \mathbb{N}^+, \mu_1 I \leq \sum_{h=1}^T \| d(k) d^T(k) \| \leq \mu_2 I$.

**Definition 2**: [23] Consider the system

$$
y(k + 1) = F(y(k)),
$$

(21)

where $y \in \mathcal{D}_y$ and $F : \mathcal{D}_y \to \mathbb{R}^n$ is a nonlinear function on the open neighborhood $\mathcal{D}_y$ of the origin and the equilibrium point of (21) is the origin. The system (21) is said to be finite-time stable, if it is Lyapunov stable and finite-time convergent where $\forall y(0) \in \mathcal{D}_y$ any solution $y(k)$ of (21) reaches the origin at some finite time moment, i.e., $y(k) = 0, \forall k > K(y(0))$ where $K : \mathcal{D}_y \to \mathbb{N}^+$ is the settling-time function.

**Definition 3**: [24] The solution of (21) is uniformly ultimately bounded with ultimate bound $\alpha_2$, if there exist positive constants $\alpha_2$ and $c_0$ where for every $0 < \alpha_1 < c_0$ there exists a constant $K(\alpha_1, \alpha_2) \in \mathbb{N}^+$, such that

$$
\| y(0) \| < \alpha_1 \Rightarrow \| y(k) \| < \alpha_2, \forall k > K(\alpha_1, \alpha_2).
$$

(22)

**Lemma 2**: Consider the nonlinear system (21). Suppose there is a Lyapunov function $V : \Omega \to \mathbb{R}$ which is positive definite in $\Omega$ where $\Omega$ is an open neighborhood of the origin, and there exist real numbers $0 < \mu_1 < 1, 0 < \alpha_1 < 1$, and $b > 0$ such that

$$
V(k + 1) = (1 - a)V(k) - b(V(k))^\mu,
$$

(23)

on $\Omega$. Then, system (21) is finite-time stable and its states converge to zero for $k \geq K(V(y(0)))$ where

$$
K(V(0)) = \left\lfloor \frac{V(0)}{a^\mu(b)} \right\rfloor + 1,
$$

is the settling-time function.

**Proof**: Note that (23) suffices to show the Lyapunov stability of the system, because it guarantees that the difference $V(k + 1) - V(k)$, along the discrete-time system trajectories is negative definite and it will be zero if and only if $V(k) = 0$.

Consider that for any finite positive initial value of the Lyapunov function $V(0)$, along with an arbitrary trajectory for the discrete-time system (21), one has

$$
V(0) = r_0(b)^{\frac{1}{1-\mu}},
$$

(24)

where a unique positive scalar $r_0 > 0$ exists satisfying (24). Replacing $V(0)$ into (23), one obtains

$$
V(1) = ((1 - a)r_0 - r_0^\mu)(b)^{\frac{1}{1-\mu}}.
$$

(25)
Defining \( r_1 = (1-a)r_0 - r_0^\mu \), (25) can be rewritten as
\[
V(1) = r_1(b)^{1-\mu}.
\]
(26)
Note that if \( r_0 \leq \left( \frac{1}{1-a} \right)^{1-\mu} \), then the above implies that \( r_1 = 0 \), since \( V(1) \) has to be non-negative from the definition of the Lyapunov function. In this case, the value of the Lyapunov function already converges to zero in the first step, i.e., for \( K^* = 1 \). Now, let \( r_0 > \left( \frac{1}{1-a} \right)^{1-\mu} \). Substituting (26) into (23), gives a similar expression for \( V'(2) \),
\[
V(2) = r_2(b)^{1-\mu},
\]
(27)
where \( r_2 = (1-a)r_1 - r_1^\mu \). In the same manner, one obtains \( V(k+1) \) as follows,
\[
V(k+1) = r_{k+1}(b)^{1-\mu},
\]
(28)
with \( r_{k+1} = (1-a)r_k - r_k^\mu \) where \( r_{k+1} \) can be rewritten as
\[
r_{k+1} = -r_k^\mu(1 - (1-a)r_k^{1-\mu}).
\]
(29)
Since \( V(k+1) < V(k) \), one has \( r_{k+1} < r_k, \forall k \), and \( V(k+1) \) is positive definite, consequently \( r_{k+1} \geq 0 \). Hence, from the right side of (29), one can see that there exists a \( k \) such that
\[
r_k \leq \left( \frac{1}{1-a} \right)^{1-\mu} \Rightarrow (1 - (1-a)r_k^{1-\mu}) \geq 0 \Rightarrow r_{k+1} = 0.
\]
(30)
Therefore, for a finite integer \( k = K^* - 1 \) satisfying (30), i.e.,
\[
r_{K^* - 1} \leq \left( \frac{1}{1-a} \right)^{1-\mu},
\]
(31)
there exists \( r_{K^*} = 0 \). Hence, it follows from (28) that
\[
V(K^* - 1) \leq \left( \frac{b}{1-a} \right)^{1-\mu} V(K^*) = 0.
\]
(32)
Using (23), one has
\[
V(1) - V(0) = -aV(0) - bV^\mu(0),
\]
\[
V(2) - V(1) = -aV(1) - bV^\mu(1),
\]
\[
\vdots
\]
\[
V(K^* - 1) - V(K^* - 2) = -aV(K^* - 2) - bV^\mu(K^* - 2),
\]
\[
V(K^*) - V(K^* - 1) = -aV(K^* - 1) - bV^\mu(K^* - 1),
\]
which leads to
\[
V(K^*) - V(0) = \sum_{k=0}^{K^*-1} -aV(k) - bV^\mu(k).
\]
(33)
Since \( V(k+1) < V(k) \), (32) can be rewritten as
\[
V(K^*) - V(0) \leq K^*( -aV(K^* - 1) - bV^\mu(K^* - 1)).
\]
(34)
Using (31), (33) results in the following settling-time
\[
K^* \leq \left[ \frac{V(0)}{a(b^{1-\mu} - b\left( \frac{1}{1-a} \right)^{1-\mu}} + 1 \right].
\]
(35)
Hence, \( r_k = 0 \) and \( V(k) = 0 \) for all \( k \geq K^*(V(0)) \) where \( K^*(V(0)) = \left[ \frac{V(0)}{a(b^{1-\mu} - b\left( \frac{1}{1-a} \right)^{1-\mu}} + 1 \right] \) is the settling-time function. Therefore, the system state converges to zero for all \( k \geq K^*(V(0)) \) that assures the finite-time stability of the system. This completes the proof. 

**Fact 1:** For every matrix \( A \) and \( B \) of the same dimensions, it is known that \( ||A|| - ||B|| \leq ||A - B|| \leq ||A|| + ||B|| \).

**IV. Finite-Time Convergence Analysis for the Proposed FTCL**

The following theorem demonstrates the finite-time convergence properties of the proposed learning method. It should be noted that in the proposed FTCL method, the stored data in \( M \) and other stacks is selected based on data recording algorithm in [25] to maximize \( \lambda_{\max}(S) \) where \( S = \sum_{h=1}^P \tilde{d}(\tau_h)\tilde{d}^T(\tau_h) \), and due to the satisfaction of \( M \) rank condition, \( S > 0 \).

**Theorem 1:** Consider the approximator for nonlinear system (1) given by (14), whose parameters are adjusted according to the update law of (20) with the regressor given by (11). Let Assumptions 1-2 hold, once the rank condition on \( M \) and
\[
\gamma < \frac{2\xi_1\lambda_{\min}(S)}{\xi_G^2 + 2\xi_G\xi_C\lambda_{\max}(S)(1 + \frac{1}{\beta^2}) + 2\xi_C^2\lambda_{\min}(S)(1 + \frac{1}{\beta^2})^2}
\]
are met, then

1) for adaptive approximators with zero MFAE, i.e., \( \xi(k) = 0 \), the proposed parameter update law (20) guarantees that \( \Theta(k) \) converges to zero within finite time steps;

2) for adaptive approximators with non-zero MFAE, i.e., \( \xi(k) \neq 0 \), (\( \|\xi(k)\| \leq b_\xi \)), the proposed parameter update law (20) guarantees that \( \Theta(k) \) is ultimately bounded, \( \Theta(k) \leq b_\Theta \), in finite time steps, with the ultimate bound
\[
b_\Theta = -\tilde{b}_\Theta - \frac{\sqrt{(2\xi_1 - 4\xi_1\xi_2)\gamma}}{2a},
\]
(36)
where
\[
a = -2\xi_1\lambda_{\min}(S) + \xi_2^2\gamma + 2\xi_G\xi_C\lambda_{\max}(S)(\frac{1}{\beta} + 1) + \frac{\gamma^2\xi_1\lambda_{\max}(S)}{\lambda_{\min}(S)} + \frac{1}{\beta^2},
\]
(37)
\[
\tilde{b}_\Theta = \frac{\xi_2^2\xi_G^2}{\lambda_{\min}(S)} + 2b_\xi(\xi_C\xi_G\lambda_{\max}(S)(\xi_G + \xi_CP(1 + \frac{1}{\beta^2})) + \xi_C\xi_G\gamma + \xi_G + \xi_2^2\xi_G).
\]
(38)
\[
\epsilon = \tilde{b}_\Theta\left[ \frac{2\xi_1\xi_G}{\lambda_{\min}(S)} + (\xi_G^2b_\xi + 2\xi_G\xi_C + P(2\xi_1\xi_G^4b_\xi + \xi_G^2Pb_\xi + 2\xi_1^2 + 2\xi_1\xi_2\lambda_{\max}(S)) + \frac{\xi_2^2\xi_G^2}{\lambda_{\min}(S)} + \frac{\xi_2^2Pb_\xi}{\beta^2}) \right].
\]
(39)

**Proof:** Consider the Lyapunov function candidate
\[
V(k) = tr\{\tilde{\Theta}^T(k)\Gamma^{-1}\tilde{\Theta}(k)\}.
\]
(40)

The rate of change of \( V(k) \) is
\[
\Delta V(k) = V(k) - V(k - 1)
\]
\[
= tr\{\tilde{\Theta}^T(k)(\Gamma^{-1}\tilde{\Theta}(k) - \tilde{\Theta}^T(k - 1)\Gamma^{-1}\tilde{\Theta}(k - 1))\}
\]
\[
= tr\{\tilde{\Theta}(k) - \tilde{\Theta}(k - 1)\} \tilde{\Theta}^T(k - 1)\tilde{\Theta}(k) - \tilde{\Theta}(k - 1)\}.
\]
(41)

Using (15), (19) and (20), (41) can be written as,
\[
\Delta V(k) = tr\{(-\Gamma^2\xi_C\xi_G(\tilde{d}(k - 1) - \tilde{d}(k - 1)\tilde{e}^T(k - 1))) + \xi_C\tilde{\Theta}(k - 1) - \tilde{d}(k - 1)\tilde{e}^T(k - 1))\}
\]
\[
\tilde{\Theta}(k - 1) - \tilde{d}(k - 1)\tilde{e}^T(k - 1))\}
\]
(42)

Hence, the system states converge to zero for all \( k \geq K^*(V(0)) \) that assures the finite-time stability of the system. This completes the proof.
Using inequalities (47)-(50) and knowing $\|D_\beta(k-1)\| < 1$, it follows that
\[
\Delta V(k) \leq a|\hat{\Theta}(k-1)|^2 + \epsilon|\hat{\Theta}(k-1)| + \epsilon, \tag{51}
\]
where $a$ and $\epsilon$ are given in (37) and (39), respectively, and
\[
\delta = \frac{2\xi_C}{(\eta + 1)}\frac{\lambda_{\min}(S)}{\lambda_{\max}(S)} + 2\beta(\xi_C\gamma\lambda_{\max}(S))\xi_G + \xi_CP(1 + \frac{1}{\beta^2}) + \xi_C\gamma\xi_G + \xi_C + \xi_G^2. \tag{52}
\]
One can see from (40) that
\[
V(k) \leq \frac{1}{\gamma}|\hat{\Theta}(k)|^2 \implies \sqrt{V(k)} \leq |\hat{\Theta}(k)|. \tag{53}
\]
Now, we have the following two cases:

1) Adaptive approximators with zero MFAEs ($\varepsilon(k) = 0$)

For adaptive approximators with zero MFAEs, i.e., $\varepsilon(k) = 0$ and $b_\varepsilon = 0$, using (53), (51) leads to
\[
\Delta V(k) \leq -\alpha_r V(k-1) - b_\gamma V^2(k-1), \tag{54}
\]
where
\[
\alpha_r = -\gamma a, \quad b_\gamma = \frac{2\sqrt{\|\hat{\Theta}(k-1)\|}}{(\eta + 1)}\frac{\lambda_{\min}(S)}{\lambda_{\max}(S)}. \tag{55}
\]
Invoking Lemma 2, for $0 < \alpha_r < 1$ and $b_\gamma > 0$ (already satisfied), $\hat{\Theta}(k)$ converges to zero within finite-time steps. The following condition shall be satisfied to keep $0 < \alpha_r$,
\[
\gamma < \frac{2\xi_C\lambda_{\min}(S)}{\xi_G^2} + 2\xi_G\xi_C\lambda_{\max}(S)(1 + \frac{1}{\beta}) + \xi_C\lambda_{\max}(S)(1 + \frac{1}{\beta^2})^2, \tag{56}
\]
and $\alpha_r < 1$ always hold since
\[
(1 - \gamma C\lambda_{\max}(S))^2 + 2\gamma^2 \xi_G\xi_C\lambda_{\max}(S)(1 + \frac{1}{\beta})^2 + \xi_G^2\gamma^2 + 2\gamma^2 \xi_C^2\lambda_{\max}(S)(\frac{2}{\beta} + \frac{1}{\beta^2}) > 0. \tag{57}
\]
Therefore, based on Lemma 2 by satisfying (56), $\hat{\Theta}(k)$ converges to zero and $V(k) = 0$ for $k \geq K^*_N(V(0))$. Similar to the procedure for obtaining (34) in Lemma 2, one obtains the associated settling-time function as follows
\[
K^*_N(V(0)) = \left[\frac{V(0)}{\alpha_r(b_\gamma + \beta_\gamma (\frac{b_\gamma}{\beta}))}\right] + 1. \tag{58}
\]
Thus, for $k \geq K^*_N(V(0))$, one has $V(k) = 0$ and $\hat{\Theta}(k) = 0$.

2) Adaptive Approximators with non-zero MFAEs ($\varepsilon(k) \neq 0$)

For approximators with non-zero MFAEs, i.e., $\varepsilon(k) \neq 0$, it is known that $\epsilon > 0$ and $a < 0$ (satisfying (56)). Thus, we proceed by bounding $\beta$ with $b_\varepsilon$, given in (38), where from (51) one obtains
\[
\Delta V(k) \leq a|\hat{\Theta}(k-1)|^2 + b_{\varepsilon}(\hat{\Theta}(k-1)| + \epsilon. \tag{59}
\]
Since, $|\hat{\Theta}(k)| \geq 0$ and $a < 0$, if (56) is satisfied, the only valid non-negative root of (59) is
\[
b_{\varepsilon} = -b_{\varepsilon} - \sqrt{(b_{\varepsilon})^2 - 4(a)\epsilon}. \tag{60}
\]
Thus, if $|\hat{\Theta}(k)| > b_{\varepsilon}$,
\[
\Delta V(k) = V(k) - V(k-1) < 0, \tag{61}
\]
whereas, after $\hat{\Theta}(k)$ enters the set
\[
S^0_{\beta} = \{ \hat{\Theta} : |\hat{\Theta}| \leq b_{\beta}\}, \tag{62}
\]
it is possible to have $\Delta V(k) > 0$. However, for discrete-time samples thereafter, $\hat{\Theta}(k)$ stay within the positive invariant set $S^u_{\hat{\Theta}}$. Therefore, provided that $|\hat{\Theta}(0)| > b_{\hat{\Theta}}$, for all $k$,

$$|\hat{\Theta}(k)| \leq b_{\hat{\Theta}}. \quad (63)$$

Hence, provided that the rank condition on $M$ is met and $\gamma$ satisfies (56), the FTCL method guarantees that $\hat{\Theta}(k)$ is uniformly ultimately bounded with $S^u_{\hat{\Theta}}$ being invariant. To obtain the finite time $K_2^u(V(0))$ that $\|\hat{\Theta}(k)\|$ reaches the invariant set $S^u_{\hat{\Theta}}$ and remains thereafter, first using (53), (59) is written as

$$\Delta V(k) \leq -a_r V(k - 1) + b_\Theta |\hat{\Theta}(k - 1)| + \epsilon. \quad (64)$$

Then, similar to the proof of Lemma 2, since, for $k = 1, \ldots, K^*_2 - 1, K^*_2, V(k) < V(k - 1)$, (64) leads to

$$V(K^*_2) - V(0) \leq K^*_2(-a_r V(K^*_2) + b_\Theta |\hat{\Theta}(0)|) + \epsilon. \quad (65)$$

Using $V(K^*_2) < \gamma^{-1}(b_\Theta)^2$, (65) leads to $K^*_2 \leq K_2^u(V(0))$ where,

$$K^*_2(V(0)) = \left[ \frac{V(0) - \gamma^{-1}(b_\Theta)^2}{a_r(\gamma^{-1}(b_\Theta)^2) - b_\Theta |\hat{\Theta}(0)| - \epsilon} \right] + 1. \quad (66)$$

Therefore, for $k \geq K^*_2(V(0))$, $\hat{\Theta}(k)$ reaches the invariant set $S^u_{\hat{\Theta}}$ and remains thereafter. This completes the proof. ■

Remark 1: As (58) and (66) show, $b_\Theta$ and $a_r$ dictate how fast $\hat{\Theta}(k)$ converges to zero and the specified bound around zero for respectively the cases with $\varepsilon(k) = 0$ and $\varepsilon(k) \neq 0$. To have a faster convergence time in (58), maximizing $b_\Theta$, given in (55), by maximizing $\lambda_{\min}(S)$ is beneficial. Maximizing $\lambda_{\max}(S)$ (maximizing $\lambda_{\min}(S)$ and minimizing $\lambda_{\max}(S)$) also leads to enlarging $a_r$ which causes faster settling-time in (66). This result completely coincides with the obtained results in [20] and even supports the continuous-time framework studies in [2,4,26]. Therefore, while applying the proposed FTCL, the data recording algorithm in [25] is used, where the appropriate data is selected to maximize $\lambda_{\min}(S)$.\n
Remark 2: In approximators with non-zero MFAE ($\varepsilon(k) \neq 0$), maximizing $\lambda_{\min}(S)$ using the data recording algorithm in [25], helps to respectively enlarge and reduce the amplitudes of $a_r$ and $\gamma$ that leads to narrow down the error bound in (36), whereas maximizing $\lambda_{\max}(S)$ matches with the concepts of concurrent learning in continuous-time [2,26]. Furthermore, choosing $P = p + q$ (satisfying $P \gtrsim (p + q)$) not only helps to maximize $\lambda_{\min}(S)$ [25], but also, keeps $\epsilon$ small.

### V. Simulation Results and Discussion

In this section, the performance of the proposed finite-time concurrent learning for both adaptive approximators with zero and non-zero MFAEs is examined in comparison with asymptotically converging concurrent learning [20] and traditional gradient descent [1] whose estimation laws are respectively given as follows,

$$\dot{\hat{\Theta}}(k + 1) = \hat{\Theta}(k) - \Gamma_C \tilde{d}(k)e^T(k), \quad (67)$$

$$\dot{\hat{\Theta}}(k + 1) = \hat{\Theta}(k) - \Gamma_C[\Sigma_G \tilde{d}(k)e^T(k) + \Sigma_C \sum_{h=1}^{P} \tilde{d}(\tau_h)e^T_h(k)], \quad (68)$$

where $\Gamma_C = \gamma_C I$, $\Gamma_C = \gamma_C I$, $\Sigma_G = \sigma I$ and $\Sigma_C = \sigma_C I$ with positive constants $\gamma_C > 0$, $\Sigma_C > 0$, $\sigma_C > 0$ and $\sigma C > 0$.

For both cases, the simulation time span is $[k_0, k_f]$ where $k_0 = 0$ and $k_f = 1000$, and the $x$ domain is defined by $D_x = [x_L, x_H]$.

### TABLE I: Learning errors comparison

| Example 1 | IAE $E_f(k)$ | IAE $E_g(k)$ | IAE $E_f(k)$ | IAE $E_g(k)$ |
|-----------|--------------|--------------|--------------|--------------|
| FTCL      | 36.21        | 22.08        | 296.02       | 410.72       |
| CL        | 59.24        | 42.32        | 354.83       | 607.32       |
| GD        | 303.89       | 141.11       | 1549         | 1389         |

where $x_L < x_H$ and $x_L, x_H \in \mathbb{R}$ and $D_x$ is quantized by $[x_L : x_H - x_{k_0} : x_H]$. In the proposed FTCL method, $\gamma$ is chosen to satisfy (35). For concurrent learning according to [20], $\gamma C \leq \frac{2\gamma + \sigma C \lambda_{\max}(S)}{2\gamma + \sigma C \lambda_{\max}(S)}$ where it is chosen as $\gamma C = \frac{2\gamma + \sigma C \lambda_{\max}(S)}{2\gamma + \sigma C \lambda_{\max}(S)}$. In all cases, the initial values and the controllers are all set to zero. A small exponential sum of sinusoidal input is injected to the system controller for ensuring the rank condition on the collected data and the data selection procedure in [25] is employed for both FTCL and concurrent learning methods. To fairly compare the speed and precision of the intended on-line learning methods for approximating $f(x)$ and $g(x)$ on the whole domain of $x$ as time evolves, the following learning errors are computed on-line.

$$E_f(k) = \int_{D_x} \|e_f(x(k))\|^2 dx, \quad E_g(k) = \int_{D_x} \|e_g(x(k))\|^2 dx.$$

In the simulations, the results of the proposed finite-time concurrent learning, the traditional concurrent learning and gradient descent methods are respectively labeled by FTCL, CL and GD.

### A. Example 1: Simulations for approximators with zero MFAE ($\varepsilon(k) = 0$)

Consider the following system

$$x(k + 1) = p_1e^{-x(k)} + p_2e^{-x(k)}\cos(x(k)) + p_3\frac{1}{1 + x(k)}u(k),$$

where the parameters $[p_1, p_2, p_3]$ are unknown and the regressors are fully known as

$$z(x(k), u(k)) = [e^{-x(k)}e^{-x(k)}\cos(x(k)), \frac{1}{1 + x(k)}u(k)],$$

with $p + q = 3$. The unknown parameters are $[p_1, p_2, p_3] = [-1, 1.5, 1]$ and $D_x$ is limited with $x_L = 0$ and $x_H = 2$. We set $P = 3$ for both FTCL and concurrent learning methods. Let $\gamma G = 0.7$ for gradient descent method, and $\sigma G = 1$, $\sigma C = 0.4$ for concurrent learning method, and $\xi G = 1$, $\xi C = 0.3$ and $\beta = 0.3$ for FTCL method.

Fig. 1 depicts the true parameters and the approximated parameters for FTCL, concurrent learning and gradient descent methods. In Fig. 1, while gradient descent could not converge to the true parameters due to the lack of persistency of excitation in the learning time length, FTCL and concurrent learning succeeded in convergence. However, FTCL resulted in faster convergence to the true parameters in comparison with concurrent learning. The on-line learning errors $E_f(k)$ and $E_g(k)$ for the FTCL, concurrent learning and gradient descent are plotted in Fig. 2 where FTCL shows faster converging to the origin. Fig. 3 shows that the state estimation error of all methods has converged to zero where the results of gradient descent illustrate that the convergence of the state estimation error to zero does not necessarily imply that the parameters’ estimation error converges to zero. The integral absolute errors (IAEs) of $E_f(k)$ and $E_g(k)$ for FTCL, concurrent learning and gradient descent methods are computed in Table 1 where FTCL with IAEs 36.21 and 22.08 respectively for $E_f(k)$ and $E_g(k)$ has resulted in the best precision of on-line learning in comparison with concurrent learning and gradient descent.
Consider 5 radial basis functions defined as
\[ r_{\text{rbf}}(i) = \exp\left(-\frac{(x(k) - c_i)^2}{2\sigma_i^2}\right), \quad i = 1, 2, \ldots, 5 \]
where the centroids \( c_i \) are uniformly picked on \( D_x = [x_L, x_H] = [-2, 2] \) and the spreads are all fixed to \( \sigma_i = 1.2 \).

Therefore, the employed regressor is
\[
z(x(k), u(k)) = \begin{bmatrix} e^{-\frac{|x(k) - c_1|^2}{2\sigma_1^2}} & \cdots & e^{-\frac{|x(k) - c_5|^2}{2\sigma_5^2}} \\ e^{-\frac{|x(k) - c_1|^2}{2\sigma_1^2}} u(k) & \cdots & e^{-\frac{|x(k) - c_5|^2}{2\sigma_5^2}} u(k)^T \end{bmatrix},
\]
with 10 independent basis functions which leads to setting \( P = 10 \).

The approximation of (69) is given as
\[
x(k + 1) = \hat{\Theta}(k) z(x(k), u(k)) = [p_1, p_2, \ldots, p_{10}] z(x(k), u(k)).
\]

Employing \( \gamma_G = 0.8 \) for gradient descent method, \( \sigma_G = 1.2 \) and \( \sigma_C = 0.1 \) for concurrent learning method, and \( \xi_G = 1, \xi_C = 0.1 \) and \( \beta = 0.65 \) for FTCL method leads to the approximated parameters depicted on Fig. 4. Due to the lack of persistence of excitation in the learning time length, gradient descent parameters in Fig. 4 could not converge to the appropriate parameters, while FTCL and concurrent learning methods using a memory stack of data, satisfying the rank condition, succeeded in convergence to the suitable parameters. The steady state approximations for the uncertainties \( f(x) \) and \( g(x) \) are given in Fig. 5. As the comparison of the learning errors \( E_f(k) \) and \( E_g(k) \) in Fig. 6 shows, the gradient descent method did not perform well in learning the uncertainty, however both FTCL and concurrent learning errors showed bounded convergence near zero, where FTCL method is faster in converging to a smaller bound near zero in comparison with concurrent learning approach. Fig. 7 shows that the state estimation error of all methods has converged to a bound very close to zero. Furthermore, based on the results of IAE for \( E_f(k) \) and \( E_g(k) \) in Table 1, FTCL results in lower errors during the whole time of on-line learning in comparison with concurrent learning and gradient descent.
VI. CONCLUSION AND FUTURE WORK

This paper addressed finite-time identification of discrete-time system dynamics where finite-time learning could speed up the learning and concurrent learning technique relaxed the persistence of excitation condition on the regressor to a rank condition on the memory stack of recorded data. Learning rate condition was obtained for finite-time convergence based on discrete and finite time analysis. It was discussed that the precision and speed of the proposed finite-time convergence based on discrete and finite time analysis. For future work, it is intended to extend the existing results for the identification of time-varying systems.

REFERENCES

[1] J. A. Farrell and M. Polycarpou, Adaptive Approximation based Control: Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches. John Wiley and Sons, 2006.

[2] G. Chowdhary, “Concurrent learning for convergence in adaptive control without persistency of excitation,” Ph.D. dissertation, Georgia Inst. Technol., Atlanta, GA, USA, Dec. 2010.

[3] H. Modares, F. L. Lewis, M.-B. Naghibi-Sistani, “Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks,” IEEE Trans. Neural Netw. Learn. Syst., vol. 24, no. 10, pp. 1513–1525, 2013.

[4] G. Chowdhary, T. Yucelen, M. Muhlegg, and E. Johnson, “Concurrent learning adaptive control of linear systems with exponentially convergent bounds,” Int. J. Adapt. Control Signal Process., vol. 27, no. 4, pp. 280-301, 2013.

[5] R. Kamalapurkar, B. Reish, G. Chowdhary, and W. E. Dixon, “Concurrent Learning for Parameter Estimation Using Dynamic State-Derivative Estimators,” IEEE Trans on Automatic Control, vol. 62, no. 7, pp. 3594-3601, 2017.

[6] F. Tatari, K. G. Vamvoudakis, M. Mazouchi, “Optimal distributed learning for disturbance rejection in networked non-linear games under unknown dynamics，“IET Control Theo. App., vol. 13, no. 17, pp. 2838-2848, 2019.

[7] S. Li, H. Du, and X. Yu, “Discrete-Time Terminal Sliding Mode Control Systems Based on Euler’s Discretization,” IEEE Trans. on Automatic Control, vol. 59, no. 2, pp. 546-552, 2014.

[8] G. Sun, Z. Ma, and J. Yu, “Discrete-Time Fractional Order Terminal Sliding Mode Control for Linear Motor,” IEEE Trans. on Industrial Electronics, vol. 65, no. 4, pp. 3386-3394, 2018.

[9] Q. Zhao, H. Xu, and S. Jagannathan, “Neural Network-Based Finite-Horizon Optimal Control of Uncertain Affine Nonlinear Discrete-Time Systems,” IEEE Trans. on Neural Net. and Lear. Syst., vol. 26, no. 3, pp. 486-499, 2015.

[10] L. Liu, Y. J. Liu, and S. Tong, “Neural Networks-Based Adaptive Finite-Time Fault-Tolerant Control for a Class of Strict-Feedback Switched Nonlinear Systems,” IEEE Trans. on Cybernetics, vol. 49, no. 7, pp. 2536-2545, 2019.

[11] Y. Liu, X. Liu, Y. Jing, X. Chen, and J. Qiu, “Direct Adaptive Preassigned Finite-Time Control With Time-Delay and Quantized Input Using Neural Network,” IEEE Trans. on Neural Net. and Lear. Sys., vol. 31, no. 4, pp. 1222-1231, 2020.

[12] Y. Li, T. Yang, and S. Tong, “Adaptive Neural Networks Finite-Time Optimal Control for a Class of Nonlinear Systems,” IEEE Trans. on Neural Net. and Lear. Sys., Date of Publication: 19 December 2019, doi: 10.1109/TNNLS.2019.2955438.

[13] V. Adetola and M. Guay, “Finite-time parameter estimation in adaptive control of nonlinear systems,” IEEE Trans. Autom. Control, vol. 53, no. 3, pp. 807-811, 2008.

[14] L. Zhao, J. Zhi, N. Yin, Y. Chen, J. Li, and J. Liu, “Performance Improvement of Finite Time Parameter Estimation with Relaxed Persistence of Excitation Condition,” Journal of Elec. Eng. and Tech., vol. 14, pp. 931-939, 2019.

[15] C. Yang, Y. Jiang, W. He, J. Na, Z. Li, and B. Xu, “Adaptive Parameter Estimation and Control Design for Robot Manipulators With Finite-Time Convergence,” IEEE Trans. on Industrial Electronics, vol. 65, no. 10, pp. 8112-8123, 2018.

[16] J. Wang, D. Efimov, A. A. Bobtsov, “On Robust Parameter Estimation in Finite-Time Without Persistence of Excitation,” IEEE Trans. on Automatic control, vol. 65, no. 4, pp. 1731-1738, 2020.

[17] J. Wang, D. Efimov, and A. A. Bobtsov, “Finite-time parameter estimation without persistence of excitation,” in Proc. 18th Eur. Control Conf. (ECC), June 2019.

[18] A. Valdidi-Moghaddam, M. Mazouchi, H. Modares, “Memory-Augmented System Identification with Finite-Time Convergence,” IEEE Control Sys. Letters, vol. 5, no. 2, pp. 571-576, 2021.

[19] D. Lehrer, V. Adetola, M. Guay, “Parameter identification methods for non-linear discrete-time systems,” in Proc. 2010 American Control Conf. (ACC), Jun.-Jul. 2010, pp. 2170-2175.

[20] O. Djanye-Boundjou and R. Ordonez, “Gradient-Based Discrete-Time Concurrent Learning for Standalone Function Approximation,” IEEE Trans on Automatic Control, vol. 65, no. 2, pp. 749-756, 2020.

[21] R. Hannah, A. K. Sanyal, S. P. Viswanathan, “Discrete-Finite-time Stable Position Tracking Control of Unmanned Vehicles,” in Proc. 58th Conf. on Decis. Control (CDC), Dec. 2019.

[22] G. Tao, Adaptive Control Design and Analysis (Adaptive and Learning Systems for Signal Processing, Communications and Control Series). New York, NY, USA: Wiley, 2003.

[23] S. P. Bhat, D. S. Bernstein, “Finite time stability of continuous autonomous systems,” SIAM Journal on Control and Opt., vol. 38, no. 3, pp. 751-766, 2000.

[24] H. K. Khalil, Nonlinear Systems, 3rd ed. US, Prentice Hall, 2002.

[25] A. Vahidi-Moghaddam, M. Mazouchi, H. Modares, “Concurrent Learning for disturbance rejection in networked non-linear games under unknown dynamics,” IET Control Theo. App., vol. 13, no. 17, pp. 2838-2848, 2019.