Adaptive neural network-based optimal control of nonlinear continuous-time systems in strict-feedback form

H. Zargarzadeh 1,*,†, T. Dierks 2 and S. Jagannathan 1

1 Department of Electrical and Computer Engineering, Missouri University of Science and Technology, Rolla, MO, USA
2 DRS Sustainment Systems, Inc., 201 Evans Lane, St Louis, MO 63121, USA

SUMMARY

This paper focuses on neural network (NN) based optimal control of nonlinear continuous-time systems in strict-feedback form when the system dynamics are known by using an adaptive backstepping approach. A single NN-based adaptive approach is designed to learn the solution of the infinite horizon continuous-time Hamilton–Jacobi–Bellman (HJB) equation while the corresponding optimal control input that minimizes the HJB equation is calculated in a forward-in-time manner without using value and policy iterations. First, the optimal control problem is solved for a generic multi-input and multi-output nonlinear system with a state feedback approach. Then the approach is extended to a single-input and single-output nonlinear system by using output feedback via a nonlinear observer. Lyapunov techniques are used to show that all signals are uniformly ultimately bounded and that the approximated control signals approach the optimal control inputs with small bounded error both for the state and output feedback-based controller designs. In the absence of NN reconstruction errors, asymptotic convergence to the optimal control is demonstrated. Finally, simulation examples are provided to validate the theoretical results. Copyright © 2013 John Wiley & Sons, Ltd.

Received 13 November 2011; Revised 2 July 2013; Accepted 17 August 2013

KEY WORDS: online nonlinear optimal control; neural network control; output feedback control; strict-feedback systems

1. INTRODUCTION

During the past few decades, stabilization of nonlinear systems has been tackled through a variety of ways [1]. One of the objectives of the nonlinear stabilization is to design an adaptive controller due to structured uncertainties [2]. In addition to stabilization, it is desired that the control law minimizes a predefined performance measure [3, 4] so that optimal control laws can be generated. It is well known that the optimal control of linear systems is obtained by solving the Riccati equation [3] in a backward-in-time manner when the system dynamics are known. In contrast, the optimal control of nonlinear continuous or discrete-time systems is a much more challenging task, even when the system dynamics are known, as it often requires solving the nonlinear Hamilton–Jacobi–Bellman (HJB) equation, which has no closed-form solution [3, 4]. Moreover, when the system dynamics are uncertain, solving the HJB equation is a major challenge similar to solving the Riccati equation for uncertain linear systems.

Thus, the optimal control of uncertain nonlinear systems is an important and difficult problem. For uncertain nonlinear systems, the adaptive control methods can still be employed not only to estimate the solution to the HJB equation but also to approximate the optimal control law [5, 6]. As a first step, researchers utilized policy iterations and Q-learning-based adaptive control schemes in [6, 7], respectively, for uncertain linear systems to derive the optimal control laws.

*Correspondence to: H. Zargarzadeh, Department of Electrical and Computer Engineering, Missouri University of Science and Technology, Rolla, MO, USA.
†E-mail: hzr47@mail.mst.edu

Copyright © 2013 John Wiley & Sons, Ltd.
On the other hand, the optimal control of nonlinear continuous-time systems in affine form with known system dynamics has been addressed initially with the aim of minimizing a predefined index functional by finding an approximate solution to the HJB equation [4, 6] using an iterative, offline, and backward-in-time [4] methodology. Subsequently, adaptive critic designs (ACD) that utilizes a forward-in-time yet iterative methodology are proposed [8] for nonlinear systems when the system dynamics are uncertain. The central theme in the ACD techniques is that the optimal control law and HJB function are approximated by online parametric structures, such as neural networks (NNs) in a forward-in-time manner by using policy and value iterations [6, 8–10]. Because of the online parametric approximators, the ACD techniques [11] relax the need for system dynamics. Recently, in [12], a new framework of ACDs with sparse kernel machines is presented by integrating kernel methods into the critic of ACDs.

While these methods [6, 8, 9] render stability, the number of iterations needed within a sampling interval for convergence is significant, and therefore, such iterative schemes are not preferred for hardware implementation [11]. In addition, it is found that inadequate number of policy or value iterations [11] can lead to instability when using these online ACD methods. In some cases [13], the mathematical proofs of convergence are not offered [6], and an initial stabilizing control is needed [8]. Additionally, the ACD schemes in general require two NNs, one NN for approximating the cost function while the other NN for estimating the control input, which may be considered computationally intensive. Therefore, in [5, 14], online approximator (OLA) based ACD techniques are introduced for nonlinear continuous-time systems in affine and nonaffine forms without using value and policy iterations.

All the available ACD techniques [5, 6, 8, 9, 13] in the literature address nonlinear continuous or discrete-time systems [11] in affine form by assuming that the states are considered measurable. In addition, an initial stabilizing control input is needed in order to learn the cost or value function. On the other hand, nonlinear systems in strict-feedback form [15, 16] are an important class of nonlinear systems structurally made by cascading affine nonlinear continuous-time systems. However, despite their similarity to affine systems, the strict-feedback nonlinear systems require a different way of designing control schemes.

The control of nonlinear continuous-time systems in strict-feedback form with uncertain dynamics is attempted in [15] by using standard linear in the unknown parameter (LIP) adaptive backstepping scheme without minimizing a performance index. Other papers [16–18] focus on the tracking control of unknown nonlinear continuous-time systems in strict-feedback form by using NN-based backstepping schemes without any sort of optimality. More recently, the inverse optimal control of strict-feedback nonlinear systems is introduced in [19] under the assumption that the system dynamics are known. In the inverse optimal control design [19], first, the control law is designed based on known system dynamics, and then it is shown that there exists a meaningful performance functional, which can be minimized by using the control law. This is in contrast with the traditional optimal control design where the control law is designed based on a given cost function.

To the best knowledge of the authors, no known adaptive optimal control scheme is available for nonlinear continuous-time systems in strict-feedback form when its dynamics are known. In addition, no known output feedback-based optimal adaptive technique is available in the literature for such systems. Therefore, in this paper, a novel optimal adaptive control scheme using an NN is introduced for nonlinear continuous-time systems in strict-feedback form with known system dynamics. First, the nonlinear continuous-time system in strict-feedback form is transformed into a nonlinear tracking error system in affine form by using the backstepping-like technique as there is no known way to directly design an optimal adaptive controller for such strict-feedback nonlinear continuous-time systems. Then a single OLA (SOLA) is utilized for the tracking error system in affine form to learn its cost function, which becomes the solution to the HJB equation, in a forward-in-time manner. The Lyapunov theory is utilized to demonstrate the convergence of this approximate optimal control scheme for the overall nonlinear continuous-time system while explicitly considering the approximation errors resulting from the use of the OLA in the backstepping approach. Initially, the state feedback-based optimal design is considered for multi-input and multi-output (MIMO) systems, and subsequently, an output feedback controller design is addressed for single-input and single-output (SISO) nonlinear continuous-time strict-feedback systems.
In this paper, policy and value iterations that are commonly found in ACD techniques [8, 9] are not utilized. Instead, value and policies are updated once in a sampling interval, thus making the scheme suitable for real-time control. An initial stabilizing control is not required in contrast to [6, 8, 11]. It is shown that the approximated control input approaches the optimal value over time, and if the NN reconstruction errors become zero as in the case of traditional adaptive control, asymptotic stability is demonstrated.

The contribution of this paper is the development of an adaptive optimal controller for a class of nonlinear continuous-time systems in strict-feedback form first by using state measurements and then with output feedback. An initial stabilizing control is not required, and stability is demonstrated using Lyapunov methods without using value and policy iterations.

The proposed optimal adaptive NN control is a third generation of NN-based learning control system, which has its own challenging issues to be solved when used in feedback control of dynamic systems. Traditionally, the first generation of the NN-based controllers needed to be trained offline by using back propagation [20] while stability is not shown. On the other hand, the second generation of NN-based controllers [20] simply met stability requirements but not optimality, while the third generation of NN-based controllers guarantee both stability and optimality objectives through online learning. Achieving optimality is often not straightforward and requires novel design methods and nonstandard NN weight update laws along with a persistence of excitation condition. By using novel design and nonstandard NN weight update laws, the HJB solution is approximated online in this work, and optimality is demonstrated.

The paper is organized as follows. Section 2 is dedicated to the optimal control of a class of nonlinear continuous-time systems in strict-feedback form by transforming the system to an equivalent nonlinear system in affine form. Section 3 introduces an online optimal stabilization scheme for affine systems by assuming state feedback. Next, Section 4 extends the results from Section 3 to an observer-based output control approach where the state measurements are relaxed. Finally, Section 5 provides numerical results for the proposed optimal controller.

In the next section, a solution to the optimal tracking control of nonlinear system in strict-feedback form is introduced.

2. THE TRACKING PROBLEM FOR STRICT-FEEDBACK SYSTEMS

Consider the MIMO nonlinear continuous-time system in the absence of disturbances described by

\[
\dot{x}_i = f_i(x_1, \ldots, x_i) + g_i(x_1, \ldots, x_i) \dot{x}_{i+1} \quad \text{for} \quad 1 \leq i \leq N - 1 \quad \text{and} \quad N \geq 2 \tag{1}
\]

\[
\dot{x}_N = f_N(x_1, \ldots, x_N) + g_N(x_1, \ldots, x_N) u, \tag{2}
\]

\[
y = x_1, \tag{3}
\]

where each \(x_i \in \mathbb{R}^m\) denotes a state vector, \(u \in \mathbb{R}^m\) represents the input vector with \(f_i(x_1, \ldots, x_i) \in \mathbb{R}^m\), and \(g_i(x_1, \ldots, x_i) \in \mathbb{R}^{m \times m}\) represent nonlinear smooth functions. For the class of nonlinear systems given by (1), the next system state is treated as the virtual control input. Nonetheless, the system is going to be controlled through the control input \(u\). The following assumption is needed before we proceed.

**Assumption 1**

It is assumed that \(g_i(x_1, \ldots, x_i) \neq 0 \quad (1 \leq i \leq N)\) is invertible, belongs to \(\Omega \in \mathbb{R}^n\), and it is bounded above and below satisfying \(g_i^\min \leq \|g_i(x_1, \ldots, x_i)\| \leq g_i^\max\) when the Frobenius norm is applied and where \(g_i^\min\) and \(g_i^\max\) are positive constants. Besides, it is assumed that system (1)-(2) is reachable.

Under the preceding conditions given in Assumption 1, the optimal control input for nonlinear system (1)-(2) can be calculated [3] through a backstepping approach.
In this case, the objective of our scheme is to design a controller \( u \) in order to have the output \( y \) to track a desired trajectory \( x_{1d} \) in an optimal manner. To this end, the backstepping approach [15] is slightly modified such that the system given by (1)–(2) tracks a predesign trajectory \( (x_{2d}, \ldots, x_{Nd}) \), and the corresponding tracking error optimally converges to zero.

To stabilize the tracking error, \( e_1 = x_1 - x_{1d} \), the backstepping approach will use \( N \) steps [1], which are presented next.

**Step 1**

It is desired that \( x_1 \) follow the smooth desired trajectory \( x_{1d}^* \). Therefore, the first system dynamics of (1) can be rewritten as

\[
\dot{x}_1 - \dot{x}_{1d} = \dot{e}_1 = -\dot{x}_{1d} + f_1 (x_1, \ldots, x_l) + g_1 (x_1, \ldots, x_l) x_{(l+1)d} + g_1 (x_1, \ldots, x_l) (x_2 - x_{2d}) = f_1 (e_1) + g_1 (x_1) x_{2d}^* + g_1 (x_1) e_2.
\]

where virtual control input \( x_{2d} \) is chosen such that \( x_{2d} = x_{2d}^* + x_{2d}^a \) with the feedforward virtual control input \( x_{2d}^a \) selected by solving \(-\dot{x}_{1d} + f_1 (x_1) + g_1 (x_1) x_{2d}^a = f_1 (e_1)\), that is,

\[
x_{2d}^a = g_1^{-1} (x_1) \{ f_1 (e_1) + \dot{x}_{1d} - f_1 (x_1) \},
\]

where \( f_1 (x_1) - f_1 (x_{1d}) \) is denoted as \( f_1 (e_1) \) for convenience. Moreover, \( x_{2d}^a \) is going to be the optimal feedback control input. One should notice that in the standard backstepping method, \( x_{2d}^a \) is chosen to be zero, whereas here it plays a key role to make the system tracking error optimally converge to zero. In other words, when we choose \( x_{2d}^a = 0 \), the standard backstepping scheme guarantees that the error converges to zero asymptotically. However, by designing \( x_{2d}^a \), it will be shown that an index function of the applied input and the tracking error will be minimized. Section 3 is devoted to show the existence of \( x_{2d}^* \) and its design. Inevitably, \( e_2 \) cannot be zero because of dynamics of the second system of (1) and the desired output \( x_1 \) trajectory. Therefore, the next steps of the design procedure should handle this issue such that the last term of (4) becomes canceled by the next system dynamics in (1) by \( i = 2 \). As the second step to the \( (N - 1) \) step remains the same, we skip to the \( i^{th} \) step.

**Step. \( i \) (\( 2 \leq i \leq N - 1 \)):** In this step, we need an optimal controller for system (1)–(2) such that \( e_i \to 0 \). To this end, the system \( i \) in (1) can be rewritten as

\[
\dot{x}_i - \dot{x}_{id} = \dot{e}_i = -\dot{x}_{id} + f_i (x_1, \ldots, x_l) + g_i (x_1, \ldots, x_l) x_{(i+1)d} + g_i (x_1, \ldots, x_l) (x_{i+1} - x_{(i+1)d}) = f_i (e_1, \ldots, e_{i-1}) + g_i (x_1, \ldots, x_l) x_{(i+1)d}^* + g_i (x_1, \ldots, x_l) e_{i-1}.
\]

where \( x_{id} \) is chosen such that \( x_{(i+1)d} = x_{(i+1)d}^* + x_{(i+1)d}^a \), with the virtual control input \( x_{(i+1)d}^a \) satisfying \(-\dot{x}_{id} + f_i (x_1, \ldots, x_l) + g_i (x_1, \ldots, x_l) x_{(i+1)d}^a = f_i (e_1, \ldots, e_{i-1}) - g_i (x_1, \ldots, x_l) e_{i-1}\), that is,

\[
x_{(i+1)d}^a = g_i^{-1} (x_1, \ldots, x_l) \{ \dot{x}_{id} - f_i (x_1, \ldots, x_l) + f_i (e_1, \ldots, e_{i-1}) - g_i (x_1, \ldots, x_l) e_{i-1} \}. \tag{7}
\]

In the preceding equation, \( x_{(i+1)d}^* \) is the optimal solution for the virtual control inputs.

**Step. \( N \):** In this step, similar to the previous steps, the system input will be designed. To this end, system (2) can be rewritten as

\[
\dot{x}_N - \dot{x}_{Nd} = \dot{e}_N = -\dot{x}_{Nd} + f_N (x_1, \ldots, x_N) + g_N (x_1, \ldots, x_N) u = f_N (e_1, \ldots, e_{N-1}) + g_N (x_1, \ldots, x_N) u^* - g_N (x_1, \ldots, x_N) e_{N-1}. \tag{8}
\]
where \( x_{id} \) is chosen such that \( u = u^* + u^a \), where the feedforward control input \( u^a \) is selected from 
\[-x_{Nd} + f_N (x_1, \ldots, x_N) + g_N (x_1, \ldots, x_N) u^a = f_N (e_1, \ldots, e_N) - g_{N-1}^T (x_1, \ldots, x_{N-1}) e_{N-1}, \]
that is,
\[u^a = g_N^{-1} (x_1, \ldots, x_N) \{ \dot{x}_{Nd} - f_N (x_1, \ldots, x_N) + f_N (e_1, \ldots, e_N) - g_{N-1}^T (x_1, \ldots, x_{N-1}) e_{N-1} \}. \]

As mentioned in the previous steps, there exists an optimal feedback control input \( u^* \) that will be designed in Section 3. Now, from (4), (6), and (8), the tracking error dynamics takes the following form:
\[
\begin{align*}
\dot{e}_1 &= f_1 (e_1) + g_1 (x_1) x_{2d}^* + g_1 (x_1) e_2, \\
... \\
\dot{e}_i &= f_i (e_1, \ldots, e_i) + g_i (x_1, \ldots, x_i) x_{(i+1)d}^* + g_i (x_1, \ldots, x_i) e_{i+1} - g_{i-1}^T (x_1, \ldots, x_{i-1}) e_{i-1}, \\
... \\
\dot{e}_N &= f_N (e_1, \ldots, e_N) + g_N (x_1, \ldots, x_N) u^* - g_{N-1}^T (x_1, \ldots, x_{N-1}) e_{N-1}.
\end{align*}
\]

Now, we are ready to propose the contribution of this section by introducing the following lemma.

**Lemma 1**

Consider the tracking error dynamics defined in (4), (6), and (8) where the objective is to have the output \( y = x_1 \) to track the desired trajectory \( x_{1d} \). Assume that the virtual and real control input vector \( U = \begin{bmatrix} x_{2d} & \cdots & x_{Nd} & u \end{bmatrix} \) is designed such that \( U = U^a + U^* \), where \( U^a = \begin{bmatrix} x_{2d}^a & \cdots & x_{Nd}^a & u^a \end{bmatrix} \) is the feedforward control input designed in (5), (7), and (9) and \( U^* = \begin{bmatrix} x_{2d}^* & \cdots & x_{Nd}^* & u^* \end{bmatrix} \) represents the feedback control input, which optimally stabilizes the following error dynamics:

\[
\begin{bmatrix}
\dot{e}_1 \\
\vdots \\
\dot{e}_N
\end{bmatrix} = \begin{bmatrix} f_1 (e_1) \\
\vdots \\
f_N (e_1, \ldots, e_N) 
\end{bmatrix} + \begin{bmatrix} g_1 (x_1) & 0 \\
\vdots & \ddots \\
0 & g_N (x_1, \ldots, x_N) 
\end{bmatrix} U^*.
\]

Then optimal control of (1) and (2) is equivalent to the optimal controller design for (13). In other words, by applying \( U = U^a + U^* \) to system (1) and (2), system dynamics (1) and (2) are transformed into the error dynamic system given by (13).

**Proof**

By choosing \( J_1 = E^T E / 2 \) with \( E^T = \begin{bmatrix} e_1^T & \cdots & e_N^T \end{bmatrix} \) as the Lyapunov candidate and taking derivative through system dynamics (4), (6), and (8), we have
\[
J_1 = E^T E = E^T \left( \begin{bmatrix} f_1 (e_1) \\
\vdots \\
f_N (e_1, \ldots, e_N) 
\end{bmatrix} + \begin{bmatrix} g_1 (x_1) & 0 \\
\vdots & \ddots \\
0 & g_N (x_1, \ldots, x_N) 
\end{bmatrix} U^* \right).
\]

Therefore, the existence of an optimal controller \( U^* \) to make (14) negative is sufficient to make the closed-loop system stable. Therefore, it can be concluded that the feedforward (virtual controller) given by (5), (7), and (9) is able to reduce the optimal tracking problem to the optimal regulation of system (13) as claimed the hypothesis of the theorem. 

\[\square\]
3. OPTIMAL TRAJECTORY AND CONTROL INPUT DESIGN

Because of Lemma 1, the objective of this section is to optimally stabilize system (13). Now, in order to design the optimal control vector defined by \([x_{2d}^*, \ldots, x_{Nd}^*, u^*]\) such that the tracking error \((e_1, \ldots, e_N)\) is stable, define the cost function to be minimized

\[
V = \int_t^\infty r(E(\tau), U^*(\tau)) d\tau,
\]

where \(E = [e_1, \ldots, e_N]^T, U^* = [x_{2d}^*, \ldots, x_{Nd}^*, u^*]^T\), and \([x_1, \ldots, x_N] = X\). In (15), \(r(E, U^*) = Q(E) + U^{*T} RU^*, Q(E) \geq 0\) is the positive semidefinite penalty on the states, and \(R > 0 \in \mathbb{R}^{M\times M}\) is a positive definite matrix with \(M = mN\).

Equations (10)–(12) along with (14) demonstrate that the optimal control of nonlinear system in strict-feedback form can be transformed into solving optimal control of the affine nonlinear system written as a function of the error vector, \(E\), represented in (13). Now, consider the optimal stabilization problem for an affine nonlinear continuous-time system written in terms of the error vector

\[
\dot{E} = F(E) + G(X)U^*.
\]

where \([f_1^T (e_1) \cdots f_N^T (e_1, \ldots, e_N)]^T = F(E)\) and \(G(X) = \text{diag} [g_1 (x_1), \ldots, g_N (x_1, \ldots, x_N)]\). It is desired that \(E\) converges to zero while cost function (15) is minimized.

Moving on, the control input \(U^*\) is required to be designed such that cost function (15) is finite. We define the Hamiltonian for cost function (11) with an associated admissible control input \(U\) to be [3]

\[
H(E, U) = r(E, U) + V_E^T (F(E) + G(X)U),
\]

where \(V_E (E)\) is the gradient of the \(V(E)\) with respect to \(E\). In the sequel, we will use the same terminology for denoting gradient of functions, that is, for any function \(\Omega(\psi), \Omega_{\psi}(\psi)\ means \ gradient \ of \ \Omega(\psi)\ with \ respect \ to \ \psi\). Using the stationarity condition \(\partial H(E, U) / \partial U = 0\) and revealed to be [3]

\[
U^*(E) = -R^{-1} G(X)^T V_E^* (E) / 2.
\]

By substituting (18) into the Hamiltonian (17), while noting \(H(E, U^*) = 0\), the HJB equation reveals to be

\[
Q(E) + V_E^T (F(E) G(X) R^{-1} G(X)^T V_E^* (E) = 0,
\]

with \(V^*(0) = 0\). For linear systems, Equation (19) yields the standard algebraic Riccati equation [3]. Before proceeding, the following technical lemma is required.

Lemma 2 [5]

Given nonlinear system (16) with associated cost function (15) and optimal control (18), let \(J(E)\) be a continuously differentiable, radially unbounded Lyapunov candidate such that \(J(E) = J_E^T (E) \dot{E} = J_E^T (E) (F(E) + G(X)U) < 0\), where \(J_E^T (E)\) is the radially unbounded partial derivative of \(J^T (E)\). Moreover, let \(\hat{Q}(E)\) be a positive definite matrix satisfying \(\| \hat{Q}(E) \| = 0\) only if \(\| E \| = 0\) and \(\hat{Q}_{\min} \leq \| \hat{Q}(E) \| \leq \hat{Q}_{\max}\) for \(\chi_{\min} \leq \| E \| \leq \chi_{\max}\) for positive constants \(\hat{Q}_{\min}, \hat{Q}_{\max}, \chi_{\min}\), and \(\chi_{\max}\). In addition, let \(\hat{Q}(E)\) satisfy \(\lim_{E \to \infty} \hat{Q}(E) = \infty\) as well as

\[
V_E^* \hat{Q}(E) J_E = r(E, u^*) = Q(E) + U^*R U^*.
\]

Then the following relation holds:

\[
J_E^T (F(E) + G(E)U^*) = -J_E^T \hat{Q}(E) J_E.
\]
In [9], the closed-loop dynamics $F(E) + G(E)U^*$ is required to satisfy a Lipschitz condition such that $\|F(E) + G(X)U^*\| \leq K$ for a constant $K$. In contrast, in this work, the optimal closed-loop dynamics are assumed to be bounded above by a function of the system states such that

$$\|F(E) + G(X)U^*\| \leq \delta(E).$$  \hfill (22)

The generalized bound $\delta(E)$ is taken as $\delta(E) = \sqrt[4]{K^*||J_E||}$ in this work, where $\|J_E\|$ can be selected to satisfy general bounds and $K^*$ is a constant assumed to be existing. For example, if $\delta(E) = K_1 \|E\|$ for a constant $K_1$, then it can be shown that selecting $J(E) = (E^T E)^{(3/2)}/5$ with $J_E(E) = (E^T E)^{(3/2)}E^T$ satisfies the bound. The assumption of a time-varying upper bound in (22) is a less stringent assumption than the constant upper bound required in [9].

The next section develops an approach to optimally stabilize the affine system, which is required for optimal tracking of original strict-feedback systems. We rewrite cost function (15) using an OLA representation as

$$V(E) = \Theta^T \varphi(E) + \varepsilon(E),$$  \hfill (23)

where $\Theta \in \mathbb{R}^L$ is the constant target OLA vector, $\varphi(E) : \mathbb{R}^n \rightarrow \mathbb{R}^L$ is a linearly independent basis vector, which satisfies $\varphi(0) = 0$, and $\varepsilon(E)$ is the OLA reconstruction error. The target OLA vector and reconstruction errors are assumed to be bounded above according to $\|\Theta\| \leq \Theta_M$ and $\|\varepsilon(E)\| \leq \varepsilon_M$, respectively [20]. In addition, it will be assumed that the gradient of the OLA reconstruction error with respect to $E$ is bounded above according to $\|\partial\varepsilon(E)/\partial E\| = \|\nabla_E \varepsilon(E)\| \leq \varepsilon'_M$.

The gradient of OLA cost function (23) is written as

$$\partial V(E)/\partial E = V_E(E) = \nabla_E^T \varphi(E) \Theta + \nabla_E \varepsilon(E).$$  \hfill (24)

Now, using (24), optimal control (14) and HJB (19) are rewritten as

$$U^*(E) = -\frac{1}{2}R^{-1}G(E)^T \nabla_E \varphi(E) \Theta - \frac{1}{2}R^{-1}G(X)^T \nabla_E \varepsilon(E)$$  \hfill (25)

and

$$H = Q(E) + \Theta^T \nabla_E \varphi(E) F(E) - \frac{1}{4} \Theta^T \nabla_E \varphi(E) D \nabla_E \varphi(E) \Theta + \varepsilon_{HJB} = 0,$$  \hfill (26)

where $D = G(E)R^{-1}G(E)^T > 0$ is bounded such that $D_{\min} \leq \|D\| \leq D_{\max}$ for known constants $D_{\min}$ and $D_{\max}$, and

$$\varepsilon_{HJB} = \nabla_E \varepsilon^T \left( F(E) - \frac{1}{2}G(X)R^{-1}G(X)^T (\nabla_E \varphi(E) \Theta + \nabla_E \varepsilon) \right)$$

$$+ \frac{1}{4} \nabla_E \varepsilon^T G(X)R^{-1}G(X)^T \nabla_E \varepsilon = \nabla E \varepsilon^T (F(E) + G(X)U^*) + \frac{1}{4} \nabla E \varepsilon^T D \nabla E \varepsilon$$  \hfill (27)

is the residual error due to the OLA reconstruction error. Asserting the bounds for optimal closed-loop dynamics (22) along with the boundedness of $G(X)$ and $\nabla_E \varepsilon$, the residual error $\varepsilon_{HJB}$ is bounded above on a compact set according to $|\varepsilon_{HJB}| \leq \varepsilon'_M \delta(E) + \varepsilon_M^2 D_{\max}$. In addition, it has been shown [9] that by increasing the dimension of the basis vector $\varphi(E)$ in the case of a single-layer NN, the OLA reconstruction error decreases.

The OLA estimate of (15) is now written as

$$\hat{V}(E) = \hat{\Theta}^T \varphi(E),$$  \hfill (28)

where $\hat{\Theta}$ is the OLA estimate of the target parameter vector $\Theta$. Similarly, the estimate of optimal control (14) is written in terms of $\hat{\Theta}$ as

$$\hat{U} = -\frac{1}{2}R^{-1}G(X)^T \nabla_E \varphi(E) \hat{\Theta}.$$  \hfill (29)
It is shown in [5] that an initial stabilizing control is not required to implement the proposed SOLA-based scheme in contrast to that in [9, 11], which requires initial control policies to be stabilized. In fact, the proposed OLA parameter tuning law described next ensures that the system states remain bounded and that (29) will become admissible.

Now, using (28), the approximate Hamiltonian can be written as

$$
\hat{H}(E, \hat{\Theta}) = Q(E) + \hat{\Theta}^T \nabla_E \varphi(E) F(E) - \frac{1}{4} \hat{\Theta}^T \nabla_E \varphi(E) D \nabla_T^2 \varphi(E) \hat{\Theta}. 
$$  \( (30) \)

Observing the definition of the OLA approximation of cost function (28) and Hamiltonian function (30), it is evident that both become zero when \( \|E\| = 0 \). Thus, once the system states have converged to zero, the cost function approximation can no longer be updated. This can be viewed as a persistency of excitation (PE) requirement for the inputs to the cost function OLA \([9, 11]\). That is, the system states must be persistently exciting long enough for the OLA to learn the optimal cost function.

Recalling the HJB equation shown in (19), the OLA estimate \( \hat{\Theta} \) should be tuned to minimize \( \hat{H}(E, \hat{\Theta}) \). However, tuning to minimize \( \hat{H}(E, \hat{\Theta}) \) alone does not ensure the stability of nonlinear system (16) during the OLA learning process. Therefore, the proposed OLA tuning algorithm is designed to minimize (30) while considering the stability of (16) and written as

$$
\dot{\hat{\Theta}} = -\alpha_1 \frac{\hat{\sigma}}{(\hat{\sigma}^T \hat{\sigma} + 1)^2} \left( Q(E) + \hat{\Theta}^T \nabla_E \varphi(E) F(E) - \frac{1}{4} \hat{\Theta}^T \nabla_E \varphi(E) D \nabla_T^2 \varphi(E) \hat{\Theta} \right) + \Sigma(E, \hat{\Theta}) \frac{\alpha_2}{2} \nabla_E \varphi(E) g(E) R^{-1} G(X)^T J_E(E),
$$  \( (31) \)

where \( \hat{\sigma} = \nabla_E \varphi(E) - \nabla_E \varphi(E) D \nabla_T^2 \varphi(E) \hat{\Theta} / 2 \), \( \alpha_1 > 0 \) and \( \alpha_2 > 0 \) are design constants, \( J_E(E) \) is described in Lemma 1, and the operator \( \Sigma(E, \hat{\Theta}) \) is given by

$$
\Sigma(E, \hat{\Theta}) = \begin{cases} 
0 & \text{if } J_E^T(E) \dot{E} = J_E^T(E)(F(E) - G(X) R^{-1} G(X)^T \nabla_T^2 \varphi(E) \hat{\Theta} / 2) < 0 \\
1 & \text{otherwise}
\end{cases}. \ (32)
$$

The first term in (32) is the portion of the tuning law, which seeks to minimize (30) and was derived using a normalized gradient descent scheme with the auxiliary HJB error defined as

$$
E_{HJB} = \hat{H}(E, \hat{\Theta})^2 / 2. \ (33)
$$

Meanwhile, the second term in the OLA tuning law (31) is included to ensure the system states remain bounded while the SOLA scheme learns the optimal cost function. The form of the operator shown in (32) was selected based on the Lyapunov’s sufficient condition for stability (i.e., if \( J(E) > 0 \) and \( J(E) = J_E^T(E) \dot{E} < 0 \), then the states \( E \) are stable). From the definition of the operator in (32), the second term in (31) is removed when nonlinear system (16) exhibits stable behavior, and learning the HJB cost function becomes the primary objective of OLA update (31). In contrast, when system (16) exhibits signs of instability (i.e., \( J_E^T(E) \dot{E} \geq 0 \)), the second term of (31) is activated and tunes the OLA parameter estimates until nonlinear system (16) exhibits stable behavior.

Now, we form the dynamics of the OLA parameter estimation error \( \hat{\Theta} = \Theta - \hat{\Theta} \). Observing \( Q(E) = -\hat{\Theta}^T \nabla_E \varphi(E) F(E) + \hat{\Theta}^T \nabla_E \varphi(E) D \nabla_T^2 \varphi(E) \hat{\Theta} / 4 - \epsilon_{HJB} \) from (25), the approximate HJB (30) can be rewritten in terms of \( \hat{\Theta} \) as

$$
\hat{H}(E, \hat{\Theta}) = -\hat{\Theta}^T \nabla_E \varphi(E) F(E) + \frac{1}{2} \hat{\Theta}^T \nabla_E \varphi(E) D \nabla_T^2 \varphi(E) \hat{\Theta} - \frac{1}{4} \hat{\Theta}^T \nabla_T^2 \varphi(E) D \nabla_T^2 \varphi(E) \hat{\Theta} - \epsilon_{HJB}. \ (34)
$$
Next, observing \( \dot{\Theta} = -\hat{\Theta} \) and \( \hat{\Theta} = \nabla_{E}\varphi(E)(\dot{E}^* + D\nabla_{E}\varepsilon/2) + \nabla_{E}\psi(E)D\nabla_{E}^{T}\varphi(E)\hat{\Theta}/2 \) where \( \dot{E}^* = F(E) + G(X)U^* \), the error dynamics of (20) are written as

\[
\dot{\Theta} = -\frac{\alpha_1}{\rho^2} \left( \nabla_{E}\varphi(E) \left( \dot{E}^* + \frac{D\nabla_{E}\varepsilon}{2} \right) + \nabla_{E}\psi(E)D\nabla_{E}^{T}\varphi(E)\hat{\Theta} \right) \\
\times \left( \hat{\Theta}^{T}\nabla_{E}\varphi(E) \left( \dot{E}^* + \frac{D\nabla_{E}\varepsilon}{2} \right) + \frac{1}{4} \hat{\Theta}^{T}\nabla_{E}\psi(E)D\nabla_{E}^{T}\varphi(E)\hat{\Theta} + \varepsilon_{HJB} \right) \\
- \Sigma(E, \hat{U}) \frac{\alpha_2}{2} \nabla_{E}\psi(E)G(X)R^{-1}G(X)^{T}J_{E}(E),
\]

where \( \rho = (\hat{\Theta}^{T}\hat{\Theta} + 1) \). Next, the stability of the SOLA-based adaptive scheme for optimal control is examined along with the stability of nonlinear system (16).

Theorem 1 (SOLA-based optimal control scheme [5])
Given the nonlinear system in affine form (13) or (16) with target HJB (19), let the tuning law for the SOLA be given by (31). Then there exist computable positive constants \( b_{JE} \) and \( b_{\hat{\Theta}} \) such that the OLA approximation error \( \hat{\Theta} \) and \( \|J_{E}(E)\| \) are uniformly ultimately bounded (UUB) [20] for all \( t \geq t_{0} + T \) with ultimate bounds given \( \|J_{E}(E)\| \leq b_{JE} \) and \( \|\hat{\Theta}\| \leq b_{\hat{\Theta}} \). Further, in the presence of OLA reconstruction errors, one can show that \( \|V^* - \hat{V}\| \leq \varepsilon_{r1} \) and \( \|U^* - \hat{U}\| \leq \varepsilon_{r2} \) for some small positive constants \( \varepsilon_{r1} \) and \( \varepsilon_{r2} \), respectively, where \( b_{\Theta} \equiv \sqrt{\eta(\varepsilon)/\beta_1} \) and \( b_{JE} \equiv \alpha_1\eta(\varepsilon)/\alpha_2\hat{x}_{\min} - \alpha_1\beta_2K^* \) with \( \beta_2 \) chosen such that \( \alpha_2\hat{x}_{\min} - \alpha_1\beta_2K^* > 0 \).

The stability of the SOLA-based optimal control scheme can be examined when there are no OLA reconstruction errors as would be the case when standard adaptive control techniques [2] are utilized. In other words, when an NN is replaced with a standard linear in the unknown parameter (LIP) adaptive control, the parameter estimation errors and the states are globally asymptotically stable.

Next, the stability of the optimal adaptive control scheme for strict-feedback system is introduced.

Theorem 2 (Optimal adaptive control scheme for strict-feedback systems)
Given the nonlinear system in strict-feedback form as (1)–(3), assume that the virtual and real control input vector \( U = \begin{bmatrix} x_{2d} \cdots x_{Nd} u \end{bmatrix} \) is designed such that \( U = U^{a} + U^{*} \) where \( U^{a} = \begin{bmatrix} x_{2d}^{a} \cdots x_{Nd}^{a}u^{a} \end{bmatrix} \) is the feedforward control input designed in (5), (7), and (9) and \( U^{*} = \begin{bmatrix} x_{2d}^{*} \cdots x_{Nd}^{*}u^{*} \end{bmatrix} \) represents the feedback control input given by (29). Let the tuning law for the SOLA be given by (31). Then the closed-loop system is UUB.

Proof
Use Lemma 1 and Theorem 1.

4. OBSERVER-BASED OUTPUT FEEDBACK CONTROL DESIGN

In the previous section, the optimal adaptive control of a class of nonlinear continuous-time systems is introduced when the states are available for measurement. Practically, the states are not measurable in a vast class of nonlinear systems. In this section, we consider the control problem of strict-feedback control of system (1)–(3) where \( f_{i}(.) \) and \( g_{i}(.) \) are known, whereas the state vector is not measured and only the output \( y = h(x) \) is given. The MIMO feedback control of strict-feedback systems will have to mitigate several challenges and will be relegated for a future publication. For example, selecting different outputs can change the relative degree of the system, which in turn can complicate the process of the controller design. Therefore, we consider system (1)–(3) in a SISO case. This problem is still difficult as no known output feedback-based optimal control scheme is available in the forward-in-time manner, although recently for linear systems some results are achieved [21] where policy and value iteration method are used to estimate the optimal control solution. While the work in [21] uses adaptive dynamic programming and only input/output data,
H. ZARGARZADEH, T. DIERKS AND S. JAGANNATHAN

This paper follows the traditional method, which introduces an observer to estimate the unmeasured states.

Now, assume that (1)–(3) is represented in a SISO representation, that is, $x_i \in \mathbb{R}$ and $u \in \mathbb{R}$. It is shown [15] that, in this case, there exists a mapping $\zeta = (\xi_1, \ldots, \xi_N) = \Phi(x_1, \ldots, x_N)$ that transforms system (1)–(4) into a new state space representation as

$$
\begin{align*}
\dot{\xi}_1 &= \xi_2 + \omega_1(y) \\
\dot{\xi}_2 &= \xi_3 + \omega_2(y) \\
&\vdots \\
\dot{\xi}_{N-1} &= \xi_N + \omega_{N-1}(y) \\
\dot{\xi}_N &= \omega_N(y) + b\beta(y)u \\
y &= \xi_1 = h(x),
\end{align*}
$$

(36)

where $\omega_i(y) \in \mathbb{R}$ are known functions of the output. The transformation $\Phi$ exists only when the relative degree of (1)–(3) (in SISO case) is equal to $N$. To overcome the need for state measurements, define the observer dynamics as

$$
\begin{align*}
\dot{\hat{\zeta}} &= A\hat{\zeta} + k(y - \hat{y}) + \omega(y) + b\beta(y)u \\
\hat{y} &= c^T\hat{\zeta},
\end{align*}
$$

(37)

where

$$
A = \begin{bmatrix} 0 & I \\ 0 & \ddots & \ddots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \quad c = \begin{bmatrix} 1 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \quad \omega = \begin{bmatrix} \omega_1(y) \\ \vdots \\ \omega_n(y) \end{bmatrix}.
$$

Therefore, with $A_o = A - kc^T$ being Hurwitz, we conclude that the closed-loop observer dynamics can decay exponentially to the origin. Therefore, by defining $\hat{\zeta} = \zeta - \hat{\zeta}$, the observer error dynamics takes the following form.

$$
\dot{\hat{\zeta}} = A_o\hat{\zeta},
$$

(38)

Now, we apply the same backstepping approach from previous section with the assumption that $\xi_i$ for $i = 2, \ldots, N$ are not measured but estimated by using observer (37). By following steps 1 through $N$, we obtain

$$
\begin{bmatrix}
\dot{\hat{\xi}}_1 \\
\vdots \\
\dot{\hat{\xi}}_N
\end{bmatrix} =
\begin{bmatrix}
\omega_1(\hat{\xi}_1) \\
\vdots \\
\omega_N(\hat{\xi}_1)
\end{bmatrix} +
\begin{bmatrix}
1 & \cdots & 0 \\
0 & \ddots & \vdots \\
\vdots & \ddots & 1 \\
0 & \cdots & 0
\end{bmatrix}
\begin{bmatrix}
1 \\
\vdots \\
1 \\
\beta(y)
\end{bmatrix}
$$

$$
U^* + A\hat{\xi} = \omega(\hat{\xi}_1) + B(y)U^* + A\hat{\xi},
$$

(39)

with $\hat{\xi}_i = \hat{\xi}_i - \xi_{id}$ that implies $e_1 = \hat{\xi}_1 = y - y_d$ as $\hat{\xi}_1 = \xi_1 = y$. Moreover, the desired trajectory (feedforward controller) is designed from the expressions

$$
-\hat{y}_{1d} + \varphi_1(y) + \hat{\xi}_{2d} = \omega_1(\hat{\xi}_1) \\
\vdots \\
-\hat{\xi}_{id} + \varphi_i(y) + \hat{\xi}_{(i+1)d} = \omega_i(\hat{\xi}_1) - \hat{\xi}_{i-1} \\
\vdots \\
-\hat{\xi}_{N_d} + \omega_N(y) + b\beta(y)u^a = \omega_N(\hat{\xi}_N) - \hat{\xi}_{N-1},
$$
which gives

\[
\hat{\sigma}_2 = \omega_1(\hat{e}_1) + y_1d - \varphi_1(y) \\
\vdots \\
\hat{\sigma}_i = \omega_i(\hat{e}_i) - \hat{e}_{i-1} + \hat{\xi}_id - \varphi_i(y) \\
\vdots \\
u^a = b^{-1}b^{-1}(y)\{\omega_N(\hat{e}_N) - \hat{e}_{N-1} + \hat{\xi}_{Nd} - \omega_N(y)\}. \tag{40}
\]

Equation (40) is identical to the equations derived for the steps 1 to N in Section 2 except we used the system output and the estimated states \(\hat{e}\) (by the observer) instead of the real value of \(\hat{e}\). This is the reason that an estimation error term \(A\hat{e}\) appears in the error dynamics given by (39). Moreover, in (40), \(U^* = [\hat{\xi}_{i2d} \cdots \hat{\xi}_{Nd} \quad u^*]\). Theorem 3 will show that the state estimation error \(\hat{e} = e - \hat{e}\) is guaranteed to be bounded, which is necessary for the overall stability of the closed-loop system. Using \(r_1(\hat{E},U^*) = Q_1(\hat{E}) + U^*_1R_1U^*_1\), where \(\hat{E} = [\hat{e}_1, \ldots, \hat{e}_N]^T\), the target HJB equation takes the following form:

\[
Q_1(\hat{E}) + V^T_1E(\hat{E})\omega(e) - \frac{1}{4}V^T_1E(\hat{E})B(y)R_1^{-1}B(y)^T V^*_1E(\hat{E}) = 0, \tag{41}
\]

with \(V_1 = \int_0^\infty r_1(\hat{E}(\tau),U^*(\tau))d\tau\) as the cost function, \(R_1 \in \mathbb{R}^{N \times N}\) with \(R_1 > 0\) as \(m = 1\), and \(Q_1(\hat{E})\) is a positive semidefinite function of \(\hat{E}\). Therefore, the optimal controller for this case can be represented as

\[
U^*_1(\hat{E}) = -\frac{1}{2}R_1^{-1}B(y)^TV^*_1E(\hat{E}). \tag{42}
\]

Now, consider an OLA representation for the cost function as

\[
V_1(\hat{E}) = \Theta^T_1\varphi_1(\hat{E}) + \varepsilon_1(\hat{E}), \tag{43}
\]

with \(\varepsilon_1(\hat{E})\) as the reconstruction error, and the update law for the OLA weights is given as

\[
\dot{\Theta} = -\gamma_1 \frac{\Theta_1}{(\Theta^T_1\Theta_1 + 1)^{\frac{3}{2}}} \left( Q_1(\hat{E}) + \Theta^T_1 \nabla_{\hat{E}} \varphi_1(\hat{E})\omega(y) - \frac{1}{4}\Theta^T_1 \nabla_{\hat{E}} \varphi_1(\hat{E})D_1 \nabla_{\hat{E}} \varphi_1(\hat{E})D_1^{-1} \Theta_1 \right) \\
+ \Sigma_1(\hat{E},\hat{U}_1)\frac{\gamma_2}{2} \nabla_{\hat{E}} \varphi_1(\hat{E})B(y)R_1^{-1}B(y)^T J_{1E}(\hat{E})
\]

with

\[
\Sigma_1(\hat{E},\hat{U}_1) = \begin{cases} 0 & \text{if } J_{1E}(\hat{E})\dot{E} = J_{1E}(\hat{E})(\omega(\hat{e}_1) - B(y)R_1^{-1}B(y)^T \nabla_{\hat{E}} \varphi_1(\hat{E})\Theta_1)/2 < 0 \\
1 & \text{otherwise} \end{cases} \tag{45}
\]

Here, \(J_{1E}(\hat{E})\) is a positive definite radially unbounded function of \(\hat{E}\), \(\gamma_1, \gamma_2 > 0\) are real design parameters, \(\Theta_1\) is the target parameter, and \(\varphi_1(\hat{E})\) is the basis function for the estimation of \(V_1(\hat{E})\). Moreover,

\[
\dot{\Theta}_1 = \nabla_{\hat{E}} \varphi_1(\hat{e}_1) - \nabla_{\hat{E}} \varphi_1(\hat{E})D_1 \nabla_{\hat{E}} \varphi_1(\hat{E})\Theta_1)/2, \tag{46}
\]

with \(D_1 = B(y)R_1^{-1}B(y)^T > 0\), where \(D_{1\text{min}} < \|D_1\| < D_{1\text{max}}\). It is finally assumed that \(\|\omega(\hat{e}_1) + B(y)U^*_1\| \leq \delta_1(\hat{E}) = \sqrt{K_{10}}\|J_{1E}\|\) with \(K_{10} > 0\).

We can now introduce Theorem 3 under the case where the states are not measured while the output is only available.
Theorem 3 (Output feedback SOLA-based optimal control scheme) 
Assume that the states of the nonlinear system (1) through (3) are not measurable while the output is only available with $m = 1$. Also assume that $x_i$ are transformed using $\mathfrak{N} \{x_1, \ldots, x_N\}$ to $\zeta$, which forms system dynamics (36). Given nonlinear system (36), observer (37), and target HJB (41), let the tuning law for the SOLA be given by (44) with the cost function estimation $\hat{V}_1(\hat{E}) = \Theta_1^T \varphi_1(\hat{E})$. Then there exist computable positive constants $b_{1JE}$, $b_{1\Theta}$, and $b_{1\xi}$ such that the OLA approximation error $\tilde{\Theta}_1 = \Theta_1 - \hat{\Theta}_1$, $||J_{1JE}(\hat{E})||$, and $\tilde{\zeta}$ are UUB for all $t \geq t_0 + T$ with ultimate bounds given $||J_{1JE}(\hat{E})|| \leq b_{1JE}$, $||\tilde{\Theta}_1|| \leq b_{1\Theta}$, and $||\tilde{\zeta}|| < b_{1\xi}$. Further, in the presence of OLA reconstruction errors, one can show that $||V_1^* - \hat{V}_1|| \leq \tilde{e}_{r1}$ and $||U_1^* - \hat{U}_1|| \leq \tilde{e}_{r2}$ for small positive constants $\tilde{e}_{r1}$ and $\tilde{e}_{r2}$, respectively, where $b_{1\Theta} = \gamma_1 \sqrt{\eta_1(\varepsilon)}/\tau_1$, $b_{1JE} = \gamma_1 \eta_1(\varepsilon)/(\gamma_2 \hat{E}_{\min} - \gamma_1 \tau_2 K_1^*)$, and $b_{1\xi} = \rho_1^{-1} \sqrt{\gamma_1 \eta_1(\varepsilon)}/\lambda T_{min}$ provided $\gamma_2/\gamma_1 > \tau_2 K_1^*/\hat{E}_{\min}$, where $\rho_1 = (\hat{\sigma}_1^T \hat{\sigma}_1 + 1)$. In addition, the following relationship $-T = A_o^T P + P A_o$ is utilized with $P$ and $T$ being an arbitrary positive definite matrix, where $\lambda T_{\min}$ being the minimum eigenvalue of $T$.

Proof
See the Appendix.

Remark 1 (PE condition requirement for the OLA convergence)
As mentioned in the description of (30) and (31), the PE condition is required in order to make the OLA weights converge and remain bounded. The PE condition is used in the stability proof of the adaptive systems to guarantee the estimated parameter convergence to their target values [2]. In the traditional adaptive control systems, the estimated parameter convergence to their target value is not a mandatory condition for the stability as the boundedness of the parameters estimation error is normally shown. With the PE condition provided, the stability proof shows that the parameters will converge asymptotically to their target values over time.

Nonetheless, this paper requires that the OLA weights converge to a small enough bound in a reasonable time. In other words, although the tracking error is proven to be stable, the optimality of the applied control input requires that $\hat{\Theta}$ converges while the PE condition is utilized to excite the dynamics to learn them. To this end, the closed-loop system and particularly $\phi(E)$ should be persistently exciting. Unfortunately, there is no classical method for adaptive systems to define the level of PE or guarantee the convergence of the estimated parameters in a finite time so that the PE condition can be turned off. Previous works in online optimal control [5, 6, 13] also require the PE condition to ensure convergence to the optimal controllers. However, the recent paper [22] shows that noise can be utilized to meet the PE condition. While in this paper we rely on the results by [22], the PE condition has been verified in some particular cases. For example, in [23], it has been shown that with RBF networks and a recurrent desired trajectory, the PE condition will be satisfied.

It is finally noted that the proposed optimal control scheme can be implemented online without performing offline calculations, and stability is guaranteed.

5. NUMERICAL RESULTS

In this section, we start with applying the results to linear systems. While we have shown that the proposed online controller will be able to solve the HJB equation online, it would be interesting to see that it can also solve the Riccati equation online for linear systems. Then a MIMO system is considered, and a state feedback optimal approach is designed and verified in simulation. Subsequently, the output feedback-based optimal scheme is evaluated in another example.
5.1. Online optimal control of linear systems

It is obvious that in the case of linear systems, the infinite horizon optimal control requires the Riccati equation to be solved instead of the HJB equation. Consider the following _unstable linear_ system given by

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} =
\begin{bmatrix}
0 & 1 \\
2 & -1
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2
\end{bmatrix} +
\begin{bmatrix}
0 \\
1
\end{bmatrix} u,
\]

\[y = x_1,
\]

which is in the strict-feedback form. It is clear that by the proposed backstepping approach, the tracking error dynamics takes the following form:

\[
\begin{bmatrix}
\dot{e}_1 \\
\dot{e}_2
\end{bmatrix} =
\begin{bmatrix}
0 & 1 \\
2 & -1
\end{bmatrix}
\begin{bmatrix}
e_1 \\
e_2
\end{bmatrix} +
\begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} U^*.
\]

By solving the Riccati equation, we can easily find that the cost function is given by \(V(E) = E^T P E = 3.924e_1^2 + 1.461e_2^2 + 1.854e_1 e_2\), where \(P\) is the solution to the Riccati equation when \(Q = I\) and \(R = 1\). The design parameters are chosen as \(\alpha_1 = 200\) and \(\alpha_2 = 0.01\). Therefore, the SOLA should converge to this solution if we choose the basis function as \(\varphi(E) = [e_1^2 e_2^2 e_1 e_2]\), as the cost function is known for the linear systems. A probe noise is also added to system dynamics to provide the PE condition.

Figure 1 depicts the evolution of the OLA weights during the online learning that shows that the estimated cost function accurately converges to the desired one in about 8 s, although the PE condition is applied for an additional 15 min in order to show that the estimated Hamiltonian will stay close to origin. Starting from zero, the weights of the online OLA are tuned to learn the optimal cost function and obviously converge to their exact values in \(V(E)\). The system tracking performance is shown in Figure 2, where \(x_{1d} = \sin(0.5t)\) is chosen as the output desired trajectory. Figure 3 shows the Hamiltonian \(\tilde{H}(E, \tilde{\Theta})\) with respect to time. The figure shows that the Hamiltonian convergence time is shorter than that of the cost weights and the tracking error.

The overshoot in Figure 2 is mainly because of the optimal controller is not initiated by an admissible controller, while plant (48) is unstable. While the update law helps the initial controller to converge to the optimal one (that is obviously stabilizing), the plant states trajectory may have overshoots/undershoots temporarily. Here, it should be mentioned that the purpose of the represented results is to show that the proposed method will converge even in the worst case. Otherwise, the user can observe much better results when cost function weights are initially chosen to provide an admissible controller. The same argument is also valid for the next two sections when the method is applied to the nonlinear systems in the state feedback and output feedback cases.

![Figure 1](image1.png)

*Figure 1. The evolution of the cost function weights \(\hat{\Theta}(t)\) with time for the linear system.*
5.2. MIMO online optimal control

Consider the following nonlinear system in the form of (1)–(2) respectively as

\[
\begin{align*}
\dot{X}_1 &= \begin{bmatrix} \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} -x_{11} \left( \frac{\pi}{2} + \tan^{-1} \left( 5x_{11} \right) \right) - \frac{5x_{11}^2}{2(1 + 25x_{11})} + 4x_{12} \\ 1 + 0.5 \cos(x_{21} + x_{11}) \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} x_{21} \\ x_{22} \end{bmatrix} \\
\dot{X}_2 &= \begin{bmatrix} \dot{x}_{21} \\ \dot{x}_{22} \end{bmatrix} = \begin{bmatrix} -4x_{21} + x_{12}^2 - 2x_{11} \\ -3x_{22} + 2x_{12}^2 - x_{11} \end{bmatrix} + \begin{bmatrix} 1 + x_{21}^2 \\ 0 \end{bmatrix} u.
\end{align*}
\]

Using HJB cost function (3) with \( Q(x) = E^T E \) and \( R = 1 \), the basis vector for the SOLA-based scheme implementation was selected as \( \varphi(E) = \begin{bmatrix} x_{e11} & x_{e12} & x_{e21} & x_{e22} & x_{e11}^2 & x_{e12}^2 & x_{e11} \tan^{-1} (5x_{e11}) & x_{e21}^2 \end{bmatrix}^T \) while the tuning parameters are selected as \( \alpha_1 = 200 \) and \( \alpha_2 = 0.01 \). Moreover, \( x_{e11} = x_{11} - x_{11d} \), \( x_{e12} = x_{12} - x_{12d} \), \( x_{e21} = x_{21} - x_{21d} \), and \( x_{e22} = x_{22} - x_{22d} \). The initial conditions of the system states are taken as \( \begin{bmatrix} x_{11} & x_{12} & x_{21} & x_{22} \end{bmatrix}^T = [2, -2, 2, 2]^T \) while all NN weights are initialized to zero. That is, no initial stabilizing control was utilized for implementation of this online design for the nonlinear system. Moreover, it is desired that the output track \( X_{1d} = \begin{bmatrix} \sin(t/50) & \sin(t/40) \end{bmatrix}^T \) as the desired trajectory.

Figure 4 depicts the evolution of the OLA weights during the online learning. Starting from zero, the weights of the online OLA are tuned to learn the optimal cost function. The system output \( X_1 = [x_{11}, x_{12}]^T \) is shown in Figure 5, and noise is added to each state to ensure the PE condition. Figure 6 depicts the stability of the internal system states \( X_2 = [x_{21}, x_{22}]^T \). Figure 7 shows the control input to the system \( \hat{U}^* \).

Finally, in the case of Figures 4–7, Figure 8 demonstrates the estimated Hamiltonian in (30). To demonstrate the importance of the secondary stabilizing term in the tuning law given by (31), the online OLA design is attempted with \( \Sigma(E, \hat{U}) = 0 \). That is, the learning algorithm only seeks to
Figure 4. The evolution of NN weights with time.

Figure 5. The convergence of system outputs to the desired trajectory.

Figure 6. The convergence of the internal system states to their desired trajectory.

Figure 7. The actual control input to the system $\hat{U}^*$. 
minimize auxiliary HJB residual (33) and does not consider system stability. Figure 9 shows the result of not considering the nonlinear system stability while learning the optimal HJB function. From this figure, it is clear that the system state quickly escapes to infinity, and the SOLA-based controller fails to learn the HJB function. Thus, the importance of the secondary term in (31), which ensures the stability of the system, is revealed.

5.3. Observer-based online optimal control output feedback control

Consider the following nonlinear system in the form of (1)–(2) respectively as

\[
\dot{\zeta}_1 = \zeta_2 - \zeta_1 \left( \frac{\pi}{2} + \tan^{-1}(5\zeta_1) \right) - \frac{5\zeta_1^2}{2(1 + 25\zeta_1^2)} + 4\zeta_1
\]

\[
\dot{\zeta}_2 = 2\zeta_1^2 - \zeta_1 + \left\{ 1 + \frac{1}{2} \cos(\zeta_1) \right\} u.
\]

\[
y = \zeta_1,
\]

which is in the form of system (36). Here, we repeat the experiment of part b with the assumption that \( z \) is not measurable. Using HJB cost function (3) with \( Q_1(\bar{E}) = \bar{E}^T \bar{E} \) and \( R_1 = 1 \), the basis vector for the SOLA-based scheme implementation was selected as \( \varphi_1(\bar{E}) = [\bar{e}_1, \bar{e}_1^2, \bar{e}_1^3, \bar{e}_1^2 \tan^{-1}(5\bar{e}_1)\bar{e}_2, \bar{e}_2^2, \bar{e}_2^2 \tan^{-1}(5\bar{e}_2)\bar{e}_2^2]^T \) while the tuning parameters were selected as \( \gamma_1 = 200, \gamma_2 = 0.01, \) and \( \lambda_{\bar{z}} = 0.04, \) and \( A_o = 0.1 \). The initial conditions of the system states were taken as \( [\zeta_1 \zeta_2]^T = [2 \ -2]^T \) while all NN weights were initialized to zero. That is, no initial stabilizing control was utilized for implementation of this online design for the nonlinear system. Moreover, it is desired that the output track \( x_d = \sin(t/50) \) as the desired trajectory.
The simulation results are given in Figures 10 and 11. In these figures, the convergence of $\hat{\xi}_1$ and $\hat{\xi}_2$ to $\xi_1$ and $\xi_2$ are depicted. We can check from (A.5) and (A.7) that by properly choosing $\lambda T_{\text{min}}$, the upper bound of $||\hat{\xi}||$ can be arbitrarily adjusted as small as desired. Therefore, after a transient response time of about 100 s, the observed state $\hat{\xi}$ is equal to $\xi$, and the online optimal controller can rely on the observed value instead of the real value. The presence of PE noise injected is very vital in the process of the cost function learning. In fact, without a significant level of the noise injected to the process, there is no guarantee that the Hamiltonian will converge to zero.

6. CONCLUSIONS

This work proposed an optimal scheme for stabilizing nonlinear MIMO strict-feedback systems by using a single OLA to solve the HJB equation forward-in-time. In the presence of known dynamics, the regulation problem was undertaken. Then by using a backstepping approach, the control input to the nonlinear system was derived by using state measurements. Next, this scheme was developed to the optimal output feedback of SISO nonlinear systems. A nonlinear observer was designed in order to estimate the unknown states in the output feedback case. The UUB stability of the overall system is guaranteed in the presence of OLA approximation errors. Simulation results were also provided to verify the theoretical conjectures. Future work should extend the results of the output feedback case of SISO to MIMO systems.

APPENDIX

Proof of Theorem 3

It should be mentioned that although this theorem uses Lyapunov stability and similar to the Theorem 1, here, the observer dynamics should be proven to be stable while the tracking error converges. This fact distinguishes the proof from that of [5] and makes the proof more difficult.
Consider the following positive definite Lyapunov candidate:

$$J_{1HJB} = \gamma_2 J_1(\hat{E}) + \frac{1}{2} \hat{\Theta}_1^T \hat{\Theta}_1 + \frac{1}{2} \hat{\xi}^T P \hat{\xi}$$  \hspace{1cm} (A.1)$$

whose first derivative with respect to time is given by

$$\dot{J}_{1HJB} = \gamma_2 J_T^T(\hat{E}) \hat{E} + \hat{\Theta}_1^T \hat{\Theta}_1 + \frac{1}{2} \hat{\xi}^T P \hat{\xi} + \frac{1}{2} \hat{\xi}^T P \hat{\xi} = \gamma_2 \hat{E} \hat{E} + \hat{\Theta}_1^T \hat{\Theta}_1 + \hat{\xi}^T (A^T P + PA) \hat{\xi}. \hspace{1cm} (A.2)$$

where $J_1(\hat{E})$ and $J_{1E}(\hat{E})$ are given in Lemma 1. With the same steps as of Theorem 1, we obtain

$$\dot{J}_{1HJB} \leq \gamma_2 J_T^T(\hat{E}) \left( \omega(e_1) - \frac{1}{2} B(y) R_1^{-1} B(y)^T \nabla_{\hat{E}} \psi_1(\hat{E}) \hat{\Theta}_1 \right)
- \Sigma_1(\hat{E}, \hat{U}) \frac{\gamma_2}{2} \hat{\Theta}_1^T \nabla_{\hat{E}} \psi_1(\hat{E}) B(y) R_1^{-1} B(y)^T J_T^T(\hat{E})
- \frac{\gamma_1}{64 \rho_1^2} \left\| \hat{\Theta}_1^T \nabla_{\hat{E}} \psi_1(\hat{E}) \right\|^4 D_{1min}^2 + \frac{\gamma_1}{\rho_1^4} \frac{1256}{D_{1min}^2} \left\| \hat{E}^* + \frac{D_1 \nabla_{\hat{E}} \epsilon}{2} \right\|^4 + \frac{\gamma_1}{\rho_1^2} \epsilon_{HJB}^2. \hspace{1cm} (A.3)$$

Now, completing the square with respect $\| \hat{\Theta}_1^T \nabla_{\hat{E}} \psi(\hat{E}) \|^2$ renders

$$\dot{J}_{1HJB} \leq -\frac{1}{2} \hat{\xi}^T T \hat{\xi} + \gamma_2 J_T^T(\hat{E}) \left( \omega(e_1) - \frac{1}{2} B(y) R_1^{-1} B(y)^T \nabla_{\hat{E}} \psi_1(\hat{E}) \hat{\Theta}_1 \right)
- \Sigma_1(\hat{E}, \hat{U}) \frac{\gamma_2}{2} \hat{\Theta}_1^T \nabla_{\hat{E}} \psi_1(\hat{E}) B(y) R_1^{-1} B(y)^T J_T^T(\hat{E})
- \frac{\gamma_1}{64 \rho_1^2} \left\| \hat{\Theta}_1^T \nabla_{\hat{E}} \psi_1(\hat{E}) \right\|^4 D_{1min}^2 + \frac{\gamma_1}{\rho_1^4} \frac{1256}{D_{1min}^2} \left\| \hat{E}^* + \frac{D_1 \nabla_{\hat{E}} \epsilon}{2} \right\|^4 + \frac{\gamma_1}{\rho_1^2} \epsilon_{HJB}^2. \hspace{1cm} (A.3)$$

Next, observing the bound $\| \psi_1(e_1) + B(y) U^* \| \leq \delta_1(\hat{E})$, which is similar to (22), and applying the Cauchy–Schwarz inequality, $\dot{J}_{1HJB}$ is upper bounded according to

$$\dot{J}_{1HJB} \leq \gamma_2 J_T^T(\hat{E}) \left( \omega(e_1) - \frac{1}{2} B(y) R_1^{-1} B(y)^T \nabla_{\hat{E}} \psi_1(\hat{E}) \hat{\Theta}_1 \right)
- \Sigma_1(\hat{E}, \hat{U}) \frac{\gamma_2}{2} \hat{\Theta}_1^T \nabla_{\hat{E}} \psi_1(\hat{E}) B(y) R_1^{-1} B(y)^T J_T^T(\hat{E})
- \frac{\gamma_1}{64 \rho_1^2} \left\| \hat{\Theta}_1^T \nabla_{\hat{E}} \psi_1(\hat{E}) \right\|^4 \tau_1 + \frac{\gamma_1}{\rho_1^4} \frac{1}{2} \left\| \hat{\Theta}_1^T \nabla_{\hat{E}} \psi_1(\hat{E}) \right\|^2 \tau_2 \delta_1^4(\hat{E}) - \hat{\xi}^T T \hat{\xi}, \hspace{1cm} (A.3)$$

with $\tau_1 = \nabla_{\hat{E}} \psi_1^4 D_{1min}^2 / 64$, $\tau_2 = 1024 / D_{2min}^4 + 3 / 2$, and $\eta_1(e) = 64 D_{1max}^4 e_1^4 / D_{1min}^2 + 3 \left( e_1^4 + e_1^2 D_{2max}^2 / 2 \right)$, and $0 < \nabla_{\hat{E}} \psi_1 \leq \| \nabla_{\hat{E}} \psi(\hat{E}) \|$ is ensured by $\| \hat{E} \| > 0$ for a constant $\nabla_{\hat{E}} \psi_{1\min}$. Now, the cases of $\Sigma_1(\hat{E}, \hat{U}_1) = 0$ and $\Sigma_1(\hat{E}, \hat{U}_1) = 1$ will be considered.

\textbf{Case 1}

For $\Sigma_1(\hat{E}, \hat{U}_1) = 0$, the first term in (A.3) is less than zero by the definition of the operator in (32). Recalling $\delta_1(\hat{E}) = \frac{\sqrt{2} K_{\psi_1}^* \| J_T^T \|}{\sqrt{2} K_{\psi_1}^*}$ and observing $\| 1 / \rho_1^2 \| \leq 1$, (A.3) is rewritten as

$$\dot{J}_{HJB} \leq -\frac{1}{2} \hat{\xi}^T T \hat{\xi} - \left( \gamma_2 \hat{E}_{\min} - \gamma_1 \tau_2 K^* \right) \left\| J_T(\hat{E}) \right\| - \frac{\gamma_1}{\rho_1^2} \left\| \hat{\Theta}_1 \right\|^4 \tau_1 + \frac{1}{\rho_1^2} \gamma_1 \eta_1(e), \hspace{1cm} (A.4)$$

and (A.4) is less than zero provided $\gamma_2 / \gamma_1 > \tau_2 K^* / \hat{E}_{\min}$, and the following inequalities hold

$$\left\| J_T(\hat{E}) \right\| > \gamma_1 \eta_1(e) / \left( \gamma_2 \hat{E}_{\min} - \gamma_1 \tau_2 K^* \right) \equiv b_{1JTE0}, \hspace{1cm} \text{or}$$

$$\left\| \hat{\Theta}_1 \right\| > \frac{\sqrt{2} \eta_1(e)}{\tau_1} \equiv b_{1\Theta0}, \hspace{1cm} \text{or} \hspace{1cm} \left\| \hat{\xi} \right\| > \frac{1}{\rho_1^2} \gamma_1 \eta_1(e) / \lambda_{T_{\min}} \equiv b_{1T}. \hspace{1cm} (A.5)$$

Copyright © 2013 John Wiley & Sons, Ltd.

**Case 2**

Next, consider the case of $\Sigma_1(\hat{E}, \hat{U}_1) = 1$, which implies that the OLA-based input $\hat{U}_1^* = -R_1^{-1}B(y)^T \nabla^T_{\hat{E}} \varphi_1(\hat{E}) \hat{\Theta}_1/2$ may not stabilizing. To begin, add and subtract $\alpha_2 J^T_1(\hat{E}) D \left( \nabla^T_{\hat{E}} \varphi(\hat{E}) \Theta + \nabla_{\hat{E}} \varepsilon \right)$ in (A.3) to obtain

$$\dot{J}_{HJB} \leq -\xi^T T \xi + \alpha_2 J^T_1(\hat{E}) \left( \omega(e_1) - \frac{1}{2} D \left( \nabla^T_{\hat{E}} \varphi(\hat{E}) \Theta + \nabla_{\hat{E}} \varepsilon \right) \right)$$

$$+ \frac{\gamma_1}{\rho_1^2} \left( \| \hat{\Theta} \|^4 \tau_1 \right) + \frac{\gamma_2}{\rho_1^2} \left( \| \hat{\Theta} \|^4 \tau_2 \right)$$

$$- \frac{\gamma_1}{\rho_1^2} \left( \| \hat{\Theta} \|^4 \tau_1 \right) + \frac{\gamma_2}{\rho_1^2} \left( \| \hat{\Theta} \|^4 \tau_2 \right)$$

Next, using Lemma 2 and recalling the boundedness of $D$, $\dot{J}_{HJB}$ is rewritten as

$$\dot{J}_{HJB} \leq -\xi^T T \xi - \gamma_2 \| \hat{\Theta} \|_{\min} \| J_1(\hat{E}) \|^2$$

$$+ \frac{\gamma_2}{\rho_1^2} \left( \| \hat{\Theta} \|^4 \tau_1 \right)$$

where $\hat{Q}_{\min} > 0$ satisfies $\hat{Q}_{\min} \leq \| \hat{Q}(\hat{E}) \|$ and is ensured by the condition $\| \hat{E} \| > 0$. As a final step, complete the square with respect to $\| J_1(\hat{E}) \|^2$ to reveal

$$\dot{J}_{HJB} \leq -\xi^T Q \xi - \frac{\alpha_2}{2} \hat{Q}_{\min} || J_1(\hat{E}) ||^2$$

and $\dot{J}_{HJB} < 0$ provided the following inequalities hold:

$$\| J_1(\hat{E}) \| > \sqrt{\frac{D_{\max}^2 \varepsilon_{\max}^2}{2 \hat{Q}_{\min}}} = b_{1,JE}^*,$$  or

$$\| \hat{\Theta} \| > \sqrt{\frac{\eta_1(e)}{\tau_1}} + \frac{\gamma_1}{\tau_1 \gamma_2 \hat{Q}_{\min}} \tau_2^2 K^{*2} = b_{1,\Theta}^*,$$  or

$$\| \xi \| > \sqrt{\frac{1}{\lambda T} \frac{\eta_1(e)}{\tau_1}} + \frac{\gamma_1}{\tau_1 \gamma_2 \hat{Q}_{\min}} \tau_2^2 K^{*2} = b_{1,\lambda}^*.$$

According to standard Lyapunov extensions [20], the inequalities in (A.7) guarantee that $\dot{J}_{HJB}$ is less than zero outside of a compact set. Thus, $\| J_1(\hat{E}) \|$ as well as the OLA parameter estimation error estimation error $|| \hat{\Theta} \||$ remains bounded for the case $\Sigma_1(\hat{E}, \hat{U}_1) = 1$. Recalling that the Lyapunov candidate $J_1(\hat{E})$ is a radially unbounded and continuously differentiable (Lemma 1), the boundedness of $\| J_1(\hat{E}) \|$ implies the boundedness of the states $\| \hat{E} \|$.

The overall bounds for the cases $\Sigma_1(\hat{E}, \hat{U}_1) = 0$ and $\Sigma_1(\hat{E}, \hat{U}_1) = 1$ are given by $\| J_1(\hat{E}) \| \leq b_{1,JE}$ and $\| \hat{\Theta} \| \leq b_{1,\Theta}$ for computable positive constants $b_{1,JE} = \max(b_{1,JE0}, b_{1,JE1})$ and $b_{1,\Theta} = \max(b_{1,\Theta0}, b_{1,\Theta1})$. Note that $b_{1,\lambda0}$ and $b_{1,\varepsilon1}$ in (A.5) and (A.7), respectively, can be reduced through appropriate selection of $\gamma_1$ and $\gamma_2$. To complete the proof, subtract (23), (28), and (29) from (24) to obtain

$$V_1^*(\hat{E}) - \hat{U}_1(\hat{E}) = \hat{\Theta}_1^T \varphi_1(\hat{E}) + \varepsilon_1(x)$$

$$U_1^* - \hat{U}_1 = -\frac{1}{2} R_1^{-1} B(y)^T \nabla^T_{\hat{E}} \varphi_1(\hat{E}) \hat{\Theta}_1 - \frac{1}{2} R_1^{-1} B(y)^T \nabla_{\hat{E}} \varepsilon_1(\hat{E}).$$
Next, observing that the boundedness of the system states ensures the existence of positive constants $\varphi_{1M}$ and $\varphi'_{1M}$ such that $||\varphi_1|| \leq \varphi_{1M}$ and $||\nabla \varphi_1|| \leq \varphi'_{1M}$, respectively, and taking norm and the limit as $\tau \to \infty$ when $\Sigma_1 (E, \hat{U}_1) = 0$ reveals

$$||V_1^* - \hat{V}_1|| \leq ||\hat{\Theta}_1|| \varphi_1 (\hat{E}) + \varepsilon M \leq b_1 \theta \varphi_{1M} + \varepsilon M \equiv \varepsilon_{r1}$$

$$||U_1^* (x) - \hat{U}_1 (x)|| \leq \frac{1}{2} \lambda_{max} (R_1^{-1}) B_{1M} b_1 \theta \varphi'_{1M} + \frac{\lambda_{max}}{2} (R_1^{-1}) B_M \varphi'_{1M} \equiv \varepsilon_{r2}.$$

ACKNOWLEDGEMENT

This research was supported in part by NSF grants ECCS 0624644 and ECCS 0901562.

REFERENCES