Abstract—In this paper, the optimal regulation and tracking control of affine nonlinear continuous-time systems with known dynamics is undertaken using a novel single online approximator (SOLA)-based scheme. The SOLA-based adaptive approach is designed to learn the infinite horizon continuous-time Hamilton-Jacobi-Bellman (HJB) equation and its corresponding optimal control input. A novel parameter tuning algorithm is derived which not only ensures the optimal cost (HJB) function and control input are achieved, but also ensures the system states remain bounded during the online learning process. Lyapunov techniques show that all signals are uniformly ultimately bounded (UUB) and the approximated control signal approaches the optimal control input with small bounded error. In the absence of OLA reconstruction errors, asymptotic convergence to the optimal control is shown. Simulation results illustrate the effectiveness of the approach.

Index Terms—Online nonlinear optimal control; Hamilton-Jacobi-Bellman; Online approximators; Lyapunov Stability.

I. INTRODUCTION

Control of nonlinear continuous-time systems has been considered by many researchers [1]-[2] using methods ranging from feedback linearization [1] to the use of online approximators (OLA’s) [2]. In many cases, it is desirable that the control law stabilizes the system while minimizing a pre-defined cost function for optimality. Traditionally, the optimal control of linear systems accompanied by quadratic cost functions is attained by solving the well known Riccati equation [3]. However, the optimal control of nonlinear continuous time systems is a much more challenging task that often requires solving the nonlinear Hamilton-Jacobi-Bellman (HJB) equation.

In general, the HJB equation is more difficult to work with than Riccati equations because they involve solving nonlinear partial differential equations [3]. To avoid finding exact solutions, neural networks (NN’s) and dynamic programming techniques have been used to investigate the discrete time nonlinear optimal regulation problems [4]-[5]. However, in each case the optimal solutions are obtained iteratively offline, and the NN reconstruction errors are considered to be negligible.

In addition, the optimal tracking control problem has been considered in recent literature through linearization of the tracking error equations [6], model predictive control with a receding horizon [7], inverse optimal control [8], directly calculating the infinite horizon HJB equation via offline scheme [9], and online learning-based technique [10]. In [6], $H_{\infty}$ optimal tracking control is considered by linearizing the error equations about the origin yielding a locally optimal control law. To overcome linearization, the authors in [9] consider the HJB equation and employ similar techniques as [4] to find an offline solution to the optimal tracking control.

In our previous work [10], a novel approach to the optimal regulation and tracking of nonlinear discrete-time affine systems was undertaken to solve the discrete-time HJB equation online and forward-in-time. Using an initial stabilizing control, an OLA was tuned online to learn the HJB equation while a second OLA was utilized that minimizes the cost (HJB) function based on the information provided by the first OLA. For affine nonlinear continuous-time systems, two online policy iteration schemes using NN’s have been introduced in [11] for optimal control. In each scheme, two NN’s, one referred as critic and second NN referred as action NN, are considered to approximate the cost (HJB) function and the corresponding optimal control policy, respectively.

By contrast, this work considers affine nonlinear continuous-time systems in the development of a novel single online approximator (SOLA)-based framework to learn the HJB function and optimal control input online and forward-in-time in contrast to [10] and [11] where two OLA’s are utilized. A novel online parameter tuning law is derived that not only ensures the optimal cost (HJB) function and control input is achieved, but also ensures the system states remain bounded during the online learning process. Lyapunov theory is utilized to demonstrate the stability of the system while explicitly considering the approximation errors resulting from the use of the OLA in contrast to the other works [4]. Further, the theoretical results in this work show that an initial stabilizing control is not required in contrast to [10] and [11] where an initial stabilizing control is necessary for stability.

II. NONLINEAR OPTIMAL CONTROL IN CONTINUOUS TIME

Consider the continuous nonlinear affine system

$$\dot{x} = f(x) + g(x)u_1$$

where $x \in \mathbb{R}^n$, $f(x) \in \mathbb{R}^n$, $g(x) \in \mathbb{R}^{m \times n}$ is bounded satisfying

$$g_m \leq ||g(x)||_F \leq g_M$$

where $g_m$, $g_M$ are known positive constants and the Frobenius norm is applied, and $u_1 \in \mathbb{R}^m$ is the control input. Without loss of generality, assume that the

1 Research supported in part by NSF ECCS#0621924, ECCS-0901562, and Intelligent Systems Center. Authors are with the Department of Electrical and Computer Engineering, Missouri University of Science and Technology, 301 W. 16th St., Rolla, MO 65409. Contact author Email: tplerks@drs-ssi.com
system is observable and controllable with \( x = 0 \) a unique equilibrium point on a compact set \( \Omega \in \mathbb{R}^n \) with \( f(0) = 0 \). Under these conditions, the optimal control input for the nonlinear system (1) can be calculated [3].

The infinite horizon HJB cost function for (1) is given by

\[
V(x(t)) = \int_t^\infty r(x(\tau), u_1(\tau))d\tau
\]

where \( r(x,u_1) = Q(x) + u_1^T R u_1 \), \( Q(x) > 0 \) is the positive definite penalty on the states, and \( R = \mathbb{R}^{m \times m} \) is a positive definite matrix. The control input, \( u_1 \), must be admissible so that the cost function (2) is finite [10].

Next, the Hamiltonian for the cost function (2) with an associated admissible control input \( u_1 \) is written as [3]

\[
H(x,u) = r(x,u) + V_s^T(x)(f(x) + g(x)u_1)
\]

where \( V_s(x) \) is the gradient of the \( V(x) \) with respect to \( x \). It is well known that the optimal control input that minimizes the cost function (2) also minimizes the Hamiltonian (3); therefore, the optimal control is found by solving the stationary condition \( \partial H(x,u_1)/\partial u_1 = 0 \) and revealed to be [3]

\[
u_1 * (x) = -R^{-1}g(x)^T V_s(x)/2.
\]

Substituting the optimal control (4) into the Hamiltonian (3) while observing \( H(x,u_1, V_s^*) = 0 \) reveals the HJB equation and the necessary and sufficient condition for optimal control to be [3]

\[
0 = Q(x) + V_s^T(x)f(x) - V_s^T(x)g(x)R^{-1}g(x)^TV_s(x)/4
\]

with \( V_s^*(0) = 0 \).

In [11], the optimal closed loop dynamics \( f(x) + g(x)u_1^* \) were required to satisfy \( \| f(x) + g(x)u_1^* \| \leq K \) for a constant \( K \). In contrast, in this work the optimal closed loop dynamics are assumed to be upper bounded by a function of the system states such that

\[
\| f(x) + g(x)u_1^* \| \leq \delta(x) = \sqrt{K^* J_{1x}}.
\]

where \( K^* \) is a constant and \( J_{1x}(x) \) is the partial derive with respect to \( x \) of a continuously differentiable, radially unbounded Lyapunov candidate, \( J_{1x}(x) \) satisfying

\[
J_{1x}(x) = J_{1x}^T(x)\dot{x} = J_{1x}^T(x)(f(x) + g(x)u_1^*) < 0.
\]

Moreover, it can be shown that there exists a positive definite matrix, \( \Gamma \in \mathbb{R}^{m \times m} \), such that

\[
J_{1x}^T(f(x) + g(x)u_1^*) = -J_{1x}^T \Gamma J_{1x}.
\]

Note \( \| J_{1x} \| \) can be selected to satisfy general bounds. For example, if \( \delta(x) = K_1 \| x \| \) for a constant \( K_1 \), then selecting \( J_{1x}(x) = (x^T x)^{1/2}/5 \) with \( J_{1x}(x) = (x^T x)^{1/2} x^T \) satisfies the bound. The assumption of a time varying upper bound in (6) is a less stringent assumption than the constant upper bound required in [11].

In the next section, a novel SOLA-based optimal control scheme will be introduced.

### III. Single Online Approximator-Based Optimal Control Scheme

Traditionally, adaptive critic based methodologies generate the optimal control using two OLAs [4], [10], [11] one to approximate the cost function and the other for control input. In contrast, in this work, only one OLA is utilized. As a result, this scheme is easier to implement, is less computationally demanding, and converges faster than adaptive critics which use two OLAs [11].

To begin the development, we express the cost function (2), which depends upon future values, using an OLA as

\[
V(x) = \Theta^T \phi(x) + c(x)
\]

where \( \Theta \in \mathbb{R}^l \) is the constant target OLA vector, \( \phi(x) : \mathbb{R}^n \rightarrow \mathbb{R}^l \) is a linearly independent basis vector which satisfies \( \phi(0) = 0 \), and \( c(x) \) is the OLA reconstruction error. The target OLA vector and reconstruction errors are assumed to be upper bounded according to \( \| \Theta \| \leq \Theta_M \) and \( \| c(x) \| \leq c_M \), respectively [2]. In addition, it will be assumed that the gradient of the OLA reconstruction error with respect to \( x \) is upper bounded according to \( \| \nabla_x c(x) \| = \| \nabla_x \phi(x) \| \leq c_M \).

Moving on, the gradient of the OLA cost function (8) is

\[
\partial V(x)/\partial x = V_s(x) = \nabla_x \phi(x) \Theta + \nabla_x c(x).
\]

Now, using (9), the optimal control (4) and HJB function (5) are rewritten as

\[
u_1 * (x) = -R^{-1}g(x)^T V_s(x)/2 - R^{-1}g(x)^T \nabla_x c(x)/2
\]

and

\[
H*(x, \Theta) = \tilde{Q}(x) + \tilde{\Theta}^T \nabla \phi(x)f(x) - \Theta^T \nabla \phi(x)D \tilde{V}_s \tilde{\phi}(x)\Theta/4 + \varepsilon_{HJB} = 0
\]

where \( D = g(x)R^{-1}g(x)^T > 0 \) is bounded such that \( D_{\min} \leq D \leq D_{\max} \) for known constants \( D_{\min} \) and \( D_{\max} \), and \( \varepsilon_{HJB} = \nabla \phi^T (f(x) - D \nabla \phi(x) \Theta + \nabla \phi^T \varepsilon) / 2 + \nabla \phi^T D \nabla \phi \varepsilon / 4 = \nabla \phi^T (f(x) + g(x)u^* + \nabla \phi^T \varepsilon) / 4 \) is the residual error due to the OLA reconstruction error. Asserting the bounds for the optimal closed loop dynamics (6) along with the boundedness of \( g(x) \) and \( \nabla \phi \), the residual error \( \varepsilon_{HJB} \) is bounded above on a compact set according to

\[
\| \varepsilon_{HJB} \| \leq c_M \delta(x) + c_M^2 D_{\max}.
\]

Moving on, the OLA estimate of (8) is now written as

\[
\tilde{V}(x) = \hat{\Theta}^T \hat{\phi}(x)
\]

where \( \hat{\Theta} \) is the OLA estimate of the target parameter vector \( \Theta \). Similarly, the estimate of the optimal control (10) is written in terms of \( \hat{\Theta} \) as

\[
\hat{u}_1(x) = -R^{-1}g(x)^T \nabla \hat{\phi}(x) \hat{\Theta}/2.
\]
Using (12) and (13), the approximate Hamiltonian is
\[
\dot{H}(x, \hat{\Theta}) = Q(x) + \hat{\Theta}^T \nabla_x \phi(x)f(x) - \hat{\Theta}^T \nabla_x \phi(x)DV^T_x \phi(x) \hat{\Theta}/4 .
\] (14)

**Remark 1:** Observing the definition of the approximated cost function (12) and the Hamiltonian (14), it is evident that both become zero when \(\|x\| = 0\). Thus, once the system states have converged to zero, the cost function approximation can no longer be updated. This can be viewed as a perspective of excitation (PE) requirement for the inputs to the cost function OLA [10], [11].

Observing the Hamiltonian in (5), the OLA estimate \(\hat{\Theta}\) should be tuned to minimize \(\dot{H}(x, \hat{\Theta})\). However, tuning to minimize \(\dot{H}(x, \hat{\Theta})\) alone does not ensure the stability of the nonlinear system (1) during the online learning. Therefore, the proposed OLA tuning algorithm is designed to minimize (14) while considering the stability of (1) and written as
\[
\dot{\hat{\Theta}} = -\frac{\alpha_2 \sigma}{(\hat{\Theta}^T \hat{\Theta} + 1)^2} \left[ \left( Q(x) + \hat{\Theta}^T \nabla_x \phi(x)f(x) - \frac{1}{4} \hat{\Theta}^T \nabla_x \phi(x)DV^T_x \phi(x) \hat{\Theta} \right) + 2 \alpha_1 \sigma \alpha_2 \nabla_x \phi(x)D\xi(x)/2 \right]
\] (15)
where \(\sigma = \nabla_x \phi(x) - \nabla_x \phi(x)DV^T_x \phi(x) \hat{\Theta}/2\), \(\alpha_1 > 0\) and \(\alpha_2 > 0\) are design constants, \(J_{1x}(x)\) is described in (6), and the operator \(\Sigma(x, \hat{\xi})\) is given by
\[
\Sigma(x, \hat{\xi}) = \begin{cases} 0 & \text{if } J_{1x}(x) = J_{1x}(x)f(x) - DV^T_x \phi(x) \hat{\Theta}/2 < 0, \\ 1 & \text{otherwise} \end{cases}.
\] (16)

The first term in the tuning law (15) seeks to minimize (14) and was derived using a normalized gradient descent scheme with the auxiliary HJB error defined as \(E_{HJB} = \dot{H}(x, \hat{\Theta})^2/2\). Meanwhile, the second term in (15) ensures the system states remain bounded while the SOLA scheme learns the optimal cost function. The form of the operator shown in (16) was selected based on the Lyapunov’s sufficient condition for stability (i.e. if \(J_{1x}(x) > 0\) and \(J_{1x}(x) = J_{1x}(x) < 0\), then \(x\) is stable).

From the operator in (16), the second term in (15) is removed when the nonlinear system (1) exhibits stable behavior, and learning the HJB cost function becomes the primary objective of the OLA update (15). In contrast, when the system (1) exhibits signs of instability (i.e. \(J_{1x}(x) \geq 0\)), the second term of (15) is activated and tunes the OLA parameter estimates until the nonlinear system (1) exhibits stable behavior. This approach will be shown to render guaranteed performance in the following Lyapunov proof. In addition, the numerical examples presented in Section V will also illustrate the importance of the second term in (15).

We now form the dynamics of the OLA parameter estimation error, \(\tilde{\Theta} = \Theta - \hat{\Theta}\). Observing
\[
Q(x) = -\hat{\Theta}^T \nabla_x \phi(x)f(x) + \Theta^T \nabla_x \phi(x)DV^T_x \phi(x) \Theta/4 - \varepsilon_{HJB}
\] from (11), the approximate HJB equation (14) can be rewritten in terms of \(\tilde{\Theta}\) as
\[
\dot{\tilde{\Theta}} = -\hat{\Theta}^T \nabla_x \phi(x)f(x) + \hat{\Theta}^T \nabla_x \phi(x)DV^T_x \phi(x) \hat{\Theta}/4 - \varepsilon_{HJB}
\] (17)
Next, observing \(\tilde{\Theta} = -\hat{\Theta}\) and \(\sigma = \nabla_x \phi(x)(x^* + DV_x \varepsilon/2) + \nabla_x \phi(x)DV^T_x \phi(x) \hat{\Theta}/2\) where \(x^* = f(x) + g(x)u^*_1\), the error dynamics of (15) are written as
\[
\dot{\hat{\Theta}} = -\frac{\alpha_2 \sigma}{\rho^2} \left[ \left( \sigma \nabla_x \phi(x)(x^* + DV_x \varepsilon/2) + \nabla_x \phi(x)DV^T_x \phi(x) \hat{\Theta} \right) + 2 \alpha_1 \sigma \alpha_2 \nabla_x \phi(x) D \xi(x)/2 \right]
\] (18)
where \(\rho = (\hat{\Theta}^T \hat{\Theta} + 1)^{1/2}\). Next, the stability of the SOLA-based adaptive scheme for optimal control is examined along with the stability of the nonlinear system (1). First, the following definition is required.

**Definition 1** [2]: An equilibrium point \(x^*_0\) is said to be uniformly ultimately bounded (UUB) if there exists a compact set \(S \subset \mathbb{R}^n\) so that for all \(x_0 \in S\) there exists a bound \(B\) and a time \(T\) such that \(\|x(t) - x^*_0\| \leq B\) for all \(t \geq t_0 + T\).

**Theorem 1**: Given the nonlinear system (1) with the target HJB equation (5), let the SOLA tuning law be given by (15). Then, there exists computable positive constants \(b_{xT}\) and \(b_\Theta\) such that \(\tilde{\Theta}\) and \(\|J_{1x}(x)\|\) are UUB for all \(t \geq t_0 + T\) with ultimate bounds given by \(\|J_{1x}(x)\| \leq b_{xT}\) and \(\|\tilde{\Theta}\| \leq b_\Theta\). Further, \(\|x^* - \tilde{x}\| \leq \varepsilon_{r1}\) and \(\|u^*_1 - \tilde{u}_1\| \leq \varepsilon_{r2}\) for small positive constants \(\varepsilon_{r1}\) and \(\varepsilon_{r2}\), respectively.

**Proof**: Consider the Lyapunov candidate
\[
J_{HJB} = \alpha_2 J_{1x}(x) + \hat{\Theta}^T \hat{\Theta}/2
\] (19)
whose derivative with respect to time is given by
\[
\dot{J}_{HJB} = \alpha_2 J_{1x}(x) \dot{x} + \hat{\Theta}^T \dot{\hat{\Theta}}/2
\] (20)
where \(J_{1x}(x)\) and \(J_{1x}(x)\) are described in (6). To begin, observe that if \(\|x\| = 0\), then \(J_{HJB} = \hat{\Theta}^T \hat{\Theta}/2\) with \(J_{HJB}(x) = 0\), and the parameter estimation error \(\|\tilde{\Theta}\|\) remains constant and bounded [2]. On the other hand, to successfully accomplish the online learned objective, the system states are required to satisfy \(\|x\| > \delta\) as described in Remark 1. Therefore, the remainder of this proof considers the case of \(\|x\| > 0\) (i.e. online learning is being performed). Then, substituting the nonlinear dynamics (1) with control input (13) applied along with the OLA estimation error dynamics (18) into (20) reveals
\[
\dot{J}_{HJB} = \alpha_2 J_{1x}(x) (f(x) - DV^T_x \phi(x) \hat{\Theta}/2)
\] (1520)

\[-\alpha_1 \left( \left( \hat{\Theta}^T \nabla_x \phi(x)(x^* + DV_x \varepsilon/2) \right) + 2 \alpha_1 \sigma \alpha_2 \nabla_x \phi(x) D \xi(x)/2 \right) \leq \rho^2 \]

1570
\[-\alpha_1 \left( \Theta^T \nabla \phi(x) D \nabla^2 \phi(x) \Theta \right)^T (8 \rho^2) \]
\[-3 \alpha_1 \Theta^T \nabla \phi(x) \left( \dot{x} + \frac{D \nabla \phi(x) \Theta}{2} \right) \left( \Theta^T \nabla \phi(x) D \nabla^2 \phi(x) \Theta \right) / (4 \rho^2) \]
\[-\alpha_2 \left( \Theta^T \nabla \phi(x) \right)^2 \left( \dot{x} \right) \frac{D \nabla \phi(x) \Theta}{2} \]
\[-\alpha_2 \Theta^T \nabla \phi(x) \left( \dot{x} \right) \frac{D \nabla \phi(x) \Theta}{2} \left( \Theta^T \nabla \phi(x) D \nabla^2 \phi(x) \Theta \right) / (2 \rho^2) \]
\[-\Sigma(x, \dot{x}) \alpha_3 \Theta^T \nabla \phi(x) D \nabla^2 \phi(x) \Theta \left( \Theta^T \nabla \phi(x) D \nabla^2 \phi(x) \Theta \right) / (2 \rho^2) \]

Next, completing the squares with respect to 
\[\Theta^T \nabla \phi(x) D \nabla^2 \phi(x) \Theta \] and 
\[\Theta^T \nabla \phi(x) \left( \dot{x} \right) \frac{D \nabla \phi(x) \Theta}{2} \] and taking the upper bound yields

\[J_{HJB} \leq \alpha_2 J_{1x}(x) \left[ (f(x) - D \nabla^2 \phi(x) \Theta)^2 / 2 \right] \]
\[-\alpha_1 \norm{\Theta^T \nabla \phi(x) \left( \dot{x} \right) \frac{D \nabla \phi(x) \Theta}{2} \norm{\theta}^2 \]
\[-\Sigma(x, \dot{x}) \alpha_3 \Theta^T \nabla \phi(x) D \nabla^2 \phi(x) \Theta \left( \Theta^T \nabla \phi(x) D \nabla^2 \phi(x) \Theta \right) / (2 \rho^2) \]

Now, completing the square with respect to 
\[\Theta^T \nabla \phi(x) \left( \dot{x} \right) \frac{D \nabla \phi(x) \Theta}{2} \] and taking the upper bound yields

\[J_{HJB} \leq \alpha_2 J_{1x}(x) \left[ (f(x) - D \nabla^2 \phi(x) \Theta)^2 / 2 \right] \]

Now, the cases of 
\[\Sigma(x, \dot{x}) = 0 \] and 
\[\Sigma(x, \dot{x}) = 1 \] will be considered. First, for 
\[\Sigma(x, \dot{x}) = 0 \] and 
\[\Sigma(x, \dot{x}) = 1 \] will be considered. First, for 
\[\Sigma(x, \dot{x}) = 0 \] and 
\[\Sigma(x, \dot{x}) = 1 \] will be considered. First, for 
\[\Sigma(x, \dot{x}) = 0 \] and 
\[\Sigma(x, \dot{x}) = 1 \] will be considered. First, for
Remark 3: The results of Theorem 1 indicate that the system states and OLA parameter estimation errors are UUB even when \( \Theta \) does not provide a stabilizing control input. This result implies that an initial stabilizing control is not required for implementation of the proposed SOLA design. The simulation results in Section V also support this claim. When there are no OLA reconstruction errors, the parameter estimation errors are shown to be globally asymptotically stable. Using similar methods as those used in Theorem 1, one can show that \( \bar{V} \to V^* \) and \( \hat{u}_1 \to u_1^* \) when \( \Sigma(x, \hat{u}_1) = 0 \).

In the following section, the SOLA-based design will be extended to include the tracking problem by effectively converting the tracking into a regulation problem.

IV. NEAR OPTIMAL TRACKING

The following optimal tracking development will consider the nonlinear system (1). Additionally, in this section it is assumed that there exists a matrix \( g(x)^T \in \mathbb{R}^{m \times n} \) such that \( g(x)g(x)^T = I \in \mathbb{R}^{m \times n} \) where \( I \) is the identity matrix. Note that when \( n = m \), \( g(x)^T = g(x)^{-1} \).

The objective for the infinite time optimal tracking problem is to design the optimal control \( u_1^* \) to ensure that the nonlinear system (1) tracks a desired trajectory \( x_d(t) \) with known time derivative, \( \dot{x}_d(t) \), in an optimal manner. To achieve our objective, the cost function (2) must be modified accordingly to ensure it remains finite. To begin the development, define the desired trajectory to be [6]

\[
\dot{x}_d = f(x_d) + g(x)u_d(x_d) \tag{26}
\]

where \( f(x_d) \) is the internal dynamics of the nonlinear system (1) rewritten in terms of the desired state \( x_d \), \( g(x) \) is defined in (1), and \( u_d(x_d) \) is the desired control input. It is useful to note that \( x_d, \dot{x}_d, f(x_d) \) and \( g(x) \) are known in (26) while \( u_d(x_d) \) is found by rearranging (26) to reveal

\[
u_d(x_d) = g(x)^T(\dot{x}_d - f(x_d)) \tag{27}
\]

Next, define the state tracking error as

\[
e = x - x_d \tag{28}
\]

and using (1) and (26), the tracking error dynamics are

\[
\dot{e} = f(x) + g(x)e - \dot{x}_d = f_1(e) + g(x)u_e \tag{29}
\]

where \( f_1(e) = f(x) - f(x_d) \) and \( u_e = u - u_d \). In order to control (29) in an optimal manner, it is required to select the control policy \( u_e \) that minimizes the infinite horizon HJB cost function

\[
V_T(e(t)) = \int_t^\infty (r_x(e(\tau), u_e(\tau)))d\tau \tag{30}
\]

where \( r_x(e, u) \) is defined similarly to \( r(x, u) \) with \( x \) and \( u_1 \) replaced with \( e \) and \( u_e \), respectively, and where \( u_e \) is required to be admissible. The Hamiltonian for the HJB tracking problem is now written as

\[
H_T(e, u_e) = r_x(e, u_e) + V_T^2(e)f_1(e) + g(x)u_e \tag{31}
\]

where \( V_T^2(e) \) is the gradient of the \( V_T(e) \) with respect to \( e \). Applying the stationary condition \( \partial H_T(e, u_e, d) / \partial u_e = 0 \) reveals the optimal control input for the tracking problem as

\[
u_e^* = -R^{-1}g(x)^TV_T^2(e)/2 \tag{32}
\]

Now, substituting (32) into (31) yields the HJB equation for the tracking problem to be

\[
0 = Q_e(e) + V_T^2(e)f_1(e) - V_T^2(e)g(x)R^{-1}g(x)^TV_T^2(e)/4 \tag{33}
\]

with \( V_T^2(0) = 0 \) where \( Q_e(e) \) and \( R_e \) are defined similarly to \( Q(x) \) and \( R \) presented in Section III, respectively.

Recalling \( u_e^* = u - u_d \), (32) can be rewritten as

\[
u_e^* = u - R^{-1}g(x)^TV_T^2(e)/2 \tag{34}
\]

It is observed that the optimal control input (34) consists of a predetermined feedforward term, \( u_d \), and an optimal feedback term that is a function of the gradient of the optimal cost function (30). Thus, to implement the optimal control (34), the SOLA based control scheme designed in Section III is utilized to learn the optimal feedback tracking control term after appropriate modifications to reflect (30)-(32).

Further, the Theorem 1 results are applicable to the HJB optimal tracking control problem since the cost function (30) effectively converts the tracking control problem into a regulation problem [9],[10].

V. SIMULATION RESULTS

To demonstrate the effectiveness of the SOLA-based designs of this work, the HJB regulation problem is solved for a nonlinear system. To implement the online SOLA-based designs, a linear in the parameter (LIP) NN is utilized as the OLA. In addition, the Lyapunov candidate from (6) was taken as \( J_1(x) = x^T x/2 \) so that \( J_d(x) = x \) in (15).

Although not shown due to page constraints, satisfactory performance of the SOLA-based optimal tracking control design was also observed.

Consider the nonlinear system in the form of (1) with \( x = [x_1 \ x_2]^T \), \( f(x) = \begin{bmatrix} -x_1 + x_2 \\ -x_1/2 - x_2(1 - \cos(2x_1 + 2))/2 \end{bmatrix} \), and \( g(x) = [0 \ \cos(2x_1 + 2)]^T \). Using the HJB cost function (2) with \( Q(x) = x^Tx \) and \( R = 1 \), the optimal cost function is given by \( V^*_z = W_{z1}^1 + W_{z2}^2 \) with \( W_{z1} = 1/2 \) and \( W_{z2} = 1 \) [11].

The basis vector for the SOLA-based scheme implementation was selected as \( \phi(x) = \begin{bmatrix} x_1 \ x_2 \ x_1x_2 \end{bmatrix} \).
The tuning parameters were selected as $\alpha_1 = 25$ and $\alpha_2 = 0.01$, and the initial conditions were taken as $x(0) = [5 - 5]^T$ while all NN weights were initialized to zero. That is, no initial stabilizing control was utilized for implementation of this online design.

Fig. 1 depicts the evolution of the OLA weights during the online learning. Starting from zero, the OLA weights are tuned to learn the optimal cost function, and the final values were $\hat{W}_{c4} = 0.49999$ and $\hat{W}_{c5} = 1$, with $[\hat{W}_{c1}, \hat{W}_{c2}, \hat{W}_{c3}, \hat{W}_{c4}, \hat{W}_{c5}] = [-0.0009 \ 0.0000 \ 0.0000 \ -0.003 \ 0.0030]$. These results confirm that the SOLA design converges to the actual optimal cost function with small bounded error as the theoretical results suggested. The system states are shown in Fig. 2 where probing noise was added to ensure the PE condition is satisfied. After 275 seconds, the PE condition was no longer required and was thus removed. Even though no initial stabilizing control was used, the system states remained bounded as Theorem 1 predicted. Further, the proposed single OLA-based controller is observed to converge faster than two OLA-based adaptive critics[11].

To illustrate the importance of the secondary term in the tuning law in (15), the online OLA design is attempted with $\Sigma(x, \dot{x}) = 0$ so that the system stability is not considered. Fig. 3 shows the system state quickly escape towards infinity, and the SOLA-based controller fails to learn the HJB function when the secondary term in the tuning law in (15) is ignored. Thus, the importance of the secondary term in (15) is revealed in terms of ensuring the stability of the system.

**VI. CONCLUSIONS**

In this work, a single OLA was utilized to design a single network adaptive critic, and the Hamilton Jacobi-Bellman equation was solved in real time and forward-in-time for the optimal control of general affine nonlinear continuous-time systems. In the presence of known dynamics, the optimal regulation and tracking control problems were undertaken. The OLA parameters were tuned online using a novel update law, and Lyapunov techniques were used to demonstrate the stability of the proposed optimal control scheme. Simulation results verify the theoretical conjectures.

**REFERENCES**