# Policy Iteration Algorithm for Online Design of Robust Control for a Class of Continuous-Time Nonlinear Systems

Ding Wang, Derong Liu, *Fellow, IEEE*, and Hongliang Li

***Abstract*—In this paper, a novel strategy is established to design the robust controller for a class of continuous-time nonlinear systems with uncertainties based on the online policy iteration algorithm. The robust control problem is transformed into the optimal control problem by properly choosing a cost function that reflects the uncertainties, regulation, and control. An online policy iteration algorithm is presented to solve the Hamilton–Jacobi–Bellman (HJB) equation by constructing a critic neural network. The approximate expression of the optimal control policy can be derived directly. The closed-loop system is proved to possess the uniform ultimate boundedness. The equivalence of the neural-network-based HJB solution of the optimal control problem and the solution of the robust control problem is established as well. Two simulation examples are provided to verify the effectiveness of the present robust control scheme.**

***Note to Practitioners*—Since the increasing complexity of industrial processes leads to the wide occurrence of nonlinearities and uncertainties, how to design the robust controller for nonlinear systems with uncertainties is a matter of great significance to control practitioners. In this paper, an optimal learning control approach is employed to handle the robust control problem by using the online policy iteration algorithm. Only a critic neural network needs to be constructed to approximate the cost function. The training of the action network is not required since the closed-form solution is available. The uniform ultimate boundedness of the closed-loop system based on the developed control policy is provided. The validity of the robust control strategy is illustrated through simulation study.**

***Index Terms*—Adaptive dynamic programming, neural networks, optimal control, policy iteration, robust control, uncertain nonlinear systems.**

## I. Introduction

Practical control systems are always subject to model uncertainties, exogenous disturbances or other changes in their lifetime. They are necessarily considered during the controller design process in order to avoid the deterioration of nominal closed-loop performance. We say a controller is robust if it works even if the actual system deviates from its nominal model on which the controller design is based. The importance of the robust control problem is evident, and it has been recognized by control scientists for several decades [1]–[3]. In [3], it was shown that the robust control problem can be solved by studying the corresponding optimal control problem, but the detailed procedure was not discussed.

When studying the nonlinear optimal control problem, we have to solve the Hamilton–Jacobi–Bellman (HJB) equation instead of the Riccati equation. Though dynamic programming is a useful method of optimal control, it is often computationally difficult to run it because of the "curse of dimensionality." Based on function approximators, such as neural networks, adaptive/approximate dynamic programming (ADP) was proposed in [4] and [5] as a method to solve optimal control problems forward-in-time. The ADP and related research have gained much attention from scholars of automatic control, artificial intelligence, operational research, and so on [6]–[21]. The ADP technique is closely related to the filed of reinforcement learning [13], one of whose fundamental algorithms is policy iteration [22]–[25].

Recently, Adhyaru *et al.* [26] proposed a HJB equation based optimal control algorithm for robust controller design for nonlinear systems. The algorithm is constructed using the least square method and performed offline. The stability analysis of the closed-loop optimal control system is not presented. In this paper, we employ an online policy iteration algorithm to tackle the robust control problem. The robust control problem is transformed into an optimal control problem with the cost function modified to account for uncertainties. Then, an online policy iteration algorithm is developed to solve the HJB equation by constructing and training a critic network. It is shown that an approximate closed-form expression of the optimal control policy is available. Hence, there is no need to build an action network. The uniform ultimate boundedness (UUB) of the closed-loop system is analyzed by using the Lyapunov approach. Since the ADP method is effective to solve optimal control problem and neural networks can be constructed to facilitate the implementation process, it is convenient to employ the policy iteration algorithm to handle robust control problem. Thus, the developed robust control approach is easy to understand and implement. It can be used to solve a broad class of nonlinear robust control problems.

## II. Problem Statement

In this paper, we study the continuous-time nonlinear systems described by

$$\dot{x}(t) = f\left(x(t)\right) + g\left(x(t)\right) u\left(x(t)\right) + \Delta f\left(x(t)\right) \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $u(x(t)) \in \mathbb{R}^m$ is the control vector, $f(\cdot)$ and $g(\cdot)$ are differentiable in their arguments with $f(0) = 0$, and $\Delta f(x(t))$ is the unknown perturbation. Here, we let $x(0) = x_0$ be the initial state.

Denote $\bar{f}(x) = f(x) + \Delta f(x)$. Suppose that the function $\bar{f}(x)$ is known only up to an additive perturbation, which is bounded by a known function in the range of $g(x)$. Note that the condition for the unknown perturbation to be in the range space of $g(x)$ is called the matching condition. Thus, we write $\Delta f(x) = g(x)d(x)$ with $d(x) \in \mathbb{R}^m$, which represents the matched uncertainty of the system dynamics. Assume that the function $d(x)$ is bounded by a known function $d_M(x)$, i.e., $\|d(x)\| \leq d_M(x)$ with $d_M(0) = 0$. We also assume that $d(0) = 0$, so that $x = 0$ is an equilibrium.

For system (1), in order to deal with the robust control problem, we should find a feedback control policy $u(x)$, such that the closed-loop system is globally asymptotically stable for all admissible uncertainties $d(x)$. Here, we will show that this problem can be converted into designing an optimal controller for the corresponding nominal system with appropriate cost function.

Considering the nominal system

$$\dot{x}(t) = f\left(x(t)\right) + g\left(x(t)\right) u\left(x(t)\right) \tag{2}$$

we assume that $f + gu$ is Lipschitz continuous on a set $\Omega$ in $\mathbb{R}^n$ containing the origin, and that the system (2) is controllable in the sense that there exists a continuous control policy on $\Omega$ that asymptotically

Manuscript received April 19, 2013; revised November 14, 2013; accepted November 27, 2013. Date of publication January 20, 2014; date of current version April 03, 2014. This paper was recommended for publication by Associate Editor R. Fierro and Editor A. Bicchi upon evaluation of the reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grants 61034002, 61233001, 61273140, 61304086, and 61374105, in part by the Beijing Natural Science Foundation under Grant 4132078, and in part by the Early Career Development Award of SKLMCCS.

The authors are with The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: ding.wang@ia.ac.cn; derong.liu@ia.ac.cn; hongliang.li@ia.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2013.2296206

stabilizes the system. It is desired to find the control policy $u(x)$ which minimizes the infinite horizon cost function given by

$$J(x_0) = \int_0^\infty \left\{ \rho d_M^2 (x(\tau)) + U(x(\tau), u(x(\tau))) \right\} \mathrm{d}\tau \tag{3}$$

where $\rho$ is a positive constant, $U$ is the utility function, $U(0,0) = 0$, and $U(x,u) \geq 0$ for all $x$ and $u$. In this paper, the utility function is chosen as the quadratic form $U(x,u) = x^\top Qx + u^\top Ru$, where $Q$ and $R$ are positive definite matrices with $Q \in \mathbb{R}^{n \times n}$ and $R \in \mathbb{R}^{m \times m}$. The cost function described in (3) gives a modification with respect to the ordinary optimal control problem, which appropriately reflects the uncertainties, regulation, and control simultaneously.

When dealing with the optimal control problem, the designed feedback control must be admissible. For any admissible control policy $\mu \in \Psi(\Omega)$, where $\Psi(\Omega)$ is the set of admissible controls on $\Omega$, if the associated cost function

$$V(x_0) = \int_0^\infty \left\{ \rho d_M^2 (x(\tau)) + U(x(\tau), \mu(x(\tau))) \right\} \mathrm{d}\tau \tag{4}$$

is continuously differentiable, then an infinitesimal version of (4) is the so-called nonlinear Lyapunov equation

$$0 = \rho d_M^2(x) + U(x, \mu(x)) + (\nabla V(x))^\top (f(x) + g(x)\mu(x)) \tag{5}$$

with $V(0) = 0$. In (5), the term $\nabla V(x)$ denotes the partial derivative of the cost function $V(x)$ with respect to $x$, i.e., $\nabla V(x) = \partial V(x)/\partial x$.

Define the Hamiltonian function of the problem and the optimal cost function as

$$\begin{aligned} H(x, \mu, \nabla V(x)) = {} & \rho d_M^2(x) + U(x, \mu) \\ & + (\nabla V(x))^\top (f(x) + g(x)\mu) \end{aligned} \tag{6}$$

and

$$J^*(x_0) = \min_{\mu \in \Psi(\Omega)} \int_0^\infty \left\{ \rho d_M^2 (x(\tau)) + U(x(\tau), \mu(x(\tau))) \right\} \mathrm{d}\tau. \tag{7}$$

The optimal cost function $J^*(x)$ satisfies the HJB equation

$$0 = \min_{\mu \in \Psi(\Omega)} H(x, \mu, \nabla J^*(x)) \tag{8}$$

where $\nabla J^*(x) = \partial J^*(x)/\partial x$. Assume that the minimum on the right-hand side of (8) exists and is unique. Then, the optimal control policy for the given problem is

$$u^*(x) = -\frac{1}{2} R^{-1} g^\top(x) \nabla J^*(x). \tag{9}$$

Substituting the optimal control policy (9) into the nonlinear Lyapunov (5), we can obtain the formulation of the HJB equation in terms of $\nabla J^*(x)$ as follows:

$$\begin{aligned} 0 = {} & \rho d_M^2(x) + x^\top Qx + (\nabla J^*(x))^\top f(x) \\ & - \frac{1}{4} (\nabla J^*(x))^\top g(x) R^{-1} g^\top(x) \nabla J^*(x) \end{aligned} \tag{10}$$

with $J^*(0) = 0$.

## III. Robust Control Scheme Based on the Online Policy Iteration Algorithm

### A. Equivalence of Problem Transformation

*Theorem 1:* For the nominal system (2) with the cost function (3), assume that the HJB (8) has a solution $J^*(x)$. Then, using this solution, the optimal control policy obtained in (9) ensures closed-loop asymptotic stability of the uncertain nonlinear system (1), provided that the following condition is satisfied:

$$\rho d_M^2(x) \geq d^\top(x) R d(x). \tag{11}$$

*Proof:* Let $J^*(x)$ be the optimal solution of the HJB (8) and $u^*(x)$ be the optimal control policy defined by (9). Now, we prove that $u^*(x)$ is a solution to the robust control problem, namely, the equilibrium point $x = 0$ of system (1) is asymptotically stable for all possible uncertainties $d(x)$. To do this, it is shown that $J^*(x)$ is a Lyapunov function.

According to (7), $J^*(x) > 0$ for any $x \neq 0$ and $J^*(x) = 0$ when $x = 0$. This means that $J^*(x)$ is a positive definite function. Using (8), we have

$$(\nabla J^*(x))^\top (f(x) + g(x)u^*(x)) = -\rho d_M^2(x) - U(x, u^*(x)). \tag{12}$$

Formula (9) implies that

$$2 (u^*(x))^\top R = -(\nabla J^*(x))^\top g(x). \tag{13}$$

Considering (12) and (13), we find that

$$\dot{J}^*(x) = -\rho d_M^2(x) - U(x, u^*(x)) - 2 (u^*(x))^\top R d(x). \tag{14}$$

By adding and subtracting the term $d^\top(x)Rd(x)$, (14) can further be changed to

$$\begin{aligned} \dot{J}^*(x) = {} & -\rho d_M^2(x) - x^\top Qx + d^\top(x)Rd(x) \\ & - (u^*(x) + d(x))^\top R (u^*(x) + d(x)) \\ \leq {} & -\left( \rho d_M^2(x) - d^\top(x)Rd(x) \right) - x^\top Qx. \end{aligned} \tag{15}$$

Observing (11), we can conclude that $\dot{J}^*(x) \leq -x^\top Qx < 0$ for any $x \neq 0$. Then, the conditions for Lyapunov local stability theory are satisfied. Thus, there exists a neighborhood $\Phi = \{x : \|x(t)\| < c\}$ for some $c > 0$ such that if $x(t) \in \Phi$, then $\lim_{t \to \infty} x(t) = 0$.

However, $x(t)$ cannot remain forever outside $\Phi$. Otherwise, $\|x(t)\| \geq c$ for all $t \geq 0$. Denote $q = \inf\{x^\top Qx\} > 0$. Clearly, $\dot{J}^*(x) \leq -x^\top Qx \leq -q$. Then

$$J^*(x(t)) - J^*(x(0)) = \int_0^t \dot{J}^*(x(\tau)) \mathrm{d}\tau \leq -qt. \tag{16}$$

From (16), we obtain that $J^*(x(t)) \leq J^*(x(0)) - qt \to -\infty$ as $t \to \infty$. This contradicts the fact that $J^*(x(t)) > 0$ for any $x \neq 0$. Therefore, $\lim_{t \to \infty} x(t) = 0$ no matter where the trajectory starts from. ■

*Remark 1:* According to Theorem 1, if we set $\rho = 1$ and $R = I_m$, where $I_m$ denotes the $m \times m$ identity matrix, then (11) becomes $d_M^2(x) \geq d^\top(x)d(x) = \|d(x)\|^2$. Hence, in this special case, the robust control problem is equivalent to the optimal control problem without introducing any additional conditions. Otherwise, the formula (11) should be satisfied in order to ensure the equivalence of problem transformation.

In light of Theorem 1, by acquiring the solution of the HJB (10) and then deriving the optimal control policy (9), we can obtain the robust control policy for system (1) in the presence of matched uncertainty. However, due to the nonlinear nature of the HJB equation, finding its solution is generally difficult. In the following, we will introduce an online policy iteration algorithm to solve the problem based on neural network techniques.

### B. Online Policy Iteration Algorithm

According to [27], the policy iteration algorithm consists of policy evaluation based on (5) and policy improvement based on (9). Its iteration procedure can be described as follows.

Step 1) Choose a small positive number $\epsilon$. Let $i = 0$ and $V^{(0)} = 0$. Then, start with an initial admissible control policy $\mu^{(0)}(x)$.

Step 2) Based on the control policy $\mu^{(i)}(x)$, solve the nonlinear Lyapunov equation

$$0 = \rho d_M^2(x) + U\left(x, \mu^{(i)}(x)\right) + \left(\nabla V^{(i+1)}(x)\right)^\top \left(f(x) + g(x)\mu^{(i)}(x)\right) \quad (17)$$

with $V^{(i+1)}(0) = 0$.

Step 3) Update the control policy via

$$\mu^{(i+1)}(x) = -\frac{1}{2} R^{-1} g^\top(x) \nabla V^{(i+1)}(x). \quad (18)$$

Step 4) If $\|V^{(i+1)}(x) - V^{(i)}(x)\| \leq \epsilon$, stop and obtain the approximate optimal control; else, let $i = i + 1$ and go back to Step 2.

The algorithm will converge to the optimal cost function and optimal control policy, i.e., $V^{(i)}(x) \to J^*(x)$ and $\mu^{(i)}(x) \to u^*(x)$ as $i \to \infty$. The convergence proofs of the policy iteration algorithm have been given in [22] and the related references therein.

### C. Neural Network Implementation

Assume that the cost function $V(x)$ is continuously differentiable. According to the universal approximation property of neural networks, $V(x)$ can be reconstructed by a single-layer neural network on a compact set $\Omega$ as

$$V(x) = \omega_c^\top \sigma_c(x) + \varepsilon_c(x) \quad (19)$$

where $\omega_c \in \mathbb{R}^l$ is the ideal weight, $\sigma_c(x) \in \mathbb{R}^l$ is the activation function, $l$ is the number of neurons in the hidden layer, and $\varepsilon_c(x)$ is the approximation error of the neural network. Then

$$\nabla V(x) = (\nabla \sigma_c(x))^\top \omega_c + \nabla \varepsilon_c(x) \quad (20)$$

where $\nabla \sigma_c(x) = \partial \sigma_c(x)/\partial x \in \mathbb{R}^{l \times n}$ and $\nabla \varepsilon_c(x) = \partial \varepsilon_c(x)/\partial x \in \mathbb{R}^n$ are the gradient of the activation function and neural network approximation error, respectively. Based on (20), the Lyapunov (5) takes the following form:

$$0 = \rho d_M^2(x) + U(x, \mu) + \left(\omega_c^\top \nabla \sigma_c(x) + (\nabla \varepsilon_c(x))^\top\right) \dot{x}. \quad (21)$$

Assume that the weight vector $\omega_c$, the gradient $\nabla \sigma_c(x)$, and the approximation error $\varepsilon_c(x)$ and its derivative $\nabla \varepsilon_c(x)$ are all bounded on a compact set $\Omega$. We also have $\varepsilon_c(x) \to 0$ and $\nabla \varepsilon_c(x) \to 0$ as $l \to \infty$ [23].

Since the ideal weights are unknown, a critic neural network can be built in terms of the estimated weights as

$$\hat{V}(x) = \hat{\omega}_c^\top \sigma_c(x) \quad (22)$$

to approximate the cost function. In (22), $\sigma_c(x)$ is selected such that $\hat{V}(x) > 0$ for any $x \neq 0$ and $\hat{V}(x) = 0$ when $x = 0$. Then, we have

$$\nabla \hat{V}(x) = (\nabla \sigma_c(x))^\top \hat{\omega}_c \quad (23)$$

where $\nabla \hat{V}(x) = \partial \hat{V}(x)/\partial x$.

The approximate Hamiltonian function can be derived as

$$H(x, \mu, \hat{\omega}_c) = \rho d_M^2(x) + U(x, \mu) + \hat{\omega}_c^\top \nabla \sigma_c(x) \dot{x} = e_c. \quad (24)$$

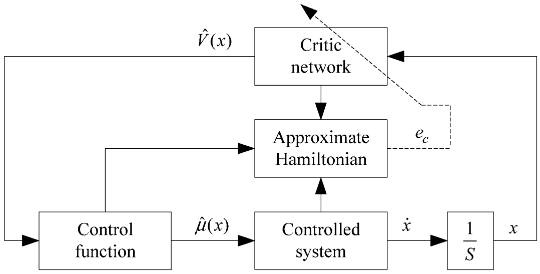

Fig. 1. The structural diagram of the algorithm (the solid line represents the signal and the dashed line represents the back-propagating path).

For training the critic network, it is desired to design $\hat{\omega}_c$ to minimize the objective function $E_c = (1/2)e_c^\top e_c$. We employ the standard steepest descent algorithm to tune the weights of the critic network, i.e.,

$$\dot{\hat{\omega}}_c = -\alpha_c \left[\frac{\partial E_c}{\partial \hat{\omega}_c}\right] = -\alpha_c e_c \left[\frac{\partial e_c}{\partial \hat{\omega}_c}\right] \quad (25)$$

where $\alpha_c > 0$ is the learning rate of the critic network.

Using (21), the Hamiltonian function becomes

$$H(x, \mu, \omega_c) = \rho d_M^2(x) + U(x, \mu) + \omega_c^\top \nabla \sigma_c(x) \dot{x} = e_{cH} \quad (26)$$

where $e_{cH} = -(\nabla \varepsilon_c(x))^\top \dot{x}$ is the residual error due to the neural network approximation.

Denote $\theta = \nabla \sigma_c(x) \dot{x}$, assume that there exists a positive constant $\theta_M$ such that $\|\theta\| \leq \theta_M$, and let the weight estimation error of the critic network be $\tilde{\omega}_c = \omega_c - \hat{\omega}_c$. Then, considering (24) and (26), we have $e_{cH} - e_c = \tilde{\omega}_c^\top \theta$. Therefore, the dynamics of weight estimation error is

$$\dot{\tilde{\omega}}_c = -\dot{\hat{\omega}}_c = \alpha_c \left(e_{cH} - \tilde{\omega}_c^\top \theta\right) \theta. \quad (27)$$

The persistency of excitation condition is required to tune the critic network ensuring $\|\theta\| \geq \theta_m$, where $\theta_m$ is a positive constant. A probing noise will be added to the system in order to satisfy the persistency of excitation condition.

When implementing the online policy iteration algorithm, for the purpose of policy improvement, we should obtain the policy that can minimize the current cost function in (19). Hence, according to (9) and (20), we have

$$\mu(x) = -\frac{1}{2} R^{-1} g^\top(x) \left((\nabla \sigma_c(x))^\top \omega_c + \nabla \varepsilon_c(x)\right). \quad (28)$$

The approximate control policy can be formulated as

$$\hat{\mu}(x) = -\frac{1}{2} R^{-1} g^\top(x) (\nabla \sigma_c(x))^\top \hat{\omega}_c. \quad (29)$$

The (29) implies that based on the trained critic network, the approximate control policy can be derived directly. The actor-critic architecture is maintained but training of the action network is not required in this case since we have closed form solution available. The structural diagram of the online policy iteration algorithm is depicted in Fig. 1.

### D. Stability Analysis

*Theorem 2:* For the controlled system (2), the weight update law for tuning the critic network is given by (25). Then, the dynamics of the weight estimation error of the critic network is UUB.

*Proof:* Select the Lyapunov function candidate as $L(t) = (1/\alpha_c)\mathrm{tr}(\tilde{\omega}_c^\top \tilde{\omega}_c)$. The time derivative of the Lyapunov function along the trajectory of error dynamics (27) is

$$\dot{L}(t) = \frac{2}{\alpha_c} \mathrm{tr}\left(\tilde{\omega}_c^\top \dot{\tilde{\omega}}_c\right) = \frac{2}{\alpha_c} \mathrm{tr}\left(\tilde{\omega}_c^\top \alpha_c \left(e_{cH} - \tilde{\omega}_c^\top \theta\right) \theta\right). \quad (30)$$

After doing some basic manipulations, we have

$$\dot{L}(t) \le -(2-\alpha_c)\left\|\tilde{\omega}_c^\top \theta\right\|^2 + \frac{1}{\alpha_c} e_{cH}^2. \tag{31}$$

Considering the Cauchy–Schwarz inequality and noticing the assumption $\|\theta\| \le \theta_M$, we can conclude that $\dot{L}(t) < 0$ as long as $0 < \alpha_c < 2$ and

$$\|\tilde{\omega}_c\| > \sqrt{\frac{e_{cH}^2}{\alpha_c(2-\alpha_c)\theta_M^2}}. \tag{32}$$

According to the Lyapunov theory, we obtain that the dynamics of the weight estimation error is UUB. The norm of the weight estimation error is bounded as well. ■

*Theorem 3:* For the controlled system (2), the weight update law of the critic network given by (25) and the approximate optimal control policy obtained by (29) ensure that, for any initial state $x_0$, there exists a time $T(x_0, M)$ such that $x(t)$ is UUB. Here, the bound $M$ is given by

$$\|x(t)\| \le \sqrt{\frac{\beta_M}{\rho\rho_0^2 + \lambda_{\min}(Q)}} \equiv M,\ t \ge T \tag{33}$$

where $\beta_M$ and $\rho_0$ are positive constants and $\lambda_{\min}(Q)$ is the least eigenvalue of $Q$.

*Proof:* Taking the time derivative of $V(x)$ along the trajectory generated by the approximate control policy $\hat{\mu}(x)$ and substituting the system dynamics (2), we can obtain

$$\dot{V} = (\nabla V(x))^\top \left(f(x) + g(x)\hat{\mu}\right). \tag{34}$$

Using (10), we can find that

$$\begin{aligned} 0 = {} & \rho d_M^2(x) + x^\top Q x + (\nabla V(x))^\top f(x) \\ & - \frac{1}{4}(\nabla V(x))^\top g(x) R^{-1} g^\top(x) \nabla V(x). \end{aligned} \tag{35}$$

Considering (35), (34) becomes

$$\begin{aligned} \dot{V} = {} & -\rho d_M^2(x) - x^\top Q x + \frac{1}{4}(\nabla V(x))^\top g(x) R^{-1} g^\top(x) \nabla V(x) \\ & + (\nabla V(x))^\top g(x)\hat{\mu}. \end{aligned} \tag{36}$$

Adding and subtracting $(\nabla V(x))^\top g(x)\mu$ and using (28) and (29), (36) becomes

$$\begin{aligned} \dot{V} = {} & -\rho d_M^2(x) - x^\top Q x - \frac{1}{4}(\nabla V(x))^\top g(x) R^{-1} g^\top(x) \nabla V(x) \\ & + \frac{1}{2}(\nabla V(x))^\top g(x) R^{-1} g^\top(x)\left(\nabla V(x) - \nabla\hat{V}(x)\right). \end{aligned} \tag{37}$$

Substituting (20) and (23) into (37), we can further obtain

$$\begin{aligned} \dot{V} = {} & -\rho d_M^2(x) - x^\top Q x - \frac{1}{4}(\nabla V(x))^\top g(x) R^{-1} g^\top(x) \nabla V(x) \\ & + \frac{1}{2}\left(\omega_c^\top \nabla\sigma_c(x) + (\nabla\varepsilon_c(x))^\top\right) g(x) R^{-1} g^\top(x) \\ & \times \left((\nabla\sigma_c(x))^\top \tilde{\omega}_c + \nabla\varepsilon_c(x)\right). \end{aligned} \tag{38}$$

Here, we denote

$$\begin{aligned} \beta = {} & \frac{1}{2}\left(\omega_c^\top \nabla\sigma_c(x) + (\nabla\varepsilon_c(x))^\top\right) g(x) R^{-1} g^\top(x) \\ & \times \left((\nabla\sigma_c(x))^\top \tilde{\omega}_c + \nabla\varepsilon_c(x)\right). \end{aligned} \tag{39}$$

Considering the fact that $R^{-1}$ is positive definite, the assumption that $\omega_c$, $\nabla\sigma_c(x)$, and $\nabla\varepsilon_c(x)$ are bounded, and Theorem 2, we can conclude that $\beta$ is upper bounded by $\beta \le \beta_M$, where $\beta_M$ is a positive constant. Therefore, $\dot{V}$ takes the following form:

$$\dot{V} \le -\rho d_M^2(x) - x^\top Q x + \beta_M. \tag{40}$$

In many cases, we can determine a quadratic bound of $d(x)$. Under such circumstances, we assume that $d_M(x) = \rho_0\|x\|$, where $\rho_0$ is a positive constant. Then, (40) becomes

$$\dot{V} \le -\left(\rho\rho_0^2 + \lambda_{\min}(Q)\right)\|x\|^2 + \beta_M. \tag{41}$$

Hence, we can observe that $\dot{V} < 0$ whenever $x(t)$ lies outside the compact set

$$\Omega_x = \left\{x : \|x\| \le \sqrt{\frac{\beta_M}{\rho\rho_0^2 + \lambda_{\min}(Q)}}\right\}. \tag{42}$$

Therefore, based on the approximate optimal control policy, the state trajectories of the closed-loop system are UUB and $\|x(t)\| \le M$. ■

In the following, the equivalence of the neural-network-based HJB solution of the optimal control problem and the solution of robust control problem is established.

*Theorem 4:* Assume that the neural-network-based HJB solution of the optimal control problem exists. Then, the control policy defined by (29) ensures closed-loop asymptotic stability of uncertain nonlinear system (1) if the formula described in (11) is satisfied.

*Proof:* Let $\hat{V}(x)$ be the solution of $0 = \rho d_M^2(x) + U(x, \hat{\mu}(x)) + (\nabla\hat{V}(x))^\top (f(x) + g(x)\hat{\mu}(x))$, and $\hat{\mu}(x)$ be the approximate optimal control policy defined by (29). Then, we have $2\hat{\mu}^\top(x)R = -(\nabla\hat{V}(x))^\top g(x)$. Now, we show that with the approximate optimal control $\hat{\mu}(x)$, the closed-loop system remains asymptotically stable for all possible uncertainties $d(x)$. According to (22) and the selection of $\sigma_c(x)$, we have $\hat{V}(0) = 0$ and $\hat{V}(x) > 0$ when $x \ne 0$. Taking the manipulations similar to the proof of Theorem 1, we can easily obtain $\hat{V}(x) \le -x^\top Q x$, which implies that $\dot{\hat{V}}(x) < 0$ for any $x \ne 0$. Thus, the proof is completed. ■

## IV. Simulation Study

*Example 1:* Consider the following continuous-time nonlinear system:

$$\begin{aligned} \dot{x} = {} & \begin{bmatrix} -x_1 + x_2 \\ -0.5x_1 - 0.5x_2\left(1 - (\cos(2x_1) + 2)^2\right) \end{bmatrix} \\ & + \begin{bmatrix} 0 \\ \cos(2x_1) + 2 \end{bmatrix}(u + 0.5px_1 \sin x_2) \end{aligned} \tag{43}$$

where $x = [x_1, x_2]^\top \in \mathbb{R}^2$ and $u \in \mathbb{R}$ are the state and control variables, respectively, and $p$ is an unknown parameter. The term $d(x) = 0.5px_1 \sin x_2$ reflects the uncertainty of the control plant. For simplicity, we assume that $p \in [-1, 1]$. Here, we choose $d_M(x) = \|x\|$ and we select $\rho = 1$ for the purpose of simulation.

We aim at obtaining a robust control policy that can stabilize system (43) for all possible $p$. This problem can be formulated into the following optimal control problem. For the nominal system, we need to find a feedback control policy $u(x)$ that minimizes the cost function

$$J(x_0) = \int_0^\infty \left\{\|x\|^2 + x^\top Q x + u^\top R u\right\} \mathrm{d}\tau \tag{44}$$

where $Q = I_2$ and $R = 2I$. Based on the procedure proposed in [28], the optimal cost function and the optimal control policy of the problem are $J^*(x) = x_1^2 + 2x_2^2$ and $u^*(x) = -(\cos(2x_1) + 2)x_2$.

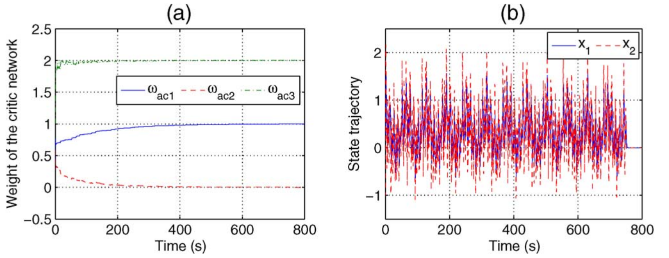

Fig. 2. Simulation results. (a) Convergence of the weight vector of the critic network ($\omega_{ac1}$, $\omega_{ac2}$, and $\omega_{ac3}$ represent $\hat{\omega}_{c1}$, $\hat{\omega}_{c2}$, and $\hat{\omega}_{c3}$, respectively). (b) Evolution of the state trajectory during the implementation process.

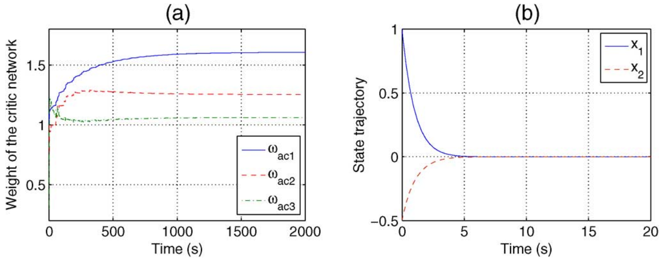

Fig. 4. Simulation results. (a) Convergence of the weight vector of the critic network. (b) The state trajectory under the robust control policy $\hat{\mu}(x)$.

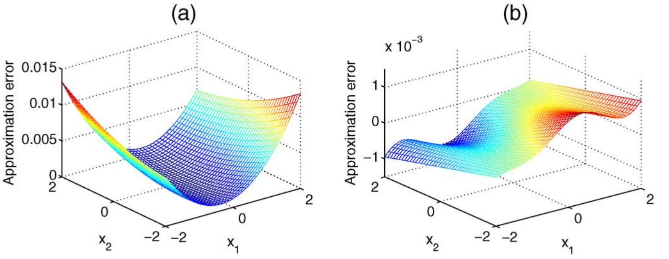

Fig. 3. Simulation results. (a) 3D plot of the approximation error of the cost function, i.e., $J^*(x) - \hat{V}(x)$. (b) 3D plot of the approximation error of the control policy, i.e., $u^*(x) - \hat{\mu}(x)$.

We adopt the online policy iteration algorithm to tackle the optimal control problem, where a critic network is constructed to approximate the cost function. We denote the weight vector of the critic network as $\hat{\omega}_c = [\hat{\omega}_{c1}, \hat{\omega}_{c2}, \hat{\omega}_{c3}]^\top$. During the simulation process, the initial weights of the critic network are chosen randomly in [0,2] and the weight normalization is not used. The activation function of the critic network is chosen as $\sigma_c(x) = [x_1^2, x_1 x_2, x_2^2]^\top$, so the ideal weight is $[1, 0, 2]^\top$. Let the learning rate of the critic network be $\alpha_c = 0.1$ and the initial state of the controlled plant be $x_0 = [1, -1]^\top$.

During the implementation process of the policy iteration algorithm, we bring in the probing noise to satisfy the persistency of excitation condition. The exponentially decreasing probing noise and sinusoidal signals with different frequencies are used. They are introduced into the control input and thus affect the system states. The weights of the critic network converge to $[0.9978, 0.0008, 1.9997]^\top$, as shown in Fig. 2(a), which displays a good approximation of the ideal ones. Actually, we can observe that the convergence of the weight has occurred after 750 s. Then, the probing signal is turned off. The evolution of the state trajectory is depicted in Fig. 2(b). We see that the state converge to zero after the probing noise is turned off.

According to (22) and (29), the approximate optimal cost function and control policy are derived. The error between the optimal cost function and the approximate one is presented in Fig. 3(a). The error between the optimal control policy and the approximate version is displayed in Fig. 3(b). Both approximation errors are close to zero, which verifies the good performance of the learning algorithm.

Next, the scalar parameter $p = 1$ is chosen for evaluating the robust control performance. Through verification, the condition of Theorem 4 is satisfied, thus the derived control policy can be employed to stabilize system (43).

*Example 2:* Consider the following continuous-time system:

$$\dot{x} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} x + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} (u + p x_2 \sin x_1) \qquad (45)$$

where $d(x) = p x_2 \sin x_1$ is the system uncertainty. Choose $a_1 = -0.5$, $a_2 = 1$, $a_3 = 0$, $a_4 = -1$, $b_1 = 0$, $b_2 = -1$, and $R = I$. Other parameters are selected the same as Example 1. Using the developed algorithm, the weights of the critic network converge to $[1.6053, 1.2545, 1.0622]^\top$, which is shown in Fig. 4(a). Then, the robust control policy can be established based on the approximate optimal control. Fig. 4(b) presents the state trajectory of the original system under the action of the robust control policy when choosing $p = 0.25$ and $x_0 = [1, -0.5]^\top$. These simulation results authenticate the availability of the robust control method.

## V. Conclusion

A novel strategy is developed to solve the robust control problem of a class of uncertain nonlinear systems. The robust control problem is transformed into an optimal control problem with appropriate cost function. The online policy iteration algorithm is presented to solve the HJB equation by constructing a critic network. Two examples are given to reinforce the theoretical results.

## References

[1] B. R. Barmish and Z. Shi, "Robust stability of a class of polynomials with coefficients depending multilinearly on perturbations," *IEEE Trans. Automat. Control*, vol. 35, no. 9, pp. 1040–1043, Sep. 1990.

[2] C. Kravaris and S. Palanki, "A Lyapunov approach for robust nonlinear state feedback synthesis," *IEEE Trans. Automat. Control*, vol. 33, no. 12, pp. 1188–1191, Dec. 1988.

[3] F. Lin, R. D. Brand, and J. Sun, "Robust control of nonlinear systems: Compensating for uncertainty," *Int. J. Control*, vol. 56, no. 6, pp. 1453–1459, 1992.

[4] P. J. Werbos, "Approximate dynamic programming for real-time control and neural modeling," in *Handbook of Intelligent Control: Neural, Fuzzy, Adaptive Approaches*, D. A. White and D. A. Sofge, Eds. New York, NY, USA: Van Nostrand Reinhold, 1992, ch. 13.

[5] P. J. Werbos, "Intelligence in the brain: A theory of how it works and how to build it," *Neural Netw.*, vol. 22, no. 3, pp. 200–212, Apr. 2009.

[6] *Handbook of Learning and Approximate Dynamic Programming*, J. Si, A. G. Barto, W. B. Powell, and D. C. Wunsch, Eds. New York, NY, USA: Wiley, 2004.

[7] D. V. Prokhorov and D. C. Wunsch, "Adaptive critic designs," *IEEE Trans. Neural Netw.*, vol. 8, no. 5, pp. 997–1007, Sep. 1997.

[8] Q. Kang, M. Zhou, J. An, and Q. Wu, "Swarm intelligence approaches to optimal power flow problem with distributed generator failures in power networks," *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 2, pp. 343–353, Apr. 2013.

[9] H. He, Z. Ni, and J. Fu, "A three-network architecture for on-line learning and optimization based on adaptive dynamic programming," *Neurocomputing*, vol. 78, no. 1, pp. 3–13, Feb. 2012.

[10] J. Fu, H. He, and X. Zhou, "Adaptive learning and control for MIMO system based on adaptive dynamic programming," *IEEE Trans. Neural Netw.*, vol. 22, no. 7, pp. 1133–1148, Jul. 2011.

[11] H. N. Wu and B. Luo, "Neural network based online simultaneous policy update algorithm for solving the HJI equation in nonlinear $H_\infty$ control," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 12, pp. 1884–1895, Dec. 2012.

[12] H. Zhang, L. Cui, and Y. Luo, "Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network ADP," *IEEE Trans. Cybern.*, vol. 43, no. 1, pp. 206–216, Feb. 2013.

[13] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits Syst. Mag.*, vol. 9, no. 3, pp. 32–50, Jul. 2009.

[14] W. S. Lin and J. W. Sheu, "Optimization of train regulation and energy usage of metro lines using an adaptive-optimal-control algorithm," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 4, pp. 855–864, Oct. 2011.

[15] T. Dierks and S. Jagannathan, "Online optimal control of affine nonlinear discrete-time systems with unknown internal dynamics by using time-based policy update," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 7, pp. 1118–1129, Jul. 2012.

[16] D. Wang, D. Liu, and Q. Wei, "Finite-horizon neuro-optimal tracking control for a class of discrete-time nonlinear systems using adaptive dynamic programming approach," *Neurocomputing*, vol. 78, no. 1, pp. 14–22, Feb. 2012.

[17] D. Liu, D. Wang, D. Zhao, Q. Wei, and N. Jin, "Neural-network-based optimal control for a class of unknown discrete-time nonlinear systems using globalized dual heuristic programming," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 3, pp. 628–634, Jul. 2012.

[18] D. Wang, D. Liu, Q. Wei, D. Zhao, and N. Jin, "Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming," *Automatica*, vol. 48, no. 8, pp. 1825–1832, Aug. 2012.

[19] D. Wang and D. Liu, "Neuro-optimal control for a class of unknown nonlinear dynamic systems using SN-DHP technique," *Neurocomputing*, vol. 121, pp. 218–225, Dec. 2013.

[20] S. K. Pradhan and B. Subudhi, "Real-time adaptive control of a flexible manipulator using reinforcement learning," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 2, pp. 237–249, Apr. 2012.

[21] Q. Wei and D. Liu, "A novel iterative $\theta$-adaptive dynamic programming for discrete-time nonlinear systems," *IEEE Trans. Autom. Sci. Eng.*, 10.1109/TASE.2013.2280974.

[22] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, no. 5, pp. 779–791, May 2005.

[23] K. G. Vamvoudakis and F. L. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, May 2010.

[24] S. Bhasin, M. Johnson, and W. E. Dixon, "A model-free robust policy iteration algorithm for optimal control of nonlinear systems," in *Proc. 49th IEEE Conf. Decision Control*, Atlanta, GA, USA, Dec. 2010, pp. 3060–3065.

[25] H. Modares, F. L. Lewis, and 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, Oct. 2013.

[26] D. M. Adhyaru, I. N. Kar, and M. Gopal, "Bounded robust control of nonlinear systems using neural network-based HJB solution," *Neural Comput. Appl.*, vol. 20, no. 1, pp. 91–103, 2011.

[27] R. S. Sutton and A. G. Barto, *Reinforcement Learning—An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

[28] V. Nevistic and J. A. Primbs, "Constrained nonlinear optimal control: A converse HJB approach," California Inst. Techn., Pasadena, CA, USA, Tech. Memo. No. CIT-CDS 96-021, Dec. 1996.

# A Generalized Result for Degradation Model-Based Reliability Estimation

Xiao-Sheng Si and Donghua Zhou, *Senior Member, IEEE*

***Abstract*—Reliability estimation based on degradation model is a feasible and low-cost alternative used to estimate reliability for highly reliable systems when the failure-time data are rare. Based on reliability estimation by degradation modeling, preventive maintenance work orders need to be timely triggered to minimize unscheduled downtime. In Trans. Autom. Sci. Eng., vol. 9, no. 1, pp. 209–212, Jan. 2012, Sun *et al.*, an approach to dynamically extract maintenance threshold is presented for maintenance scheduling, in which the reliability threshold for maintenance is determined by maximizing the expected availability and the reliability estimation is achieved by a modified two-stage degradation modeling approach. Although this approach is novel and useful, its reliability estimation is an asymptotic solution in long time scale. In this paper, we generalize the above result by considering a general degradation path model and provide the exact and explicit formulation for reliability estimation. Additionally, a maximum-likelihood estimation method for parameters in the presented model is proposed based on the historical degradation observations. Finally, an example is provided for illustration.**

***Note to Practitioners* —With advances in information and sensing technologies, the past decade has witnessed an increasingly growing research interest on various aspects of degradation model-based reliability estimation. In Trans. Autom. Sci. Eng., vol. 9, no. 1, pp. 209–212, Jan. 2012, Sun *et al.*, an approach to dynamically extract maintenance threshold is presented for maintenance scheduling. However, the formulation for reliability estimation by a modified two-stage degradation modeling approach presented in the above paper is just an asymptotic solution in long-time scale. The main contribution of this paper is that we provide the exact and explicit formulation for reliability estimation, which accounts for the impact of random effect parameter on reliability estimation. This is necessary, in general, because the approximation on reliability estimation will affect the timeliness of maintenance decision. Therefore, the results in this paper are useful for engineers and developers to implement dynamic threshold-based maintenance in the above paper.**

***Index Terms*—Brownian motion, degradation, reliability.**

## I. Introduction

Traditional methods for estimating system reliability depend on the time-to-failure data or lifetime data. However, most of critical systems and new products are forbidden to run to failure, and the cost of obtaining the failure data through the accelerated life test is very high. On the other hand, the system deteriorates over time inevitably since it operates with certain load under various environments. In contrast to the failure data, the degradation observations related to the health

Manuscript received February 25, 2013; accepted April 24, 2013. Date of publication May 23, 2013; date of current version April 03, 2014. This paper was recommended for publication by Associate Editor M. K. Jeong and Editor M. Jeong upon evaluation of the reviewers' comments. This work was supported in part by the National 973 Project under Grant 2010CB731800 and Grant 2009CB32602, and the National Science Foundation of China (NSFC) under Grant 61174030, Grant 61210012, Grant 61025014, Grant 61104223, and Grant 61290324. *(Corresponding author: D. Zhou.)*

X.-S. Si is with the Department of Automation, Xi'an Institute of High-Tech, Xi'an, Shaanxi 710025, China, and also with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: sxs09@mails.tsinghua.edu.cn).

D. Zhou is with the Department of Automation, TNList, Tsinghua University, Beijing 100084, China (e-mail: zdh@mail.tsinghua.edu.cn).

Digital Object Identifier 10.1109/TASE.2013.2260740