Neural Networks 55 (2014) 30–41

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Discrete-time online learning control for a class of unknown nonaffine nonlinear systems using reinforcement learning☆

Xiong Yang, Derong Liu*, Ding Wang, Qinglai Wei

The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

ARTICLE INFO

Article history:
Received 14 August 2013
Received in revised form 8 February 2014
Accepted 20 March 2014
Available online 28 March 2014

Keywords:
Adaptive critic design
Neural network
Nonaffine nonlinear system
Online learning
Reinforcement learning

ABSTRACT

In this paper, a reinforcement-learning-based direct adaptive control is developed to deliver a desired tracking performance for a class of discrete-time (DT) nonlinear systems with unknown bounded disturbances. We investigate multi-input–multi-output unknown nonaffine nonlinear DT systems and employ two neural networks (NNs). By using Implicit Function Theorem, an action NN is used to generate the control signal and it is also designed to cancel the nonlinearity of unknown DT systems, for purpose of utilizing feedback linearization methods. On the other hand, a critic NN is applied to estimate the cost function, which satisfies the recursive equations derived from heuristic dynamic programming. The weights of both the action NN and the critic NN are directly updated online instead of offline training. By utilizing Lyapunov's direct method, the closed-loop tracking errors and the NN estimated weights are demonstrated to be uniformly ultimately bounded. Two numerical examples are provided to show the effectiveness of the present approach.

1. Introduction

Adaptive control theory has been an active area of research for several decades, which aims to find stable controllers for nonlinear dynamic systems (Chemachema, 2012; Chen & Khalil, 1995; Ge, Hang, & Zhang, 1999; Lewis, Yesildirek, & Liu, 1996; Liu, Venayagamoorthy, & Wunsch, 2003; Nakanishi & Schaal, 2004; Narendra & Mukhopadhyay, 1994). Nevertheless, stability is only a bare minimum requirement in a system design. The optimality based on a prescribed cost function is usually taken into consideration for control problems of nonlinear systems. In other words, control schemes should be proposed to guarantee the stability of the closed-loop system, while keeping the cost function as small as possible.

In order to derive such a controller, large amounts of significant methods have been proposed. Among these approaches, dynamic programming (DP) has been widely applied to generate optimal control for nonlinear systems by employing Bellman's principle of optimality (Bellman, 1957). The method guarantees to perform optimization backward-in-time. However, a serious shortcoming about DP is that the computation is untenable to be run with the increasing dimension of nonlinear systems, which is the well-known "curse of dimensionality". Moreover, the backward direction of search obviously prohibits the wide use of DP in real-time control. On the other hand, with considerable investigations engaged in artificial neural networks (NNs), researchers find NNs can successfully be applied to intelligent control due to their properties of nonlinearity, adaptivity, self-learning, and fault tolerance (Haykin, 2008; Yu, 2009). Consequently, NNs are extensively utilized for universal function approximation in adaptive dynamic programming (ADP) algorithms, which were proposed by Werbos (1991, 1992, 2007, 2008), as methods to solve optimal control problems forward-in-time. There are several synonyms used for ADP including "adaptive dynamic programming" (Liu, Wang, & Yang, 2013; Liu & Wei, 2013; Liu, Zhang, & Zhang, 2005; Murray, Cox, Lendaris, & Saeks, 2002; Wang, Liu, & Wei, 2012; Wang, Liu, Wei, Zhao, & Jin, 2012; Wang, Zhang, & Liu, 2009; Wei & Liu, 2012; Zhang, Wei, & Liu, 2011), "approximate dynamic programming" (Al-Tamimi, Lewis, & Abu-Khalaf, 2008), "adaptive critic designs" (ACDs) (Prokhorov & Wunsch, 1997), "neuro-dynamic programming" (NDP) (Bertsekas & Tsitsiklis, 1996), and "neural dynamic programming" (Si & Wang, 2001). Furthermore, according to Prokhorov and Wunsch

☆ This work was supported in part by the National Natural Science Foundation of China under Grants 61034002, 61233001, 61273140, 61304086, and 61374105, and in part by Beijing Natural Science Foundation under Grant 4132078.

* Corresponding author. Tel.: +86 10 82544761; fax: +86 10 82544799.
E-mail addresses: xiong.yang@ia.ac.cn (X. Yang), derongliu@gmail.com, derong.liu@ia.ac.cn (D. Liu), ding.wang@ia.ac.cn (D. Wang), qinglai.wei@ia.ac.cn (Q. Wei).

http://dx.doi.org/10.1016/j.neunet.2014.03.008
0893-6080/

(1997) and Werbos (1992), ADP algorithms are mainly classified as follows: heuristic dynamic programming (HDP), dual heuristic programming (DHP), globalized dual heuristic programming (GDHP). When the action is introduced as an additional input to the critic, ACDs are referred to action dependent version of the ACDs, such as action dependent HDP (ADHDP), action dependent DHP (ADDHP), and action dependent GDHP (ADGDHP).

Unfortunately, most of ADP algorithms are implemented either by an offline process via iterative schemes or need a priori knowledge of dynamics of nonlinear systems. Since the exact knowledge of nonlinear systems is often unavailable, it brings about great challenges to implement these algorithms. In order to overcome the difficulty, reinforcement learning (RL) is introduced to cope with optimal control problems. RL is a class of approaches used in machine learning to methodically revise the actions of an agent based on responses from its environment (Sutton & Barto, 1998). A distinct difference between the traditional supervised NN learning and RL is that, there is no prescribed behavior or training model proposed to RL schemes. If the cost function is viewed as the reinforcement signal, then ADP algorithms become RL approaches. Therefore, ADP algorithms are actually a class of RL methods (Lewis & Vamvoudakis, 2011; Lewis, Vrabie, & Vamvoudakis, 2012). Since RL shares considerable common features with ADP algorithms, it is often employed for adaptive optimal controller designs.

Applications of RL methods to feedback control have been widely investigated in the literature (Bhasin et al., 2013; He & Jagannathan, 2005; Lewis, Lendaris, & Liu, 2008; Lewis & Vamvoudakis, 2011; Liu, Yang, & Li, 2013; Vamvoudakis & Lewis, 2010, 2011; Yang & Jagannathan, 2012; Yang, Liu, & Huang, 2013; Yang, Si, Tsakalis, & Rodriguez, 2009). In He and Jagannathan (2005), an RL-based output feedback control was developed for multi-input–multi-output (MIMO) unknown affine nonlinear DT systems. By using Lyapunov's direct approach, the estimated state errors, the tracking errors and the NN estimated weights were all guaranteed to be uniformly ultimately bounded (UUB). After that, in Yang et al. (2009), a direct HDP was proposed to obtain online learning control for MIMO unknown affine nonlinear DT systems. With the aid of Lyapunov's direct method, the uniform ultimate boundedness of both the closed-loop tracking errors and the NN estimated weights was derived. Just as mentioned above, in this literature, the authors took the cost function as the reinforcement signal. Recently, in Vamvoudakis and Lewis (2010), an online algorithm based on RL for affine nonlinear continuous-time (CT) systems was proposed. By employing the algorithm, both the optimal cost and the optimal control were well approximated in real time, while guaranteeing the uniform ultimate boundedness of the closed-loop system. In addition, the NN estimated weights were guaranteed to be UUB by using Lyapunov's direct method. More recently, in Vamvoudakis and Lewis (2011), RL methods were also applied to multi-player differential games for nonlinear CT systems. Based on Lyapunov's direct method, the uniform ultimate boundedness of both the closed-loop system and the NN estimated weights was demonstrated.

However, all of them deal with feedback control problems of RL methods for *affine* nonlinear systems. To the best of our knowledge, there are rather few investigations on feedback control of RL approaches for *nonaffine* nonlinear systems, especially MIMO unknown nonaffine nonlinear DT systems. Though there exist some researches about nonaffine nonlinear DT systems (Deng, Li, & Wu, 2008; Noriega & Wang, 1998; Yang, Vance, & Jagannathan, 2008), most of them focus on feedback control problems of nonlinear autoregressive moving average with exogenous inputs (NARMAX) systems. This form is less convenient than the state-form of nonaffine nonlinear systems for purpose of adaptive control using NNs. On the other hand, since the output of *affine* nonlinear systems is linear with respect to the control input, it is easy to design a controller to follow prescribed trajectories by using feedback linearization methods. Nevertheless, feedback linearization approaches cannot be implemented for *nonaffine* nonlinear systems, for the output of this type of systems depends nonlinearly on the control signal. It gives rise to great difficulties for researchers to design an efficient controller of such a nonaffine nonlinear system, which aims at achieving desired trajectories. Furthermore, in real engineering, control approaches of affine nonlinear systems do not always hold and control methods for nonaffine nonlinear systems are necessary. Therefore, control problems of RL methods for unknown nonaffine nonlinear systems are very significant in both theory and applications.

The objective of this paper is to develop an online direct adaptive control based on RL methods by delivering a desired tracking performance for MIMO unknown nonaffine nonlinear DT systems with unknown bounded disturbances. Two NNs are employed in the controller design: an action NN is utilized to generate the control signal. Meanwhile, by using Implicit Function Theorem, the action NN approximation is well designed to cancel the nonlinearity of unknown nonlinear DT systems, for purpose of utilizing feedback linearization methods. A critic NN is used to estimate the prescribed cost function, which satisfies the recursive equations derived from HDP. The weights of both the action NN and the critic NN are directly updated online instead of preliminary offline training. By using Lyapunov's direct method, the closed-loop tracking errors and the NN estimated weights are verified to be UUB.

The main contributions of the paper include the following:

1. To the best of our knowledge, it is the first time that an online RL-based direct adaptive control is developed for the state-form of MIMO unknown nonaffine nonlinear DT systems with unknown bounded disturbances.
2. Compared with He and Jagannathan (2005), Yang et al. (2009), and Yang and Jagannathan (2012), we consider nonaffine nonlinear DT systems with unknown system drift dynamics. A significant difference between these literature and the present paper is that, in our case, the adaptive control is developed based on Implicit Function Theorem and RL methods since feedback linearization methods cannot be directly implemented for nonaffine nonlinear DT systems.

The rest of the paper is organized as follows. Section 2 provides the problem statement and preliminaries. Section 3 develops an online adaptive control by using RL approaches. Section 4 shows the stability analysis and the performance of the closed-loop systems. Section 5 presents two simulation results to verify the effectiveness of the established theory. Finally, Section 6 gives several concluding remarks.

For convenience, we introduce the notations, which will be used throughout the paper.

- $\mathbb{R}$ denotes the real numbers, $\mathbb{R}^m$ and $\mathbb{R}^{m\times n}$ denote the real m-vectors and the real $m \times n$ matrices, respectively. $\otimes$ denotes the Kronecker product. If there is no special explanation, T is a transposition symbol.
- Ω is a compact set of $\mathbb{R}^{mn}$, $C^m(\Omega) = \left\{f^{(m)} \in C | f: \Omega \to \mathbb{R}^m\right\}$. Let $\Omega_i \subset \Omega$ $(i = 1, 2)$, $\Omega_1 \times \Omega_2 = \left\{(x, y) | x \in \Omega_1, y \in \Omega_2\right\}$ stands for the Cartesian product of Ω_1 and Ω_2.
- $\|\cdot\|$ stands for any suitable norm. When z is a vector, $\|z\|$ denotes the Euclidean norm of z. When A is a matrix, $\|A\|$ denotes the 2-norm of A.