Automatica 50 (2014) 3281–3290

Contents lists available at ScienceDirect

Automatica

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

Brief paper

Data-based approximate policy iteration for affine nonlinear continuous-time optimal control design☆

Biao Luo[a], Huai-Ning Wu[b,1], Tingwen Huang[c], Derong Liu[a]

[a] State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, PR China

[b] Science and Technology on Aircraft Control Laboratory, Beihang University (Beijing University of Aeronautics and Astronautics), Beijing 100191, PR China

[c] Texas A&M University at Qatar, PO Box 23874, Doha, Qatar

ARTICLE INFO

Article history:
Received 24 October 2013
Received in revised form
2 August 2014
Accepted 3 August 2014
Available online 25 October 2014

Keywords:
Nonlinear optimal control
Reinforcement learning
Off-policy
Data-based approximate policy iteration
Neural network
Hamilton–Jacobi–Bellman equation

ABSTRACT

This paper addresses the model-free nonlinear optimal control problem based on data by introducing the reinforcement learning (RL) technique. It is known that the nonlinear optimal control problem relies on the solution of the Hamilton–Jacobi–Bellman (HJB) equation, which is a nonlinear partial differential equation that is generally impossible to be solved analytically. Even worse, most practical systems are too complicated to establish an accurate mathematical model. To overcome these difficulties, we propose a data-based approximate policy iteration (API) method by using real system data rather than a system model. Firstly, a model-free policy iteration algorithm is derived and its convergence is proved. The implementation of the algorithm is based on the actor–critic structure, where actor and critic neural networks (NNs) are employed to approximate the control policy and cost function, respectively. To update the weights of actor and critic NNs, a least-square approach is developed based on the method of weighted residuals. The data-based API is an off-policy RL method, where the "exploration" is improved by arbitrarily sampling data on the state and input domain. Finally, we test the data-based API control design method on a simple nonlinear system, and further apply it to a rotational/translational actuator system. The simulation results demonstrate the effectiveness of the proposed method.

1. Introduction

The nonlinear optimal control problem has been widely studied in the past few decades, and a large number of theoretical results (Bertsekas, 2005; Hull, 2003; Lewis, Vrabie, & Syrmos, 2013) have been reported. However, the main bottleneck for their practical application is that the so-called Hamilton–Jacobi–Bellman (HJB) equation should be solved. The HJB equation is a first order nonlinear partial differential equation (PDE), which is difficult or impossible to solve, and may not have global analytic solutions even in simple cases. For linear systems, the HJB equation results in an algebraic Riccati equation (ARE). In 1968, Kleinman (1968) proposed a famous iterative scheme for solving the ARE, where it was converted to a sequence of linear Lyapunov matrix equations. In Saridis and Lee (1979), the thought of the iterative scheme was extended to solve the HJB equation, which was successively approximated by a series of generalized HJB (GHJB) equations that are linear PDEs. To solve the GHJB equation, Beard, Saridis, and Wen (1997) proposed a Galerkin approximation approach where a detailed convergence analysis was provided. By using a neural network (NN) for function approximation, the iterative scheme was further extended to constrained input systems (Abu-Khalaf & Lewis, 2005). In Lin, Loxton, and Teo (2014); Wang, Gui, Teo, Loxton, and Yang (2009), the control parameterization method does not require the solution of the HJB equation, and can handle state constraints, which furnishes an open-loop control rather than a feedback control. However, most of these approaches are model-based which require an accurate mathematical model of the system.

With the fast development of science technologies, many industrial systems (such as systems in aeronautics and astronautics, chemical engineering, mechanical engineering, electronics, electric power, traffic and transportation) become more and more complicated due to their large scale and complex manufacturing

☆ The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Kok Lay Teo under the direction of Editor Ian R. Petersen.

E-mail addresses: biao.luo@ia.ac.cn (B. Luo), whn@buaa.edu.cn (H.-N. Wu), tingwen.huang@qatar.tamu.edu (T. Huang), derong.liu@ia.ac.cn (D. Liu).

1 Tel.: +86 10 82317301; fax: +86 10 82317332.

http://dx.doi.org/10.1016/j.automatica.2014.10.056
0005-1098/

techniques, equipment and procedures. One of the most prominent features for these systems is the presence of vast volume of data accompanied by the lack of an effective physical process model that can support control design. Moreover, the accurate modeling and identification of these systems are extremely costly or impossible to conduct. On the other hand, with the development and extensive applications of digital sensor technologies, and the availability of cheaper measurement and computing equipments, more and more system information could be extracted for direct control design. Thus, the development of data-based control approaches for practical systems is a promising, but still challenging research area.

Over the past few decades, the thought of reinforcement learning (RL) techniques has been introduced to study the optimal control problems (Al-Tamimi, Lewis, & Abu-Khalaf, 2008; Lewis & Liu, 2013; Lewis, Vrabie, & Vamvoudakis, 2012; Luo, Wu, & Huang, in press; Luo, Wu, & Li, in press, 2014; Si & Wang, 2001; Vrabie & Lewis, 2009). RL methods have the ability to find an optimal control policy in an unknown environment, which makes RL a promising method for data-based control design. For discrete-time systems, many RL based optimal control approaches have been developed, such as, heuristic dynamic programming (HDP) (Al-Tamimi et al., 2008), direct HDP (Si & Wang, 2001), dual heuristic programming (Heydari & Balakrishnan, 2013), and globalized DHP algorithm (Wang, Liu, Wei, Zhao, & Jin, 2012). For continuous-time systems, Vrabie and Lewis (2009) proposed a policy iteration algorithm to solve the nonlinear optimal control problem online along a single state trajectory. Vamvoudakis and Lewis (2010) gave an online policy iteration algorithm which tunes synchronously the weights of both actor and critic NNs for the nonlinear optimal control problem. In Liu, Wang, and Li (2014), approximate dynamic programming (ADP) was employed to design a stabilizing control strategy for a class of continuous-time nonlinear interconnected large-scale systems. But those methods are partially model-based (Vrabie & Lewis, 2009) or completely model-based (Liu et al., 2014; Vamvoudakis & Lewis, 2010). Recently, some data-based RL methods have been reported. For example, data-based policy iteration (Jiang & Jiang, 2012) and Q-learning (Lee, Park, & Choi, 2012) algorithms were developed for linear systems. Zhang, Cui, Zhang, and Luo (2011) presented a data-driven robust approximate optimal tracking control scheme for nonlinear systems, but it requires a prior model identification procedure and the ADP method is still model-based. Till present, the development of model-free RL methods and theories for nonlinear continuous-time optimal control problem remains an open issue, which motivates the present study.

In this paper, we consider the optimal control problem of continuous-time nonlinear systems with completely unknown model, and develop a model-free approximate policy iteration (API) method for learning the optimal control policy from real system data. The rest of the paper is arranged as follows. The problem description and some preliminary results are presented in Sections 2 and 3. A data-based API method is developed in Section 4 and its effectiveness is tested in Section 5. Finally, a brief conclusion is given in Section 6.

Notation. $\mathbb{R}$, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ are the set of real numbers, the n-dimensional Euclidean space and the set of all real matrices, respectively. $\|\cdot\|$ denotes the vector norm or matrix norm in $\mathbb{R}^n$ or $\mathbb{R}^{n\times m}$, respectively. The superscript T is used for the transpose and I denotes the identify matrix of appropriate dimension. $\nabla \triangleq \partial/\partial x$ denotes a gradient operator notation. For a symmetric matrix M, $M > (\geq) 0$ means that it is a positive (semi-positive) definite matrix. $\|v\|_M^2 \triangleq v^T M v$ for some real vector v and symmetric matrix $M > (\geq) 0$ with appropriate dimensions. $C^1(\mathcal{X})$ is a function space on $\mathcal{X}$ with first derivatives are continuous. Let $\mathcal{X}$ and $\mathcal{U}$ be compact sets, denote $\mathcal{D} \triangleq \{(x, u, x')|x, x' \in \mathcal{X}, u \in \mathcal{U}\}$. For column vector functions $s_1(x, u, x')$ and $s_2(x, u, x')$, where $(x, u, x') \in \mathcal{D}$ define the inner product $\langle s_1(x, u, x'), s_2(x, u, x')\rangle_{\mathcal{D}} \triangleq \int_{\mathcal{D}} s_1^T(x, u, x') s_2(x, u, x') d(x, u, x')$ and the norm $\|s_1(x, u, x')\|_{\mathcal{D}} \triangleq \langle s_1(x, u, x'), s_1(x, u, x')\rangle_{\mathcal{D}}^{1/2}$.

2. Problem description

Let us consider the following continuous-time affine nonlinear system:

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t), \quad x(0) = x_0 \tag{1}$$

where $[x_1 \ \ldots \ x_n]^T \in \mathcal{X} \subset \mathbb{R}^n$ is the state, x_0 is the initial state and $u = [u_1 \ \ldots \ u_m]^T \in \mathcal{U} \subset \mathbb{R}^m$ is the control input. Assume that $f(x) + g(x)u(t)$ is Lipschitz continuous on a set $\mathcal{X}$ that contains the origin, $f(0) = 0$, and that the system is stabilizable on $\mathcal{X}$, i.e., there exists a continuous control function such that the system is asymptotically stable on $\mathcal{X}$. In this paper, system dynamics $f(x)$ and $g(x)$ are *unknown* continuous vector or matrix functions of appropriate dimensions.

The optimal control problem under consideration is to find a state feedback control law $u(t) = u^*(x)$ such that the system (1) is closed-loop asymptotically stable, and the following infinite horizon cost function is minimized:

$$V(x_0) \triangleq \int_0^\infty \left(Q(x(t)) + \|u(t)\|_R^2\right) dt \tag{2}$$

where $R > 0$ and $Q(x)$ is a positive definite function, i.e., for $\forall x \neq 0$, $Q(x) > 0$, $Q(x) = 0$ only when $x = 0$. Then, the optimal control problem is briefly presented as

$$u(t) \triangleq u^*(x) \triangleq \arg\min_u V(x_0). \tag{3}$$

3. Preliminary works

From the optimal control theory (Bertsekas, 2005; Lewis et al., 2013), if the mathematical model of system (1) is completely known, the optimal control problem (3) with cost function (2) can be converted to solve the following HJB equation:

$$[\nabla V^*(x)]^T f(x) + Q(x) - \frac{1}{4}[\nabla V^*(x)]^T g(x) R^{-1} g^T(x) \nabla V^*(x) = 0 \tag{4}$$

where $V^*(x) \in C^1(\mathcal{X})$, $V^*(x) \geq 0$ and $V^*(0) = 0$. Then, the optimal controller (3) is given by

$$u^*(x) = -\frac{1}{2} R^{-1} g^T(x) \nabla V^*(x). \tag{5}$$

It is noted that the optimal control policy (5) depends on the solution $V^*(x)$ of the HJB equation (4). However, the HJB equation is a nonlinear PDE that is impossible to be solved analytically. To obtain its approximate solution, in Saridis and Lee (1979), the HJB equation (4) was successively approximated by a sequence of GHJB equations as follows:

$$[\nabla V^{(i+1)}]^T (f + g u^{(i)}) + Q(x) + \|u^{(i)}\|_R^2 = 0 \tag{6}$$

with

$$u^{(i)} = -\frac{1}{2} R^{-1} g^T(x) \nabla V^{(i)}(x). \tag{7}$$

By providing an initial admissible control policy $u^{(0)}$ (see Definition 1 for admissible control), it has been proven in Saridis and Lee (1979) that the solution of the iterative GHJB equation (6) will converge to the solution of the HJB equation (4), i.e., $\lim_{i\to\infty} V^{(i)} = V^*$ and $\lim_{i\to\infty} u^{(i)} = u^*$.