Local Policy Iteration Adaptive Dynamic Programming for Discrete-Time Nonlinear Systems

Qinglai Wei[1(✉)], Yancai Xu[1], Qiao Lin[1], Derong Liu[2], and Ruizhuo Song[2]

[1] The State Key Laboratory of Management and Control for Complex Systems,
Institute of Automation, Chinese Academy of Sciences, University of Chinese
Academy of Sciences, Beijing 100190, China
{qinglai.wei,yancai.xu,linqiao2014}@ia.ac.cn
[2] The School of Automation and Electrical Engineering,
University of Science and Technology Beijing, Beijing 100083, China
{derong,ruizhuosong}@ustb.edu.cn

Abstract. Adaptive dynamic programming is a hot research topic nowadays. Therefore, the paper concerns a new local policy adaptive iterative dynamic programming (ADP) algorithm. Moreover, this algorithm is designed for the discrete-time nonlinear systems, which are used to solve problems concerning infinite horizon optimal control. The new local policy iteration ADP algorithm has the characteristics of updating the iterative control law and value function within one subset of the state space. Morevover, detailed iteration process of the local policy iteration is presented thereafter. The simulation example is listed to show the good performance of the newly developed algorithm.

Keywords: Nonlinear systems · Approximate dynamic programming · Local policy iteration · Optimal control · Discrete time

1 Introduction

Adaptive dynamic programming (ADP) is always a hot research area since proposed by Werbos [1]. ADP is a very useful and significant intelligent way to solve nonlinear system problems. With the aim of getting optimal control law, the corresponding iterative learning methods are applied to analyze the convergence and optimality characteristics of ADP [2–7].

It has to be admitted that the iterative control laws and the iterative value functions usually have to be updated in the whole state space [8–18], which are also as "global policy iteration algorithms". Moreover, the global policy iteration algorithms have the disadvantages of low efficiency during applications. Most of time, the algorithm has to pause to wait for the accomplishment of a search of the whole state area. Correspondingly, the computation efficiency goes down in the global policy iteration algorithm. The constraint has hindered the development of this research area. Therefore, useful policy iteration algorithms need to be proposed to increase computation efficiency.

F. Cong et al. (Eds.): ISNN 2017, Part II, LNCS 10262, pp. 148–153, 2017.
DOI: 10.1007/978-3-319-59081-3_18

This paper has proposed a new “local policy iteration algorithm” concerning the discrete nonlinear systems. It proves its usage to iterative in a small area. The algorithm has the ability to update the iterative control laws and also the iterative value functions within the given area of the state space. Despite the fact of iterative control laws updating within a preset state space, the system still has the ability to keep stable under any kind of iterative control law. At the end, the simulation part shows the good performance of this newly developed method.

2 Problem Statement

We assume a deterministic discrete-time nonlinear system here

$$s_{k+1} = F(s_k, c_k),\ \ k = 0, 1, 2, \ldots, \tag{1}$$

where $s_k \in \mathbb{R}^n$ is the state vector. Besides, $c_k \in \mathbb{R}^m$ is the control vector. Assume s_0 as the initial state and $F(s_k, c_k)$ as the system function. Assume $\underline{c}_k = (c_k, c_{k+1}, \ldots)$ as an arbitrary sequence of controls. The performance index function can be defined as

$$J(s_0, \underline{c}_0) = \sum_{k=0}^{\infty} U(s_k, c_k), \tag{2}$$

for state s_0 under the control sequence $\underline{c}_0 = (c_0, c_1, \ldots)$. The utility function $U(x_k, c_k)$ is a positive definite function for s_k and c_k. It is noted that $\underline{c}_k$ changes from k to ∞.

We aim to find an optimal scheme. The scheme has the ability to minimize performance index function (2) while stabilizing system (1).

Assume the control sequence set as $\underline{\mathfrak{U}}_k = \big\{\underline{c}_k \colon \underline{c}_k = (c_k, c_{k+1}, \ldots), \forall c_{k+i} \in \mathbb{R}^m, i = 0, 1, 2, \ldots\big\}$.

Then, for an arbitrary control sequence $\underline{c}_k \in \underline{\mathfrak{U}}_k$, the optimal performance index function is

$$J^*(s_k) = \inf_{\underline{c}_k} \left\{ J(s_k, \underline{c}_k) \colon \underline{c}_k \in \underline{\mathfrak{U}}_k \right\}. \tag{3}$$

Based on Bellman principle of optimality, $J^*(s_k)$ meet the requirement of the discrete-time HJB formula

$$J^*(s_k) = \inf_{c_k} \left\{ U(s_k, c_k) + J^*(F(s_k, c_k)) \right\}. \tag{4}$$

Define the law of optimal control as

$$c^*(s_k) = \arg\inf_{c_k} \left\{ U(s_k, c_k) + J^*(F(s_k, c_k)) \right\}. \tag{5}$$

Therefore, the HJB Eq. (4) is

$$J^*(s_k) = U(s_k, c^*(s_k)) + J^*(F(s_k, c^*(s_k))). \tag{6}$$

Overall, there exists the curse of dimensionality. So it is very difficult to obtain the numerical results for the traditional dynamic programming algorithms. Considering this situation, we have proposed a new ADP algorithm thereafter.

3 Descriptions of This New Local Iterative ADP Algorithm

We have designed a new local iterative ADP algorithm. This section gives a detailed description of the algorithm. It is designed to have the ability to get the optimal control law for system (1) correspondingly. Assume $\{\Theta_s^i\}$ as the state sets, $\Theta_s^i \subseteq \Omega_s$, $\forall\, i$. The value iteration functions and the control laws of the newly developed algorithm have to be updated iteratively.

For all $s_k \in \Omega_s$, assume $v_0(s_k)$ as an admissible control law. Besides, assume $V_0(x_s)$ as the initial iterative value function for all $s_k \in \Omega_s$. The function satisfies the generalized HJB (GHJB) equation

$$V_0(s_k) = U(s_k, v_0(s_k)) + V_0(s_{k+1}), \tag{7}$$

where $s_{k+1} = F(s_k, v_0(s_k))$. Then, for all $s_k \in \Theta_s^0$, the local iterative control law $v_1(s_k)$ is computed as

$$v_1(s_k) = \arg\min_{c_k} \{U(s_k, c_k) + V_0(s_{k+1})\} \tag{8}$$

and let $v_1(s_k) = v_0(s_k)$, for all $s_k \in \Omega_s \backslash \Theta_s^0$.

For all $s_k \in \Omega_s$, assume $V_1(s_k)$ as the iterative value function. Therefore, $V_1(s_k)$ satisfies the GHJB equation

$$V_1(s_k) = U(s_k, v_1(s_k)) + V_1(F(s_k, v_1(s_k))). \tag{9}$$

For $i = 1, 2, \ldots$, assume $V_i(s_k)$ as the iterative value function. So $V_i(s_k)$ can satisfy the following GHJB equation

$$V_i(s_k) = U(s_k, v_i(s_k)) + V_i(F(s_k, v_i(s_k))). \tag{10}$$

For all $s_k \in \Theta_x^i$, the iterative control law $v_{i+1}(s_k)$ should be computed as

$$\begin{aligned} v_{i+1}(s_k) &= \arg\min_{c_k} \{U(s_k, c_k) + V_i(s_{k+1})\} \\ &= \arg\min_{c_k} \{U(s_k, c_k) + V_i(F(s_k, c_k))\}, \end{aligned} \tag{11}$$

and for all $s_k \in \Omega_s \backslash \Theta_s^i$, let $v_{i+1}(s_k) = v_i(s_k)$.

The local policy iteration algorithm will be updated within the preset subset of state space according to Eqs. (7) and (11). The given subset is part of whole state space. Therefore, during iterations, once local data of state space is got, the newly developed algorithm can be performed immediately. The advantage is that the algorithm can save lots of time while competing all the data of the whole space in traditional algorithms. Therefore, the computation efficiency can be improved greatly and save a lot of trouble. Besides, if the preset subset of state space is enlarged to all, local policy iteration algorithms equal to the global policy iteration ones.

4 Simulation Examples

First, we have chosen a discretized nonaffine nonlinear system as follows

$$\begin{aligned} s_{1(k+1)} &= (1-\varDelta T)s_{1k} + \varDelta T s_{2k} c_k, \\ s_{2(k+1)} &= (1-\varDelta T)x_{2k} + \varDelta T(1+s_{1k}^2)c_k + \varDelta T c_k^3. \end{aligned} \tag{12}$$

We choose the utility function as $Q = I_1$ and $R = I_2$. Thereafter, We choose the state space as $\varOmega_s$. While I_1 and I_2, are denoted as the identity matrices with suitable dimensions. Let the initial state be $s_0 = [1, -1]^\mathsf{T}$. Based on Algorithm 1 in [16].

The iterative value functions and iterative control laws should be updated accordingly. After 30 iterations, the algorithm has reached corresponding computing precision of $\varepsilon = 0.001$. Figure 1(a) shows that the iterative value function

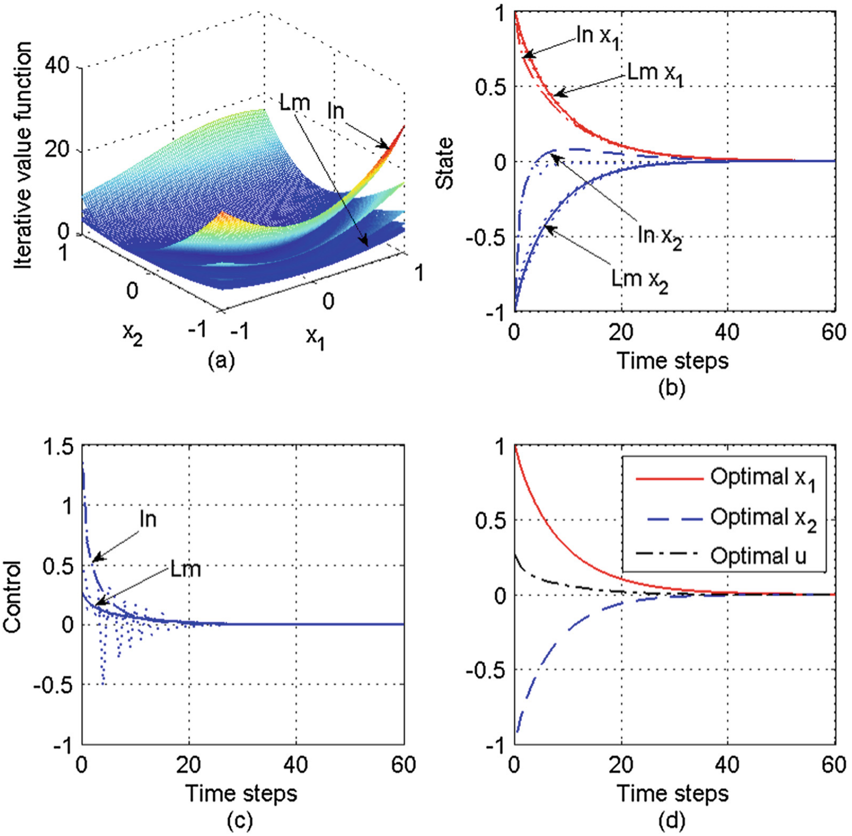

Fig. 1. Simulation results of the new local policy iteration algorithm. (a) Corresponding iterative value function. (b) Corresponding state trajectories. (c) Corresponding control trajectories. (d) Corresponding optimal state and control trajectories.

is monotonically nonincreasing. More importantly, the value function converges to the optimum. Figure 1(b) illustrates the trajectories of simulation states while Fig. 1(c) shows the simulation functions. In Fig. 1(d), we have shown the optimal trajectories of control and also states correspondingly.

5 Conclusion

We proposed a new local policy iteration ADP algorithm in this paper. The algorithm has the ability to greatly improve the computation efficiency of traditional ADP algorithm concerning discrete time nonlinear systems. Therefore, it can reduce computation time greatly which contrast to traditional global policy iteration algorithms. The characteristic concerning this newly developed algorithm is that the iteration control laws and iterative iteration control laws are updated within a preset area of the state space. Besides, the simulation results have proven its effectiveness of the newly developed algorithm.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China under Grants 61233001, 61273140, 61374105, and 61304079.

References

1. Werbos, P.: Advanced forecasting methods for global crisis warning and models of intelligence. Gen. Syst. Yearb. **22**, 25–38 (1977)
2. Fu, Y., Fu, J., Chai, T.: Robust adaptive dynamic programming of two-player zero-sum games for continuous-time linear systems. IEEE Trans. Neural Netw. Learn. Syst. **26**, 3314–3319 (2015). doi:10.1109/TNNLS.2015.2461452
3. Abouheaf, M., Lewis, F., Vamvoudakis, K., Haesaert, S., Babuska, R.: Multi-agent discrete-time graphical games and reinforcement learning solutions. Automatica **50**(12), 3038–3053 (2014)
4. Zargarzadeh, H., Dierks, T., Jagannathan, S.: Optimal control of nonlinear continuous-time systems in strict-feedback form. IEEE Trans. Neural Netw. Learn. Syst. **26**(10), 2535–2549 (2015)
5. Wei, Q., Liu, D.: Data-driven neuro-optimal temperature control of water gas shift reaction using stable iterative adaptive dynamic programming. IEEE Trans. Industr. Electron. **61**(11), 6399–6408 (2014)
6. Heydari, A.: Revisiting approximate dynamic programming and its convergence. IEEE Trans. Cybern. **44**(12), 2733–2743 (2014)
7. Lewis, F., Vrabie, D., Vamvoudakis, K.: Reinforcement learning and feedback control: using natural decision methods to design optimal adaptive controllers. IEEE Control Syst. **32**(6), 76–105 (2012)
8. Wei, Q., Liu, D., Lin, H.: Value iteration adaptive dynamic programming for optimal control of discrete-time unknown nonlinear systems with disturbance using ADP. IEEE Trans. Neural Netw. Learn. Syst. **27**(2), 444–458 (2016)
9. Wei, Q., Liu, D., Yang, X.: Inifinite horizon self-learning optimal control of nonaffine discrete-time nonlinear systems. IEEE Trans. Neural Netw. Learn. Syst. **26**(4), 879–886 (2015)

10. Wei, Q., Song, R., Yan, P.: Data-driven zero-sum neuro-optimal control for a class of continuous-time unknow nonlinear systems with disturbance using ADP. IEEE Trans. Neural Netw. Learn. Syst. **27**(2), 444–458 (2016)
11. Wei, Q., Wang, F., Liu, D., Yang, X.: Finite-approximation-error based discrete-time iterative adaptive dynamic programming. IEEE Trans. Cybern. **44**(12), 2820–2833 (2014)
12. Wei, Q., Liu, D., Shi, G., Liu, Y.: Optimal multi-battery coordination control for home energy management systems via distributed iterative adaptive dynamic programming. IEEE Trans. Ind. Electron. **42**(7), 4203–4214 (2015)
13. Wei, Q., Liu, D., Shi, G.: A novel dual iterative Q-learning method for optimal battery management in smart residential environments. IEEE Trans. Ind. Electron. **62**(4), 2509–2518 (2015)
14. Wei, Q., Liu, D.: A novel iterative θ-adaptive dynamic programming for discrete-time nonlinear systems. IEEE Trans. Autom. Sci. Eng. **11**(4), 1176–1190 (2014)
15. Wei, Q., Liu, D.: Adaptive dynamic programming for optimal tracking control of unknown nonlinear systems with application to coal gasification. IEEE Trans. Autom. Sci. Eng. **11**(4), 1020–1036 (2014)
16. Liu, D., Wei, Q.: Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems. IEEE Trans. Neural Netw. Learn. Syst. **25**(3), 621–634 (2014)
17. Xu, X., Hou, Z., Lian, C., He, H.: Online learning control using adaptive critic designs with sparse kernel machines. IEEE Trans. Neural Netw. Learn. Syst. **24**(5), 762–775 (2013)
18. Liu, D., Yang, X., Wang, D., Wei, Q.: Reinforcement-learning-based robust controller design for continuous-time uncertain nonlinear systems subject to input constraints. IEEE Trans. Cybern. **45**(7), 1372–1385 (2015)