Data-Driven Based Optimal Output-Feedback Control of Continuous-Time Systems

Zican Li, Tao Wu, Jing Na*, Jun Zhao, Guanbin Gao and Guido Herrmann

***Abstract*—In this paper, we propose a novel method to solve the optimal output-feedback control problem of continuous-time (CT) linear systems based on a data-driven based reinforcement learning (RL). An output-feedback Riccati equation is first derived by further tailoring its counterpart of state-feedback optimal control. Then, based on this modified Riccati equation, we further derive an output Lyapunov function, where only the system output rather than the unknown state is involved. This allows to obtain the optimal output-feedback gain based on the measured output only. Then, an online data-driven based policy iteration is suggested to obtain the feedback gain *K* and matrix *P*. Finally, a simulation example is given to prove the effectiveness of the proposed algorithm.**

***Index Terms*—Optimal control; Output-feedback control, Data-driven, Policy iteration, Riccati equation.**

I. INTRODUCTION

The purpose of controller design is usually to find a control strategy that stabilizes a closed-loop control system. However, in the control system design, stability is usually the minimum requirement [1]. In this respect, optimal control [2, 3] aims at finding a control that minimizes a predefined cost function, besides the stability. Hence, optimal control has always been a major discipline in the control theory.

In traditional optimal control designs, e.g., optimal regulation control problem [4] and optimal tracking control problems [5, 6], the system dynamics are always assumed to be known, and then offline calculations should be carried out to solve the derived optimal equations. For instance, for linear systems, we can get the solution of the well-known linear quadratic regulation (LQR) problem by solving the associated algebraic Riccati equation (ARE). For nonlinear systems, the derived optimal HJB equation is even more difficult to solve though the system dynamics are fully known.

In recent years, many scholars have revisited optimal control designs [7] by incorporating the principle of adaptation into dynamic programming. This idea leads to a new method named approximate dynamic programming (ADP), which was originally proposed by Werbos [8]. In this framework, neural networks (NNs) are trained to approximate the optimal cost function and then adopted for solving optimal control problem [9, 10]. To derive optimal state-feedback controllers for both linear [11-15] and nonlinear systems [16-20], the idea of reinforcement learning (RL) has also been further exploited, which result in some novel policy iteration (PI) algorithms. However, it is known that in most of above ADP methods, only the state-feedback control problem has been addressed, which means that the system states should be fully known or measurable. This assumption is stringent in some practical applications, where only the system output is available or measurable.

In fact, the output-feedback optimal control problem has not been fully addressed in the ADP literature. The work in [21] used an iterative algorithm to solve the output ARE, which can be conducted *offline* only. To achieve online solution of optimal output-feedback control, a recent paper [22] suggested an improved PI-based RL algorithm by further tailoring the idea of integral reinforcement learning (IRL). However, the Lyapunov function used in the IRL is based on the system state; this approach in turn leads to a two-step optimal control synthesis, where an observer [23] has to be used to reconstruct immeasurable system states.

Inspired by the above facts, a novel data-driven based policy iteration approach is suggested in this paper to address the output-feedback optimal control problem, where only the measured system output and the input gain matrix are needed. Firstly, after revisiting the LQR problem, an output-feedback Riccati equation is derived by further tailoring its counterpart of the state-feedback control. Then, based on this modified Riccati equation, we can derive an output Lyapunov function, where only the system output rather than the unknown system state is involved. This equation allows to develop a data-driven based policy iteration algorithm to get the output-feedback gain and the solution of modified output ARE online. In this new method, we do not need the immeasurable system state. Consequently, the observer used in [22] to estimate the immeasurable system states is avoided in this paper, which could improve the computational cost.

The rest of this paper is organized as follows. In Section II, we briefly revisit the standard LQR problem, and then we construct an output Riccati equation based on output-feedback. In Section III, a method of data-driven based policy iteration is developed to solve the optimal output-feedback control gain *K* and matrix *P*. In Section IV, a simulation is given to indicate that the effectiveness of the proposed algorithm. Conclusions are given in Section V.

*This work was supported by National Natural Science Foundation of China (NSFC) under Grant 61573174, and a Newton fund jointly supported by Royal society, UK and NSFC of China under Grant IE150833/61611130213). (*Corresponding author*: *Jing Na*; E-mail: najing25@ 163.com)

Zican Li, Tao Wu, Jing Na, Jun Zhao, Guanbin Gao are with the Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming, 650500 China.

Guido Herrmann is with with the Department of Mechanical Engineering, University of Bristol, BS8 ITR, UK. (E-mail: G.Herrmann@bristol.ac.uk).

II. Problem Formulation

A Preliminaries

The following continuous-time linear system will be studied in this paper

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \tag{1}$$

where $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^p$ are the system state vector and output, and $u \in \mathbb{R}^m$ denotes the control input, respectively. $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ define the system drift dynamics and input matrix, $C \in \mathbb{R}^{n\times p}$ is the output matrix.

The purpose of optimal control problem for system (1) is to find appropriate control action *u* such that the following cost function can be minimized

$$J(x(t)) = \int_t^\infty [x^T(\tau)Qx(\tau) + u(\tau)Ru(\tau)]d\tau \tag{2}$$

where $Q = Q^T \geq 0$ and $R = R^T > 0$ are the weights matrix corresponding to the system state and control input.

Without loss of generality, we suppose that the pair (*A*, *B*) is controllable, and the pair (*A*, *C*) is observable. If the full knowledge of the system states are measurable, the system can be controlled by using the measured system states, which leads to the so-called state-feedback control, i.e. the conventional linear quadratic regulation (LQR) problem, where the state-feedback control action for system (1) can be given as

$$\begin{aligned} u &= -K_x x \\ K_x &= R^{-1}B^T P \end{aligned} \tag{3}$$

where the positive definite matrix *P* is the solution for the following algebraic Riccati equation (ARE)

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \tag{4}$$

By Bellman's optimality principle and the optimal control given in (3), the optimal cost function can be displayed as

$$J(x(t)) = \int_t^\infty x^T(t)(Q + K_x R K_x)x(t)dt = x^T(t)Px(t) \tag{5}$$

Many algorithms have been reported to solve the ARE given in (4), e.g., offline iteration [17], and online integral reinforcement learning [22]. In fact, the state feedback optimal control has been widely addressed in the literature. In particular, many recent attention has been paid on solving this problem by using the idea of ADP.

B Problem Formulation

In contrast to state-feedback control, the output-feedback optimal control for system (1) is more challenging since only the system output *y* can be used for the optimal control implementation. Hence, the aim of this paper is to design an output optimal feedback controller for system (1) to minimize the above cost function (2).

For the static output-feedback control, the control action can be derived directly by the output measurements associated with a constant feedback gain, such that

$$u = -Ky \tag{6}$$

where $K \in \mathbb{R}^{m\times p}$ is a constant gain matrix to be determined.

To synthesis the feedback gain *K* in (6), we first find that the control (6) can be further rewritten as $u = -Ky = -KCx$. Then, substituting (6) into (5), the optimal cost function can be written as

$$J(x(t)) = \int_t^\infty x^T\left(Q + C^T K^T R K C\right)x dt = x^T(t)Px(t) \tag{7}$$

The derivative of (7) regarding time can be calculated as

$$\dot{x}^T Px + x^T P\dot{x} + x^T\left(Q + C^T K^T R K C\right)x = 0 \tag{8}$$

Substituting the output-feedback control (6) into (1), we know that Eq. (8) can be further presented as

$$x^T\left(\left(A - BKC\right)^T P + P\left(A - BKC\right)^T + Q + C^T K^T R K C\right)x = 0 \tag{9}$$

Since the above equation holds for all $x \in \mathbb{R}^n$, then the following output feedback ARE can be obtained

$$\left(A - BKC\right)^T P + P\left(A - BKC\right)^T + Q + C^T K^T R K C = 0 \tag{10}$$

If we can solve the above output ARE (10), then the feedback gain *K* can be derived accordingly. However, since both the unknown matrices *K* and *P* are involved in (10), it could be difficult to solve (10) directly.

In viewing of the state-feedback control (3) and the output feedback control (6), we can have

$$KC = K_x = R^{-1}B^T P \tag{11}$$

due to the fact $u = -Ky = -KCx$ holds mathematically.

Hence, substituting (11) into (10), we can verify that the ARE (10) can be reduced to the standard ARE (4). Hence, based on the analysis given in [24], we know that

Theorem 1: [22] Consider linear system (1), if we can find a control gain matrix *K* fulfilling the matching condition (11) with the standard ARE (4), then the output-feedback control given in (6) is globally optimal.

Proof: The detailed proof has been provided in [24] .

As pointed in above analysis and statements in [22], the existence of the above output-feedback gain *K* depends on the matching condition (11), which may be stringent, i.e., there might be no *K* to satisfy (11) and (4). Hence, we should relax this matching condition. As shown in [22], we could relax the matching condition (11) into the following condition by introducing an extra matrix *L*

$$KC = R^{-1}\left(B^T P + L\right) \tag{12}$$

where *L* is an arbitrary matrix, which is chosen to obtain a feasible control gain. Then similar to the analysis shown in

Theorem 1, we can substitute (12) into ARE (10), such that it follows

$$A^T P + PA + Q - PBR^{-1}B^T P + L^T R^{-1} L = 0 \tag{13}$$

Hence, we have the following results:

Theorem 2: [22] Consider linear system (1), then the output feedback control (6) is stabilizable if and only if:

1) (*A, B*) is stabilizable and (*A, C*) is detectable;
2) There exist matrices *K* and *L* such that the conditions in (12) and (13) are fulfilled, and *P* is the solution of Lyapunov equation (13).

Proof: A similar proof of Theorem 2 was provided in [22] .

Comparing (11) and (4) with (12) and (13), it is found that the matrix *L* shows the difference between the optimal state-feedback control and the optimal output-feedback control. Hence, one can conclude that the solutions of (12) and (13) give a suboptimal output-feedback control *K*. Based on this fact, an offline policy iteration algorithm based on the Lyapunov equation (12) and (13) was suggested in [22].

Algorithm 1 [22]: **Offline Policy Iteration for Output-feedback Control**

1) Start with an admissible control policy K_0 and $L = 0$
2) (**Policy evaluation**) Given a control input gain K_i , find the gain matrix P_i using the equation

$$\left(A - BK_i C\right)^T P_i + P_i \left(A - BK_i C\right) + Q + C^T \left(K_i\right)^T R \left(K_i\right) C = 0 \tag{14}$$

3) (**Policy improvement**) The control policy and the matrix L can be updated by using

$$\begin{aligned} K_{i+1} &= R^{-1}\left(B^T P_i + L_i\right) C^T \left(CC^T\right)^{-1} \\ L_{i+1} &= RK_{i+1} C - B^T P_i \end{aligned} \tag{15}$$

An offline solution for the output-feedback control can be obtained via the above PI algorithm, which demands the precise dynamics of system (1), e.g., system matrices *A* and *B*. If it achieves convergence, then the equation (12) and (13) give necessary and sufficient conditions. Following this observation, the recent work [22] developed an integral RL (IRL) algorithm to obtain the suboptimal output-feedback policy without knowing the concrete knowledge of system drift dynamics *A*. However, the method proposed in [22] requires the system state *x* to be known in the iteration, which is not the case in the output-feedback control designs. Hence, an observer should be developed in [22], which leads to a two-step control implementation with fairly demanding computational costs. In contrary to [22], we will propose an alternative PI algorithm based on the reinforcement learning to find the optimal output-feedback control solution.

III. SOLVING OUTPUT-FEEDBACK OPTIMAL CONTROL VIA DATA-DRIVEN METHOD

Inspired by [11], we will introduce an alternative method to obtain the solution (10) by using data-driven adaptive method the based output *y* rather than the system state *x*. Hence, the proposed approach is clearly different to Algorithm 1. For this purpose, we will first make further manipulations on the equation (10) or (14), which will be given as follows.

Comparing with the Algorithm 1, to avoid the use of system state *x*, we can find that there is a constant matrix $C^T C$, which can be used to manipulate equation (10). For the ease of simple notation, we define $A_{c_i} = A - BK_i C$, and then multiply both sides of the equation by $C^T C$, it follows that

$$C^T C(A_{c_i}^T P_i + P_i A_{c_i}) C^T C = -C^T C(Q + C^T K_i^T R K_i C) C^T C \tag{16}$$

The both sides of equation (16) are further multiplied by system state variable x , such that

$$x^T C^T C(A_{c_i}^T P_i + P_i A_{c_i}) C^T C x = -x^T C^T C(Q + C^T K_i^T R K_i C) C^T C x \tag{17}$$

According to the fact $y = Cx$, the equation (17) can be further represented as

$$y^T C(A_{c_i}^T P_i + P_i A_{c_i}) C^T y = -y^T C(Q + C^T K_i^T R K_i C) C^T y \tag{18}$$

Comparing (18) with (9), it is clearly shown that the system state *x* can be replaced by the output *y*, and thus we can address the optimal gains based on the output *y*, which leads to the exact output-feedback optimal control. Consequently, the required observer to provide the estimate of the system state *x* used in [22] is not necessary. Moreover, as shown in (18), the system input gain *B* does not appear explicitly, which could relax the requirements on the system.

In the following, we will develop an online algorithm to obtain a suboptimal output-feedback solution. Inspired by [11], we apply $\mathrm{vec}(\bullet)$ operator on both sides of (18) and use the Kronecker product. Then, we can further reformulate (18) as

$$\begin{aligned} &2(C^T y \otimes A_{c_i} C^T y)^T vec(P_i) \\ &\quad = -(C^T y \otimes C^T y)^T vec(Q + C^T K_i^T R K_i C) \end{aligned} \tag{19}$$

The equation (19) is established by applying $\mathrm{vec}(\bullet)$ operator and Kronecker product on (18). To obtain the optimal solution *P*, we can further rewrite (18) in a more compact form as

$$\omega_i \chi_i = \xi_i \tag{20}$$

where $\omega_i \in \mathbb{R}^{l \times n^2}$ and $\xi_i \in \mathbb{R}^{l \times 1}$ are known or measured system dynamics, χ_i is an unknown vector of the matrix *P*, which need to be solved. According to (19) and (20), the concrete

formulations of ω_i , χ_i and ξ_i can be given as

$$\begin{aligned} \omega_i(y,A,B,C,K_i) &= 2\left[C^T y \otimes A_{c_i} C^T y\right]^T \\ \chi_i(P_i) &= vec(P_i) \\ \xi_i(y) &= -\left[C^T y \otimes C^T y\right]^T vec(Q + C^T K_i^T R K_i C) \end{aligned} \tag{21}$$

Considering the fact $P_i^T = P_i$, we can further redefine the equation (21) as

$$\varpi_i \overline{\chi}_i = \xi_i \tag{22}$$

where

$$\begin{aligned} \varpi_i &= [\omega_{j1}, \omega_{j2} + \omega_{j5}, \omega_{j3} + \omega_{j9}, \omega_{j4} + \omega_{j13}, \omega_{j6}, \\ &\quad \omega_{j7} + \omega_{j10}, \omega_{j8} + \omega_{j14}, \omega_{j11}, \omega_{j12} + \omega_{j15}, \omega_{j16}] \\ \overline{\chi}_i &= \left[P_{i_{11}}, P_{i_{12}}, P_{i_{13}}, P_{i_{14}}, P_{i_{22}}, P_{i_{23}}, P_{i_{24}}, P_{i_{33}}, P_{i_{34}}, P_{i_{44}}\right]^T \end{aligned} \tag{23}$$

where $\varpi_i \in \mathbb{R}^{j \times \frac{n}{2}(n+1)}$ and $\overline{\chi}_i \in \mathbb{R}^{\frac{n}{2}(n+1)}$. The constant j is given by $l \geq j \geq 1$ and $j \in \mathbb{Z}^+$, where $i \in \mathbb{Z}^+$ is the number of the iteration. Hence, the problem now is to obtain the unknown values P_i involved in (23) online, which can then be used to derive the optimal output-feedback gain K_i and L_i.

Under the condition that ϖ_i has a full column rank, the unknown values $\overline{\chi}_i$ in (22) during each iteration can be determined by

$$\overline{\chi}_i = \left(\varpi_i^T \varpi_i\right)^{-1} \varpi_i^T \xi_i \tag{24}$$

Then based on the idea of reinforcement learning and PI, we can get P_i and L_i through the following Algorithm 2:

Algorithm 2: Online Policy Iteration for Output-feedback Control

1) Set the initial conditions as $L = 0$ and $K = 0$.
2) Get the optimal solution of P_i from equation (24).
3) Update the matrices K_i and L_i using (15).
4) Set $k \leftarrow k+1$, and echo Step 2 and Step 3 until $\|P_i - P_{i-1}\| \leq \sigma$ for a small threshold constant σ.
5) Use the control policy $u = -Ky$ as the optimal control action on the system.

The implementation of the proposed online computational adaptive optimal output-feedback control method given in Algorithm 2 can be detailed in Fig.1.

Remark 1: As shown in the above policy iteration procedure of adaptive optimal tracking control algorithm, we use the input and output data *u*, *y* to online calculate the solution of ARE. In particular, in this section, by using the vec(•) operator and the Kronecker product, the optimal output feedback control policy can be derived online, without using the system state *x*. Hence, we do not need to design an extra observer to solve the optimal control policy.

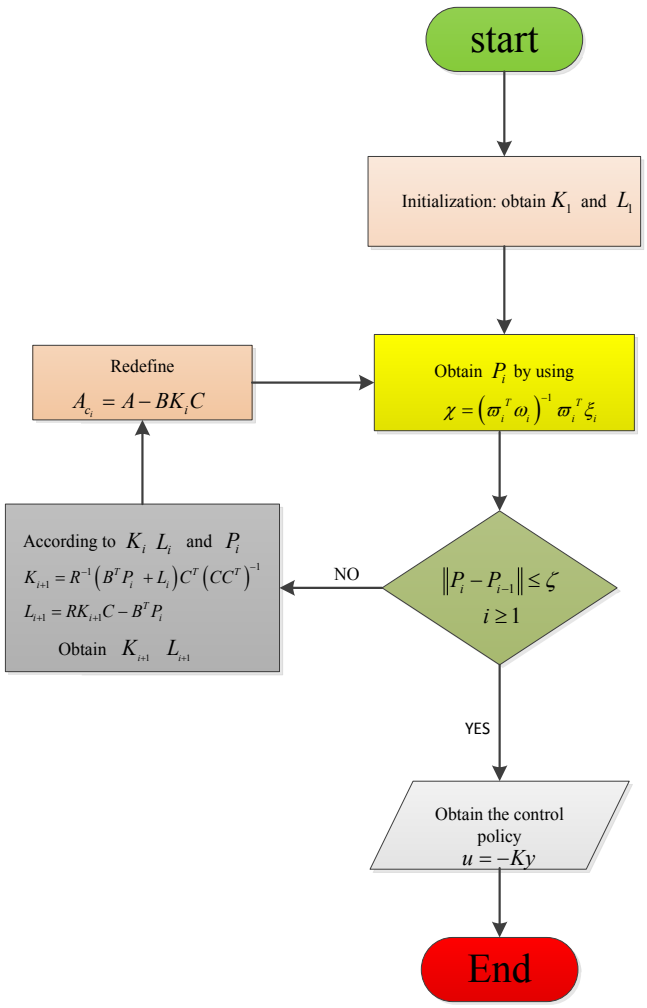

Fig. 1: Implementation of the proposed Algorithm 2.

Next, the convergence of Algorithm 2 will be shown to guarantee under some rank condition.

Lemma 1: [11] For all $l \geq l_0$ with $l_0 > 0$, if the rank of matrix ϖ_i in (22) equals to $\frac{n}{2}(n+1)$, then the basis function vector ϖ_i has full column rank for any $i \in \mathbb{Z}^+$.

Then, following a similar analysis as shown in [11], we can show that by using Algorithm 2 with rank condition in Lemma 1, the sequences $\{P_i\}_{i=0}^{\infty}$ and $\{K_i\}_{i=1}^{\infty}$ obtain from equation (24) will converge to its optimal values P^* and K^*. Moreover, by using the obtained output-feedback control policy K on system (1), we can show that the system state can converge to zero, which will not be detailed due the imposed page limit.

IV. SIMULATION

In this section, a simulation example is given to show the effectiveness of the proposed method for solving the

output-feedback control problem. For this purpose, we consider the continuous-time liner system (1), where the system matrices are given as follows

$$A=\begin{bmatrix} 0.090 & 0.180 & 0.460 & 0 \\ 0.300 & 0.007 & 0.040 & -0.050 \\ -0.010 & -0.100 & -0.090 & 0 \\ 0 & 0.030 & 0 & -0.300 \end{bmatrix}$$

$$B=\begin{bmatrix} 1.420 \\ 0.200 \\ 0.100 \\ 0.1410 \end{bmatrix} \tag{25}$$

$$C=\begin{bmatrix} 5 & 0.049 & 5 & 0.100 \\ 0.010 & 0.200 & 1 & 0 \\ 0.100 & 1 & 0.100 & 1 \end{bmatrix}$$

In order to conduct the simulation, the initial value of the matrix L can be selected as $L=[0\ \ 0\ \ 0\ \ 0]$. The initial value of output-feedback gain K can be chosen as $K=[0\ \ 0\ \ 0]$. Moreover, the weighting matrix Q and R in the cost funciton (2) are given by

$$Q=\begin{bmatrix} 6 & 0 & 0 & 0 \\ 0 & 6 & 0 & 0 \\ 0 & 0 & 1.5 & 0 \\ 0 & 0 & 0 & 15 \end{bmatrix} \tag{26}$$

$$R=11$$

We first use the offline Algorithm 1 to solve this problem. According to equation (14) and (15), the opitmal output-feedback gain K^* can be obtained as

$$K^*=[-0.0238 \quad 0.1527 \quad 0.5289] \tag{27}$$

Note that to obtain the above solutions, we need to carry out the *offline* calculations, and both the system drift dynamics A and input dynamics B should be fully known.

We now use Algorithm 2 to solve the problem. According to equation (22), ϖ_i and ξ_i can be gained based on the measured system output y and input u. Then we can further get $\bar{\chi}=\left[P_{11},P_{12},P_{13},P_{14},P_{22},P_{23},P_{24},P_{33},P_{34},P_{44},\right]^T$, which we need to online calculate based on the measured system output and input.

With the equation (24) and (15) and the proposed Algorithm 2, where the threshold constant is set as $\sigma=1\times e^{-10}$. Then we can obtain the output-feedback gain K and optimal control gain matrix P as

$$K=[0.0701 \quad 0.1398 \quad 0.5347] \tag{28}$$

$$P=\begin{bmatrix} 1.1621 & -3.4362 & 4.5561 & 10.2876 \\ -3.4362 & 5.1383 & 7.3415 & 0.6072 \\ 4.5561 & 7.3415 & -2.8648 & 1.4184 \\ 10.2876 & 0.6072 & 1.4184 & 1.1646 \end{bmatrix}$$

The convergence performance of the output-feedback gain K and optimal control matrix P are given in Fig. 2 and Fig. 3, respectively. As it is shown, the convergence can be obtained after 7 iteration, which can prove the effectiveness of the suggested Algorithm 2.

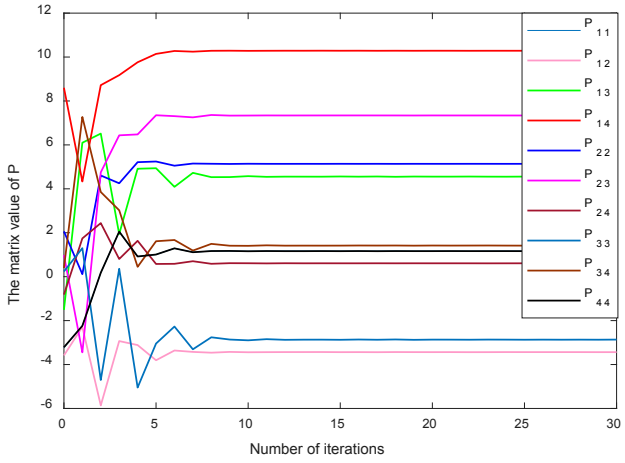

Fig.2 Evolution of the parameters of matrix P.

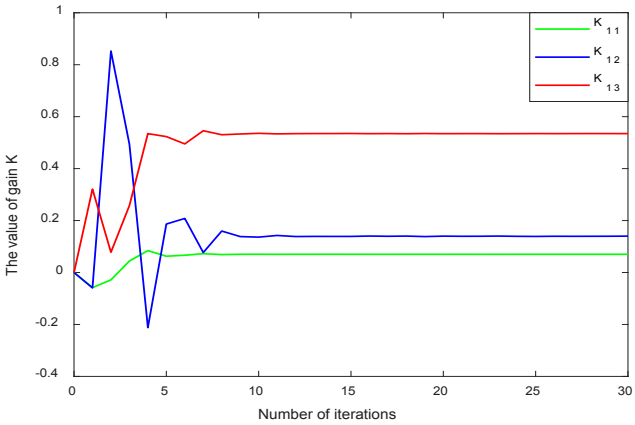

Fig.3 Evolution of the parameters of matrix K.

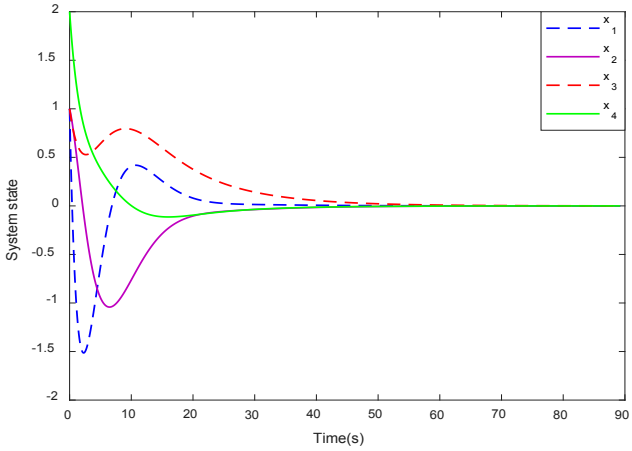

Fig.4 The convergence trajectory of system state.

Comparing (27) with (28), we can find that the obtained solution in (28) by using the online Algorithm 2 is close to the solution given in (27), which is obtained via the offline Algorithm 1.

Finally, Fig. 4 gives the controlled system states with the initial condition $x_0 = [1,1,1,2]^T$. One may find from Fig. 4 that the system states all converge to zero, which validates that the proposed control can guarantee the stability of the closed-loop control system.

V. Conclusion

A new method has been proposed to address the optimal output-feedback control problem of continuous-time linear systems in this paper. Firstly, the output-feedback problem and the state-feedback control has been compared, and then the output-feedback Riccati equation is transformed into an alternative form, which allows to derive its solution without using the system states. Then, we propose a method of data-driven based policy iteration to resolve the derived Riccati equation. The innovation of the proposed algorithm lies in that neither the system state nor the observer design is used in the proposed control algorithm. Finally, a numerical simulation example is given to indicate the effectiveness of the developed method. In our future work, we will study the optimal output-feedback control problem for nonlinear systems.

References

[1] H. Zhang, D. Liu, Y. Luo, and D. Wang, *Adaptive Dynamic Programming for Control*: Springer Publishing Company, Incorporated, 2015.

[2] F. L. Lewis, D. Vrabie, and V. L. Syrmos, "Optimal Control, 3rd Edition," 2012.

[3] A. K. Alhejji and M. R. Sayeh, "Dynamic neural network-observer-based adaptive inverse optimal control design for unknown nonlinear systems," *International Journal of Industrial Electronics & Drives,* vol. 2, p. 10.1504/IJIED.2015.072826, 2015.

[4] D. L. Lukes, "Optimal regulation of nonlinear dynamical systems," *Siam J.contr.optim,* vol. 7, p. 37, 1969.

[5] H. Zhang, Q. Wei, and Y. Luo, "A novel infinite-time optimal tracking control scheme for a class of discrete-time nonlinear systems via the greedy HDP iteration algorithm," *IEEE Trans Syst Man Cybern B Cybern,* vol. 38, pp. 937-42, 2008.

[6] L. Yang, S. Yang, and R. Burton, "Discrete-time H2-optimal output tracking control for an experimental hydraulic positioning control system," *International Journal of Advanced Mechatronic Systems,* vol. 1, 2009.

[7] D. P. Bertsekas, "Dynamic Programming & Optimal Control," vol. 47, 2005.

[8] W. Miller, R. Sutton, and P. Werbos, *A Menu of Designs for Reinforcement Learning Over Time*: MIT Press, 1990.

[9] N. Jing and G. Herrmann, "Online adaptive approximate optimal tracking control with simplified dual approximation structure for continuous-time unknown nonlinear systems," *IEEE/CAA Journal of Automatica Sinica,* vol. 1, pp. 412-422, 2014.

[10] Y. Lv, J. Na, Q. Yang, X. Wu, and Y. Guo, "Online adaptive optimal control for continuous-time nonlinear systems with completely unknown dynamics," *International Journal of Control,* vol. 89, pp. 99-112, 2016.

[11] Y. Jiang and Z. P. Jiang, *Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics*: Pergamon Press, Inc., 2012.

[12] J. Y. Lee, B. P. Jin, and Y. H. Choi, *Integral Q-learning and explorized policy iteration for adaptive optimal control of continuous-time linear systems*: Pergamon Press, Inc., 2012.

[13] F. L. Lewis, H. Modares, A. Karimpour, M. B. Naghibisistani, and B. Kiumarsi, "Reinforcement Q-learning for optimal tracking control of linear discrete-time systems with unknown dynamics," *Automatica,* vol. 50, pp. 1167–1175, 2015.

[14] F. L. Lewis and K. G. Vamvoudakis, "Reinforcement Learning for Partially Observable Dynamic Processes: Adaptive Dynamic Programming Using Measured Output Data," *IEEE Trans Syst Man Cybern B Cybern,* vol. 41, pp. 14-25, 2011.

[15] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Brief paper: Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica,* vol. 45, pp. 477-484, 2009.

[16] 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 Transactions on Neural Networks & Learning Systems,* vol. 23, pp. 1118-1129, 2012.

[17] D. Liu and Q. Wei, "Policy Iteration Adaptive Dynamic Programming Algorithm for Discrete-Time Nonlinear Systems," *IEEE Trans Neural Netw Learn Syst,* vol. 25, pp. 621-634, 2014.

[18] K. G. Vamvoudakis and F. L. Lewis, "Online actor critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica,* vol. 46, pp. 878-888, 2010.

[19] 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 Transactions on Neural Networks & Learning Systems,* vol. 23, p. 1884, 2012.

[20] X. Xu, Z. Hou, C. Lian, and H. He, "Online learning control using adaptive critic designs with sparse kernel machines," *IEEE Transactions on Neural Networks & Learning Systems,* vol. 24, pp. 762-775, 2013.

[21] Y. Jiang and Z. P. Jiang, "Approximate Dynamic Programming for Output Feedback Control," in *CCC*, 2010, pp. 5815-5820.

[22] L. M. Zhu, H. Modares, O. P. Gan, F. L. Lewis, and B. Yue, "Adaptive Suboptimal Output-Feedback Control for Linear Systems Using Integral Reinforcement Learning," *IEEE Transactions on Control Systems Technology,* vol. 23, pp. 264-273, 2014.

[23] F. Abdollahi, H. A. Talebi, and R. V. Patel, "A stable neural network-based observer with application to flexible-joint manipulators," *IEEE Trans Neural Netw,* vol. 17, pp. 118-129, 2006.

[24] J. Gadewadikar, F. L. Lewis, and M. Abukhalaf, "Necessary and Sufficient Conditions for H Static Output-Feedback Control," *Journal of Guidance Control & Dynamics,* vol. 29, pp. 915-920, 2006.