Information Sciences 484 (2019) 84–94

Contents lists available at ScienceDirect

Information Sciences

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

New insight into the simultaneous policy update algorithms related to H_∞ state feedback control

Zhou-Yang Liu, Huai-Ning Wu*

Science and Technology on Aircraft Control Laboratory, School of Automation Science and Electrical Engineering, Beihang University (Beijing University of Aeronautics and Astronautics), Beijing 100191, China

A R T I C L E I N F O

Article history:
Received 19 June 2018
Revised 14 December 2018
Accepted 24 January 2019
Available online 25 January 2019

2010 MSC:
49N90
49J15

Keywords:
Simultaneous policy update algorithm
H_∞ control
Continuous-time algebra Riccati equation
Kantorovich's theorem
Reinforcement learning

A B S T R A C T

In this paper, we present a new insight into the simultaneous policy update algorithm (SPUA) for solving H_∞ control problems. From the perspective of practicability, a checkable criterion is presented for the initialization of SPUAs such that the initial guess can be chosen with clear rules and lie within a local domain of convergence. To further get a better convergence property, a novel initialization is developed such that the starting matrix does not need to lie in a neighborhood of the solution of H_∞ control problems in some cases and its convergence is established rigorously by mathematical induction principle. Subsequently, an improved SPUA is proposed, and its model-free variant based on reinforcement learning (RL) is also able to learn the solution online without any system dynamics. Finally, numerical results illustrate the effectiveness of the proposed methods.

1. Introduction

Over the past decades, numerical and iterative methods to solve optimization problems have received increasing attention in control community [2,21,27], especially for reinforcement learning (RL) and approximate/adaptive dynamic programming (ADP) methodologies. RL is a branch of artificial intelligence (AI) and involves learning from interaction. Sutton and Barto [27] suggested a RL process as how to map situations to actions so as to maximize a numerical reward signal. From the perspective of optimal control, maximizing a reward tends to mean minimizing a cost, which is used frequently in the context of ADP. Hence, RL is also referred to as ADP in control community [10,21,33]. By borrowing from the ideas of RL, many RL-based approaches have been developed to solve the traditional dynamic programming problem in a forward-in-time way and obtain optimal controllers from online data measurements. See, for example, [31,39] for review papers and [5,13,14,17,29,32] for some recently developed results related to H_2 optimal control problems.

However, practical control systems such as servomechanism, industrial processes, etc may be influenced by many factors such as malfunction of the equipment, time-delay, plant uncertainties and types of disturbance [37]. In addition to H_2 optimal performance, the designers also need to take full account of robustness of control systems [26,38]. A fundamental problem of theoretical and practical interest, that lies at the heart of control theory, is the design of H_∞ controllers that yield acceptable performance under the effect of various disturbances [1]. Such a problem has been mathematically well-defined,

* Corresponding author.
E-mail addresses: young_qd@foxmail.com (Z.-Y. Liu), whn@buaa.edu.cn (H.-N. Wu).

https://doi.org/10.1016/j.ins.2019.01.060
0020-0255/

whose aim is to ensure that the energy gain from the disturbances to the objective outputs is less than a prespecified level. Generally speaking, the infinite horizon H_∞ control problem can be casted as a problem of solving H_∞-type AREs for linear systems [4] and Hamilton–Jacobi–Isaacs (HJI) equations for nonlinear systems [28]. In contrast to H_2-type AREs arising from optimal control, the distinctive feature is that the quadratic term in H_∞-type AREs is indefinite in general, which is a source of difficulty in studying such problems.

In the existing results there are two important categories of iterative methods to compute the stabilizing solution of H_∞-type AREs namely, double iterative loops method [9,30] and single iterative loop method [7,11,16,22,35]. The former contains two iterative loops: (1) the outer loop is aimed at reducing the problem of solving an H_∞-type ARE to one of generating successive iterations of solutions of H_2-type AREs, and (2) the inner loop is aimed at the iterative solution of an H_2-type ARE by solving a series of Lyapunov equations. Based on this work, Vrabie and Lewis [30] proposed an online value iteration (VI) based RL algorithm to solve each H_2-type ARE, and obtained the solution of the H_∞-type ARE without requiring the knowledge of internal system dynamics. Notice that these methods can be considered as a two-player zero-sum game [9], with inner loop updating control policies and outer loop updating disturbance policies. However these methods may lead to redundant iterations and result in low efficiency. To overcome the deficiency, Wu and Luo [34,35] proposed the simultaneous policy update algorithm (SPUA), namely the single iterative loop method, where two players updated their policies simultaneously in the same iterative loop, thus the convergence was greatly improved. Based on this work, the SPUA has been widely extended to learn the stabilizing solution online for linear systems [3,7,11,22] and nonlinear systems [12,15,16,19,24,36]. It is worth mentioning that the crucial point of the single iterative loop method is to find an appropriate initial matrix which ensures the local convergence of SPUAs. Such an initial matrix relies on an open ball, whose center is the stabilizing solution of the H_∞-type ARE under consideration. However, for practical problems, it is almost impossible to obtain the stabilizing solution in advance. Hence it's usually useless when one tries to ascertain whether or not for a given initial matrix the iterative process will converge, and also, a question naturally arises: whether a more global scheme exists, which can be convergent for any initial matrix. These questions above motivate us to have a deeper look on the SPUAs.

In this paper, initially we develop an offline iterative algorithm with two novel alternative initializations to obtain the stabilizing solution of the H_∞-type ARE. The first one, as with the result in [35], is based on Kantorovich's theorem and thus is typical of local convergence algorithm, but it yields such checkable conditions that the initial guess guaranteeing the convergence of iteration can be chosen with clear rules. In order to obtain a better convergence, the second one is then developed under some additional conditions. Its properties of monotonicity and convergence are established rigorously by mathematical induction principle. Subsequently a new SPUA is proposed and its online variant based on off-policy RL is also given. The online SPUA does not require any knowledge of the system dynamics and thus is a fully model-free method.

The reminder of the paper is organized as follows. In Section 2, we briefly introduce the standard linear H_∞ control problem for continuous-time systems and the results of original SPUA. Section 3 gives several novel offline iterative algorithms for the solution of H_∞-type ARE and their convergence proofs. A practical online model-free algorithm is provided in Section 4. Numerical examples are given to demonstrate the application of the proposed methods in Section 5. Finally, Section 6 draws conclusion of the paper.

Notations: Throughout this paper, $\mathbb{R}$, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ denote the set of real numbers, n-dimensional Euclidean space and $n\times m$ matrices, respectively. The symbols T is used to denote the transpose and I_n stands for $n\times n$ identity matrix. $\bar{\sigma}(\cdot)$ denotes the maximum singular value of a matrix. $\|\cdot\|$ denotes the Euclidean norm and we also use it to denote the matrix norm $\bar{\sigma}(\cdot)$. The use of upper-case letters for matrices and lower-case letters for vectors means that no confusion should rise. For a symmetric matrix P, $P>(\geq 0)$ means that it is positive definite (positive semidefinite). $\mathbb{P}\triangleq\{P\in\mathbb{R}^{n\times n}|P\geq 0\}$ is a Banach space with the norm $\bar{\sigma}(P)$. $L_2[0,\infty)$ is a space of square integrable function on $[0,\infty)$, i.e., for $\forall w(t)\in L_2(0,\infty)$, $\int_0^\infty \|w(t)\|^2dt<\infty$. The symbol $\otimes$ stands for the Kronecker product. The vec operator transforms a matrix into a vector by stacking its columns one underneath the other. The matrix A is called Hurwitz in the sense that the eigenvalues of the matrix pencil have negative real parts.

2. Problem formulation and preliminaries

This paper considers a class of linear time-invariant (LTI) continuous-time systems described by the following form:

$$\begin{cases}\dot{x}(t)=Ax(t)+B_1u(t)+B_2w(t)\\ z(t)=\begin{bmatrix}Cx(t)\\ Du(t)\end{bmatrix}\end{cases}\tag{1}$$

where $x(t)\in\mathbb{R}^n$ is the state vector, $u(t)\in\mathbb{R}^m$ is the control input, $w(t)\in\mathbb{R}^q$ denotes the external disturbance and $w(t)\in L_2[0,\infty)$, and $z(t)\in\mathbb{R}^p$ is a vector of control output signals related to the performance of the system. Somewhere in the sequel the time variable dependence is omitted, to simplify the notations. A, B_1, B_2, C and D are constant matrices with appropriate dimensions. In addition it is assumed that the pair (A, B_1) is stabilizable and the pair (C, A) has no unobservable modes on the imaginary axis.

The objective of this paper is to design a state feedback controller $u=Kx$ such that the following requirements are satisfied:

1. The closed-loop system is asymptotically stable in the absence of disturbance.

2. The L_2-gain from disturbance input w to objective output z is less than or equal to γ, i.e.,

$$\int_0^\infty (x^TQx + u^TRu)dt \le \gamma^2 \int_0^\infty w^Twdt \tag{2}$$

for all $w \in L_2[0,\infty)$, where $Q = C^TC \ge 0, R = D^TD > 0$ and $\gamma > 0$ is a known scalar measuring the ability of disturbance attenuation.

It was shown in [4] that the requirements above can be satisfied if and only if the ARE

$$\mathcal{F}(P) \triangleq A^TP + PA - PB_1R^{-1}B_1P + \gamma^{-2}PB_2B_2^TP + Q = 0 \tag{3}$$

has a stabilizing solution $P \ge 0$, i.e., $A - B_1R^{-1}B_1^TP + \gamma^{-2}B_2B_2^TP$ is Hurwitz. In this case, the gain matrix of a suitable controller is given by

$$K = -R^{-1}B_1^TP. \tag{4}$$

To find such a stabilizing solution, a Newton iterative sequence $\{P_i\}_{i=0}^\infty$ is constructed in [35] as

$$P_{i+1} = P_i - (\mathcal{F}'_{P_i})^{-1}\mathcal{F}(P_i), \quad i = 0, 1, 2, \ldots \tag{5}$$

where $\mathcal{F}$ is a Fréchet differentiable operator in Banach space $\mathbb{P}$ and $\mathcal{F}'_{P_i}$ is the first Fréchet derivative of $\mathcal{F}$ at a matrix P_i and assumed to be invertible. In view of Theorem 1 in [35], the iteration (5) is equivalent to

$$\bar{A}_i^TP_{i+1} + P_{i+1}\bar{A}_i = -\bar{Q}_i, \quad i = 0, 1, 2, \ldots \tag{6}$$

where $\bar{A}_i = A - B_1R^{-1}B_1P_i + \gamma^{-2}B_2B_2^TP_i$ and $\bar{Q}_i = Q + P_iB_1R^{-1}B_1P_i - \gamma^{-2}P_iB_2B_2^TP_i$.

The success of such an iteration is guaranteed by Theorem 1 in [23], which shows that an open ball $\mathbb{P}^*$ exists with the stabilizing solution P being the center. That is to say, if P_0 is chosen in $\mathbb{P}^*$, the Newton iterative sequence $\{P_i\}_{i=0}^\infty$ exists and converges to the stabilizing solution of ARE (3). In addition, the radius of $\mathbb{P}^*$ is $(2-\sqrt{2})/2MN$ where N is a constant such that $\mathcal{F}'$ is Lipschitz continuous in some region $\mathbb{P}$; that is,

$$\|\mathcal{F}'_X - \mathcal{F}'_Y\| \le N\|X - Y\|, \quad X, Y \in \mathbb{P} \tag{7}$$

and M is determined by

$$\|(\mathcal{F}'_P)^{-1}\| \le M. \tag{8}$$

Generally speaking, the study about the convergence of Newton's method is usually centered on two types: the first type is, based on the information around an initial matrix, to give criteria ensuring the convergence of Newton's method; while the second one is, based on the information around a solution, to find estimates of the radii of convergence balls. Apparently, Theorem 1 in [23] is the latter. However, it makes the SPUA difficult to implement in practice. Concerning the first type of methods, one of the most important results is the celebrated Kantorovich's theorem [6]. This theorem gives a simple and transparent sufficient condition for the existence and uniqueness of solutions of nonlinear operator equations in Banach spaces, which is given as follows:

Lemma 1 [6]**.** *Let $P_0 \in \mathbb{P}$ be such that $\mathcal{F}'_{P_0}$ exists. Assume that*

$$1)\ \|(\mathcal{F}'_{P_0})^{-1}\| \le B^*, \tag{9}$$

$$2)\ \|(\mathcal{F}'_{P_0})^{-1}\mathcal{F}(P_0)\| \le \eta, \tag{10}$$

$$3)\ \|\mathcal{F}'_{P_1} - \mathcal{F}'_{P_2}\| \le K^*\|P_1 - P_2\|,\ P_1, P_2 \in \mathbb{P}, \tag{11}$$

with

$$h = B^*K^*\eta \le \frac{1}{2} \tag{12}$$

are satisfied and that

$$\mathbb{P}_1 = \{P \mid \|P - P_0\| \le \sigma\} \subset \mathbb{P}, \quad \sigma = \frac{1-\sqrt{1-2h}}{B^*K^*}. \tag{13}$$

Then the Newton iterative sequence $\{P_i\}_{i=0}^\infty$ is well defined and converges to P of $\mathcal{F}(P)$ in $\mathbb{P}_1$.

3. Offline SPUAs for the solution of H_∞-type ARE

In this section, several offline iterative algorithms using the perfect knowledge of system dynamics are provided for the solution of ARE (3). Also necessary analyses about the advantages and weaknesses of each of them are included as a reference for the choice of SPUAs in practice.

3.1. Offline SPUAs starting from an initial matrix P_0

In what follows, we show how the basic step of the SPUA can be analyzed using the Kantorovich's theorem. First of all, it is observed from Lemma 1 that the Fréchet derivative $\mathcal{F}'_{P_0}$ is required, which is usually difficult to compute directly. To this end, we arrive at the following definition.

Definition 1 [18]**.** Let $\mathcal{F}$ be a differentiable Fréchet operator of an matrix variable $P \in \mathbb{P}$. The Jacobian matrix of $\mathcal{F}$ at P is given by

$$\mathcal{F}'_P = \frac{\partial \text{vec } \mathcal{F}}{\partial (\text{vec } P)^T}. \tag{14}$$

Observed that the Fréchet derivative $\mathcal{F}'_{P_0}$ can be calculated in the form of the Jacobian matrix of $\mathcal{F}$ at P_0 in actual computation [20], which yields

$$\mathcal{F}'_{P_0} = \frac{\partial \text{vec } \mathcal{F}}{\partial (\text{vec } P)^T}\bigg|_{P=P_0} = \bar{A}_0^T \otimes I_n + I_n \otimes \bar{A}_0^T \tag{15}$$

where P_0 is a candidate initial matrix. It follows from (15) that $\mathcal{F}'_{P_0}$ is inverse if $\bar{A}_0$ is nonsingular. Therefore, there exists a constant B^* such that $\|(\mathcal{F}'_{P_0})^{-1}\| = B^*$.

Next, the Lipschitz constant K^* is immediately obtained that

$$\|\mathcal{F}'_{P_1} - \mathcal{F}'_{P_2}\| \le K^*\|P_1 - P_2\|, \;\; P_1, P_2 \in \mathbb{P}_1 \subset \mathbb{P}, \tag{16}$$

where $K^* = 2\|B_1R^{-1}B_1^T - \gamma^{-2}B_2B_2^T\|$.

Defining $\eta = \|(\mathcal{F}'_{P_0})^{-1}\| \cdot \|\mathcal{F}(P_0)\|$, there exists a suitable initial matrix P_0 such that

$$h = B^*K^*\eta \le \frac{1}{2}. \tag{17}$$

Then the hypotheses of Kantorovich's theorem are satisfied with (13) holds. We are now ready to describe our theorem.

Theorem 1. *Starting with any symmetric matrix P_0 for which $\bar{A}_0$ is nonsingular and (13) and (17) are satisfied, the iterative procedure (6) produces sequences belonging to $\mathbb{P}_1$ that converge to the stabilizing solution P of ARE (3).*

Proof. According to the results in [35], the sequence generated by (6) is equivalent to Newton sequence obtained by (5). Hence, for an initial matrix P_0 defined in Theorem 1, we obtain from Lemma 1 that the sequence $\{P_i\}_{i=0}^{\infty}$ is convergent, i.e., $P_i \to P$, when $i \to \infty$. □

Remark 1. Theorem 1 shows not only that the iterative procedure (6) will converge to a unique solution bus also that a closed ball $\mathbb{P}_1$ in (17) with center P_0 exists. That is, a specified radius is given to guarantee the success of iterative sequence (6) starting from an suitable P_0. Obviously, the conditions of Theorem 1 can be verified easily by calculating the parameters B^*, K^* and η. Thus it provides a checkable condition for choosing a suitable initial matrix. Meanwhile, the original SPUA [35] fails to ascertain whether or not for an initialization the iterative process will converge. Hence, Theorem 1 shows an advantage over the initialization of the SPUA.

It's worth noting that Theorem 1 and the original SPUA are both based on Kantorovich's theorem, thus only a local convergence can be guaranteed; in other words, (13) and (17) may be failed to satisfy if the initial matrix is chosen too far away from the solution of ARE (3). However, in practice, we are always desired to get a better convergence property. In what follows, we will show that in some cases the starting matrix P_0 does not need to lie in a neighborhood of the solution of ARE (3). Before presenting our result, we will first establish the following lemma:

Lemma 2 [8]**.** *Consider the Lyapunov equation*

$$SA + A^TS = W \tag{18}$$

where A and W are given $n \times n$ matrices and W is symmetric. Assume that A is Hurwitz, then:

1. *The Lyapunov Eq. (18) has a unique solution.*
2. *If $W > 0$ then $S > 0$ and, if $W \ge 0$ then $S \ge 0$.*

Theorem 2. *Under the assumption that $B_1R^{-1}B_1^T \ge \gamma^{-2}B_2B_2^T$, letting $\bar{A}_0$ be chosen to be Hurwitz, the iterative procedure (6) determines a Newton iterative sequence $\{P_i\}_{i=1}^{\infty}$ for which $\bar{A}_i$ is Hurwitz for $i = 1, 2, \ldots,$ $P_1 \ge P_2 \ge \ldots,$ $\lim_{i\to\infty} P_i = P$.*

Proof. The proof follows by mathematical induction principle. We know that there exists a P_0 such that $\bar{A}_0$ is Hurwitz, Now it is assumed inductively that we have already defined P_i with $\bar{A}_i$ Hurwitz, it follows from Lemma 2 that (6) has a unique solution P_{i+1}. We are now to show that $\bar{A}_{i+1}$ is Hurwitz. To this end note the following identity which holds for any symmetric matrices X and $\hat{X}$:

$$X(A - SX) + (A - SX)^TX + XSX = X(A - S\hat{X}) + (A - S\hat{X})^TX + \hat{X}S\hat{X} - (X - \hat{X})S(X - \hat{X}), \tag{19}$$

where $S = B_1 R^{-1} B_1^T - \gamma^{-2} B_2 B_2^T$. Letting $X = P$, and $\hat{X} = P_i$, it follows from (3) that

$$P(A - SP_i) + (A - SP_i)^T P + P_i S P_i - (P - P_i) S (P - P_i) + Q = 0. \tag{20}$$

By subtracting (20) from (6), we have

$$(P_{i+1} - P)(A - SP_i) + (A - SP_i)^T (P_{i+1} - P) = -(P - P_i) S (P - P_i). \tag{21}$$

As $\bar{A}_i = A - SP_i$ is Hurwitz and $S \geq 0$, it follows from Lemma 2 that $P_{i+1} \geq P$.

Next, by using (19) again with $X = P_{i+1}$, $\hat{X} = P_i$, it follows from (6) that

$$P_{i+1}(A - SP_{i+1}) + (A - SP_{i+1})^T P_{i+1} + P_{i+1} S P_{i+1} + (P_{i+1} - P_i) S (P_{i+1} - P_i) + Q = 0. \tag{22}$$

Subtracting (20) with P_i replaced by P_{i+1} gives

$$(P_{i+1} - P)(A - SP_{i+1}) + (A - SP_{i+1})^T (P_{i+1} - P) = -(P_{i+1} - P_i) S (P_{i+1} - P_i) - (P_{i+1} - P) S (P_{i+1} - P). \tag{23}$$

Since $P_{i+1} \geq P$ and $S \geq 0$, $A - SP_{i+1}$ is Hurwitz, and then P_{i+1} is stabilizing.

Next, it will be shown that the monotonicity property of the sequence $\{P_i\}_{i=1}^{\infty}$. Associated with (6) gives

$$P_{i+2}(A - SP_{i+1}) + (A - SP_{i+1})^T P_{i+2} + P_{i+1} S P_{i+1} + Q = 0. \tag{24}$$

By subtracting (24) from (22), we have

$$(P_{i+1} - P_{i+2})(A - SP_{i+1}) + (A - SP_{i+1})^T (P_{i+1} - P_{i+2}) = -(P_{i+1} - P_i) S (P_{i+1} - P_i). \tag{25}$$

Since $A - SP_{i+1}$ is Hurwitz, $P_{i+1} - P_{i+2} \geq 0$. Thus $\{P_i\}_{i=1}^{\infty}$ is a nonincreasing sequence of symmetric matrices bounded below by P, and then the limit $P = lim_{i \to \infty} P_i$ exists. □

Remark 2. Theorem 2 replaces the Kantorovich's assumptions with a more concise condition. Once the assumption of Theorem 2 is met, any symmetric matrix P_0 such that A_0 being Hurwitz guarantees that Newton's method converges to the stabilizing solution P of ARE (3). In this situation it's more convenient to be implemented practically than Theorem 1. Observe also that the iterative procedure (6) has a monotonicity property beginning with P_1, and such a fact reflects the first iterative step is capable of making a adjustment to the initial matrix P_0 and thus increases the flexibility of the choice of it. For instance, $P_0 \geq P_1$ and $P_0 \geq 0$ is not necessary. This point is also reflected in Theorem 1, where P_0 only needs to make A_0 nonsingular.

Based on the above analyses, a SPUA with two alternative initializations is given as follows.

Algorithm 1 SPUA starting from an initial matrix P_0.

Initialization: Choose an suitable initial matrix P_0 according to Theorem 1 or Theorem 2, and set $i = 0$.

Iterative process:

1: Solve the following Lyapunov equation for the positive semidefinite solution P_{i+1}.

$$\bar{A}_i^T P_{i+1} + P_{i+1} \bar{A}_i = -\bar{Q}_i, \; i = 0, 1, 2, \ldots \tag{26}$$

where $\bar{A}_i = A - B_1 R^{-1} B_1 P_i + \gamma^{-2} B_2 B_2^T P_i$ and $\bar{Q}_i = Q + P_i B_1 R^{-1} B_1 P_i - \gamma^{-2} P_i B_2 B_2^T P_i$.

2: Set $i = i + 1$ and repeat step 1 until $\|P_i - P_{i-1}\| \leq \varepsilon$ (ε is a small positive real number). Output P_i as the stabilizing solution of ARE (3).

Remark 3. After comparing the two candidate initializations, we conclude that the former based on Theorem 1 is suitable to a more general case, at the price of the local convergence and a heavier computational burden. The latter based on Theorem 2 is easier for implementation and shares a better convergence property, at the expense of a smaller application scope. Fortunately, these two initializations are not exclusive. When the perfect knowledge of system is known, one can verify whether the assumption of Theorem 2 holds. If not, choose Theorem 1 to be the initialization of the SPUA.

3.2. Offline SPUA starting from initial policies

In this part, the SPUA will be generalized to a form of offline algorithms based on policy iteration. For fixed control policy u and disturbance policy w, define the value function as

$$V(x, u, w) = \int_t^{\infty} (x^T Q x + u^T R u - \gamma^2 w^T w) dt. \tag{27}$$

It is well known that the H_∞ control problem could be solved by recasting it as a two-player zero-sum game problem, with the goal obtaining the optimal control policy and the worst case disturbance policy, i.e., a saddle point stabilizing equilibrium (u^*, w^*):

$$u^* = K^* x, \tag{28}$$

$$w^* = L^* x, \tag{29}$$

where $K^* = -R^{-1}B_1^T P$ and $L^* = \gamma^{-2}B_2^T P$. Thus the iterative procedure (26) can be viewed as a process that two persons play to learn the stabilizing solution P of ARE (3). The procedure is described in Algorithm 2.

Algorithm 2 SPUA starting from initial policies.

Initialization: Let $K_0 \in \mathbb{R}^{m\times n}, L_0 \in \mathbb{R}^{q\times n}$ be stabilizing gain matrices (i.e., $\bar{A}_0 = A + B_1K_0 + B_2L_0$ is Hurwitz). Set $i = 0$.
Iterative process:

1: (Policy evaluation). Solve for the real symmetric positive definite solution P_{i+1} of the Lyapunov equation

$$\bar{A}_i^T P_{i+1} + P_{i+1}\bar{A}_i - \gamma^2 L_i^T L_i + K_i^T R K_i + Q = 0. \tag{30}$$

2: (Policy improvement). Update the feedback gain matrix by

$$K_{i+1} = -R^{-1}B_1^T P_{i+1} \tag{31}$$

and the disturbance gain matrix by

$$L_{i+1} = \gamma^{-2}B_2^T P_{i+1}. \tag{32}$$

3: (Output optimal solution). If $\|K_{i+1} - K_i\| \leq \varepsilon$ and $\|L_{i+1} - L_i\| \leq \varepsilon$, stop and output P_{i+1}, K_{i+1} and L_{i+1} as the solution of ARE (3), the optimal feedback gain and the worst case disturbance gain, respectively, else, set $i = i + 1$ and go to step 1.

Remark 4. Algorithm 2 is a policy iteration (PI) procedure. Indeed step 1 evaluates the value P_{i+1} associated with feedback gain matrix K_i and disturbance gain matrix L_i, which is known as policy evaluation. Given this estimated value P_{i+1}, a new pair of gain matrices is improved by step 2, which is known as policy improvement. At each iteration the gain matrices of the two policies are updated simultaneously, that's why Algorithm 2 belongs to one of SPUAs.

Remark 5. The convergence of Algorithm 2 can be guaranteed by modifying Theorems 1 and 2 appropriately. It's straightforward to extend the convergence analysis of Theorem 2 to Algorithm 2 since the initialization is still satisfied by choosing suitable K_0 and L_0. However the result of Theorem 1 is much more challenging to be applied since some additional requirements, i.e., Kantorovich's assumptions need to be satisfied.

Fortunately, an appropriate initialization applying to the case of Theorem 1 can be obtained by the following lemma:

Lemma 3. *Let K_0 and L_0 be some stabilizing gain matrices, there can be found a solution $P_0 \geq 0$ of the following Lyapunov equation*

$$\bar{A}_0^T P_0 + P_0\bar{A}_0 = \gamma^2 L_0^T L_0 - K_0^T R K_0 - Q \tag{33}$$

to satisfy the convergence condition (13) and (17) of Theorem 1.

Proof. K_0 and L_0 being stabilizing implies that A_0 is Hurwitz and also nonsingular, then it follows from Lemma 2 that (29) has a unique solution P_0. In the meantime, such a solution satisfying (13) and (17) can be found by picking appropriate K_0 and L_0. □

4. Online model-free off-policy SPUA for the solution of H_∞-type ARE

Algorithms 1 and 2 are offline approaches, which require the perfect knowledge of system dynamics. In this part, we will develop an online model-free RL algorithm for H_∞ control problem with completely unknown dynamics. First, we assume initial stabilizing gain matrices K_0 and L_0 to be known. Define $V_{i+1} = x^T P_{i+1} x$, $u_i = K_i x$ and $w_i = L_i x$ as the value function, control policy and disturbance policy, respectively, for each iterative step $i \geq 0$.

Then, consider the following closed-loop system composed of (1), the initial feedback control policy $u = u_0$ and the disturbance policy $w = w_0$,

$$\dot{x} = Ax + B_1u_0 + B_2w_0 = Ax + B_1u_i + B_2w_i + B_1(u_0 - u_i) + B_2(w_0 - w_i). \tag{34}$$

Taking the time derivative of V_{i+1} along the trajectories of (30) and performing some manipulations gives

$$\frac{d}{dt}(x^T P_{i+1} x) = -x^T Q_i x - 2(u_0 - K_i x)^T R K_{i+1} x + 2(w_0 - L_i x)^T L_{i+1} x, \tag{35}$$

where $Q_i = Q - \gamma^2 L_i^T L_i + K_i^T R K_i$.

It is worth pointing out that the exact knowledge of system matrices A, B_1 and B_2 are no longer required in (31) and the problem under consideration becomes how to find P_{i+1}, K_{i+1} and L_{i+1} simultaneously using online measurements.

Now, by integrating both sides of (31) on any given interval $[t, t+\delta t]$, we have

$$x^T(t+\delta t)P_{i+1}x(t+\delta t) - x^TP_{i+1}x + 2\int_t^{t+\delta t}(u_0 - K_ix)^TRK_{i+1}xd\tau - 2\int_t^{t+\delta t}(w_0 - L_ix)^TL_{i+1}xd\tau + \int_t^{t+\delta t}x^TQ_ixd\tau = 0. \quad (36)$$

Therefore, we establish the online model-free SPUA as follows:

Remark 6. Algorithm 3 is in fact an off-policy RL method for solving H_∞ control problem with system dynamics completely unknown. It can be observed from Algorithm 3 that each iteration is based on the same online data generated by initial policies u_0 and w_0, which are also called the behavior polices and may be unrelated to the evaluating policies $K_{i+1}x$ and $L_{i+1}x$. Such a way is named off-policy in reinforcement learning. On the other hand, the information about nominal plant is in fact buried in K_{i+1}, L_{i+1} and the online measured data, so does the system uncertainty. Thus Algorithm 3 is not only a completely model-free method but also has robustness.

The next theorem will give the convergence analysis of the Algorithm 3.

Algorithm 3 Online model-free off-policy SPUA.

1: **Initialization**: Collect real system data online for sample set by applying initial polices $u_0 = K_0x + e$ and $w_0 = L_0x + e$, where K_0 and L_0 are stabilizing matrices and e is the exploration signals. Set $i = 0$.
2: **Policy evaluation and improvement**: Solve for P_{i+1}, K_{i+1} and L_{i+1} from equation (36).
3: **iterative process**: If $\|K_{i+1} - K_i\| \le \varepsilon$ and $\|L_{i+1} - L_i\| \le \varepsilon$, stop and output P_{i+1}, K_{i+1} and L_{i+1} as the solution of ARE (3), the optimal feedback gain and the worst case disturbance gain, respectively, else, set $i = i + 1$ and go to Step 2.

Theorem 3. *Let K_0 and L_0 be a pair of stabilizing gain matrices. Then under Theorem 2 or Lemma 3, the sequences of $\{P_i\}_{i=1}^{\infty}$, $\{K_i\}_{i=1}^{\infty}$ and $\{L_i\}_{i=1}^{\infty}$ obtained from (32) converge to the solution of ARE (3), the optimal feedback gain and the worst case disturbance gain, respectively, as $i \to \infty$.*

Proof. From (30) to (32), it follows that P_{i+1}, K_{i+1} and L_{i+1} obtained from (30) to (32) must satisfy (32). In addition, by Lemma 2, the solution P_{i+1} of Lyapunov Eq. (30) is unique, then K_{i+1} and L_{i+1} is also unique. Therefore, according to Theorem 2 or Lemma 3, the sequences of $\{P_i\}_{i=1}^{\infty}$, $\{K_i\}_{i=1}^{\infty}$ and $\{L_i\}_{i=1}^{\infty}$ obtained from (32) in Algorithm 3 are convergent as $i \to \infty$, which completes the proof. □

Remark 7. It is necessary to discuss the differences between Algorithm 3 and other model-free algorithms [3,7,11,22]. The main common feature of them is that, all of them are online PI methods without requiring the perfect system dynamics. Anyway, they are essentially different on the proof of convergence. Existing model-free algorithms mainly rely on the results in [35], i.e., the original SPUA, to prove that they are equivalent to Newton's method. However, the original SPUA is in fact an algorithm starting from an initial matrix P_0 and is true under the Kantorovich's assumptions. Applying such a result to the algorithms starting from a pair of initial polices, just as mentioned in Remark 5, is not straightforward. To overcome this difficulty, we provide an offline SPUA based on given initial policies and its rigorous theoretical guarantee. Then Theorem 3 is proposed to show that the convergence of Algorithm 3 can be guaranteed by Theorem 2 or Lemma 3. Therefore, Algorithm 3 has a more explicit theoretical basis than other model-free PI methods.

Remark 8. Since Algorithm 3 is a model-free method, there exists a problem that the initial gain matrices restrained by Theorem 2 or Lemma 3 can't be verified in advance. Therefore, the initialization has to be determined by trial and error in practice. Till present, it is still a difficult issue in related optimization problems and needs further investigations.

For computational simplicity, the implementation of Algorithm 3 can be done in a least squares scheme via Kronecker product representation. Due to space limitations in the manuscript the details are omitted here, see more in [3,7,11,22].

5. Simulation examples

In this section, we will demonstrate merits of the developed methods by designing an H_∞ controller for the longitudinal dynamics of an aerial vehicle model, which can be written in the following form [25]:

$$\dot{x} = \begin{bmatrix} -1.0189 & 0.9051 \\ 0.8223 & -1.0774 \end{bmatrix}x + \begin{bmatrix} -0.0022 \\ -0.1756 \end{bmatrix}u + \begin{bmatrix} -0.0011 \\ -0.0878 \end{bmatrix}w.$$

The longitudinal dynamics has two states given as $x = [\alpha\ q]^T$, where α is the aircraft angle of attack, q is the pitch rate. The control input is the elevator deflection δ_e and the disturbance w is wind gusts on α and q. For the performance (2), the weight matrices are selected as $Q = C^TC = I$, $R = D^TD = I$ and $\gamma = 5$. The value of stop criterion is set as $\varepsilon = 10^{-8}$.

Table 1
B^*, K^*, η and σ for different initial matrix.

	B^*	K^*	η	h	σ
P_0^1	2.2475	0.0611	0.0148	0.0020	0.0148
P_0^2	2.5046	0.0611	1.7080	0.2612	2.0200
P_0^3	2.0579	0.0611	3.9764	0.4997	7.7597
P_0^4	1.5505	0.0611	11.4502	1.0841	–

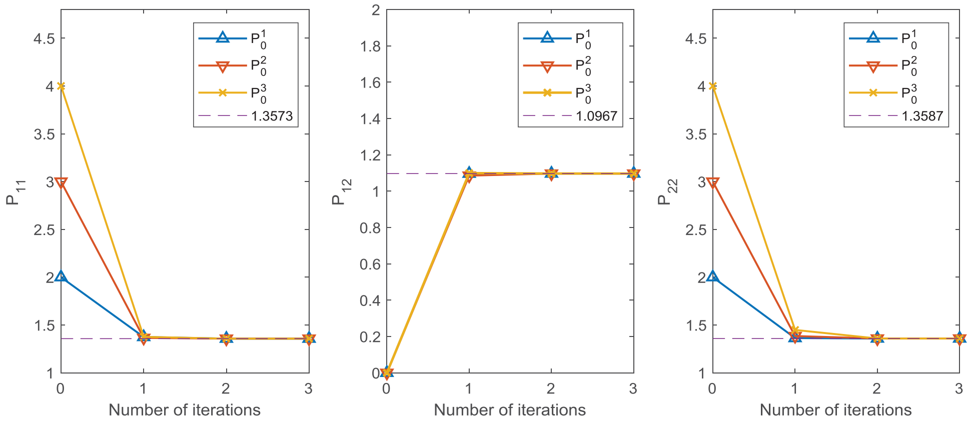

Fig. 1. Convergence of P_{11}, P_{12} and P_{22} to their optimal value with different initial value.

By solving the associated ARE with the Matlab command CARE, we obtain

$$P = \begin{bmatrix} 1.3573 & 1.0967 \\ 1.0967 & 1.3587 \end{bmatrix}. \tag{37}$$

The optimal feedback gain is

$$K = \begin{bmatrix} 0.1956 & 0.2410 \end{bmatrix}, \tag{38}$$

and the worst case disturbance gain is

$$L = \begin{bmatrix} -0.0039 & -0.0048 \end{bmatrix}. \tag{39}$$

5.1. Simulation for Algorithm 1 based on Theorem 1

We use the proposed Theorem 1 to obtain the solution of ARE (3) iteratively. For comparison, four initial matrices are chosen as follows:

$$P_0^1 = \begin{bmatrix} 1.35 & 1.09 \\ 1.09 & 1.35 \end{bmatrix}, \quad P_0^2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad P_0^3 = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}, \quad P_0^4 = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}. \tag{40}$$

According to (9)–(13), the parameters B^*, K^*, η, h and σ are listed in Table 1, where σ defined in (13) is a distance metric between the initial matrix and the stabilizing solution and the symbol '–' means infeasibility.

According to Table 1, it is easy to see that the initial candidate P_0^1, P_0^2 and P_0^3 conform to Theorem 1 since $h = B^*K^*\eta \leq 1/2$. Also, we find that there is a progressive increase in h with σ and this fact means that the initialization of Theorem 1 depends crucially on how close the initial candidate is to a solution. Indeed, as reflected by P_0^4, the initialization is failed if the initial candidate is chosen too far away from the solution of ARE (3).

Next, the results of iterative process are shown in Fig. 1, where p_{ij} denotes the (i, j)th block of P. It can be seen that the exact solution is achieved after 3 iterations and the first step of iteration plays an important role in forcing the iterative value to the stabilizing solution. In fact, the parameter η as reflected in Table 1 measures the step size of the first iteration [20].

5.2. Simulation for Algorithm 1 based on Theorem 2

In this case we will illustrate the convergence of Theorem 2. First it is easy to verify that the assumption of Theorem 2 is satisfied. The results of iteration process are then given with different initial matrix P_0, as depicted in Fig. 2. In this case,

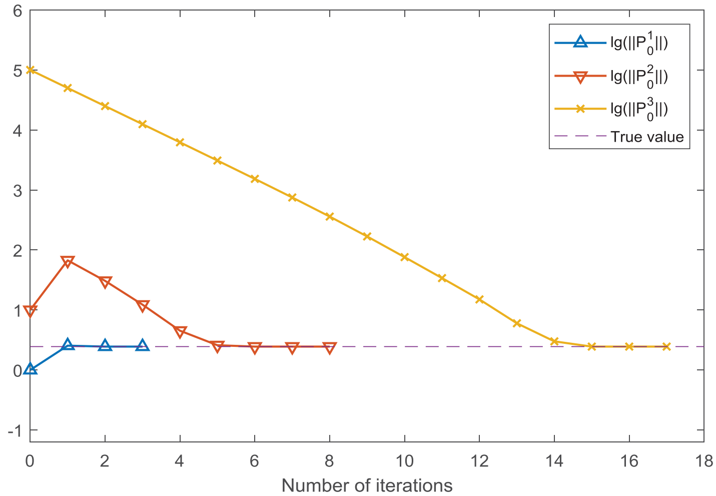

Fig. 2. Convergence of P_0^1, P_0^2 and P_0^3 to their optimal value.

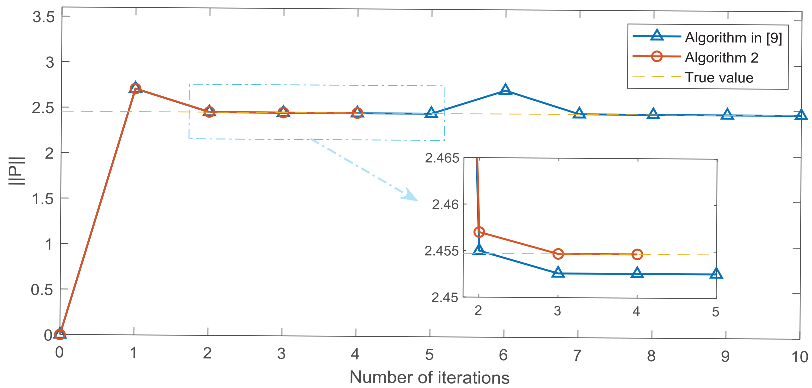

Fig. 3. Convergence of P_0 to its optimal value.

the following initial matrices are considered:

$$P_0^1 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad P_0^2 = \begin{bmatrix} -10 & 0 \\ 0 & -10 \end{bmatrix}, \quad P_0^3 = \begin{bmatrix} 10^5 & 0 \\ 0 & 10^5 \end{bmatrix}. \tag{41}$$

It can be observed from Fig. 2 that the iterative process starting from P_0^1, P_0^2 and P_0^3 is convergent and reaches the exact solution after 3, 8 and 17 iterations, respectively. Note also that the initialization is effective even if the starting matrix P_0^3 lies far away from the solution of ARE, thus it shows a better convergence property than the result of Wu and Luo [35] and Theorem 1. In addition, it can be observed from P_0^1 and P_0^2 that the initial matrix can be negative semi-definite, or even indefinite, which reflects the flexibility of our method on selecting an initial matrix.

5.3. Simulation for Algorithms 2

Since the aircraft longitudinal dynamics is asymptotically stable, the initial stabilizing gain matrices can be set as $K = [0\ \ 0]$ and $L = [0\ \ 0]$. In this situation the initialization is compatible with the algorithm of Lanzon et al. [9]. Thus it is convenient for their comparison.

Once the initial condition is well defined, the convergence information is depicted as Fig. 3. It can be observed that Algorithm 2 converges to the exact solution after 4 iterations and the method of Lanzon et al. [9] needs more iterations. This is mainly due to the fact that the method of Lanzon et al. [9] has two iterative loops and thus a large computing effort is involved.

5.4. Simulation for Algorithm 3

Next we show the effectiveness of the presented model-free off-policy SPUA. In this case the dynamic matrices A, B_1, and B_2 are assumed to be completely unknown. The initial state and the initial gain matrices are chosen as $x_0 = [5\ \ -2]^T$,

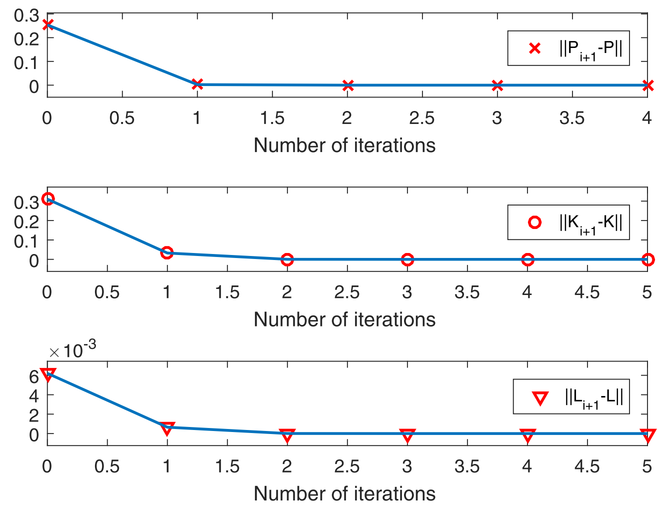

Fig. 4. Convergence of P_{i+1}, K_{i+1} and L_{i+1} to their optimal value P, K and L during the learning process.

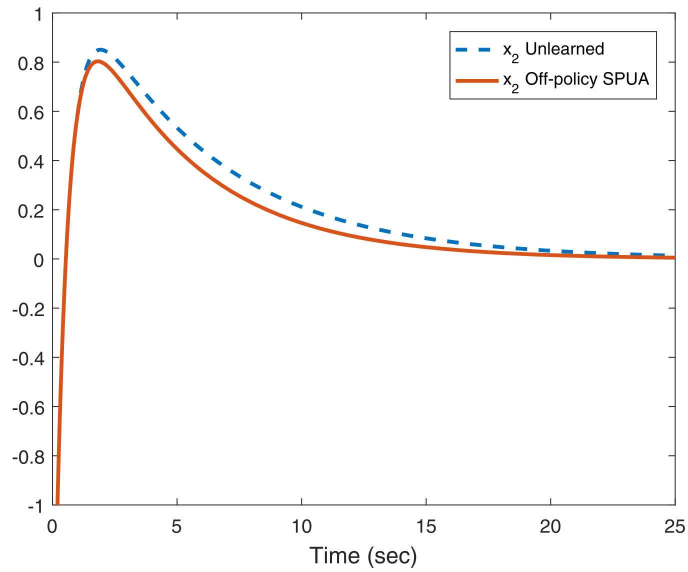

Fig. 5. The system output $y = x_2$ before and after learning.

$K_0 = [0\ 0]$ and $L_0 = [0\ 0]$, respectively. In each iteration, 100 data samples, generated by initial policies along with random noise in interval $[0, 0.1]$, are used to perform the off-policy SPUA. The learned optimal solution is achieved after 4 iterations. The convergence of P_{i+1}, K_{i+1} and L_{i+1} is illustrated in Fig. 4 and the system output $y = x_2$ is plotted in Fig. 5.

6. Conclusion

In this paper, we have introduced three algorithms for solving H_∞ state feedback control problem for LTI continuous-time systems: the offline SPUA starting from an initial matrix P_0, the offline SPUA starting from a pair of initial policies and the model-free off-policy SPUA. Their convergence is established with the help of the Kantorovich's theorem or mathematical induction principle. Moreover, the model-free off-policy SPUA learns the solution of the H_∞-type ARE without requiring any system dynamics. Finally numerical experiments on an aerial vehicle model are made to show the effectiveness of the SPUAs. However, it is necessary to point out that the convergence may occur even if the starting matrix does not satisfy the hypotheses of Theorem 1 or 2. This is mainly due to the fact that the proposed initializations are only sufficient criteria guaranteeing the convergence of iteration. In the future work, we will investigate this issue further and extend our methods to deal with challenging nonlinear, hybrid or time-varying systems.

Acknowledgments

This work was supported in part bythe National Natural Science Foundation for Distinguished Young Scholars of China under grant 61625302 and in part by the National Natural Science Foundation of China under grant 61473011 and grant 61721091. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] T. Başar, P. Bernhard, H_∞ Optimal Control and Related Minimax Design Problems, Birkhäuser Boston, Boston, MA, USA, 2008.
[2] D.P. Bertsekas, Dynamic Programming and Optimal Control, Athena Scientific, Belmont, MA, forth edition, 2007.
[3] Y. Fu, J. Fu, T. Chai, Robust adaptive dynamic programming of two-player zero-sum games for continuous-time linear systems, IEEE Trans. Neural Netw. Learn. Syst. 26 (12) (2015) 3314–3319, doi:10.1109/tnnls.2015.2461452.
[4] M. Green, D.J.N. Limebeer, Linear Robust Control, Prentice Hall, Englewood Cliffs, NJ, USA, 1995.
[5] Y. Jiang, Z.-P. Jiang, Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics, Automatica 48 (10) (2012) 2699–2704, doi:10.1016/j.automatica.2012.06.096.
[6] L.V. Kantorovich, G.P. Akilov, Functional Analysis, Pergamon Press, Oxford, U.K., 1982.
[7] B. Kiumarsi, F.L. Lewis, Z.-P. Jiang, H_∞ control of linear discrete-time systems: Off-policy reinforcement learning, Automatica 78 (2017) 144–152, doi:10.1016/j.automatica.2016.12.009.
[8] P. Lancaster, L. Rodman, Algebraic Riccati Equation, Oxford Univ. Press, Oxford, 1995.
[9] A. Lanzon, Y. Feng, B.D.O. Anderson, M. Rotkowitz, Computing the positive stabilizing solution to algebraic Riccati equations with an indefinite quadratic term via a recursive method, IEEE Trans. Autom. Control 53 (10) (2008) 2280–2291, doi:10.1109/tac.2008.2006108.
[10] F.L. Lewis, D. Liu, Reinforcement Learning and Approximate Dynamic Programming for Feedback Control, John Wiley & Sons, Hoboken, NJ, USA, 2009.
[11] H. Li, D. Liu, D. Wang, Integral reinforcement learning for linear continuous-time zero-sum games with completely unknown dynamics, IEEE Trans. Autom. Sci. Eng. 11 (3) (2014) 706–714, doi:10.1109/TASE.2014.2300532.
[12] B. Luo, T. Huang, H.-N. Wu, X. Yang, Data-driven H_∞ control for nonlinear distributed parameter systems, IEEE Trans. Neural Netw. Learn. Syst. 26 (11) (2015) 2949–2961, doi:10.1109/TNNLS.2015.2461023.
[13] B. Luo, D. Liu, H.-N. Wu, Adaptive constrained optimal control design for data-based nonlinear discrete-time systems with critic-only structure, IEEE Trans. Neural Netw. Learn. Syst. 29 (6) (2018b) 2099–2111, doi:10.1109/TNNLS.2017.2751018.
[14] B. Luo, D. Liu, H.-N. Wu, D. Wang, F.L. Lewis, Policy gradient adaptive dynamic programming for data-based optimal control, IEEE Trans. Cybern. 47 (10) (2017) 3341–3354, doi:10.1109/TCYB.2016.2623859.
[15] B. Luo, H.-N. Wu, Computationally efficient simultaneous policy update algorithm for nonlinear H_∞ state feedback control with galerkin's method, Int. J. Robust Nonlinear Control 23 (9) (2013) 991–1012, doi:10.1002/rnc.2814.
[16] B. Luo, H.-N. Wu, T. Huang, Off-policy reinforcement learning for H_∞ control design, IEEE Trans. Cybern. 45 (1) (2015) 65–76, doi:10.1109/TCYB.2014.2319577.
[17] B. Luo, H.-N. Wu, T. Huang, Optimal output regulation for model-free quanser helicopter with multistep Q-learning, IEEE Trans. Ind. Electron. 65 (6) (2018a) 4953–4961, doi:10.1109/TIE.2017.2772162.
[18] J.R. Magnus, H. Neudecker, Matrix Differential Calculus with Applications in Statistics, John Wiley & Sons, Chichester, England, third edition, 2007.
[19] H. Modares, F.L. Lewis, Z.-P. Jiang, H_∞ tracking control of completely unknown continuous-time systems via off-policy reinforcement learning, IEEE Trans. Neural Netw. Learn.Syst. 26 (10) (2015) 2550–2562, doi:10.1109/tnnls.2015.2441749.
[20] J.M. Ortega, Numerical Analysis: A Second Course, Academic Press, New York, 1972.
[21] W.B. Powell, Approximate Dynamic Programming: Solving the Curses of Dimensionality, John Wiley & Sons, Hoboken, NJ, USA, 2007.
[22] C. Qin, H. Zhang, Y. Luo, Model-free H_∞ control design for unknown continuous-time linear system using adaptive dynamic programming, Asian J. Control 18 (2) (2015) 609–618, doi:10.1002/asjc.1102.
[23] L.B. Rall, A note on the convergence of Newton's method, SIAM J. Numer. Anal. 11 (1) (1974) 34–36, doi:10.1137/0711004.
[24] R. Song, F.L. Lewis, Q. Wei, H. Zhang, Off-policy actor-critic structure for optimal control of unknown systems with disturbances, IEEE Trans. Cybern. 46 (5) (2016) 1041–1050.
[25] B.L. Stevens, F.L. Lewis, E.N. Johnson, Aircraft control and simulation: dynamics, controls design, and autonomous systems, John Wiley & Sons, Hoboken, NJ, USA, three edition, 2015.
[26] X. Su, F. Xia, L. Wu, C.L.P. Chen, Event-triggered fault detector and controller coordinated design of fuzzy systems, IEEE Trans. Fuzzy Syst. 26 (4) (2018) 2004–2016, doi:10.1109/tfuzz.2017.2757459.
[27] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT Press Cambridge, Cambridge, U.K., 1998.
[28] A.J. van der Schaft, L_2-gain analysis of nonlinear systems and nonlinear state-feedback H_∞ control, IEEE Trans. Automat. Contr 37 (6) (1992) 770–784, doi:10.1109/9.256331.
[29] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, F.L. Lewis, Adaptive optimal control for continuous-time linear systems based on policy iteration, Automatica 45 (2) (2009) 477–484, doi:10.1016/j.automatica.2008.08.017.
[30] D. Vrabie, F. Lewis, Adaptive dynamic programming for online solution of a zero-sum differential game, J. Control Theory Appl. 9 (3) (2011) 353–360, doi:10.1007/s11768-011-0166-4.
[31] D. Wang, H. He, D. Liu, Adaptive critic nonlinear robust control: a survey, IEEE Trans. Cybern. 47 (10) (2017) 3429–3451, doi:10.1109/tcyb.2017.2712188.
[32] D. Wang, D. Liu, H. Li, B. Luo, H. Ma, An approximate optimal control approach for robust stabilization of a class of discrete-time nonlinear systems with uncertainties, IEEE Trans. Syst. Man Cybern.-Syst. 46 (5) (2016) 713–717, doi:10.1109/TSMC.2015.2466191.
[33] P. Werbos, Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Science, Ph.D. thesis. Harvard University, 1974.
[34] H.-N. Wu, B. Luo, Neural network based online simultaneous policy update algorithm for solving the HJI equation in nonlinear H_∞ control, IEEE Trans. Neural Netw. Learn. Syst. 23 (12) (2012) 1884–1895, doi:10.1109/TNNLS.2012.2217349.
[35] H.-N. Wu, B. Luo, Simultaneous policy update algorithms for learning the solution of linear continuous-time H_∞ state feedback control, Inf. Sci. 222 (2013) 472–485, doi:10.1016/j.ins.2012.08.012.
[36] X. Yang, D. Liu, B. Luo, C. Li, Data-based robust adaptive control for a class of unknown nonlinear constrained-input systems via integral reinforcement learning, Inf. Sci. 369 (2016) 731–747, doi:10.1016/j.ins.2016.07.051.
[37] S. Zhao, G. Yang, Z. Zhang, C. Chen, Analysis and improvement of the influence of measurement noise on MVC based controller performance assessment, Int. J. Innov. Comput.Inf. Control 14 (2) (2018a) 697–716.
[38] Y. Zhao, Y. Lyu, T. Suzuki, K. Yamada, The parameterization of all robust stabilizing multi-period repetitive controllers for MIMO TD plants with the specified input-output characteristic, Int. J. Innov. Comput.Inf. Control 14 (2) (2018b) 387–403.
[39] Y. Zhu, D. Zhao, Comprehensive comparison of online ADP algorithms for continuous-time optimal control, Artif. Intell. Rev. 49 (4) (2018) 531–547, doi:10.1007/s10462-017-9548-4.