processes

MDPI

Article

Improved Q-Learning Method for Linear Discrete-Time Systems

Jian Chen [1,†], Jinhua Wang [1,2,†] and Jie Huang [1,*]

[1] College of Electrical Engineering and Automation, Fuzhou University, Fuzhou 350108, China; jianchen@fzu.edu.cn (J.C.); fdjinhua_wang@126.com (J.W.)
[2] Fujian Key Laboratory of New Energy Generation and Power Conversion, Fuzhou 350108, China
* Correspondence: jie.huang@fzu.edu.cn
† These authors contributed equally to this work.

Received: 5 February 2020; Accepted: 16 March 2020; Published: 22 March 2020

Abstract: In this paper, the Q-learning method for quadratic optimal control problem of discrete-time linear systems is reconsidered. The theoretical results prove that the quadratic optimal controller cannot be solved directly due to the linear correlation of the data sets. The following corollaries have been made: (1) The correlation of data is the key factor in the success for the calculation of quadratic optimal control laws by Q-learning method; (2) The control laws for linear systems cannot be derived directly by the existing Q-learning method; (3) For nonlinear systems, there are some doubts about the data independence of current method. Therefore, it is necessary to discuss the probability of the controllers established by the existing Q-learning method. To solve this problem, based on the ridge regression, an improved model-free Q-learning quadratic optimal control method for discrete-time linear systems is proposed in this paper. Therefore, the computation process can be implemented correctly, and the effective controller can be solved. The simulation results show that the proposed method can not only overcome the problem caused by the data correlation, but also derive proper control laws for discrete-time linear systems.

Keywords: Q-learning; reinforcement learning; model-free control; optimal control; least squares regression; ridge regression

1. Introduction

This article reconsiders the Q-learning method for quadratic optimal control problem of discrete-time linear systems. Therefore, it is necessary to explain the quadratic optimal control problem of discrete-time linear systems. There is a discrete-time linear system which can be described as $\mathbf{x}(l+1) = \mathbf{A}\mathbf{x}(l) + \mathbf{B}\mathbf{u}(l)$, where $\mathbf{x}(l)$ is states of system, $\mathbf{u}(l)$ is the control value, $\mathbf{A}$ is state matrix, $\mathbf{B}$ is control matrix. The quadratic optimal control problem for the system is to minimize the quadratic performance index function $J = \sum_{i=l}^{\infty}(\mathbf{x}^T(i)\mathbf{Q}\mathbf{x}(i) + \mathbf{u}^T(i)\mathbf{R}\mathbf{u}(i))$, by designing a proper state feedback matrix $\mathbf{K}$ and choosing control law $\mathbf{u}(l) = -\mathbf{K}\mathbf{x}(l)$. According to the optimal control theory, the optimal value of feedback matrix $\mathbf{K}$ should satisfy following equation: $\mathbf{K} = (\mathbf{R} + \mathbf{B}^T\mathbf{P}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{P}\mathbf{A}$, where the matrix $\mathbf{P}$ should be the solution of Riccati equation $\mathbf{P} = \mathbf{A}^T\mathbf{P}\mathbf{A} + \mathbf{Q} - \mathbf{A}^T\mathbf{P}\mathbf{B}(\mathbf{R} + \mathbf{B}^T\mathbf{P}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{P}\mathbf{A}$. So, if matrices $\mathbf{A}$ and $\mathbf{B}$ are known, by setting proper matrices $\mathbf{Q}$ and $\mathbf{R}$, and resolving matrix from Riccati equation, the optimal feedback matrix $\mathbf{K}$ can be solved. It can be seen from the above process that the precondition of the calculation process is the precise acquaintance of controlled systems' mathematical models.

There are great amount of papers which discussed the design of model-free controllers via Q-learning method for linear discrete-time systems, while the independence of data during the computation process is not get enough attention. In this paper, the effect of the data correlation

is discussed, specifically aiming to the computation processes of optimal controllers for linear discrete-time systems adopting Q-learning method. Based on the theoretical analysis, the data independence condition should be satisfied to implement the existing Q-learning method. The data sets sampled from the linear discrete-time systems is destined relevant, which means that the controllers cannot be solved by the existing Q-learning method. To design proper controllers for linear discrete-time systems, an improved Q-learning method is proposed in this paper. The improved method adopts the ridge regression instead of the least squares regression. The ridge regression can deal with the multiple collinearity of the sample sets. Therefore, the proper controllers can be solved by the proposed method.

In this paper, following results have been achieved. First, limited to linear discrete-time systems, the computation process cannot be executed correctly by the existing Q-learning method. Second, relative corollaries have been made for several types of systems, to indicate in detail the necessary for data independence condition for different situations. Third, an improved Q-learning method has been proposed to solve the problem of multiple collinearity. At last, results of simulations show the effectiveness of the proposed method.

2. Q-Learning Method for Model-Free Control Schemes

Generally, there is an obvious difference between optimal control [1] and adaptive control [2,3]. Adaptive controller often needs online dynamic data of systems. On the contrary, optimal controllers are designed off-line by solving Hamilton-Jacobi-Bellman (HJB) equations. In optimal theory, the solution of HJB equations is the real valued function with minimum cost for specific dynamic system and corresponding cost function. Therefore, the solution of HJB equations is also the optimal controller of the controlled system. In the HJB equations, all information of the controlled system should be known, which means the precise mathematical model of the system should be established. To solve HJB equations, many methods have been researched, and the earlier mentioned Riccati equation is derived from HJB equations. According former expoundation, the optimal controller cannot be designed under model-free condition. However, the development of theory made the combination of the two different control schemes. The reinforcement learning theory plays an important part in the process.

As one of the most important reinforcement learning methods, Q-learning has been proved by Watkins [4] that under certain conditions the algorithm converges to the optimum cation-values. Therefore, the calculation of optimal controllers became realizable for complete or partial model-free systems by adopting online calculation method. With data sets sampled from systems, optimal controllers can be designed by Q-learning method. Many researchers took the view and many achievements have been made. Compared with other popular techniques such as SVM and KNN [5], reinforcement learning methods represented by Q-learning are unsupervised algorithms. They do not need the target information of the train data, on the contrary, they evaluate their actions by the value functions aiming to the current status. Their directions of calculation are decided by the value functions, in this sense, the reinforcement learning methods are also gradient-free optimization methods [6] when they are used to solve optimization problems. Some optimal estimation algorithms such as Kalman filter and its extended forms [7] have been adopted by many researchers as observers in adaptive control schemes. Their observation results still rely on the structure and parameters of systems, so these algorithms are beyond the boundary of model-free methods.

Bradtke used the Q-learning method to solve quadratic control problem for discrete-time linear systems [8]. He set up the model-free quadratic optimal control solution based on data, and studied the convergence of Q-learning method for certain type of systems. From then on, the important position of Q-learning in model-free quadratic optimal control is established. And the Q-learning schemes for the continuous-time systems also had been proposed [9,10], without the convergence condition.

As a kind of typical data-based iterative algorithm, Q-learning method has been used to solve problems in many fields, including but not limited to following examples. An adaptive Q-learning

algorithm is adopted to obtain the optimal solution between exploration and exploitation of electricity market [11]. An effective Q-learning algorithm is proposed to solve the traffic signal control problem [12], and satisfying results have been obtained. A distributed Q-learning (QD-learning) [13] algorithm is investigated to get the solution of the sensor networks collaboration problem. A time-based Q-learning algorithm is adopted to get the optimal control of the energy systems [14,15].

Compared with other fields, the quadratic optimal control problems attract main attention in this paper. To design feedback controllers for discrete and continuous-time systems, the principles of using reinforcement learning method are fully described [16]. According to the paper, reinforcement learning method of policy iteration and value iteration can be used to design adaptive controllers which converge to the optimal control theory. Adopting Q-learning schemes for adaptive linear quadratic controllers [17], the difference between linear and nonlinear discrete-time systems has been compared. Simulations show that controllers for both kind of the systems can be solved with almost same convergence speed. Some researchers focused on output feedback control schemes [18–21]. The method to design optimal or nearly optimal controllers has been studied for linear discrete-time systems subject to actuator saturation condition [21–24]. Other researchers considered stochastic linear quadratic optimal control as an effective method [25–28].

3. Problem Description and Improved Method

Theoretically, the existing Q-learning algorithm can calculate optimal or approximate optimal controllers as a model-free method. For discrete-time linear systems, there is an obvious limitation caused by multi-collinearity of data sets sampled from systems, which is shown by analysis. To solve the problem, the existing Q-learning algorithm need to be modified, as the same situation as the existing stability analysis method for uncertain retarded systems [29]. By adopting ridge regression, an improved algorithm has been proposed in this paper, which can overcome the multi-collinearity problem and obtain the proper controller. To explain the limitation of the existing Q-learning method, the design principle of the existing Q-learning model-free quadratic optimal controllers is shown in following section.

3.1. Design Process of Quadratic Optimal Controller by Existing Q-Learning Method

To design a quadratic optimal controller, a linear discrete-time system is considered to be:

$$x(l+1) = \mathbf{A}x(l) + \mathbf{B}u(l) \tag{1}$$

where $x \in R^n$ is a vector of state variables, $u \in R^p$ is control input vector, $\mathbf{A}$ is $n \times n$ state matrix, $\mathbf{B}$ is input $n \times p$ matrix. And $(\mathbf{A}, \mathbf{B})$ is a complete controllability pair. There is the single step performance index is expressed as:

$$v(l) = x^T(l)\mathbf{Q}x(l) + u^T(l)\mathbf{R}u(l) \tag{2}$$

where $\mathbf{Q} = \mathbf{Q}^T \geq 0$ is $n \times n$ weight matrix, $\mathbf{R} > 0$ is $p \times p$ weight matrix, $(\mathbf{Q}^{\frac{1}{2}}, \mathbf{A})$ is complete observability pair, $0 \leq \gamma \leq 1$ is discount factor. Please note that:

$$\mathbf{V}(x(l)) = \sum_{i=1}^{\infty} \gamma^{i-1} v(i) \tag{3}$$

The problem of quadratic optimal control can be described as follows. For system formed as Equation (1), a control law $u = \mathbf{K}x$ should be designed to minimize $\mathbf{V}(x(0)) = \sum_{i=1}^{\infty} \gamma^{i-1} v(i)$. Based on Formula (3), $\mathbf{V}(x(0))$ means the total performance index for infinite time quadratic control systems. To minimize the total performance index, the Q function is defined as:

$$Q(x(l), u(l)) = v(l) + \gamma \mathbf{V}(x(l+1)) \tag{4}$$

Formula (4) shows that the Q function is not a new function but just another expression of total performance index $\mathbf{V}(x(l))$. Then the Q function can be calculated as following formula:

$$\begin{aligned} Q(x(l),u(l)) &= \begin{bmatrix} x(l) & u(l) \end{bmatrix}^T \mathbf{H} \begin{bmatrix} x(l) & u(l) \end{bmatrix} = \begin{bmatrix} x(l) & u(l) \end{bmatrix}^T \begin{bmatrix} \mathbf{H}_{xx} & \mathbf{H}_{xu} & \mathbf{H}_{ux} & \mathbf{H}_{uu} \end{bmatrix} \begin{bmatrix} x(l) & u(l) \end{bmatrix} \\ &= \begin{bmatrix} x(l) & u(l) \end{bmatrix}^T \begin{bmatrix} \mathbf{Q}+\gamma\mathbf{A}^T\mathbf{P}\mathbf{A} & \gamma\mathbf{A}^T\mathbf{P}\mathbf{B} & \gamma\mathbf{B}^T\mathbf{P}\mathbf{A} & \mathbf{R}+\gamma\mathbf{B}^T\mathbf{P}\mathbf{B} \end{bmatrix} \begin{bmatrix} x(l) & u(l) \end{bmatrix} \end{aligned} \tag{5}$$

The feedback law can be calculated by following formula according to the extremum condition of $Q(x(l),u(l))$,which is $\partial Q/\partial u = 0$.

$$\mathbf{K} = -\mathbf{H}_{uu}^{-1}\mathbf{H}_{ux} = -\gamma(\mathbf{R}+\mathbf{B}^T\mathbf{P}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{P}\mathbf{A} \tag{6}$$

where $\mathbf{P}$ is the solution of Riccati equation:

$$\mathbf{P} = \gamma\mathbf{A}^T\mathbf{P}\mathbf{A} + \mathbf{Q} - \gamma\mathbf{A}^T\mathbf{P}\mathbf{B}(\mathbf{R}+\gamma\mathbf{B}^T\mathbf{P}\mathbf{B})\mathbf{B}^T\mathbf{P}\mathbf{A} \tag{7}$$

And $V(x(l))$ should satisfy following equation: $V(x(l)) = x^T(l)\mathbf{P}x(l)$.

The design method of model-based controller is that the control law will be established by solving Equation (7). According to the optimal control theory, the values of the matrices and will decide the different weights between the quantity of states and control values [26]. The values of matrices of Q and R often depend on performance index, experience of designers and constraint conditions. The quadratic optimal controller based on Q-learning is model-free, and shown as follows. Equation (5) is represented as:

$$\begin{aligned} Q(x(l),u(l)) &= \begin{bmatrix} x(l) & u(l) \end{bmatrix}^T \mathbf{H} \begin{bmatrix} x(l) & u(l) \end{bmatrix} = \overline{x}^T\mathbf{H}\overline{x} \\ &= \sum_{i=1}^{n+p} h_{ij}\overline{x}_i^2 + \sum_{i=1}^{n+p}\sum_{j=i+1}^{n+p} 2h_{ij}\overline{x}_i\overline{x}_j = \varphi^T(l)\mathbf{\Theta}(\mathbf{H}) \end{aligned} \tag{8}$$

where $\varphi^T = \begin{bmatrix} \overline{x}_1^2 & \overline{x}_1\overline{x}_2 & \cdots & \overline{x}_1\overline{x}_{n+p} & \overline{x}_2^2 & \overline{x}_2\overline{x}_3 & \cdots & \overline{x}_2\overline{x}_{n+p} & \cdots & \overline{x}_{n+p}^2 \end{bmatrix}$, and the parameters vector $\mathbf{\Theta}(\mathbf{H}) = \begin{bmatrix} h_{11} & 2h_{12} & \cdots & 2h_{1(n+p)} & h_{22} & 2h_{23} & \cdots & 2h_{2(n+p)} & \cdots & h_{(n+p)(n+p)} \end{bmatrix}$. The Q function possesses the recurrence relation shown by following formula:

$$Q(x(l),u(l)) = v(l) + \gamma Q(x(l+1),u(l+1)) \tag{9}$$

Therefore, the following formula can be deduced:

$$(\varphi(l) - \gamma\varphi(l+1))^T\mathbf{\Theta}(\mathbf{H}) = v(l) \tag{10}$$

There are N equations formed by Formula (10) can be built if $\mathbf{N}$ sets of data on state and control have been collected. Please note that: $\Phi = \begin{bmatrix} (\varphi^1(l) - \gamma\varphi^1(l+1))^T \vdots (\varphi^N(l) - \gamma\varphi^N(l+1))^T \end{bmatrix}$

With the least squares method, the solution of the parameters vector $\mathbf{\Theta}(\mathbf{H})$ can be calculated as following formula when the sufficient excitation condition of system has been satisfied.

$$\mathbf{\Theta}(\mathbf{H}) = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T v \tag{11}$$

According to the value of $\mathbf{\Theta}(\mathbf{H})$, the feedback matrix $\mathbf{K}$ can be obtained by Formula (6), and the corresponding control law can be solved.

The complete Q-learning method is an iterative procedure, and it can be fulfilled by the following steps:

- Step 1: An initial feedback matrix $\mathbf{K}_0$ is set to control the system;

- Step 2: A set of sufficient excited data are obtained;
- Step 3: The least squares method is used to solve $\boldsymbol{\Theta}(\mathbf{H})$;
- Step 4: The new control law is calculated by Formula (6);
- Step 5: The process from step 1 to step 4 will be repeated until $\mathbf{K}$ converges to a stable value.

According to Formula (11), the least squares regression solution has been adopted by the existing Q-learning method to solve $\mathbf{H}$. The Q-learning method for quadratic optimal problem should be a kind of completely or at least partially model-free control. Theoretically, the data sampled from systems are the only needed conditions during the learning procedure instead of the structure or parameters of systems. For linear discrete-time systems, the matrix $\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi}$ is singular. Therefore, the least squares regression calculation cannot be implemented.

3.2. Analysis for the Multi-Collinearity of Data Sampled from Linear Discrete-Time Systems

An assumed system defined by Formula (1) has n states and p inputs. Then $\boldsymbol{\Phi}$ of Formula (11) is a matrix composed by n rows and $(n+p)(n+p+1)/2$ columns from n sets of data. When a system with 2 states and 1 input is assumed, the size of matrix $\boldsymbol{\Phi}$ is $l \times 6$ for l sets of data. The ith row of $\boldsymbol{\Phi}$ is expressed as following form:

$$\boldsymbol{\Phi}^T(i) = \begin{bmatrix} (x_1^i(l^i))^2 - \gamma(x_1^i(l^i+1))^2 \\ x_1^i(l^i)x_2^i(l^i) - \gamma x_1^i(l^i+1)x_2^i(l^i+1) \\ x_1^i(l^i)u^i(l^i) - \gamma x_1^i(l^i+1)u^i(l^i+1) \\ (x_2^i(l^i))^2 - \gamma(x_2^i(l^i+1))^2 \\ x_2^i(l^i)u^i(l^i) - \gamma x_2^i(l^i+1)u^i(l^i+1) \\ (u^i(l^i))^2 - \gamma(u^i(l^i+1))^2 \end{bmatrix} \tag{12}$$

For the quadratic optimal state control problems, there is $u = \mathbf{K}x$. Therefore, Formula (12) can be represented as:

$$\boldsymbol{\Phi}^T(i) = \begin{bmatrix} (x_1^i(l^i))^2 - \gamma(x_1^i(l^i+1))^2 \\ x_1^i(l^i)x_2^i(l^i) - \gamma x_1^i(l^i+1)x_2^i(l^i+1) \\ k_1[(x_1^i(l^i))^2 - \gamma(x_1^i(l^i+1))^2] + k_2[x_1^i(l^i)x_2^i(l^i) - \gamma x_1^i(l^i+1)x_2^i(l^i+1)] \\ (x_2^i(l^i))^2 - \gamma(x_2^i(l^i+1))^2 \\ k_1[x_1^i(l^i)x_2^i(l^i) - \gamma x_1^i(l^i+1)x_2^i(l^i+1)] + k_2[(x_1^i(l^i))^2 - \gamma(x_1^i(l^i+1))^2] \\ k_1^2[(x_1^i(l^i))^2 - \gamma(x_1^i(l^i+1))^2] + 2k_1k2[x_1^i(l^i)x_2^i(l^i) - \gamma x_1^i(l^i+1)x_2^i(l^i+1)] + \cdots \; k_2^2[(x_2^i(l^i))^2 - \gamma(x_2^i(l^i+1))^2] \end{bmatrix} \tag{13}$$

Shown by Formula (13), the third element of row vector $\boldsymbol{\Phi}(i)$ is a linear combination of the first and the second element; the fifth element is a linear combination of the second and the fourth element; the sixth element is a linear combination of first, the second and the forth element. Therefore, the matrix $\boldsymbol{\Phi}$ is column-related. Generally, for n order linear discrete-time systems, in the $(n+p)(n+p+1)/2$ columns of matrix $\boldsymbol{\Phi}$, the number of linearly independent columns is no more than $n(n+1)/2$. Thus, it is unable to solve the $(n+p)(n+p+1)/2$ parameters directly by the least squares method.

Following corollaries can be deduced by the analysis.

Corollary 1 (Linear discrete-time systems)**.** *For the linear discrete-time systems, the quadratic optimal controllers cannot be solved by the Q-learning algorithm adopting the least squares method shown as Formula (11).*

Corollary 2 (Linear continuous-time systems)**.** *For the linear continuous-time systems described as $\dot{x} = Ax + Bu$, the quadratic optimal controllers cannot be solved by the Q-learning algorithm adopting the least squares method which has the similar form to Formula (11).*

For the mentioned continuous systems, the quadratic performance index is: $V(x(t)) = \int_t^\infty e^{-\gamma(\tau-t)}(x^TQx + u^TRu)dt$. If Q function is defined as:

$$\begin{aligned} Q(x(t),u(t)) &= \int_t^{t+T} e^{-\gamma(\tau-t)}(x^TQx + u^TRu)d\tau + e^{-\gamma T}Q(x(t+T),u(t+T)) \\ &= R(x(t),u(t)) + e^{-\gamma T}Q(x(t+T),u(t+T)) \end{aligned} \tag{14}$$

Therefore, the Q function can be represented as:

$$Q(x(t),u(t)) = \begin{bmatrix} x(t) \\ u(t) \end{bmatrix}^T \mathbf{H} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} = \varphi^T(t)\boldsymbol{\Theta}(\mathbf{H}) \tag{15}$$

where $\varphi^T(t)$ and $\boldsymbol{\Theta}(\mathbf{H})$ are similar to the definitions in Formula (8). With the state feedback control law $u = \mathbf{K}x$, following equation is insoluble by the existing Q-learning method based on data.

$$(\varphi(t) - e^{-\gamma T}\varphi(t+T))\boldsymbol{\Theta}(\mathbf{H}) = \mathbf{R}(x(t),u(t)) \tag{16}$$

3.3. Improved Q-Learning Method Adopting Ridge Regression

To solve the multi-collinearity problem caused by matrix $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$, an improved Q-learning method adopting ridge regression has been proposed in this paper. Both ridge regression and generalized inverse are common methods for inversion of singular or ill-conditioned matrices [30–32]. According to former studies [32,33], the principles of the two methods are different. With generalized inverse method, the order of model built by matrix $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ will be reduced. In another word, the generalized inverse method is fitter for overparameterization [34] than general irreversible situation. While the ridge regression method maintains the original model order and directly interferes the main diagonal elements of the matrix $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$, the regression error will be reduced. Therefore, for the multi-collinearity of matrix $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$, the ridge regression is the better solution method.

As an improved least squares method, ridge regression was proposed by Hoerl in 1970. It is an effective supplement for the least squares regression. Ridge regression obtains higher estimate accuracy than the least squares regression, in the meantime, it is a kind of biased estimation method.

The principle of ridge regression is described as follows. There will be $\left|\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right| = 0$ or $\left|\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right| \approx 0$ when $\boldsymbol{\Phi}$ is a rank-deficient matrix. Therefore, the result of Formula (11) will be unsolved or very unstable. A positive matrix $\lambda\mathbf{I}$ is added with $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ to make a new matrix $\left(\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \lambda\mathbf{I}\right)$, where $\mathbf{I}$ is an unit matrix with the same dimensions as $\boldsymbol{\Phi}$, and $\lambda > 0$. As a full rank matrix, $\left(\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \lambda\mathbf{I}\right)$ is invertible. According to Formula (11), $\Theta(\mathbf{H})$ can be solved by following formula when ridge regression is adopted, where λ is a minimal positive value.

$$\Theta(\mathbf{H}) = (\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \lambda\mathbf{I})^{-1}\boldsymbol{\Phi}^T\nu \tag{17}$$

The principle of ridge regression is simple and understandable. There are many kinds of selection method for parameter λ, because the most optimal value of λ is dependent on the estimated model. In simulations, the optimized λ can be selected by calculating the ridge trace.

The state feedback control law $u = \mathbf{K}x$ can be obtained based on the $\Theta(\mathbf{H})$. The improved Q-learning method is still an iterative procedure, which can be realized as following steps.

- Step 1: An initial feedback matrix $\mathbf{K}_0$ is given to control the system;
- Step 2: A set of sufficient excited data are obtained;
- Step 3: The $\Theta(\mathbf{H})$ is solved based on the ridge regression method Formula (14);
- Step 4: The new control law is calculated by Formula (6);
- Step 5: The process from step 1 to step 4 will be repeated until $\mathbf{K}$ converges to a stable value.

4. Simulations

To make numerical exposition, two simulation examples are shown in following part. Both examples are linear discrete-time system. The simulations show the effectiveness of the improved algorithm.

4.1. Example 1

Example 1: In many optimal control papers the objects are one-order or two-order systems. To show the effectiveness of the proposed method on high order systems, a three-order linear discrete-time system is chosen as following form:

$$\dot{x} = \begin{bmatrix} 1 & 0.0997 & 0.0045 \\ 0 & 0.9909 & 0.0861 \\ 0 & -0.1722 & 0.7326 \end{bmatrix} x + \begin{bmatrix} 0.7502 \\ -0.4955 \\ 0.0861 \end{bmatrix} u$$

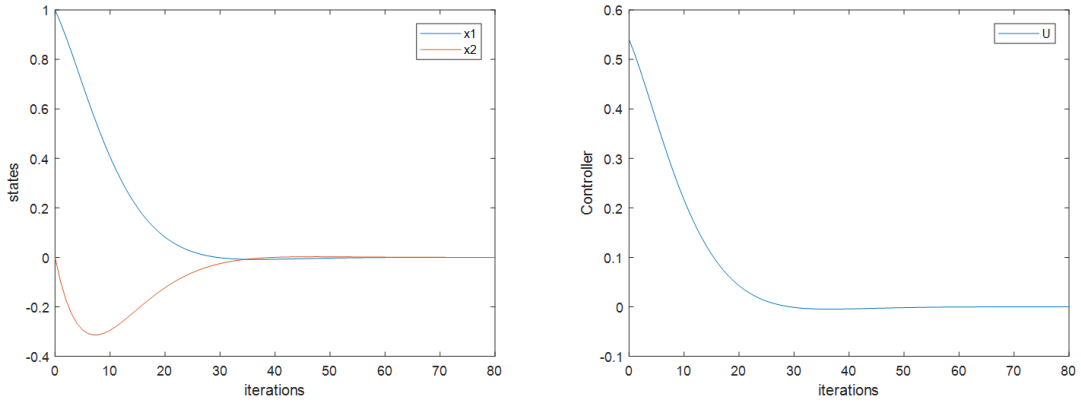
The weight matrices of performance criteria are: $\mathbf{Q} = \mathbf{I}$, $\mathbf{R} = 1$. The discount factor is $\gamma = 1$. The initial feedback matrix is set as $\mathbf{K} = \begin{bmatrix} 0.5 & 0 & 0 \end{bmatrix}$, and then data can be acquired by solving state equation. The data acquisition process is shown as follows.

To guarantee the sufficiency of excitation, three random numbers will be produced by program as state $x(l)$ at the lth sample time; $u(l)$ will be calculated according to values of $x(l)$; $x(l+1)$ can be derived by state equation, and also the corresponding $u(l+1)$ will be obtained.

For the linear discrete-time system, matrices $\mathbf{A}$ and $\mathbf{B}$ are just used to generate data which is provided for calculation. The feedback matrix $\mathbf{K}$ can be solved without the knowledge of system.

The existing Q-learning calculation process has been calculated based on 200 sets of data sampled from the system. While the step 2 cannot be executed, because of the matrix $\mathbf{\Phi}^T\mathbf{\Phi}$ is irreversible. To avoid the data deficiencies factor, the process is recalculated based on 1500 sets of data, and the same situation appears.

An effective feedback matrix $\mathbf{K} = \begin{bmatrix} 1.0 & 0.27 & 0.21 \end{bmatrix}$ can be obtained from the same 200 sets of data based on the improved Q-learning method. Figure 1 shows the closed-loop response and the control effort with the initial state of system $x_0 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$. In this figure, ordinates are values of states or control variables, and the horizontal ordinates are the computation times of the discrete-time system. The system is stable under the calculated feedback matrix, which means that the effective controller can be solved by the proposed method. There is a distinct different between the solved feedback matrix and the optimal theoretical value, the reason of that is the limitation of the information provided by the sampled data. With more data sets, the calculated feedback matrix will converge to the optimal theoretical value, as shown in example 2.



(a) (b)

Figure 1. Closed-loop response and the control effort of system 2. (**a**) Trajectories of state values. (**b**) Trajectory of control value.

4.2. Example 2

Example 2: A DC motor discrete-time model described as two-order state equation is:

$$\dot{x} = \begin{bmatrix} 1 & 0.0952 \\ 0 & 0.8187 \end{bmatrix} x + \begin{bmatrix} 0.0955 \\ 0.1813 \end{bmatrix} u$$

The weight matrices of performance criteria are: $\mathbf{Q} = \mathbf{I}$,$\mathbf{R} = 1$. The discount factor is $\gamma = 1$.

To compare the optimal theoretical feedback matrix with the calculated one, a controller is designed by the traditional optimal control theory based on the mathematical model. So the matrices $\mathbf{P}$ and $\mathbf{K}^*$ are solved by Equation (7): $\mathbf{P} = \begin{bmatrix} 8.8808 & 1.4297 \\ 1.4297 & 2.9104 \end{bmatrix}$, $\mathbf{K}^* = \begin{bmatrix} 0.9031 & 0.5294 \end{bmatrix}$. In addition, the solution of Equation (5) is: $\mathbf{H} = \begin{bmatrix} 9.8809 & 2.0159 & 1.1073 \\ 2.0159 & 3.2541 & 0.6492 \\ 1.1073 & 0.6492 & 1.2262 \end{bmatrix}$. The $\Theta(\mathbf{H})$ can be calculated according to Equation (8): $\Theta(\mathbf{H}) = \begin{bmatrix} 9.8809 & 4.0319 & 2.2146 & 3.2541 & 1.2984 & 1.2262 \end{bmatrix}$.

To get the feedback matrix by the existing Q-learning method, the initial feedback matrix is set as $\mathbf{K}_0 = \begin{bmatrix} 0.9031 & 0.5294 \end{bmatrix}$. To avoid the data deficiencies factor, the system has been sufficiently excited, and 1500 sets of date is sampled. According to Formula (11), the equation $(\mathbf{\Phi}^T\mathbf{\Phi})\Theta(\mathbf{H}) = \mathbf{\Phi}^T\nu$ is obtained as following form:

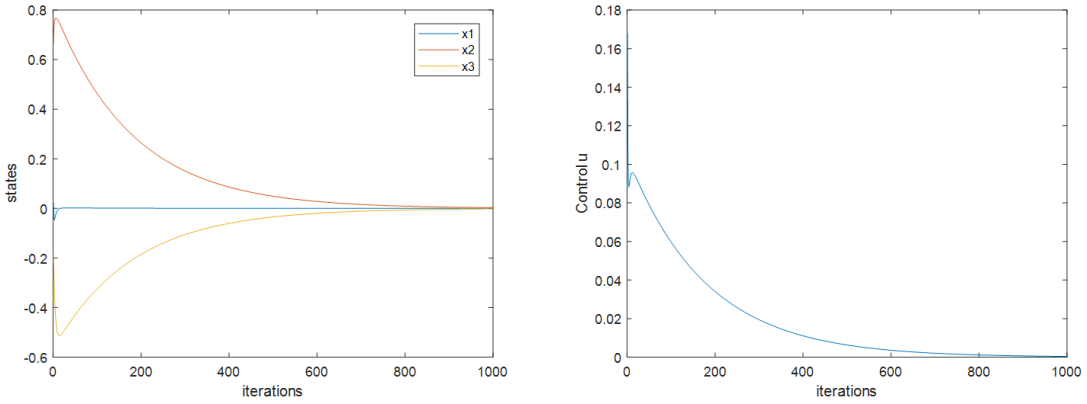
$$\begin{bmatrix} 3.8771 & 9.9710 & -8.7804 & 8.3758 & -13.4391 & 15.0447 \\ 9.9710 & 39.8898 & -30.1240 & 55.0816 & -65.1862 & 61.7167 \\ -8.7804 & -30.1240 & 23.8783 & -36.7265 & 46.6489 & -46.2619 \\ 8.3758 & 55.0816 & -36.7265 & 117.7324 & -112.0756 & 92.5045 \\ -13.4391 & -65.1862 & 46.6489 & -112.0756 & 118.2058 & -104.7108 \\ 15.0447 & 61.7167 & -46.2619 & 92.5045 & -104.7108 & 97.2166 \end{bmatrix} \Theta(\mathbf{H}) = \begin{bmatrix} 87.3195 \\ 362.9179 \\ -271.0004 \\ 574.5259 \\ -631.9216 \\ 579.3007 \end{bmatrix}$$

To verify the above equation, the $\mathbf{H}$ calculated by Riccati equation and the $\Theta(\mathbf{H})$ are substituted into it, and the two sides of the equation are equal. Therefore, the procedure of the calculation is proved correct. However, the $\Theta(\mathbf{H})$ cannot be solved by Equation (15), the analysis is as following. The eigenvalues of $\mathbf{\Phi}^T\mathbf{\Phi}$ is calculated as: $\lambda(\mathbf{\Phi}^T\mathbf{\Phi}) = \begin{Bmatrix} 0 & 0 & 0 & 0.7968 & 26.4209 & 373.5823 \end{Bmatrix}$.

The rank of square matrix $\mathbf{\Phi}^T\mathbf{\Phi}$ is 3, which means that the matrix is singular and noninvertible. Therefore the $\Theta(\mathbf{H})$ cannot be calculated via the least squares regression adopted by the existing Q-learning method.

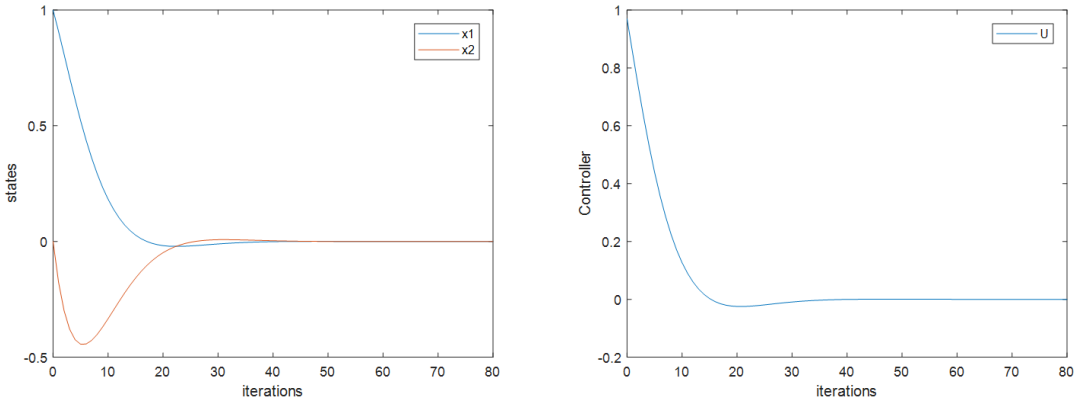
While with the improved Q-learning method, the feedback matrix can be calculated as $\mathbf{K} = \begin{bmatrix} 0.5406 & 0.0115 \end{bmatrix}$. According to the ridge trace of $\mathbf{H}$ and ν, the parameter λ in Formula (14) is set as 0.01. Figure 1 shows the closed-loop response and the control effects with the initial state of the system as $x_0 = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$. Similar to example 1, the Figure 1 shows that the system is closed-loop stable. Like in Figure 2, ordinates of Figure 1 are values of states or control variables, and the horizontal ordinates are the computation times of the discrete-time system. In Figure 3 the ordinates and the horizontal ordinates are the same.

The initial feedback matrix is set as $\mathbf{K}_0 = \begin{bmatrix} 0.5 & 0 \end{bmatrix}$, and the data acquisition process is the same as shown in example 1. The calculation process can be iterated repeatedly until $\mathbf{K}$ converges to a stable value or the system achieves satisfying performance. When the system obtains best performance, the calculated feedback matrix is $\mathbf{K} = \begin{bmatrix} 0.9171 & 0.4143 \end{bmatrix}$, and it is very close to the optimal theoretical value $\mathbf{K}^* = \begin{bmatrix} 0.9031 & 0.5294 \end{bmatrix}$. The simulation results are shown in Figure 3. According to the simulation, for the linear discrete-time systems, the quadratic optimal controllers can be designed by online iterative calculation.



(a) (b)

Figure 2. Closed-loop response and the control effort of system 1. (**a**) Trajectories of state values. (**b**) Trajectory of control value.



(a) (b)

Figure 3. Closed-loop response and the control effort of system 2 with $K = [0.91710.4143]$. (**a**) Trajectories of state values. (**b**) Trajectory of control value.

5. Further Analysis for Nonlinear Systems

The analysis and the simulation results both present that the quadratic optimal controllers for the linear discrete-time systems cannot be solved by Equation (11). The corresponding analysis should

be done to guide the design of Q-learning quadratic optimal controllers for nonlinear discrete-time systems and nonlinear continuous-time systems.

For nonlinear discrete-time systems described as $\mathbf{x}(l+1) = \mathbf{f}(\mathbf{x}(l), \mathbf{u}(l))$, the index of the optimized performance is defined as: $V(\mathbf{x}(l)) = \sum_{k=l}^{\infty} \gamma^{k-l} \nu(\mathbf{x}(k), \mathbf{u}(k))$. And the Q function is defined as: $Q(\mathbf{x}(l), \mathbf{u}(l)) = \nu(\mathbf{x}(l), \mathbf{u}(l)) + \gamma Q(\mathbf{x}(l+1), \mathbf{u}(l+1))$. Therefore, the Q function can be noted as:

$$Q(\mathbf{x}(l), \mathbf{u}(l)) = \varphi^T(\mathbf{x}(l), \mathbf{u}(l)) \Theta_Q \tag{18}$$

where Θ_Q is the vector of unknown parameters, $\varphi^T(\mathbf{x}(l), \mathbf{u}(l))$ is sampled data set composed by state variables $\mathbf{x}$ and control variables $\mathbf{u}$. The extreme condition derived from Formula (18) is shown as following form: $\frac{\partial Q(\mathbf{x}(l), \mathbf{u}(l))}{\partial \mathbf{u}(l)} = \frac{\partial \varphi_Q^T(\mathbf{x}(l), \mathbf{u}(l))}{\partial \mathbf{u}(l)} \Theta_Q = 0$. The optimized solution of the above equation is noted as:

$$\mathbf{u}(l) = \varphi_u^T(\mathbf{x}(l)) \Theta_u \tag{19}$$

where Θ_u is the unknown parameters vector, $\varphi_u^T(\mathbf{x}(l))$ is the sampled data set composed by state vector $\mathbf{x}$.

Theoretically, the controller can be solved by Formula (18) under the condition derived as Formula (19). While the elements of vector $\varphi_Q^T(\mathbf{x}(l), \mathbf{u}(l))$ should be linear independent to solve Θ_Q adopting the least squares method. So the linear correlation of the vector $\varphi_Q^T(\mathbf{x}(l), \mathbf{u}(l))$ should be discussed to ensure the calculation can be implemented correctly.

For nonlinear continuous-time systems described as: $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, the index of the optimized performance is defined as: $V(\mathbf{x}(l)) = \int_t^{\infty} e^{-\gamma(\tau - t)} \nu(\mathbf{x}, \mathbf{u}) d\tau$. And the Q function is defined as following form.

$$\begin{aligned} l\mathbf{Q}(\mathbf{x}(l), \mathbf{u}(l)) &= \int_t^{t+T} e^{-\gamma(\tau - t)} \nu(\mathbf{x}, \mathbf{u}) + e^{-\gamma T} \mathbf{Q}(\mathbf{x}(t+T), \mathbf{u}(t+T)) \\ &= \mathbf{R}(\mathbf{x}(l), \mathbf{u}(l)) + e^{-\gamma T} \mathbf{Q}(\mathbf{x}(t+T), \mathbf{u}(t+T)) \end{aligned} \tag{20}$$

Therefore, the Q function can be noted as:

$$Q(\mathbf{x}(t), \mathbf{u}(t)) = \varphi_Q^T(\mathbf{x}(t), \mathbf{u}(t)) \Theta_Q \tag{21}$$

where Θ_Q is the unknown parameters vector, $\varphi^T(\mathbf{x}(t), \mathbf{u}(t))$ is sampled data set composed by state variables and control variables . The extreme condition derived from Formula (20) is shown as the following form: $\frac{\partial Q(\mathbf{x}(t), \mathbf{u}(t))}{\partial \mathbf{u}(t)} = \frac{\partial \varphi_Q^T(\mathbf{x}(t), \mathbf{u}(t))}{\partial \mathbf{u}(t)} \Theta_Q = 0$. The optimized solution of the above equation noted as:

$$\mathbf{u}(t) = \varphi_u^T(\mathbf{x}(t)) \Theta_u \tag{22}$$

where Θ_u is the unknown parameters vector, $\varphi_u^T(\mathbf{x}(t))$ is the sampled data set composed by state vector $\mathbf{x}$.

Similar to nonlinear discrete-time systems, the quadratic optimal controllers for nonlinear continuous-time systems can be solved by Formula (20) under the extreme condition formed as Formula (21). And also the elements of vector $\varphi_Q^T(\mathbf{x}(t), \mathbf{u}(t))$ should be linear independent to solve Θ_Q via the least square method. So the linear correlation of the vector $\varphi_Q^T(\mathbf{x}(t), \mathbf{u}(t))$ should also be discussed to ensure the calculation can be implemented correctly.

In general, the problem about the multi-collinearity of data sets sampled from nonlinear systems no matter discrete-time or continuous-time needs to be discussed in further research to determine whether the existing Q-learning algorithm should be modified when it is used to deal with nonlinear systems. This will make positive significance for nonlinear systems control schemes such as microbial fuel cell systems [35,36]. The relative analysis and simulations will be developed in our further researching.

6. Conclusions

The design of optimal quadratic controllers for linear discrete-time systems based on Q-learning method is claimed as a kind of model-free method [37,38]. In this paper, both the theoretical analysis and the simulation results demonstrate that the existing design method of model-free controllers based on Q-learning algorithm ignores the linear independence of sampled data sets. For the linear discrete-time or continuous-time systems, the model-free quadratic optimal controllers cannot be established directly by the Q-learning method until the linear independence of data sets is guaranteed. The matrix $\mathbf{\Phi}$ is inevitably column-related considering the quadratic optimal control law $u = \mathbf{K}x$. So other solution methods should be investigated besides the least squares method to solve parameters matrix $\Theta(\mathbf{H})$.

An improved Q-learning method based on ridge regression for linear discrete-time systems is proposed in this paper to deal with the multi-collinearity problem of the sampled data sets. Simulations show the effectiveness of the proposed method. There are deficiencies of the improved Q-learning method. First, the proposed method relies on the ridge trace of the linear correlation matrix, so the computation speed is limited. Second, the proposed method considers with neither disturbance nor saturation of controller. Further research will focus on solving these problems.

Author Contributions: Conceptualization and formal analysis, J.W.; methodology, software, and writing, J.C.; review and editing, funding acquisition, J.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grants 61603094.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lewis, F.L.; Vrabie, D.; Vamvoudakis, K.G. Reinforcement Learning and Feedback Control: Using Natural Decision Methods to Design Optimal Adaptive Controllers. *Control Syst. IEEE* **2012**, *32*, 76–105.
2. Bellman, R.; Kalaba, R. On adaptive control processes. *IRE Trans. Autom. Control* **1959**, *4*, 1–9. [CrossRef]
3. Ramadge, G. Discrete time multivariable adaptive control. *IEEE Trans. Autom. Control* **1980**, *25*, 335–340.
4. Watkins, C.J.; Dayan, P. Q-learning. *Mach. Learn.* **1992**, *8*, 279–292. [CrossRef]
5. Huang, K.; Li, S.; Kang, X.; Fang, L. Spectral–Spatial Hyperspectral Image Classification Based on KNN. *Sens. Imaging* **2016**, *17*, 1–13. [CrossRef]
6. Qi, X.; Zhu, Y.; Zhang, H. A new meta-heuristic butterfly-inspired algorithm. *J. Comput. Sci.* **2017**, *23*, 226–239. [CrossRef]
7. Pires, D.S.; de Oliveira Serra, G.L. Methodology for Evolving Fuzzy Kalman Filter Identification. *Int. J. Control Autom. Syst.* **2019**, *17*, 793–800. [CrossRef]
8. Bradtke, S.J.; Ydstie, B.E.; Barto, A.G. Adaptive Linear Quadratic Control Using Policy Iteration. In Proceedings of the 1994 American Control Conference—ACC '94, Baltimore, MD, USA, 29 June–1 July 1994. [CrossRef]
9. Balakrishnan, S.N.; Ding, J.; Lewis, F.L. Issues on Stability of ADP Feedback Controllers for Dynamical Systems. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **2008**, *38*, 913–917. [CrossRef]
10. Baird, L.C., III. Reinforcement learning in continuous time: Advantage updating. In Proceedings of the IEEE World Congress on IEEE International Conference on Neural Networks IEEE, Orlando, FL, USA, 28 June–2 July 1994. [CrossRef]
11. Rahimiyan, M.; Mashhadi, H.R. An adaptive-learning algorithm developed for agent-based computational modeling of electricity market. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **2010**, *40*, 547–556.
12. Prashanth, L.A.; Bhatnagar, S. Reinforcement learning with function approximation for traffic signal control. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 412–421. [CrossRef]
13. Kar, S.; Moura, J.M.; Poor, H.V. QD-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus innovations. *IEEE Trans. Signal Process.* **2013**, *61*, 1848–1862.

14. Huang, T.; Liu, D. A self-learning scheme for residential energy system control and management. *Neural Comput. Appl.* **2013**, *22*, 259–269. [CrossRef]
15. Sun, Q.; Zhou, J.; Guerrero, J.M.; Zhang, H. Hybrid three-phase/single-phase microgrid architecture with power management capabilities. *IEEE Trans. Power Electron.* **2015**, *30*, 5964–5977.
16. Al-Tamimi, A.; Lewis, F.L.; Abu-Khalaf, M. Model-free Q-learning designs for linear discrete-time zero-sum games with application to H-infinity control. *Automatica* **2007**, *43*, 473–481. [CrossRef]
17. Chun, T.Y.; Lee, J.Y.; Park, J.B.; Choi, Y.H. Comparisons of continuous-time and discrete-time Q-learning schemes for adaptive linear quadratic control. In Proceedings of the SICE Annual Conference (SICE), Akita, Japan, 20–23 August 2012. [CrossRef]
18. Gao, W.; Jiang, Z.P. Data-driven adaptive optimal output-feedback control of a 2-DOF helicopter. In Proceedings of the American Control Conference (ACC), Boston, MA, USA, 6–8 July 2016. [CrossRef]
19. Lewis, F.L.; Vrabie, D. Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE Circuits Syst. Mag.* **2009**, *9*, 32–50. [CrossRef]
20. Lewis, F.L.; Vamvoudakis, K.G. Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data. *IEEE Trans. Syst. Man Cybern. Part B Cybern. A Publ. IEEE Syst. Man Cybern. Soc.* **2010**, *41*, 14–25.
21. Rizvi, S.A.A.; Lin, Z. An iterative Q-learning scheme for the global stabilization of discrete-time linear systems subject to actuator saturation. *Int. J. Robust Nonlinear Control* **2019**, *29*, 2660–2672.
22. Abu-Khalaf, M.; Lewis, F.L. Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach. *Automatica* **2005**, *41*, 779–791. [CrossRef]
23. Kiumarsi, B.; Lewis, F.L. Actor–Critic-Based Optimal Tracking for Partially Unknown Nonlinear Discrete-Time Systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *26*, 140–151. [CrossRef] [PubMed]
24. Modares, H.; Lewis, F.L. Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning. *Automatica* **2014**, *50*, 1780–1792. [CrossRef]
25. Caines, P.E.; Levanony, D. Stochastic epsilon-optimal linear quadratic adaptation: An alternating controls policy. *SIAM J. Control Optim.* **2019**, *57*, 1094–1126. [CrossRef]
26. la Fé-Perdomo, I.; Beruvides, G.; Quiza, R.; Haber, R.; Rivas, M. Automatic Selection of Optimal Parameters Based on Simple Soft-Computing Methods: A Case Study of Micromilling Processes. *IEEE Trans. Ind. Inform.* **2019**, *15*, 800–811. [CrossRef]
27. Damm, T.; Mena, H.; Stillfjord, T. Numerical solution of the finite horizon stochastic linear quadratic control problem. *Numer. Linear Algebra Appl.* **2017**, *24*, e2091. [CrossRef]
28. Li, G.; Zhang, W. Discrete-Time Indefinite Stochastic Linear Quadratic Optimal Control with Second Moment Constraints. *Math. Probl. Eng.* **2014**. [CrossRef]
29. Dey, R.; Martinez Garcia, J.C. Improved delay-range-dependent stability analysis for uncertain retarded systems based on affine Wirtinger-inequality. *Int. J. Robust Nonlinear Control* **2017**, *27*, 3028–3042. [CrossRef]
30. Dinc, E. Linear regression analysis and its application to the multivariate spectral calibrations for the multiresolution of a ternary mixture of acffeine, paracetamol and metamizol in tablets. *J. Pharm. Biomed. Anal.* **2003**, *33*, 605–615. [CrossRef]
31. Belsley, D.A. A Guide to using the collinearity diagnostics. *Comput. Sci. Econ. Manag.* **1991**, *4*, 33–50. [CrossRef]
32. Yaoqiong, Z.; Ning, D.; Bo, Y. A new biased-estimator for a class of ill-conditioned seemingly unrelated regression systems. *Int. J. Appl. Math. Stat.* **2013**, *41*, 71–78. [CrossRef]
33. Chinea-Rios, M.; Sanchis Trilles, G.; Casacuberta, F. Discriminative ridge regression algorithm for adaptation in statistical machine translation. *Pattern Anal. Appl.* **2019**, *22*, 1293–1305. [CrossRef]
34. Dormann, C.F.; Elith, J.; Bacher, S.; Buchmann, C.; Carl, G.; Carré, G.; Marquéz, J.R.; Gruber, B.; Lafourcade, B.; Leitão, P.J.; et al. Collinearity: A review of methods to deal with it and a simulation study evaluating their performance. *Ecography* **2013**, *36*, 27–46. [CrossRef]
35. Patel, R.; Deb, D. Parametrized control-oriented mathematical model and adaptive backstepping control of a single chamber single population microbial fuel cell. *J. Power Sources* **2018**, *396*, 599–605. [CrossRef]
36. Patel, R.; Deb, D. Nonlinear adaptive control of microbial fuel cell with two species in a single chamber. *J. Power Sources* **2019**, *434*, 226739. [CrossRef]
37. Rizvi, S.A.A.; Lin, Z. Output feedback Q-learning for discrete-time linear zero-sum games with application to the h-infinity control. *Automatica* **2018**, *95*, 213–221. [CrossRef]

38. Wang, T.; Zhang, H.; Luo, Y. Stochastic linear quadratic optimal control for model-free discrete-time systems based on Q-learning algorithm. *Neurocomputing* **2018**, *30*, 5964–5977.