# Q-Learning Algorithms in Control Design of Discrete-Time Linear Periodic Systems by Lifting Technique

Phuong Nam Dao [1], Hong Quang Nguyen [2,*]
[1] Hanoi University of Science and Technology, Hanoi, Vietnam
[2] Thai Nguyen University of Technology, Thai Nguyen City, Vietnam
[*]Email: quang.nguyenhong@tnut.edu.vn

***Abstract*—This article investigates a lifting method enhanced modified Q-Learning to be known as the special case of reinforcement learning (RL) for optimal control design of a class of periodic systems. Due to the purpose of investigating periodic systems with many sub-equations by only one dynamic equation, a lifting method is utilized to transfer the periodic LQR to time-invariant LQR in an augmented system. After obtaining the corresponding model to be described by only one dynamic equation, partition technique is developed to achieve easier optimal control design. Due to the difficulty in analytically solving Hamilton-Jacobi-Bellman equation, adaptive reinforcement learning (ARL) is studied using iteration algorithm. The model-free Q-learning solution with the advantage of considering the Bellman function of two variables is proposed with the expanded system and the convergence analysis is discussed by considering the poles position on the complex plane as well as Lyapunov stability theory. The proposed Q-Learning method is realized online to find the optimal controller based on the system states' data collection, and the computation of Bellman function and control policy is only in one step of the proposed algorithm. The tracking and performance of the proposed methods are illustrated for spacecraft systems with appropriate simulation results.**

***Index Terms*—lifting technique, Q-learning, discrete-time linear periodic systems, LQR, adaptive reinforcement learning**

## I. Introduction

Over the past decades, robotics control systems have attracted many researchers with numerous approaches mentioned, such as the backstepping technique [1,2], separation method [3], and so on. Typically, almost all control designs for robotic systems are developed by traditional nonlinear control techniques being extended from Lyapunov stability theory. As a result, it is challenging to make control objectives being different from conventional tracking control. In recent years, the optimal control solution is developed with the advantages of overcoming the challenges of input, full state constraint, actuator saturation, etc. The fact is that these disadvantages are formulated in the performance index as well as online adaptive reinforcement learning [5,6]. The optimal control-based approaches with the consideration of optimizing the performance index are also discussed in recent time, such as adaptive dynamic programing in wheeled inverted pendulum [4], reinforcement learning in robots [14], in switched systems [15]. The starting point of optimal control application in control design can be considered the numerical solution method of the Ricatti equation for discrete-time periodic systems [7]. The Q-learning technique is an extension of classical adaptive reinforcement learning with the idea of utilizing the special Q-function to be obtained from the Hamilton-Jacobi-Bellman (HJB) equation [8,9].

It should be noted that because of the advantage of the Q-learning technique in handling completely uncertain systems[10], there are many application and development studies of Q-learning that can be implemented in [8,9,11,13]. However, most of the existing work is focused on time-invariant systems [16,17]. The fact is that the optimal function and corresponding optimal control need to be known as time-varying functions in varying systems. Hence, it is challenging to implement the PI, VI policies algorithm in control design. The linear periodic systems are a class of time-varying linear systems known much in practical applications [5,12]. Yang has investigated the novel lifting methodology in [18] with the advantage of transforming the linear periodic discrete-time system into a linear time-invariant discrete-time system, ensuring the implementation of the LQR method. However, to our knowledge, the development of Q-learning techniques for linear periodic discrete-time systems has not yet been completed. Our work presents a new lifting technique based on Q-learning to implement the periodic LQR problem. The standard Q-learning can not be directly applied to the periodic LQR problem. Therefore we need to modify the standard Q-learning so that its convergence is still guaranteed. The remaining work is organized as follows. The optimal control design for discrete-time linear periodic systems and the lifting methodology are discussed in Section 2. The proposed Q-learning and theoretical discussions are described in Section 3. On the other hand, the simulation results are developed in

Manuscript received November 21, 2020; revised February 7, 2021.


doi: 10.18178/ijmerr.10.11.620-625

Section 4. Finally, the conclusions are pointed out in Section 5.

**Notation:** Throughout this article, ⊗ denotes the Kronecker product. *blkdiag*(.) is the block diagonal matrix, Discrete-time linear periodic system (DTLP).

## II. Problem Statements

This section describes the mathematical model of the linear periodic discrete-time system. The corresponding lifting technique is investigated to easily obtain the optimal control law by transforming the optimal control problem of discrete-time periodic systems into improved Linear Time-Invariant (LTI) systems.

### A. Optimal Controller for a DTLP System

The DTLP systems can be represented as:

$$x_{k+1} = A_k x_k + B_k u_k \tag{1}$$

where $x_k \in \mathbb{R}^n$ , $,u_k \in \mathbb{R}^m$ are defined as the state variables vector, control inputs vector, respectively. Additionally, it is assumed that the initial state variables vector $x_0$ has been known. The system matrices $A_k$ $B_k$ are satisfied $A_{k+p} = A_k \in \mathbb{R}^{n\times n}$ and $B_{k+p} = B_k \in \mathbb{R}^{n\times m}$ , where $p$ is a positive constant natural number to be known as the number of samples in each period.

The following formula proposes the traditional optimal Controller:

$$u_k = -L_k^* x_k \tag{2}$$

This work enables us to minimize the following infinite performance index:

$$J = \frac{1}{2}\sum_{k=0}^{\infty}\left[x_k^T Q_k x_k + u_k^T R_k u_k\right] \tag{3}$$

where $Q_k = Q_{k+p} \geq 0, R_k = R_{k+p} > 0$ . Authors in [3] proposed the optimal feedback to be computed by solving the discrete-time periodic Riccati equation (DPRE):

$$\begin{aligned} &A_k^T P_k A_k - P_k \\ &-A_k^T P_k B_k \left(R_k + B_k^T P_k B_k\right)^{-1} B_k^T P_k A_k + Q_k = 0 \end{aligned} \tag{4}$$

It has been known that the eqn. (4) has a corresponding solution of the matrix $P_k^*$ for each couple of matrices $Q_k \geq 0$ , $R_k > 0$ . A group of $p$ Riccati equations is continuously solved, leading us to obtain $p$ positive semidefinite matrices $P_k$ , $k = 1,\ldots, p$ . Hence, the optimal feedback matrix at each sample time is computed as:

$$L_k^* = -\left(R_k + B_k^T P_k^* B_k\right)^{-1} B_k^T P_k^* A_k \tag{5}$$

We have known that the eqn. DPRE (4) can be solved using the algorithms in [7,17]. We achieve the optimal feedback matrix being different from that at different sampling periods time. It is also necessary to know the accurate models to solve (4), but this requirement is not easy in practice.

### B. Lifting Technique in Control Design

There are some easy lifting methodologies to be proposed in [6,12] but they are challenging to design the Controller. Yang [18] proposed a remarkable lifting technique for developing a controller. The following theorem will describe this problem:

**Theorem 1.** ( [18]) For a DTLP system with a period $p$ , let's define

$$\bar{A} = \begin{bmatrix} 0 & \cdots & 0 & A_0 \\ 0 & \cdots & 0 & A_1 A_0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & A_{p-1}\ldots A_1 A_0 \end{bmatrix}$$

$$\bar{B} = \begin{bmatrix} B_0 & 0 & \cdots & 0 \\ A_1 B_0 & B_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ A_{p-1}\ldots A_1 B_0 & A_{p-1}\ldots A_2 B_1 & \cdots & B_{p-1} \end{bmatrix}$$

$$\bar{x}_{K+1} = \begin{bmatrix} x_{pk+1} \\ x_{pk+2} \\ \vdots \\ x_{pk+p} \end{bmatrix} \qquad \bar{x}_K = \begin{bmatrix} x_{p(k-1)+1} \\ x_{p(k-1)+2} \\ \vdots \\ x_{p(k-1)+p} \end{bmatrix}$$

$$\bar{x}_0 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ x_0 \end{bmatrix} \qquad \bar{u}_K = \begin{bmatrix} u_{pk} \\ u_{pk+1} \\ \vdots \\ u_{pk+p-1} \end{bmatrix}$$

The eqn. (1) can be rewritten as

$$\bar{x}_{K+1} = \bar{A}\bar{x}_K + \bar{B}\bar{u}_K \tag{6}$$

where $\bar{A} \in \mathbb{R}^{pn\times pn}$ and $\bar{B} \in \mathbb{R}^{pm\times pm}$ . Then, we can realize the traditional LQR method for the augmented LTI system. We aim to find the control input $\bar{u}_K$ to minimize the following quadratic performance index, including new state and control input vectors

$$J = \sum_{K=0}^{\infty}\left[\bar{x}_K^T \bar{Q}\bar{x}_K + \bar{u}_K^T \bar{R}\bar{u}_K\right] \tag{7}$$

where

$\bar{Q} = blkdiag\left(Q_0, Q_1, ..., Q_{p-1}\right), \bar{R} = blkdiag\left(R_0, R_1, ..., R_{p-1}\right)$ .

Because the matrices $\bar{A}$ , $\bar{B}$ , $\bar{Q}$ , $\bar{R}$ are constant, the periodic LQR problem can be discussed as the traditional LQR problem with following controller:

$$\bar{u}_K = -\bar{L}^*\bar{x}_K = \left(\bar{R} + \bar{B}^T \bar{P}^* \bar{B}\right)^{-1} \bar{B}^T \bar{P}^* \bar{A}\bar{x}_K \tag{8}$$

where $\bar{P}^*$ is the solution of the traditional Riccati equation

$$\bar{A}^T \bar{P}\bar{A} - \bar{P} - \bar{A}^T \bar{P}\bar{B}\left(\bar{R} + \bar{B}^T \bar{P}\bar{B}\right)^{-1} \bar{B}^T \bar{P}\bar{A} + \bar{Q} = 0 \tag{9}$$

**Remark 1.** It can be seen that by using the lifting method in Theorem 1, the optimal feedback matrix (8) it can solve one Riccati equation (9) than solving $p$ Riccati equations. However, it is necessary to require accurate knowledge of the improved system dynamic for solving offline the Riccati equation (9).

## III. Novel Q-learning Algorithm

In this section, we propose the method to deal with the LQR problem online using data collection along with the state variables in the absence of system dynamics. We consider the particular structure of the augmented system. Next, we investigate the Q-function and the development of policy improvement. Moreover, we also consider the feasibility of previous adaptive reinforcement learning algorithms. Finally, we present the control system using a modified Q-learning strategy for DTLP systems.

### *A. Partition Method based Optimal Control*

Authors in [18] pointed out the method to split the matrices into subsystems obtaining easier analysis. Consider $\bar{P}^* = (\bar{P}^*)^T$ being a unique solution to (9) as

$$\bar{P}^* = \begin{bmatrix} \bar{P}_{11}^* & \bar{P}_{12}^* \\ \bar{P}_{21}^* & \bar{P}_{22}^* \end{bmatrix} \in \mathbb{R}^{pn \times pn}$$

where $\bar{P}_{11}^* \in \mathbb{R}^{(p-1)n \times (p-1)n}$, $\bar{P}_{12}^* \in \mathbb{R}^{(p-1)n \times n}$, $\bar{P}_{21}^* \in \mathbb{R}^{n \times (p-1)n}$, $\bar{P}_{22}^* \in \mathbb{R}^{n \times n}$.

Based on the results in [18], we have $\bar{P}_{11}^* = (\bar{P}_{11}^*)^T = \bar{Q}_1, \bar{P}_{12}^* = (\bar{P}_{21}^*)^T = 0)$. Thus, it leads to:

$$\bar{P}^* = \begin{bmatrix} \bar{Q}_1 & 0 \\ 0 & \bar{P}_{22}^* \end{bmatrix} \tag{10}$$

The optimal feedback matrix can be yielded as:

$$\bar{L}^* = \begin{bmatrix} 0 & \bar{L}_{12}^* \end{bmatrix} \in \mathbb{R}^{pm \times pn} \tag{11}$$

where

$$\bar{L}_{12}^* = \left(\bar{R} + \bar{B}^T \bar{P}^* \bar{B}\right)^{-1} \left(\bar{B}_1^T \bar{Q}_1 \bar{A}_1 + \bar{B}_2^T \bar{P}_{22}^* \bar{A}_2\right) \in \mathbb{R}^{pm \times n}$$

Several Assumptions are mentioned in this work to analyze the convergence of the proposed algorithm in the next sessions

**Assumption 1** A couple of matrices $\left(\bar{A}_2, \bar{B}_2\right)$ is controllable.

***Remark 2.*** The improved system (6) has a couple of matrices $\left(\bar{A}, \bar{B}\right)$ being stabilizable.

**Remark 3.** For the model-free approach, Landelius [9] proposed several essential optimal control methods for discrete-time systems. The Q-Learning is also discussed for the time-invariant LQR problem in [4]. However, all algorithms require the condition $\left(\bar{A}, \bar{B}\right)$ to be controllable. We can not apply these algorithms to our LQR problem (6) and (7) because the matrices are only stabilizable. Additionally, the tracking of the existing algorithms has not been guaranteed in this work. Therefore, to tackle this disadvantage of controllability conditions, an enhanced Q-learning is presented.

**Remark 4.** The optimal Controller (8) can only move n non-zero poles being controllable.

Due to the purpose is to consider the proposed algorithm in the next chapter, the state variable is splited as:

$$\bar{x}_K = \begin{bmatrix} \bar{x}_K^1 \\ \bar{x}_K^2 \end{bmatrix} \in \mathbb{R}^{pn} \tag{12}$$

where $\bar{x}_K^1 \in \mathbb{R}^{(p-1)n}$, $\bar{x}_K^2 \in \mathbb{R}^n$. According to (11,12,8) the control input can be given as:

$$\bar{u}_K = -\begin{bmatrix} 0 & \bar{L}_{12}^* \end{bmatrix} \begin{bmatrix} \bar{x}_K^1 \\ \bar{x}_K^2 \end{bmatrix} = -\bar{L}_{12}^* \bar{x}_K^2 \tag{13}$$

It should be noted that the optimal Controller (13) only depends on $\bar{x}_K^2$. Therefore, this result is the critical idea to modify the existing Q-learning algorithms, and the convergence is still satisfied as applying for the augmented system (6).

### *B. Q-function in Control Design*

The unity function of the improved system can be chosen as:

$$r\left(\bar{x}_K, \bar{u}_K\right) = \bar{x}_K^T \bar{Q} \bar{x}_K + \bar{u}_K^T \bar{R} \bar{u}_K$$

A policy $\bar{u}_K = -\bar{L}\bar{x}_K$ stabilizing system (6) is led to Bellman Function as:

$$V_L(\bar{x}_K) = \bar{x}_K^T \bar{P} \bar{x}_K \tag{14}$$

Q-function $Q_L(x_K, u_K)$ can be considered as a framework of the value function $V_L(x_k)$ and the unity function $r\left(\bar{x}_K, \bar{u}_K\right)$

$$Q_L(\bar{x}_K, \bar{u}_K) = r(\bar{x}_K, \bar{u}_K) + V_L(\bar{x}_{K+1}) \tag{15}$$

According to (6,14,15), the Q-function can be given explicitly:

$$\begin{aligned} &Q_L(\bar{x}_K, \bar{u}_K) \\ &= \bar{x}_K^T \left(\bar{Q} + \bar{A}^T \bar{P} \bar{A}\right) \bar{x}_K + \bar{u}_K^T \left(\bar{B}^T \bar{P} \bar{B} + \bar{R}\right) \bar{u}_K \\ &+ \bar{x}_K^T \bar{A}^T \bar{P} \bar{B} u_K + \bar{u}_K^T \bar{B}^T \bar{P} \bar{A} x_K \end{aligned} \tag{16}$$

It leads to the relation as:

$$\begin{aligned} &\bar{x}_K^T \left(\bar{Q} + \bar{A}^T \bar{P} \bar{A}\right) \bar{x}_K \\ &= (\bar{x}_K^1)^T \bar{Q}_1 \bar{x}_K^1 + (\bar{x}_K^2)^T \left(\bar{Q}_2 + \bar{A}_1^T \bar{Q}_1 \bar{A}_1 + \bar{A}_2^T \bar{P}_{22} \bar{A}_2\right) \bar{x}_K^2 \\ &\bar{u}_K^T \bar{B}^T \bar{P} \bar{A} \bar{x}_K = \bar{u}_K^T [0 \left(\bar{B}_1^T \bar{Q}_1 \bar{A}_1 + \bar{B}_2^T \bar{P}_{22} \bar{A}_2\right) \bar{x}_K^2] \end{aligned} \tag{17}$$

Eq (17) does not depend on $\bar{x}_K^1$. Hence, it is formulated as

$$\bar{u}_K^T \bar{B}^T \bar{P} \bar{A} \bar{x}_K = \bar{u}_K^T \bar{B}^T \bar{P} \bar{A} M \bar{x}_K^2 \tag{18}$$

With $M = \begin{bmatrix} 0_{(p-1)n\times n} \\ I_{n\times n} \end{bmatrix}$

Therefore, we can obtain that:

$$Q_L(\bar{x}_K, \bar{u}_K) = (\bar{x}_K^1)^T \bar{Q}_1 \bar{x}_K^1 + \phi_K^T H \phi_K \tag{19}$$

where kernel matrix $H \in \mathbb{R}^{(pm+n)\times(pm+n)}$ is given as:

$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$$

where:

$$H_{11} = \bar{Q}_2 + \bar{A}_1^T \bar{Q}_1 \bar{A}_1 + \bar{A}_2^T \bar{P}_{22} \bar{A}_2 \in \mathbb{R}^{n\times n},$$
$$H_{12} = H_{21}^T = M^T \bar{A}^T \bar{P} \bar{B} \in \mathbb{R}^{n\times pm},$$
$$H_{22} = \bar{B}^T \bar{P} \bar{B} + \bar{R} \in \mathbb{R}^{pm\times pm},$$
$$\phi_K = \begin{bmatrix} (\bar{x}_K^2)^T & \bar{u}_K^T \end{bmatrix}^T \in \mathbb{R}^{n+pm}$$

### C. The Proposed Q-learning Strategy

The Q-function can be written in the recursive form to evaluate the algorithm as

$$\begin{aligned} &Q_L(\bar{x}_K, \bar{u}_K) = \\ &r(\bar{x}_K, \bar{u}_K) + Q_L(\bar{x}_{K+1}, -\bar{L}\bar{x}_{K+1}) \end{aligned} \tag{20}$$

Note that $\phi_K^T H \phi_K = \bar{\phi}_K vec(H)$ were $(\bar{\phi}_K = \left(\phi_K^T \otimes \phi_K^T\right)$ and $H$ is the symmetric matrix, (19) becomes

$$Q_L(\bar{x}_K, \bar{u}_K) = (\bar{x}_K^1)^T \bar{Q}_1 \bar{x}_K^1 + \bar{\phi}_K vec(H) \tag{21}$$

Then, we rewrite (20) to yield

$$\begin{aligned} &r(\bar{x}_K, \bar{u}_K) = (\bar{\phi}_K - \bar{\phi}_{K+1}) vec(H) \\ &+((\bar{x}_K^1)^T \bar{Q}_1 \bar{x}_K^1 - (\bar{x}_{K+1}^1)^T \bar{Q}_1 \bar{x}_{K+1}^1 \end{aligned} \tag{22}$$

We define

$$\gamma_K = r(\bar{x}_K, \bar{u}_K) - ((\bar{x}_K^1)^T \bar{Q}_1 \bar{x}_K^1 + (\bar{x}_{K+1}^1)^T \bar{Q}_1 \bar{x}_{K+1}^1$$

And (22) becomes

$$(\bar{\phi}_K - \bar{\phi}_{K+1}) vec(H) = \gamma_K \tag{23}$$

Which is a linear equation that can be written as

$$Z vec(H) = Y \tag{24}$$

With $Z \in \mathbb{R}^{N\times(pm+n)(pm+n)}$ and $Y \in \mathbb{R}^N$ being the data matrices defined by:

$$Z = \begin{bmatrix} (\bar{\phi}_K - \bar{\phi}_{K+1})^T \\ (\bar{\phi}_{K+1} - \bar{\phi}_{K+2})^T \\ \cdots \\ (\bar{\phi}_{K+N-1} - \bar{\phi}_{K+N})^T \end{bmatrix} \quad Y = [\gamma_K, \ \gamma_{K+1}, \ \cdots \ \gamma_{K+N-1}]^T$$

Now, the proposed Q-Leaning strategy for online implementation is presented as follows:

**Algorithm 1:**

**1. Initialization:** The stabilizing policy $\bar{u}_K^0$ is chosen to guarantee the admissible control condition

**2. Policy Evaluation:** The equation is solved by the Least-Squares method:

$$(\bar{\phi}_K - \bar{\phi}_{K+1}) vec(H^j) = \gamma_K \tag{25}$$

**3. Policy Update:** Update control policy using

$$\bar{L}_{12}^j = \left(H_{22}^j\right)^{-1} H_{21}^j \tag{26}$$

$$\bar{u}_K^{j+1} = -\bar{L}^j \bar{x}_K = -\begin{bmatrix} 0 & \bar{L}_{12}^j \end{bmatrix} \bar{x}_K \tag{27}$$

In this policy evaluation stage, the Bellman equation (20) is realized for a term of $vec(H^j)$ under the data collection and system states to obtain the data matrices. The solution of (21) is solved by using the Least-Squares (LS)

$$vec(H^j) = \left(Z^T Z\right)^{-1} Z^T Y \tag{28}$$

The PE condition [1,4,15] need to be satisfied to guarantee the convergence of Algorithm 1 in the optimal policy. The intersection of the proposed algorithm is expressed in the following theorem.

**Theorem 2.** Let a couple of $(\bar{A}_2, \bar{B}_2)$ being controllable, $(\bar{A}, \bar{Q}^{1/2})$ be observable, and $\bar{u}_K^0$ be an initially stabilizing control. Hence, the convergence of the proposed algorithm is described as The sequence $\{H^j\}_{j=0}^{\infty}$ convergences to the optimal matrix kernel $H^*$ as $j \to \infty$ and the feedback gain $\bar{L}^j \to \bar{L}^*$ as $j \to \infty$.

**Remark 5.** Algorithm 1 is implemented online in real-time using only the state variables data collected along the state trajectories without requiring any system matrices knowledge.

## IV. Simulation Results

In this section, we implement the spacecraft attitude control design using proposed algorithms. The spacecraft can be described by the continuous-time linear periodic system $\frac{d}{dt}x = Ax + B(t)u$.

where $x = [q_1, q_2, q_3, \omega_1, \omega_2, \omega_3]^T \in \mathbb{R}^6$, $\omega_1, \omega_2, \omega_3$ are the body rate concerning the local vertical and local horizontal (LVLH) frame is represented in the body frame and $q_1, q_2, q_3$ are the rotation of the body frame relative to the LVLH frame. $m_1, m_2, m_3$ do the magnetic coils induce the magnetic moment in spacecraft coordinates. The CTLP system is discretized with sampling time to get the DTLP system. It is assumed that the number of samples in one orbital period is p=10. The weighting matrices are chosen as $Q_k = Q_0 = 100 I_6, R_k = R_0 = I_3$. The initial state is appropriately chosen, and the probing noise is selected as random noise. Using the proposed method in the above

sections, the simulation results in Fig. 1,2 describe the effectiveness of tracking of matrices, trajectories, respectively.

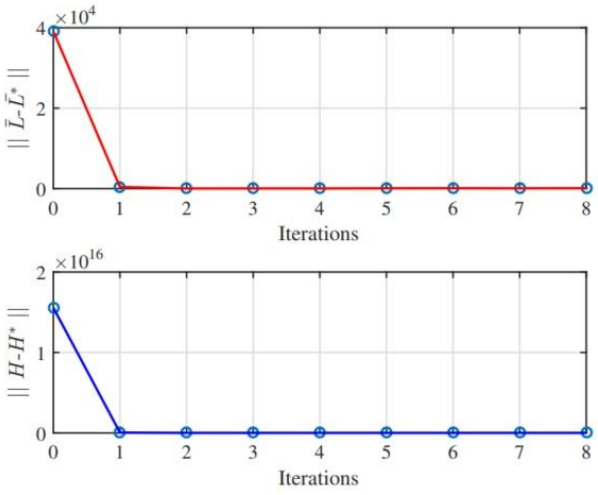


Figure 1. The convergence of a matrix $\bar{L}, \overline{H}$ to its optimal values

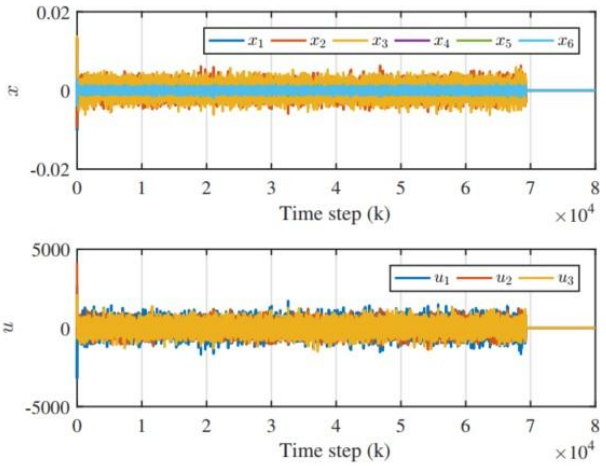


Figure 2. The response of trajectory tracking and control input

## V. Conclusions

This paper proposed a lifting methodology enhanced Q-learning algorithm to tackle the challenge of computation in traditional LQR problem. This proposed method is developed by two steps using lifting and partition technique. First, because the model of periodic systems are described by many sub-equations, it implies that the Ricatti equation is established with high dimension. Therefore, a lifting method is employed to transfer the periodic LQR to time-invariant LQR in an augmented system. Second, partition technique is implemented to obtain easier optimal control design by Q-learning algorithm. It is noted that this algorithm is able to compute Bellman function and control policy in one step. The proposed method deals with the requirement of controllability condition and convergence. Moreover, the lifting technique is utilized in this work with the purpose of handling DTLP systems. On the other hand, the algorithm is computed online in the absence of the system matrices. The simulation studies show that the convergence of not only Bellman function but also control policy with good tracking effectiveness of the proposed method. In the future, we will investigate the problem with off-policy Q learning technique as well as in general robotic systems. On the other hand, practical experiments will developed for robotic systems with reinforcement learning control scheme.

## Conflict of Interest

The authors declare that the submitted work has no conflict of interest

## Acknowledgment

This research was funded by the Thai Nguyen University of Technology, No. 666, 3/2 street, Thai Nguyen City, Viet Nam.

## References

[1] T. Nguyena, T. Hoang, M. Pham, N. Dao, "A gaussian wavelet network-based robust adaptive tracking controller for a wheeled mobile robot with unknown wheel slips," *International Journal of Control*, vol. 92, no. 11, pp. 2681-2692, 2019.

[2] Y. C. Liu, P. N. Dao, and K. Y. Zhao, "On robust control of nonlinear teleoperators under dynamic uncertainties with variable time delays and without relative velocity," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1272-1280, 2019.

[3] N. T. Binh, N. A. Tung, D. P. Nam, and N. H. Quang, "An adaptive backstepping trajectory tracking control of a tractor trailer wheeled mobile robot," *International Journal of Control, Automation and Systems*, vol. 17, no. 2, pp. 465-473, 2019.

[4] P. N. Dao, Y. C. Liu, "Adaptive reinforcement learning strategy with sliding mode control for unknown and disturbed wheeled inverted pendulum," *International Journal of Control, Automation and Systems*, pp. 1-12, 2020.

[5] C. X. Mu, et al. "Data-driven tracking control with adaptive dynamic programming for a class of continuous-time nonlinear systems," *IEEE Transactions on Cybernetics*, vol. 47, no. 6, pp. 1460-1470, 2016.

[6] H. N. Wu and Z. Y. Liu, "Data-driven guaranteed cost control design via reinforcement learning for linear systems with parameter uncertainties," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019.

[7] T. Sun, X. M. Sun, "An adaptive dynamic programming scheme for nonlinear optimal control with unknown dynamics and its application to turbofan engines," *IEEE Transactions on Industrial Informatics*, 2020.

[8] J. Yi, et al. "Optimal output regulation of linear discrete-time systems with unknown dynamics using reinforcement learning," *IEEE Transactions on Cybernetics*, 2019.

[9] L. Jinna, et al. "Adaptive interleaved reinforcement learning: robust stability of affine nonlinear systems with unknown uncertainty," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[10] Lewis, L. Frank, and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control." *IEEE Circuits and Systems Magazine*, vol. 9, no. 3, pp. 32-50, 2009.

[11] L. Jinna, et al. "Off-policy interleaved $ Q $-learning: Optimal control for affine nonlinear discrete-time systems," *IEEE Transactions on Neural Networks and Learning Systems* , vol. 30, no. 5, 2018, pp. 1308-1320.

[12] Y. Xiong, H. B. He, and X. N. Zhong, "Approximate dynamic programming for nonlinear-constrained optimizations," *IEEE Transactions on Cybernetics*, 2019.

[13] Rizvi, S. A. Asad, and Z. L. Lin. "Output feedback Q-learning control for the discrete-time linear quadratic regulator problem," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 5, pp. 1523-1536, 2018.

[14] P. N. Dao, P. T. Loc, T. Q. Huy, "Sliding variable-based online adaptive reinforcement learning of uncertain/disturbed nonlinear mechanical systems," *Journal of Control, Automation and Electrical Systems*, pp. 1-10.
[15] P. N. Dao, H. Q. Nguyen, M. D. Ngo, S. J. Ahn, "On stability of perturbed nonlinear switched systems with adaptive reinforcement learning," *Energies*, vol. 13, no. 19, 5069, 2020.
[16] Wei, Qinglai, et al. "Adaptive dynamic programming for discrete-time zero-sum games." *IEEE Transactions on Neural Networks and Learning Systems* , vol. 29, no. 4, pp. 957-969, 2017.
[17] Y. Yaguang, "An efficient algorithm for periodic Riccati equation with periodically time-varying input matrix," *Automatica*, vol. 78, pp. 103-109, 2017.
[18] Y. Yaguang, "An efficient LQR design for discrete-time linear periodic system based on a novel lifting method," *Automatica*, vol. 87, pp. 383-388, 2018.



**Phuong Nam Dao** received the Ph.D. degree in Electrical Engineering from Hanoi University of Science and Technology, Hanoi, Vietnam in 2013. Currently, he holds the position as lecturer at Hanoi University of Science and Technology, Vietnam. His research interests include control of robotic systems and robust/adaptive, optimal control.

**Hong Quang Nguyen** received the Master's degree in control engineering and automation from Hanoi University of Science and Technology (HUST), Viet Nam, 2012 and a Ph.D. from Thai Nguyen University of Technology (TNUT), Vietnam, 2019. He is currently a lecturer at the Faculty of Mechanical, Electrical, and Electronic Technology, Thai Nguyen University of Technology (TNUT). His research interests include electrical drive systems, control systems and its applications, adaptive dynamic programming control, robust nonlinear model predictive control, motion control, and mechatronics.