An Efficient LQR Design for Discrete-Time Linear Periodic System Based on a Novel Lifting Method

Yaguang Yang

March 12, 2018

Abstract

This paper proposes a novel lifting method which converts the standard discrete-time linear periodic system to an augmented linear time-invariant system. The linear quadratic optimal control is then based on the solution of the discrete-time algebraic Riccati equation associated with the augmented linear time-invariant model. An efficient algorithm for solving the Riccati equation is derived by using the special structure of the augmented linear time-invariant system. It is shown that the proposed method is very efficient compared to the ones that use algorithms for discrete-time periodic algebraic Riccati equation. The efficiency and effectiveness of the proposed algorithm is demonstrated by the simulation test for the design problem of spacecraft attitude control using magnetic torques.

Keywords: Linear periodic discrete-time system, periodic algebraic Riccati equation, LQR, spacecraft attitude control, magnetic torque.
1 Introduction

Many engineering systems are naturally periodic, for example, spacecraft attitude control using magnetic torques \[1\], helicopter rotors control system \[2\], wind turbine control system \[3\], networked control system \[4\], and multirate sampled data system \[5\]. It has been known for about six decades that linear periodic time-varying system can be converted to some equivalent linear time-invariant systems \[6, 7\]. The most popular and widely used methods that convert the linear periodic time-varying model into linear time-invariant models are the so-called lifting methods proposed in \[8, 9\]. Although these reduced linear time-invariant models are nice for analysis but they are not very convenient for control system design. For example, the Linear Quadratic Regulator (LQR) design for linear periodic system has been focused on the periodic system not on the equivalent linear time-invariant systems proposed in \[8, 9\]. This strategy leads to extensive research on the solutions of the periodic Riccati equations (see \[10, 11, 12, 13, 14\] and references therein). For the discrete-time linear periodic system, two efficient algorithms for Discrete-time Periodic Algebraic Riccati Equation (DPARE) are emerged \[15, 16\].

In this paper, we propose a novel lifting method that converts the linear periodic system to an augmented Linear Time-Invariant (LTI) system. We show that the LQR design method can be directly applied to this LTI system. Moreover, by making full use of the structure of the augmented LTI system, we can derive a very efficient algorithm. We compare the new algorithm to the ones proposed in \[15, 16\]. In addition to some simple analysis on the efficiency, we demonstrate the efficiency and effectiveness of the new algorithm by the simulation test for the design problems of spacecraft attitude control using magnetic torques.

The remainder of the paper is organized as follows. Section 2 briefly summarizes the algorithms of \[15, 16\] so that we can compare the proposed algorithm to the existing ones and analyze the efficiency of these algorithms. Section 3 proposes a novel lifting method and applies some standard discrete-time algebraic Riccati equation result to the augmented LTI model. This leads to a very efficient algorithm for the LQR design for the linear periodic system. Section 4 demonstrates the efficiency and effectiveness of the algorithm by some numerical test. Conclusions are summarized in the last section.

2 Periodic LQR design based on linear periodic system

In this section, we briefly review two efficient algorithms for solving DPARE developed in \[15, 16\]. This will help us later in the comparison of the proposed method to the existing methods.

Let \( p \) be an integer representing the total number of samples in one period in a periodic discrete-time system. We consider the following discrete-time linear periodic system given as follows:

\[
x_{k+1} = A_k x_k + B_k u_k, \tag{1}
\]

where \( A_k = A_{k+p} \in \mathbb{R}^{n \times n} \) and \( B_k = B_{k+p} \in \mathbb{R}^{n \times m} \) are periodic time-varying matrices. For this discrete-time linear periodic system (1), the LQR state feedback control is to find the optimal \( u_k \) to minimize the following quadratic cost function

\[
\lim_{N \to \infty} \left( \min \frac{1}{2} x_N^T Q_N x_N + \frac{1}{2} \sum_{k=0}^{N-1} x_k^T Q_k x_k + u_k^T R_k u_k \right) \tag{2}
\]

where

\[
Q_k = Q_{k+p} \geq 0, \quad R_k = R_{k+p} > 0, \tag{3}
\]

and the initial condition \( x_0 \) is given. It is well-known that the LQR design for problem (1) can be solved by using the periodic solution of the discrete-time periodic algebraic Riccati equation \( [12] \). Two efficient algorithms \[15, 16\] have been developed to solve \( p n \)-dimensional matrix Riccati equations to
find $p$ positive semidefinite matrices $P_k$, $k = 1, \ldots, p$. Given, $P_k$, the periodic feedback controllers are given by the following equations:

$$u_k = -(R_k + B_k^T P_k B_k)^{-1} B_k^T P_k A_k x_k. \tag{5}$$

We summarize these two algorithms as follows: Let

$$E_k = \begin{bmatrix} I & B_k R_k^{-1} B_k^T \\ 0 & A_k^T \end{bmatrix} = E_{k+p}, \tag{6}$$

$$F_k = \begin{bmatrix} A_k & 0 \\ -Q_k & I \end{bmatrix} = F_{k+p}. \tag{7}$$

If $A_k$ is invertible, then $E_k$ and $F_k$ are invertible, and

$$E_k^{-1} = \begin{bmatrix} I & -B_k R_k^{-1} B_k^T A_k^{-T} \\ 0 & A_k^{-T} \end{bmatrix} = E_{k+p}^{-1}$$

and

$$F_k^{-1} = \begin{bmatrix} A_k^{-1} & 0 \\ -Q_k A_k^{-1} & I \end{bmatrix} = F_{k+p}^{-1}.$$

Let $y_k$ be the costate of $x_k$, $z_k = [x_k^T, y_k^T]^T$, and

$$
\Pi_k = E_k^{-1} F_k + E_k^{-1} F_k E_k^{-1} F_k + \cdots + E_k^{-1} F_k + E_k^{-1} F_k = \Pi_{k+p},
\tag{8}
$$

$$\Gamma_k = F_k E_k F_k^{-1} E_k + \cdots + F_k E_k F_k^{-1} E_k = \Gamma_{k+p}. \tag{9}
$$

The solutions of $p$ discrete-time periodic algebraic Riccati equations are symmetric positive semidefinite matrices, $P_k$, $k = 1, \ldots, p$, which are related to the solutions of either one of the two linear systems of equations \[15\] \[10\]:

$$z_{k+p} = \Pi_k z_k, \tag{10}$$

$$z_k = \Gamma_k z_{k+p}. \tag{11}$$

Therefore, $P_k$, $k = 1, \ldots, p$, can be obtained by two methods. The first method uses Schur decomposition:

$$
\begin{bmatrix} T_{11k} & T_{12k} \\ T_{21k} & T_{22k} \end{bmatrix}^T \Pi_k \begin{bmatrix} T_{11k} & T_{12k} \\ T_{21k} & T_{22k} \end{bmatrix} = \begin{bmatrix} S_{11k} & S_{12k} \\ 0 & S_{22k} \end{bmatrix},
\tag{12}
$$

where $S_{11k}$ is upper-triangular and has all of its eigenvalues inside the unique circle. The periodic solution $P_k$, $k = 1, \ldots, p$, is given by \[15\]

$$P_k = T_{21k} T_{11k}^{-1}. \tag{13}$$

The second method uses Schur decomposition:

$$
\begin{bmatrix} W_{11k} & W_{12k} \\ W_{21k} & W_{22k} \end{bmatrix}^T \Gamma_k \begin{bmatrix} W_{11k} & W_{12k} \\ W_{21k} & W_{22k} \end{bmatrix} = \begin{bmatrix} U_{11k} & U_{12k} \\ 0 & U_{22k} \end{bmatrix},
\tag{14}
$$

where $W_{11k}$ is upper-triangular and has all of its eigenvalues outside the unique circle. The periodic solution $P_k$, $k = 1, \ldots, p$, is given by \[10\]

$$P_k = W_{21k} W_{11k}^{-1}. \tag{15}$$

**Remark 2.1.** When $A_k$ and $Q_k$ are constant matrices, the second method is much efficient because $F_k$ becomes a constant matrix and $F_k^{-1} = \cdots = F_{k+p-1}^{-1} = F^{-1}$, which makes the computation of \[9\] much more efficient than the computation of \[8\].
3 Periodic LQR design based on linear time-invariant system

We propose a lifting method in this section to convert the discrete-time linear periodic system into an augmented linear time-invariant system. Thereby, the periodic LQR design is reduced to the LQR design for the augmented linear time-invariant system.

To simplify our discussion, let us consider a periodic system with $p = 3$. We will use $k$ for the discrete-time in the periodic system and $K$ for the discrete-time in the augmented system.

$$
x_1 = A_0x_0 + B_0u_0,
\quad x_2 = A_1x_1 + B_1u_1,
\quad x_3 = A_2x_2 + B_2u_2,
\quad x_4 = A_0x_3 + B_0u_3,
\quad x_5 = A_1x_4 + B_1u_4,
\quad x_6 = A_2x_5 + B_2u_5,
\quad x_7 = A_0x_6 + B_0u_6,
\quad \vdots
$$

We can easily regroup the periodic system and rewrite it as the following form:

$$
\begin{align*}
\bar{x}_1 &= \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & A_0 \\ 0 & 0 & A_1A_0 \\ 0 & 0 & A_2A_1A_0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_0 & 0 & 0 \\ A_1B_0 & B_1 & 0 \\ A_2A_1B_0 & A_2B_1 & B_2 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix} \\
= \bar{A}\bar{x}_0 + \bar{B}\bar{u}_0, \\
\bar{x}_2 &= \begin{bmatrix} x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} 0 & 0 & A_0 \\ 0 & 0 & A_1A_0 \\ 0 & 0 & A_2A_1A_0 \end{bmatrix} \begin{bmatrix} x_3 \\ x_4 \\ x_5 \end{bmatrix} + \begin{bmatrix} B_0 & 0 & 0 \\ A_1B_0 & B_1 & 0 \\ A_2A_1B_0 & A_2B_1 & B_2 \end{bmatrix} \begin{bmatrix} u_3 \\ u_4 \\ u_5 \end{bmatrix} \\
= \bar{A}\bar{x}_1 + \bar{B}\bar{u}_1,
\end{align*}
$$

in general, for $k \geq 0$ ($K \geq 0$), we have

$$
\bar{x}_{K+1} := \begin{bmatrix} x_{pk+1} \\ x_{pk+2} \\ x_{pk+3} \end{bmatrix} = \begin{bmatrix} 0 & 0 & A_0 \\ 0 & 0 & A_1A_0 \\ 0 & 0 & A_2A_1A_0 \end{bmatrix} \begin{bmatrix} x_{pk+1} \\ x_{pk+2} \\ x_{pk+3} \end{bmatrix} + \begin{bmatrix} B_0 & 0 & 0 \\ A_1B_0 & B_1 & 0 \\ A_2A_1B_0 & A_2B_1 & B_2 \end{bmatrix} \begin{bmatrix} u_{pk} \\ u_{pk+1} \\ u_{pk+2} \end{bmatrix} \\
= \bar{A}\bar{x}_K + \bar{B}\bar{u}_K,
$$

where

$$
\bar{x}_0 = \begin{bmatrix} x_{-2} \\ x_{-1} \\ x_0 \end{bmatrix} := \begin{bmatrix} 0 \\ 0 \\ x_0 \end{bmatrix}, \quad \bar{u}_0 = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix}.
$$

It is worthwhile to note that (17) is a linear time-invariant system. We can easily extend the result to the general case. Let

$$
\bar{x}_K = \begin{bmatrix} x_{p(k-1)+1} \\ x_{p(k-1)+2} \\ \vdots \\ x_{p(k-1)+p} \end{bmatrix}, \quad \bar{x}_0 := \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \bar{u}_K = \begin{bmatrix} u_{pk} \\ u_{pk+1} \\ \vdots \\ u_{pk+p-1} \end{bmatrix}.
$$

4
We will use the following notation:

$$
\bar{x}_{K+1} = \begin{bmatrix}
    x_{pk+1} \\
    x_{pk+2} \\
    \vdots \\
    x_{pk+p}
\end{bmatrix}, \quad \bar{u}_{K+1} = \begin{bmatrix}
    u_{p(k+1)} \\
    u_{p(k+1)+1} \\
    \vdots \\
    u_{p(k+1)+p-1}
\end{bmatrix}.
$$

**Theorem 3.1** Given a linear periodic discrete-time system with period of \( p \) as follows:

\[
x_{pk+1} = A_0 x_{p(k-1)+1} + B_0 u_{pk}, \\
x_{pk+2} = A_1 x_{p(k-1)+2} + B_1 u_{pk+1}, \\
\vdots \\
x_{pk+p} = A_{p-1} x_{p(k-1)+p} + B_{p-1} u_{pk+p-1}.
\]

Then, this discrete-time periodic system is equivalent to the linear time-invariant system given as follows:

\[
\bar{x}_{K+1} := \begin{bmatrix}
    x_{pk+1} \\
    x_{pk+2} \\
    \vdots \\
    x_{pk+p}
\end{bmatrix} = \begin{bmatrix}
    0 & \cdots & 0 & A_0 \\
    0 & \cdots & 0 & A_1 A_0 \\
    \vdots & \vdots & \vdots & \vdots \\
    0 & \cdots & 0 & A_{p-1} \ldots A_2 A_1 A_0
\end{bmatrix} \begin{bmatrix}
    x_{p(k-1)+1} \\
    x_{p(k-1)+2} \\
    \vdots \\
    x_{p(k-1)+p}
\end{bmatrix} + \begin{bmatrix}
    B_0 \\
    A_1 B_0 \\
    \vdots \\
    A_{p-1} \ldots A_1 B_0
\end{bmatrix} \begin{bmatrix}
    u_{pk} \\
    u_{pk+1} \\
    \vdots \\
    u_{pk+p-1}
\end{bmatrix},
\]

where \( A \in \mathbb{R}^{pn \times pn} \) and \( \bar{B} \in \mathbb{R}^{pn \times pm} \). Moreover, the structure of \( \bar{B} \) matrix guarantees the causality of the system \([19]\).

It is worthwhile to emphasize that there is no overlap between \( \bar{x}_{K+1} \) and \( x_K \); in addition, there is no overlap between \( \bar{u}_{K+1} \) and \( u_K \). This is the major difference between the proposed lifting method and the existing lifting methods in \([8,9]\) (see also \([13]\)). This feature makes it possible to apply existing design methods for linear time-invariant system \([19]\) which is equivalent to the linear periodic system \([18]\). In the remainder of the paper, we will investigate the LQR design for the system \([19]\). The LQR state feedback control is to find the optimal \( \bar{u}_K \) to minimize the following quadratic cost function

\[
\lim_{N \to \infty} \left( \min \frac{1}{2} \bar{x}_N^T Q_N \bar{x}_N + \frac{1}{2} \sum_{K=0}^{N-1} \bar{x}_K^T \bar{Q}_K \bar{x}_K + \bar{u}_K^T \bar{R}_K \bar{u}_K \right)
\]

where

\[
\bar{Q}_K := \text{diag}(Q_0, \ldots, Q_{p-1}) \geq 0, \quad \bar{R}_K := \text{diag}(R_0, \ldots, R_{p-1}) > 0,
\]

are constant matrices and the initial condition \( \bar{x}_0 \) is given. It is straightforward to see that the optimal control problem described by \([19]\) and \([20]\) is time-invariant but equivalent to the time-varying periodic system described by \([11]\) and \([2]\). Moreover, the optimal feedback matrix of the system \([19, 20]\) is given as follows:

\[
\bar{u}_K = - (\bar{R} + \bar{B}^T \bar{P} \bar{B})^{-1} \bar{B}^T \bar{P} \bar{A} \bar{x}_K,
\]

where \( \bar{P} \) is the solution of the following time-invariant algebraic Riccati equation:

\[
\bar{A}^T \bar{P} \bar{A} - \bar{A}^T \bar{P} \bar{B} (\bar{R} + \bar{B}^T \bar{P} \bar{B})^{-1} \bar{B}^T \bar{P} \bar{A} + \bar{Q} = 0.
\]
Notice that \( \bar{A} \) is not invertible, this algebraic Riccati equation cannot be directly solved by using the algorithms for time-invariant algebraic Riccati equation proposed in [17, 18], but it can be solved using the algorithm proposed in [19]. In this paper, we propose a more efficient algorithm than the one of [19]. The new algorithm makes full use of the specific structure of \( \bar{A} \) in which the first \((p-1)n\) columns are zeros. Denote

\[
\bar{Q} := \bar{Q}_K = \begin{bmatrix}
\text{diag}(Q_0, \ldots, Q_{p-2}) & 0 \\
0 & Q_{p-1}
\end{bmatrix} = \text{diag}(Q_1, Q_2),
\]

where \( Q_1 = \text{diag}(Q_0, \ldots, Q_{p-2}) \in \mathbb{R}^{(p-1)n \times (p-1)n} \) and \( Q_2 = Q_{p-1} \in \mathbb{R}^{n \times n} \),

\[
\bar{R} := \bar{R}_K = \begin{bmatrix}
\text{diag}(R_0, \ldots, R_{p-2}) & 0 \\
0 & R_{p-1}
\end{bmatrix} = \text{diag}(R_1, R_2),
\]

where \( R_1 = \text{diag}(R_0, \ldots, R_{p-2}) \in \mathbb{R}^{(p-1)n \times (p-1)n} \) and \( R_2 = R_{p-1} \in \mathbb{R}^{m \times m} \). Let

\[
\bar{A} = \begin{bmatrix}
0 & \cdots & 0 & A_0 \\
0 & \cdots & 0 & A_1 A_0 \\
\vdots & \vdots & \vdots & \vdots \\
0 & \cdots & 0 & A_{p-2} \cdots A_1 A_0 \\
0 & \cdots & 0 & A_{p-1} \cdots A_2 A_1 A_0
\end{bmatrix} = \begin{bmatrix}
0 & 0 \\
0 & A_0 \\
0 & 0 \\
0 & \vdots \\
0 & A_{p-2} \cdots A_1 A_0 \\
0 & A_{p-1} \cdots A_2 A_1 A_0
\end{bmatrix},
\]

\[
\bar{F} = \begin{bmatrix}
\bar{A}_1 & \cdots & \bar{A}_2
\end{bmatrix}^T \in \mathbb{R}^{n \times n},
\]

\[
\bar{B} = \begin{bmatrix}
\bar{B}_0 & 0 & \cdots & 0 & 0 \\
A_1 \bar{B}_0 & \bar{B}_1 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
A_{p-2} \cdots A_1 \bar{B}_0 & A_{p-2} \cdots A_2 \bar{B}_1 & \cdots & \bar{B}_{p-2} & 0 \\
A_{p-1} \cdots A_1 \bar{B}_0 & A_{p-1} \cdots A_2 \bar{B}_1 & \cdots & \bar{B}_{p-1} & \bar{B}_{p-1}
\end{bmatrix},
\]

\[
\bar{P} = \begin{bmatrix}
\bar{P}_{11} & \bar{P}_{12} \\
\bar{P}_{21} & \bar{P}_{22}
\end{bmatrix},
\]

where \( \bar{B}_1 \in \mathbb{R}^{(p-1)n \times m} \) and \( \bar{B}_2 \in \mathbb{R}^{n \times m} \), and

\[
\bar{Y} = \bar{P} \bar{B}^T R + \bar{B}^T \bar{P} \bar{B}^{-1} \bar{B}^T \bar{P}.
\]

Substituting (24), (25), (26), (27), (28), and (29) into (28) yields

\[
\begin{bmatrix}
0 \\
F^T
\end{bmatrix} \bar{P} \begin{bmatrix}
0 & F \\
P_{11} & P_{12} \\
P_{21} & P_{22}
\end{bmatrix} - \begin{bmatrix}
0 \\
F^T
\end{bmatrix} \bar{Y} \begin{bmatrix}
0 & F \\
P_{11} & P_{12} \\
P_{21} & P_{22}
\end{bmatrix} + \begin{bmatrix}
\bar{Q}_1 & 0 \\
0 & \bar{Q}_2
\end{bmatrix} = 0,
\]

or equivalently

\[
\begin{bmatrix}
0 & 0 \\
0 & F^T \bar{P} \bar{F}
\end{bmatrix} - \begin{bmatrix}
\bar{P}_{11} & \bar{P}_{12} \\
\bar{P}_{21} & \bar{P}_{22}
\end{bmatrix} - \begin{bmatrix}
0 & 0 \\
0 & F^T \bar{Y} \bar{F}
\end{bmatrix} + \begin{bmatrix}
\bar{Q}_1 & 0 \\
0 & \bar{Q}_2
\end{bmatrix} = 0.
\]

This proves \( \bar{P}_{12} = \bar{P}_{21} = 0 \) and \( \bar{P}_{11} = \bar{P}_{11}^T = \bar{Q}_1 \). By examining the lower right block of (31), we have

\[
F^T \bar{P} \bar{F} = \bar{A}_1^T \bar{Q}_1 \bar{A}_1 + \bar{A}_2^T \bar{P}_{22} \bar{A}_2 = \mathbb{R}^{n \times n},
\]

(32)
and
\[
\begin{align*}
F^T Y F &= \begin{bmatrix} A_1^T, A_2^T \end{bmatrix} \begin{bmatrix} Q_1 B_1 \\ P_{22} B_2 \end{bmatrix} \begin{bmatrix} R + B_1^T Q_1 B_1 + B_2^T P_{22} B_2 \end{bmatrix}^{-1} \begin{bmatrix} B_1^T Q_1 \\ B_2^T P_{22} \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \\
&= \begin{bmatrix} \tilde{A}_1^T Q_1 B_1 + \tilde{A}_2^T P_{22} B_2 \end{bmatrix} \begin{bmatrix} R + B_1^T Q_1 B_1 + B_2^T P_{22} B_2 \end{bmatrix}^{-1} \begin{bmatrix} B_1^T Q_1 A_1 + B_2^T P_{22} A_2 \end{bmatrix}.
\end{align*}
\]

Let
\[
\begin{align*}
\hat{A} &= \hat{A}_2 \in \mathbb{R}^{n \times n}, \\
\hat{B} &= \hat{B}_2 \in \mathbb{R}^{n \times pm}, \\
\hat{Q} &= \hat{Q}_2 + \hat{A}_1^T \hat{Q}_1 \hat{A}_1 \in \mathbb{R}^{n \times n}, \\
\hat{R} &= \hat{R} + \hat{B}_1^T \hat{Q}_1 \hat{B}_1 \in \mathbb{R}^{pm \times pm}, \\
\hat{S} &= \hat{A}_1^T \hat{Q}_1 \hat{B}_1 \in \mathbb{R}^{n \times pm}, \\
\hat{P} &= \hat{P}_{22} \in \mathbb{R}^{n \times n}.
\end{align*}
\]

We can rewrite the lower right block of (31) as follows:
\[
\hat{A}^T \hat{P} \hat{A} - \hat{P} - \left( \hat{A}^T \hat{P} \hat{B} + \hat{S} \right) \left( \hat{B}^T \hat{P} \hat{B} + \hat{R} \right)^{-1} \left( \hat{B}^T \hat{P} \hat{A} + \hat{S}^T \right) + \hat{Q} = 0.
\]

The Riccati equation (35) is a special case discussed in [20]. An efficient Matlab function \texttt{dare} that implements an algorithm of [20] is available to solve (35).

**Remark 3.1** Comparing to the methods described in the previous section which need to solve \( p \) \( n \)-dimensional discrete-time Riccati equations, we need only to solve one \( n \)-dimensional discrete-time Riccati equation using the method proposed in this section.

To compare the efficiency of the method to the ones developed in [15, 16], we will not use the Matlab function \texttt{dare} because \texttt{dare} calculates more information than the solution of the Riccati equation (35). Let \( \hat{B} = \hat{B}, \hat{R} = \hat{R} \),
\[
\hat{A} = \hat{A} - \hat{B} \hat{R}^{-1} \hat{S}^T,
\]
and
\[
\hat{Q} = \hat{Q} - \hat{S} \hat{R}^{-1} \hat{S}^T.
\]

Riccati equation (34) can be solved by either eigen-decomposition or Schur decomposition for the following generalized eigenvalue problem [20, page 1748, equation (8)]:
\[
\lambda \begin{bmatrix} I & \hat{B} \hat{R}^{-1} \hat{B}^T \\ 0 & \hat{A}^T \end{bmatrix} - \begin{bmatrix} \hat{A} & 0 \\ -\hat{Q} & I \end{bmatrix} := \lambda E - F.
\]

If \( \hat{A} \) is invertible, the problem is reduced to solve the following eigenvalue problem [16, equation (30)]:
\[
Z = F^{-1} E = \begin{bmatrix} \hat{A}^{-1} & \hat{A}^{-1} \hat{B} \hat{R}^{-1} \hat{B}^T \\ \hat{Q} \hat{A}^{-1} & \hat{A}^T + \hat{Q} \hat{A}^{-1} \hat{B} \hat{R}^{-1} \hat{B}^T \end{bmatrix}.
\]

Using Schur decomposition for (39), we have
\[
\begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}^T Z \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} \\ 0 & S_{22} \end{bmatrix}.
\]
where $S_{11}$ is upper-triangular and has all of its eigenvalues outside the unique circle. The solution of the discrete algebraic Riccati equation (35) is given by

$$\hat{P} = W_0 W_1^{-1}.$$  \hfill (41)

We summarize the proposed algorithm as follows:

**Algorithm 3.1**

*Data:* $A_0, \ldots, A_{p-1}, B_0, \ldots, B_{p-1}, Q_0, \ldots, Q_{p-1}, R_0, \ldots, R_{p-1}.$

1. **Step 1:** Form

$$A_1 = \begin{bmatrix} A_0 & 0 & \ldots & 0 \\ A_1 & B_0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A_{p-2} & A_{p-1} & \ldots & B_{p-1} \end{bmatrix}, \quad B_1 = \begin{bmatrix} B_0 & 0 & \ldots & 0 \\ A_1 & B_0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A_{p-2} & A_{p-1} & \ldots & B_{p-1} \end{bmatrix}.$$  \hfill (42a)

$$A_2 = A_{p-1} \ldots A_2 A_1 A_0, \quad B_2 = \begin{bmatrix} A_{p-1} \ldots A_2 A_1 B_0, & A_{p-1} \ldots A_2 B_1, & \ldots & B_{p-1} \end{bmatrix},$$  \hfill (42b)

$$Q_1 = \text{diag}(Q_0, \ldots, Q_{p-2}), \quad Q_2 = Q_{p-1}, \quad \hat{Q}_1 = \text{diag}(R_0, \ldots, R_{p-2}), \quad \hat{Q}_2 = R_{p-1}. \hfill (42c)$$

2. **Step 2:** Form $\hat{A}, \hat{B}, \hat{Q}, \hat{R},$ and $\hat{S}$ using (42).

3. **Step 3:** Find the solution $\hat{P}$ of the discrete-time algebraic Riccati equation (35) using the algorithm of [21] implemented as dare or using the algorithm of [10], i.e., (41) and (44).

4. **Step 4:** The solution of the discrete-time algebraic Riccati equation (23) is given by

$$\hat{P} = \text{diag}(\hat{Q}_1, \hat{P}). \hfill (43)$$

Given $x_K$, we can calculate the feedback control by (22). Applying this feedback control to (19), we obtain the next state $x_{K+1}$.

**Remark 3.2** The proposed lift method for discrete linear periodic time-varying system can be used for other design methods such as $H_\infty$ and robust pole assignment design.

### 4 Implementation and numerical simulation

In this section, we discuss some details of implementation that will reduce some computation time comparing to the directly implementation described in the previous section. We also report the test of the proposed algorithm for some real world problems discussed in [10, 22]. We compare the test result with the ones obtained in [10].

#### 4.1 Implementation consideration

The most expensive calculations in Algorithm 3.1 are the calculation of $\hat{Q}, \hat{R},$ and $\hat{S}$ in Step 2, and the calculation of $\hat{R}^{-1} = \hat{R}^{-1}$ in Step 3. It is easy to check (cf. [21]): (1) direct calculation of $Q$ requires $O(2(p-1)^2 n^3) + O(2(p-1)^2 n^3) + O(n^2)$ flops, (2) direct calculation of $\hat{R}$ requires $O(2p(p-1)^2 n^2 m) + O(2p^2(p-1)nm^2) + O(p^2 m^2)$ flops, (3) direct calculation of $\hat{S}$ requires $O(2(p-1)^2 n^3) + O(2(p-1)^2 n^3)$ flops, and (4) directly calculation of $\hat{R}^{-1}$ requires $O(p^2 m^2)$ flops. For extremely large $p$, i.e., very long period of the system, the majority of the computation is the computation of $\hat{R}$ and $\hat{R}^{-1}$.

Let $Q_A = Q_A^2 A_1 \in \mathbb{R}^{(p-1)n \times n}$ and $Q_B = Q_B^2 B_1 \in \mathbb{R}^{(p-1)n \times pm}$. We will use using Matlab notation for sub-matrices. Since $Q_1, \hat{Q}_2,$ and $\hat{R}$ are positive diagonal matrices, we can calculate $\hat{Q}_1, \hat{S}_2,$ and $\hat{R}$ in (41) more efficiently as follows:
for $i = 1 : (p - 1)n$

$$Q_A(i,:) = Q_1^T(i,i)A_1(i,:);$$

end

$$\hat{Q} = Q_A^TQ_A$$

for $i = 1 : n$

$$\hat{Q}(i,i) = \hat{Q}(i,i) + Q_2(i,i);$$

end

for $i = 1 : (p - 1)n$

$$Q_B(i,:) = Q_1^T(i,i)B_1(i,:);$$

end

$$\hat{R} = Q_A^TQ_B$$

for $i = 1 : pm$

$$\hat{R}(i,i) = \hat{R}(i,i) + \hat{R}(i,i);$$

end

$$\hat{S} = Q_A^TQ_B$$

It is easy to check (cf. [21]) the flops for the following calculations: (1) the calculation of $\hat{Q}$ requires $O((p-1)n) + O((p-1)n^2) + O(2(p-1)^2n^3) + O(n)$ flops, (2) the calculation of $\hat{R}$ requires $O(p(p-1)nm) + O(2p^2(p-1)nm^2) + O(pm)$, (3) the calculation of $\hat{S}$ requires $O(2(p-1)pn^2m)$ flops, (4) this does not save the computation of $R^{-1}$.

4.2 Simulation test for the problem of [16]

The first simulation test problem is the spacecraft attitude control design using only magnetic torques discussed in [16]. The number of states of this system is $n = 6$. The number of control inputs of this system is $m = 3$. The controllability of this problem is established in [23]. In this simulation test, we use the same discrete-time linear periodic model as in [16] with the same parameters, such as the spacecraft inertia matrix, orbital inclination, orbital altitude, weight matrices $Q$ and $R$, and the same initial conditions.

Using $p = 100$, $p = 500$, and $p = 1000$, we run all three algorithms (one proposed in this paper, one proposed in [16], and one proposed in [15]) for this design and recorded the CPU times for all three algorithms. The result is presented in Table 1.

| Samples per period | Algorithm 3.1 | Yang Algorithm [16] | Hench-Laub Algorithm [15] |
|--------------------|---------------|---------------------|--------------------------|
| 100                | 0.0097 (s)    | 0.0757 (s)          | 0.2711 (s)               |
| 500                | 0.2528 (s)    | 1.6042 (s)          | 6.5435 (s)               |
| 1000               | 4.2821 (s)    | 6.3155 (s)          | 25.8996 (s)              |

Table 1: CPU time comparison for problem in [16]

Clearly, the proposed method is significantly cheaper than the algorithms in [15] [16].
4.3 Simulation test for the problem of \[22\]

The second simulation test problem is a combined method for the spacecraft attitude and desaturation control design using both reaction wheels and magnetic torques discussed in \[22\]. The number of states of this system is \(n = 9\). The number of control inputs of this system is \(m = 6\). The controllability of problem is guaranteed because three reaction wheels are assumed to be available.

Using the parameters provided in \[22\], for \(p = 100\), \(p = 500\), and \(p = 1000\), we obtained solutions for the corresponding algebraic Riccati equations and recorded the CPU times for all three algorithms. The result is presented in Table 2.

| Samples per period | Algorithm 3.1 | Yang Algorithm \[16\] | Hench-Laub Algorithm \[15\] |
|--------------------|---------------|----------------------|-----------------------------|
| 100                | 0.0284 (s)    | 0.1120 (s)           | 0.3807 (s)                  |
| 500                | 3.6376 (s)    | 2.5629 (s)           | 9.0144 (s)                  |
| 1000               | 38.4912 (s)   | 10.0629 (s)          | 36.0690 (s)                 |

Table 2: CPU time comparison for problem in \[22\]

For this problem, \(m = 6\) is twice as large as the previous problem, the proposed algorithm is faster than the algorithms developed in \[15, 16\] when the total number of samples in one period is moderate (\(p = 100\) samples per period), but when the total number of samples in one period increases (to \(p = 500\) or \(p = 1000\) samples per period), the advantage of the proposed algorithm will be lost because the computation of the inverse of \(\mathbf{R} \in \mathbb{R}^{6000 \times 6000}\) is \(\mathcal{O}(p^3 m^3)\) which is very expensive.

5 Conclusion

In this paper, we propose a new lifting method to convert the discrete-time linear periodic system to a discrete-time linear time-invariant system. The LQR design for the discrete-time linear periodic system is then reduced to the LQR design for the discrete-time linear time-invariant system. By applying the standard algorithm to the discrete-time algebraic Riccati equation associated with the augmented LTI system and using the special structure of the augmented LTI system, we derived an efficient algorithm for the LQR design for the discrete time linear periodic system. We show that the new algorithm is very efficient compared to the existing ones for the discrete-time periodic algebraic Riccati equation, in particular when the number of samples in one period is moderate. We demonstrated this property by the numerical test for the design problems of spacecraft attitude control using magnetic torques. The proposed lift method for discrete linear periodic time-varying system can be used for other design methods such as \(H_\infty\) and robust pole assignment designs.

References

[1] Lovera, M., and Astolfi, A., “Spacecraft attitude control using magnetic actuators,” *Automatica*, Vol. 40, 2004, pp. 1405-1414.

[2] Arcara, M., Bittanti, S., and Lovera, M., “Periodic control of helicopter rotors for attenuation of vibrations in forward flight,” *IEEE Transactions on Control System Technology*, Vol. 8, 2000, pp. 883-894.

[3] Stol, K.A., “Time-varying control of wind turbines,” *Proceedings of American Control Conference*, 2003, pp. 3796-3796.

[4] Zhang, L., and Hristu-Varsakelis, D., “Communication and control co-design for networked control systems”, *Automatica*, Vol. 42, 2006, pp.953-958.

[5] Khargonekar, P.P., and Sivashankar, N., “\(H_2\) optimal control for sampled-data systems”, *Systems & Control Letters*, Vol. 17, 1991, pp.425-436.
[6] Jury, E.I and Mullin, F.J. “A note on the operational solution of linear difference equations”, *J. Franklin Inst.*, Vol. 266, 1958, pp.189-205.

[7] Jury, E.I and Mullin, F.J. “The analysis of sampled data control systems with a periodically time-varying sampling rate”, IRE Transactions on Automatic Control, Vol. AC-4, 1959, pp. 15-21.

[8] Meyer, R.A. and Burrus, C.S. “A unified analysis of multirate and periodically time-varying digital filters”, IEEE Transactions on Circuits and System, 22(3), 1975, pp 162-168.

[9] Grasselli, O.M., and Longhi, S., “Pole-placement for non-reachable periodic discrete-time system,” *Mathematics of Control, Signals and Systems*, Vol.4, 1991, pp. 439-455.

[10] Bittanti, S., Colaneri, P., and Guardabassi, G., “Analysis of the periodic Lyapunov and Riccati equations via canonical decomposition,” *SIAM J. Control and Optimization*, Vol. 24, no. 6, 1986, pp. 1138-1149.

[11] Bittanti, S., Colaneri, P., and Nicolao, G.D., “A note on the maximal solution of the periodic Riccati equation,” *IEEE Transactions on Automatic Control*, Vol. 15, No. 12, 1989, pp.1316-1319.

[12] Bittanti, S., “Periodic Riccati equation,” *The Riccati Equation*, edited by S. Bittanti, et. al, Springer, Berlin, 1991, pp. 127-162.

[13] Varga, A., “On solving periodic Riccati equations,” *Numerical Linear Algebra with Applications*, Vol. 15, No. 12, 2008, pp.809-835.

[14] Varga, A., “Computational issues for linear periodic system: paradigms, algorithms, open problems”, *International Journal of Control*, Vol. 86, No. 7, 2013, pp. 1227-1239.

[15] Hench, J.J., and Laub, A.J., “Numerical solution of the discrete-time periodic Riccati equation”, *IEEE Transactions on Automatic Control*, Vol. 39, No.6, 1994, pp. 1197-1210.

[16] Yang, Y., “An Efficient Algorithm for Periodic Riccati Equation with Periodically Time-Varying Input Matrix”, *Automatica*, accepted, 2017, pp. ?-?.

[17] Vaughan D.R. “A nonrecursive algebraic solution for the discrete Riccati equation,” *IEEE Transactions on Automatic Control*, Vol. 15, No. 5, 1970, pp.597-599.

[18] Laub, A.J., “A Schur method for solving algebraic Riccati equations,” *IEEE Transactions on Automatic Control*, Vol. 24, No. 6, 1979, pp. 913-921.

[19] Pappas, T., Laub, A.J., and Sandell, N.R., “On the numerical solution of the discrete-time algebraic Riccati equation,” *IEEE Transactions on Automatic Control*, Vol. 25, No.4, 1980, pp. 631-641.

[20] Arnold, W., Laub, A.J., “Generalized eigenproblem algorithms and software for algebraic Riccati equation,” *Proceedings of the IEEE*, Vol. 72, No.12, 1984, pp. 1746-1754.

[21] Golub, G.H., and Van Loan, C. F., *Matrix computations, 4th Edition*, Johns Hopkins University Press, Baltimore, USA, 1993.

[22] Yang, Y., “Spacecraft Attitude and Reaction Wheel Desaturation Combined Control Method,” to appear in *IEEE Transactions on Aerospace and Electronic System*, available in [arXiv:1507.06963](http://arxiv.org/abs/1507.06963) 2016.

[23] Yang, Y., “Controllability of spacecraft using only magnetic torques,” *IEEE Transactions on Aerospace and Electronic System*, Vol. 52, No. 2, 2016, pp. 954-961.