Quantum Computing for the time-varying Linear Quadratic Regulator

Abstract: This paper deals with the time-varying Linear-Quadratic Regulator(LQR) problem which is a special case of the general optimal control problem. An explicit value of the control that yields an optimal solution of the LQR depends on the integrabiltiy of the Matrix Riccati equation. Unfortunately, there exists no general method to solve analytically the Riccati equation. Analytic solutions depend on the relationship between the coefficients of the Riccati equation. In this paper, we use a quantum computer algorithm to derive the solution of the matrix Riccati equation. Since quantum computer algorithms have the potential to implement faster approximate solutions to the Riccati equations compared with strictly classical algorithms, this yields a faster implementation and explicit value of the control.

1 Introduction

The Linear-Quadratic Regulator(LQR) in which the dynamics are linear, the costs are quadratic, appears to be the simplest case in dynamic programming problem for continuous system [1][2] [3]. It is the most well-studied problem in Optimal Control Theory and seems to yield the most important and influential results in optimal control theory to date. LQR finds its applications in many branches of science ranging from engineering, econometry [4] and mathematics to biomedical [5] and management sciences [6]. It has been applied successfully to problems in economic stabilization policy [7], in circuits [8], in social science [9],in statistics [10], in robotics [11] , in water delivery canal [12] and in solving aviation full tracking problem in aircraft system identification and control [13] . For the most of optimal control problem, it is hard indeed impossible to compute an analytical value of the control function. We mostly rely on numerical techniques to approximate the value of the control. Generally speaking , computational algorithms are inevitable in solving optimal control problems . The LQR is a notable exception in which the control can be computed analytically. The control can be expressed explicitly in term of the solution of a matrix

Riccati equation. Different variant of LQR have been considered in the past and several algorithms have been proposed to estimate the control function. The finite-horizon, invariant, continuous time was considered by many authors. In general, The control is derived using the discrete matrix Riccati equation or continuous Riccati equation [14][15]. In [16] The authors used a method that consists of turning the matrix Riccati equation into a Lyapunov differential equation which can be solved using the tensor product. Another method was proposed in [17] , the authors proposed a solution of the Riccati equation K(t) as a ratio of two functions that is K(t)=(P(t))/(f(t)) where f(t) and P(t) are solutions of first order linear differential equations. The infinite-horizon, invariant-time, discrete time or continuous time has been studied by several authors . Unlike the finite-horizon, the control is computed by solving the algebraic Riccati equation [18] [19] [20].

In this paper, we consider the finite-horizon, time-variant, continuous time LQR. We investigate the use of quantum algorithms to generate approximate value of the control function. It has been shown [21] that under certain conditions, quantum algorithms to find the solution of systems of linear differential equations can yield an exponential improvement in execution time compared with the best-known classical algorithms.

Our approach involves a change of variable which turns a matrix Riccati equation into an approximation using linear differential equations with second order non-constant coefficients. We can then solve matrix equations equivalent to the set of linear differential equations using a version of the Harrow-Hassidim-Lloyd (HHL) quantum algorithm [21], for the common case of Hermitian matrices.

The paper is organized as follow. Chapter 2 and chapter 3 are respectively an overview of the Optimal Control Theory and the time-varying LQR. In chapter 4, we explain the approach on how to turn the matrix Riccati equation into a set of matrix equations equivalent to a set of system of differential equations. chapter 5 describes the algorithm for finding the control function. Chapter 6 is dedicated to the conclusion.

2 Optimal Control Theory

The field of Optimal Control, as the name suggests, is a branch of mathematics that deals with analyzing a system to find solutions that cause it to behave optimally for the cost we are willing to pay. If a system is controllable [22] given an initial state and some assumptions, then we can reach a desired state of the system by finding the appropriate control with minimum cost. Control is handled through a feedback into u that depends on the state of the system . A basic optimal control problem can be stated as follows: Given the system of differential equations along with an initial condition,

$$(S)\ \frac{dx}{dt} = f(x(t), u(t)), \quad x(t_0) = x_0 \tag{1}$$

where $x(t)$ is the state of the system, $x(t) \in \mathbb{R}^n$, and $u(t) \in \mathbb{R}^m$ is the control.

The goal is to find a control $u(t)$ over $[t_0, t_f]$ which for any x_0, minimizes the cost function

$$T(x,u) = \int_{t_0}^{t_f} L(x(t), u(t))dt$$

To better frame the optimal control problem, let's consider a simple example.

Example 1. *Consider the circuit below that consists of resistor, inductor and a source(RL*

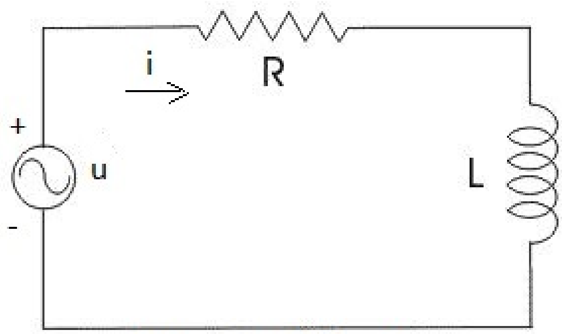

Figure 1. RL circuit

circuit).

The circuit in figure 1 is very frequent in electronic device for filtering signals.
The behavior of the resistor is specified by a constant R called resistance.
The behavior of the inductor is specified by a constant L called inductance.
i is the current across drop the circuit. u is the control and represents the voltage across the source.
According to the Kirchoff's law of current, the sum of the voltage drop across the circuit in figure 1 equals the voltage across the source, therefore,

$$V_R + V_L = u$$

$$Ri + L\frac{di}{dt} = u$$

Let i_o be the initial value of the current that is $i(0) = i_0$
we deal with the system:

$$Ri + L\frac{di}{dt} = u$$
$$i(0) = i_0$$

Suppose that we want to switch the current i from i_0 to another value i_1 at $t = T$ that is $i(T) = i_1$ with a minimum cost.
the goal is expressed by the cost functional.

$$J(i,u) = \frac{1}{2}\int_0^T c(i(t) - i_1)^2 dt + \frac{1}{2}\int_0^T c_u(u(t))^2 dt$$

where c and c_u are positive constants and T is the fixed final time $T > 0$
The first integral is the state cost and the second integral is the control cost.
We also can assume that that u belongs to a set of admissible control

$$U_{ad} = \{u \in L^2([0,T]) | k_1 \leq u \leq k_2, t \in [0,1]\}$$

where k_1 and k_2 are constant real numbers.
Question: What values of $u(t)$ allow this switch with minimum cost?

2.1 General statement of the optimal control problem. The Pontryagin principle.

The basic optimal control problem $(\mathcal{P})$ can be stated as follow: Given the system of differential equations along with an initial condition:

$$\frac{dx}{dt} = f(x(t), u(t)), \qquad x(t_0) = x_0 \tag{2}$$

where $x(t)$ is the state of the system $x(t) \in \mathbb{R}^n$ and $u(t)$ is the input of the system $u(t) \in \mathbb{R}^m$. The goal is to find a control $u(t)$ over $[t_0, t_f]$ that minimizes the cost functional

$$J(x, u) = \int_{t_0}^{t_f} L(x(t), u(t))dt.$$

To solve an optimal control problem, we can use the Pontryagin principle which represents some necessary conditions the optimal control $u^*(t)$ and the optimal state $x^*(t)$ need to satisfy.

Theorem 1 (Pontryagin's maximum principle)**.** *If $x^*(t)$ and $u^*(t)$ are optimal for the problem (P), then there exist a function* $\lambda(t) = \begin{pmatrix} \lambda_1(t) \\ \lambda_2(t) \\ . \\ . \\ . \\ \lambda_n(t) \end{pmatrix}$ *and a function H defined as:*

$H(x(t), u(t), \lambda) = \lambda^T f(x, u) - L(x, u)$ that satisfy the three properties.

a)$H(x^*(t), u^*(t), \lambda(t)) \geq H(x^*(t), u(t), \lambda(t))$
for all control u at each time t.

b)$\frac{d\lambda}{dt} = -\nabla_x H(x^*, u^*, \lambda)$

c)$\lambda(t_f) = 0$

λ^T represents the transpose of λ and H is called the Hamiltonian.
The Pontryagin's maximum principle yields to the controls that represent the candidates for the optimal controls. Those candidates need to be tested. The following theorem gives a sufficient condition for a candidate to be optimal.

Theorem 2. *Let $U(x_0)$ be the set of admissible controls of u and X an open subset of $\mathbb{R}^n$. If there exists a function $J_1 : X \to \mathbb{R}$ of class C^1 such that the three statements below are true*
i)If $u \in U$ generates the solution $x(t)$ of (7) and $x(t) \in X$ for all $t \in [t_0, t_1^*)$, then $lim_{t\to t_1} J_1(x(t)) \leq lim_{t\to t_1^*} J_1(x^*(t)) = 0$, for some $t_1^* \geq t_1$
ii)$L(x^*(t), u^*(t)) + grad^T J_1(x^*(t)) f(x^*(t), u^*(t)) = 0$ for all $t \in [t_0, t_1^*)$ for some $t_1^* \geq t_1$

iii) $L(x,u) + grad^T J_1(x) f(x,u) \geq 0$ *for all* $x \in X$ *and* $u \in U$.
Then the control $u^*(t)$ *generating the solution* $x^*(t)$ *for all* $t \in [t_0, t_1^*]$ *with* $x^*(t_0) = x_0$, *is optimal with respect to* X.

The proof of the theorem 1 and 2 can be found in [22].

Remark 1. *The proof of theorem 2 suggests that the test function* $J_1(x(t))$ *can be chosen so that :* $J_1(x_0) = \int_{t_0}^{t_f} L(x(t), u(t))$.

Remark 2. *In case we deal with a nonautonomous system that is a system in the form*

$$\frac{dx}{dt} = f(x,t,u) \quad x(t_0) = x_0$$

then we always can turn such system into an autonomous system.
We can define

$$\hat{x} = \begin{pmatrix} x_1(t) \\ x_2(t) \\ . \\ . \\ . \\ x_n(t) \\ x_{n+1}(t) \end{pmatrix}$$

where $x_{n+1}(t) = t$ *then we deal with the following autonomous system*

$$\frac{d\hat{x}}{dt} = \begin{pmatrix} f(\hat{x}, u) \\ 1 \end{pmatrix} = \hat{f}(\hat{x}, u) \quad \hat{x}(t_0) = \hat{x_0} = (x_0, t_0)$$

Also if the cost integrand depends on t *that is* $J(x,u) = \int_{t_0}^{t_f} L(x,t,u)dt$ *then we can rewrite the cost function: as*

$$\hat{J}(\hat{x}, u) = \int_{t_0}^{t_f} \hat{L}(\hat{x}, u)dt$$

Now we can refine the maximum's Pontryagin principle to the autonomous system

$$\frac{d\hat{x}}{dt} = \hat{f}(\hat{x}, u) \quad \hat{x}(t_0) = \hat{x_0}$$

with cost function given by:

$$\hat{J}(\hat{x}, u) = \int_{t_0}^{t_f} \hat{L}(\hat{x}, u)dt$$

and then $\hat{\lambda} = \begin{pmatrix} \lambda_1(t) \\ \lambda_2(t) \\ . \\ . \\ . \\ \lambda_n(t) \\ \lambda_{n+1}(t) \end{pmatrix} \in \mathbb{R}^{n+1}$

3 The time-dependent LQR-Riccati Equation.

We suppose that

$$f(x(t), u(t)) = A(t)x(t) + B(t)u(t)$$

then

$$\hat{f}(\hat{x}(t), u(t)) = \begin{pmatrix} A(t)x(t) + B(t)u(t) \\ 1 \end{pmatrix}$$

and

$$\hat{L}(\hat{x}(t), u(t)) = \frac{1}{2}(x^T(t)Q(t)x(t) + u^T(t)R(t)u(t))$$

then

$$\frac{d\hat{x}}{dt} = \begin{pmatrix} A(t)x(t) + B(t)u(t) \\ 1 \end{pmatrix} \tag{3}$$

with initial state $x(t_0) = x_0$ and the interval $[t_0, t_f]$ is specified and $x(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ . \\ . \\ . \\ x_n(t) \end{pmatrix}$

$u(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ . \\ . \\ . \\ u_m(t) \end{pmatrix}$

The cost to be minimized is: $\hat{J}(\hat{x}, u) = \frac{1}{2}\int_{t_0}^{t_f}(x^T(t)Q(t)x(t) + u^T(t)R(t)u(t))dt$ where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ $Q \in \mathbb{R}^{n\times n}$ $R \in \mathbb{R}^{m\times m}$
The matrix Q is symmetric that is $Q^T = Q$.
The matrix R is symmetric and positive definite that is $x^T Rx > 0$ if $x \neq 0$.
The functions $A(t)$, $B(t)$, $Q(t)$, and $R(t)$ are of class C^1.
We can use the Pontryagin's maximum principle to find $u(t)$.
It is already shown that $u(t)$ is given by

$$u(t) = -R^{-1}(t)B^T(t)P(t)x(t)$$

where $P(t)$ is a solution of the Riccati equation

$$\frac{dP(t)}{dt} = P(t)B(t)R^{-1}(t)B^T(t)P(t) - A^T(t)P(t) - P(t)A(t) - Q(t) \tag{4}$$

satisfying the initial condition $P(t_f) = 0$. (See proof in the Appendix(Theorem 5)).

Theorem 3. *The function $u^*(t) = -R^{-1}(t)B^T(t)P(t)x^*(t)$ is the optimal solution at x_0 and the minimum value of J is given by :*

$$J_{min} = \frac{1}{2}(x^*)^T(t_0)P(t_0)x^*(t_0)$$

where x^* is the corresponding optimal solution of (2).

the proof can be found in the Appendix.

4 Quantum computing to solve the Matrix Riccati equation

consider the Matrix Riccati equation:

$$\frac{dY}{dt} = YA(t)Y + YB(t) + C(t)Y + D(t) \tag{5}$$

where $Y \in \mathbb{R}^{n\times n}$, $A(t) \in \mathbb{R}^{n\times n}$, $B(t) \in \mathbb{R}^{n\times n}$ $C(t) \in \mathbb{R}^{n\times n}$ and $D(t) \in \mathbb{R}^{n\times n}$.

Theorem 4. *If $B = 0$ and if A is invertible, then the matrix Riccati equation (5) can be turned into the second order matrix linear differential equation.*

$$V'' - (ACA^{-} + A'A^{-1})V + ADV = 0 \tag{6}$$

using the change of variable

$$Y = -A^{-1}V'V^{-1} \tag{7}$$

where V is invertible.

The proof of the theorem is straightforward and can be found in the Appendix.
In the control problem, at some point, we need to solve the following matrix Riccati equation along with the initial.

$$\frac{dP}{dt} = PB_1R^{-1}B_1^TP - A_1^TP - PA_1 - Q(t)$$

$$P(t_f) = 0.$$

We will assume that $R(t) = A_1(t) = I$ where I represents the $n \times n$ identity.

So

$$\frac{dP}{dt} = PB_1B_1^TP - 2P - Q$$

Let's make the change of variable

$$P = -(B_1B_1^t)^{-1}V'V^{-1}$$

according to the previous theorem.
This leads to the following equation:

$$V'' - (-2B_1B_1^T(B_1B_1^T)^{-1} - (B_1B_1^T)'(B_1B_1^T)^{-1})V' - B_1B_1^TQV = 0$$

$$V'' + (2I + (B_1B_1^T)'(B_1B_1^T)^{-1})V' - B_1B_1^TQV = 0$$

Choose B_1 and Q such that $B_1B_1^TQ = I$ and $2I + (B_1B_1^T)'(B_1B_1^T)^{-1} = -S$ where S is a constant diagonal matrix So $Q = (B_1B_1^T)^{-1}$.

Notice that since $P(t_f) = 0$ then $V^{'}(t_f) = 0$.
We need to solve the initial problem:

$$V^{''} - SV^{'} - V = 0 \tag{8}$$

$$V^{'}(t_f) = 0$$

where S is a constant diagonal matrix

$$S = \begin{bmatrix} \alpha_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \alpha_n \end{bmatrix}$$

Let

$$V = (V_{ij}) \;\; 1 \leq i \leq n \;\; 1 \leq i \leq n$$

Then the previous equation yields the the following equation:

$$V^{''}_{ij} - \alpha_i V^{'}_{ij} - V_{ij} = 0 \tag{9}$$

Then V_{ij} and V_{ii} have the same general solution for any $1 \leq j \leq n$ So we can focus on V_{ii} and then derive V_{ij} from V_{ii}.
Solving

$$V^{''}_{ii} - \alpha_i V^{'}_{ii} - V_{ii} = 0 \tag{10}$$

Assuming that $V^{'}_{ii}(t_f) = 0$.
We want to Leverage the HHL algorithm so we need to turn the previous equations into a system of equations. Introduce

$$W_{ii} = V^{'}_{ii}$$

$$W^{'}_{ii} = V_{ii} + \alpha_i W_{ii}$$

So we obtain the following system:

$$\begin{array}{rcr} V^{'}_{ii} & = & W_{ii} \\ W^{'}_{ii} & = & V_{ii} + \alpha_i W_{ii} \end{array} \tag{11}$$

Let $X_{ii} = \begin{bmatrix} V_{ii} \\ V^{'}_{ii} \end{bmatrix}$.
So system (11)can be written as:

$$\frac{X_{ii}}{dt} = M_i X_{ii} \tag{12}$$

where

$$M_i = \begin{bmatrix} 0 & 1 \\ 1 & \alpha_i \end{bmatrix}$$

and

$$X_{ii} = \begin{bmatrix} V_{ii} \\ V^{'}_{ii} \end{bmatrix}$$

One way to find the solution of system (12) along with the initial condition $V_{ii}^{'}(t_f) = 0$, is to turn it to the following system:

$$e^{-M_i t} X_{ii} = e^{-M_i t_f} X_{ii}(t_f)$$

or

$$e^{-M_i(t-t_f)} X_{ii} = \begin{bmatrix} V_{ii}(t_f) \\ 0 \end{bmatrix}$$

Computing e^{-tM_i}

$$M_i = \begin{bmatrix} 0 & 1 \\ 1 & \alpha_i \end{bmatrix}$$

M_i has two eigenvalues: λ_i^1 and λ_i^2 expressed as:

$$\lambda_i^1 = \frac{\alpha_i + \sqrt{(\alpha_i)^2 + 4}}{2}$$

$$\lambda_i^2 = \frac{\alpha_i - \sqrt{(\alpha_i)^2 + 4}}{2}$$

The eigenvector associated with λ_i^1: $w_i^1 = \begin{bmatrix} 1 \\ \lambda_i^1 \end{bmatrix}$

The eigenvector associated with λ_i^2: $w_i^2 = \begin{bmatrix} 1 \\ \lambda_i^2 \end{bmatrix}$

The passage matrix is given by :

$$P_i = \begin{bmatrix} 1 & 1 \\ \lambda_i^1 & \lambda_i^2 \end{bmatrix}$$

Then

$$P_i^{-1} = \frac{1}{\lambda_i^2 - \lambda_i^1} \begin{bmatrix} \lambda_i^2 & -1 \\ \lambda_i^1 & 1 \end{bmatrix}$$

Since $M_i = P_i D_i P_i^{-1}$ where $D_i = \begin{bmatrix} \lambda_i^1 & 0 \\ 0 & \lambda_i^2 \end{bmatrix}$ Therefore

$$e^{-tM_i} = P_i e^{-tD_i} P_i^{-1}$$

After calculation,

$$e^{-tM_i} = \begin{bmatrix} \frac{\lambda_i^2 e^{-\lambda_i^1 t} - \lambda_i^1 e^{-\lambda_i^2 t}}{\lambda_i^2 - \lambda_i^1} & \frac{-e^{-\lambda_i^1 t} + e^{-\lambda_i^2 t}}{\lambda_i^2 - \lambda_i^1} \\ \frac{-e^{-\lambda_i^1 t} + e^{-\lambda_i^2 t}}{\lambda_i^2 - \lambda_i^1} & \frac{-\lambda_i^1 e^{-\lambda_i^1 t} + \lambda_i^2 e^{-\lambda_i^2 t}}{\lambda_i^2 - \lambda_i^1} \end{bmatrix}$$

then

$$e^{-M_i(t-t_f)} = \begin{bmatrix} \frac{\lambda_i^2 e^{-\lambda_i^1 (t-t_f)} - \lambda_i^1 e^{-\lambda_i^2 (t-t_f)}}{\lambda_i^2 - \lambda_i^1} & \frac{-e^{-\lambda_i^1 (t-t_f)} + e^{-\lambda_i^2 (t-t_f)}}{\lambda_i^2 - \lambda_i^1} \\ \frac{-e^{-\lambda_i^1 (t-t_f)} + e^{-\lambda_i^2 (t-t_f)}}{\lambda_i^2 - \lambda_i^1} & \frac{-\lambda_i^1 e^{-\lambda_i^1 (t-t_f)} + \lambda_i^2 e^{-\lambda_i^2 (t-t_f)}}{\lambda_i^2 - \lambda_i^1} \end{bmatrix}$$

Now solving the system

$$e^{-M_i(t-t_f)} X_{ii} = X_{ii}(t_f) = \begin{bmatrix} V_{ii} \\ 0 \end{bmatrix}$$

$1 \leq i \leq n$, is equivalent of finding the solution of the following system:

$$H \begin{bmatrix} X_{11} \\ \vdots \\ X_{nn} \end{bmatrix} = \begin{bmatrix} X_{11}(t_f) \\ \vdots \\ X_{nn}(t_f) \end{bmatrix} \tag{13}$$

where

$$H = \begin{bmatrix} e^{-M_1(t-t_f)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & e^{-M_i(t-t_f)} \end{bmatrix} \tag{14}$$

Notice that H is sparse. It is also Hermitian since e^{-tM_i} is Hermitian for all $1 \leq i \leq n$. Since there are $2n$ variables then the algorithm for solving linear systems of equations shows that the quantum computers could solve the system in time scale of order $log(2n)$ giving an exponential speed up over classical computers.

The next section describes an algorithm for finding the control u in the case $n = 2$.

5 Algorithm

If $B_1 B_1^T Q \neq I$ and $2I + (B_1 B_1^T)'(B_1 B_1^T)^{-1}$ is not a constant diagonal matrix then display error message: Halt program.

$$\alpha_1 = ent_{11}(2I + (B_1 B_1^T)'(B_1 B_1^T)^{-1})$$

$$\alpha_2 = ent_{22}(2I + (B_1 B_1^T)'(B_1 B_1^T)^{-1})$$

$$\lambda_1^1 \leftarrow \frac{\alpha_1 + \sqrt{(\alpha_1)^2 + 4}}{2}$$

$$\lambda_1^2 \leftarrow \frac{\alpha_1 - \sqrt{(\alpha_1)^2 + 4}}{2}$$

$$\lambda_2^1 \leftarrow \frac{\alpha_2 + \sqrt{(\alpha_2)^2 + 4}}{2}$$

$$\lambda_2^2 \leftarrow \frac{\alpha_2 - \sqrt{(\alpha_2)^2 + 4}}{2}$$

$$e^{-M_i(t-t_f)} \leftarrow \begin{bmatrix} \frac{\lambda_i^2 e^{-\lambda_i^1(t-t_f)} - \lambda_i^1 e^{-\lambda_i^2(t-t_f)}}{\lambda_i^2 - \lambda_i^1} & \frac{-e^{-\lambda_i^1(t-t_f)} + e^{-\lambda_i^2(t-t_f)}}{\lambda_i^2 - \lambda_i^1} \\ \frac{-e^{-\lambda_i^1(t-t_f)} + e^{-\lambda_i^2(t-t_f)}}{\lambda_i^2 - \lambda_i^1} & \frac{-\lambda_i^1 e^{-\lambda_i^1(t-t_f)} + \lambda_i^2 e^{-\lambda_i^2(t-t_f)}}{\lambda_i^2 - \lambda_i^1} \end{bmatrix}$$

$$H \leftarrow \begin{bmatrix} e^{-M_1(t-t_f)} & 0 \\ 0 & e^{-M_2(t-t_f)} \end{bmatrix}$$

Using HHL, Solve

$$H \begin{bmatrix} X_{11} \\ X_{22} \end{bmatrix} = \begin{bmatrix} X_{11}(t_f) \\ X_{22}(t_f) \end{bmatrix}$$

that is

$$H\begin{bmatrix}V_{11}\\V_{11}^{'}\\V_{22}\\V_{22}^{'}\end{bmatrix}=\begin{bmatrix}V_{11}(t_f)\\0\\V_{22}(t_f)\\0\end{bmatrix}$$

Find $V_{11}(t)$ and $V_{22}(t)$

V_{12} is obtained from V_{11} by replacing $V_{11}(t_f)$ by $V_{12}(t_f)$.

V_{21} is obtained from V_{22} by replacing $V_{22}(t_f)$ by $V_{21}(t_f)$.

So

$$V \leftarrow (V_{ij})_{1\le i,j\le n}$$

$$P \leftarrow -(B_1^T)^{-1}B_1^{-1}V^{'}V^{-1}$$

~~Find $V_{11}(t_f)$, $V_{12}(t_f)$, $V_{21}(t_f)$, $and V_{22}(t_f)$ by solving $P(t_f)=0$~~.

Finally the control u is given by:

$$u \leftarrow -R^{-1}B_1^TP(t)\begin{bmatrix}x_1(t)\\x_2(t)\end{bmatrix}$$

Since $R=I$ then

$$u \leftarrow -B_1^TP(t)\begin{bmatrix}x_1(t)\\x_2(t)\end{bmatrix}$$

End of the algorithm

6 Example

Consider the optimal control problem :

$$\frac{dx}{dt}=A_1x+B_1u \quad t\in[0,1]$$

so $t_f=1$

$$J(x,u)=\frac{1}{2}\int_0^1(x^TQx+u^TRu)dt$$

where

$$A_1(t)=I=\begin{bmatrix}1&0\\0&1\end{bmatrix},\quad B_1(t)=\begin{pmatrix}e^{t/2}&-e^{t/2}\\e^{t/2}&e^{t/2}\end{pmatrix},\quad R=I,\quad so\ Q=\begin{pmatrix}\frac{1}{2}e^{-t}&0\\0&\frac{1}{2}e^{-t}\end{pmatrix}$$

Find the control u that minimizes $J(x,u)$ using the algorithm.

7 conclusion

References

[1] Dimitri P. Bertsekas, Dynamic Programming and Optimal Control by Dimitri P. Bertsekas, Vol.1, fourth edition,2017, ISBNs: 1-886529-43-4

[2] Todorov, Emanuel. "Optimal control theory." (2006).

[3] Bellman, Richard E.. Dynamic Programming, Princeton: Princeton University Press, 2010. https://doi.org/10.1515/9781400835386

[4] Mahmoud, M. S., Hanna, M. T. (1982). Linear optimal control of national econometric models. International Journal of Systems Science, 13(10), 1061–1081. https://doi.org/10.1080/00207728208926411

[5] Irma Y. , Rubén Morales-Menéndez and Sergio O., Linear quadratic control problem in biomedical engineering, Elsevier, Vol. 20, pp 1195-1200, 2005, https://doi.org/10.1016/S1570-7946(05)80041-0.

[6] Tung Le and Hai L. Vu and Yoni Nazarathy and Bao Vo and Serge Hoogendoorn, Linear-Quadratic Model Predictive Control for Urban Traffic Networks, Procedia - Social and Behavioral Sciences, vol.80, pp512-530,2013, https://doi.org/10.1016/j.sbspro.2013.05.028.

[7] R. Pindyck, An application of linear quadratic tracking problem to economic stabilization policy, in IEEE Transaction on Automatic Control, Vol.17, No. 3, pp. 287-300, June 1972, doi:10.1109/TAC.197.1100010

[8] Rajakumar, V., Anbukumar, K., Selwynraj Arunodayaraj, I. (2017). Power Quality Enhancement Using Linear Quadratic Regulator Based Current-controlled Voltage Source Inverter for the Grid Integrated Renewable Energy System. Electric Power Components and Systems, 45(16), 1783–1794. https://doi.org/10.1080/15325008.2017.1378773

[9] Cole, D. J., Pick, A. J., Odhams, A. M. C. (2006). Predictive and linear quadratic methods for potential application to modelling driver steering control. Vehicle System Dynamics, 44(3), 259–284. https://doi.org/10.1080/00423110500260159

[10] Izumi, S., Xin, X. Reduction of data amount in data-driven design of linear quadratic regulators. Control Theory Technol. 22, 532–542 (2024). https://doi.org/10.1007/s11768-024-00220-y

[11] Sairoel Amertet, Girma Gebresenbet and Hassan Mohammed Alwan, Optimizing the performance of a wheeled mobile robots for use in agriculture using a linear-quadratic regulator, Robotics and Autonomous Systems,vol.174, (2024)104642, https://doi.org/10.1016/j.robot.2024.104642.

[12] J.M. Lemos and L.F. Pinto,Distributed Linear-Quadratic Control of Serially Chained Systems: Application to Water Delivery canal, in IEEE Control Systems Magazine, Vol.32, No.6, pp26-38, Dec.2012, doi;10.1109/MCS.2012.2214126.

[13] Dul F., Lichota P., Rusowicz A., Generalized Linear Quadratic Control for a Full Tracking Problem in Aviation. Sensors.2020;20(10):2955.https://doi.org/10.3390/s 20102955

[14] W.T. Reid. Riccati Differential Equations, Academic Press, New York, 1980.

[15] https://mathshistory.st-andrews.ac.uk/Biographies/Riccati

[16] J. Nazarzadeh, M. Razzaghi, K.Y. Nikravesh, Solution of the matrix Riccati equation for the linear quadratic control problems, Mathematical and Computer Modelling, Volume 27, Issue 7, 1998, Pages 51-55, https://doi.org/10.1016/S0895-7177(98)00035-1.

[17] N.Lovren and M.Tomic, Analytic solution of the Riccati equation for the homing missile linear quadratic control problem, J.Guidance, Cont. Dynamics 17, 619-621,(1994).

[18] Douglas J. Bender and Alan J. Laub, The linear-quadratic optimal regulator for descriptor systems: Discrete-time case, Automatica,vol. 23, no.1, pp 71-85, 1987, https://doi.org/10.1016/0005-1098(87)90119-1.

[19] Huang, Jian-Ping and Zhou, Hua-Cheng, Infinite Horizon Linear Quadratic Optimal Control Problems for Singular Volterra

Integral Equations, SIAM Journal on Control and Optimization,Vol. 63, No1, PP 57-85, 2025, doi=10.1137/24M1641713, https://doi.org/10.1137/24M1641713.

[20] Weiping Wu , Jianjun Gao , Jun-Guo Lu and Xun Li, On continuous-time constrained stochastic linear–quadratic control, Automatica, vol. 114, pp108809,2020, https://doi.org/10.1016/j.automatica.2020.108809.

[21] A. W. Harrow, A. Hassidim and S. Lloyd, "Quantum algorithm for linear systems of equations," Physical Review Letters, vol. 103, no. 15, pp. 150502, 2009. https://doi.org/10.1103/PhysRevLett.103.150502

[22] Leitman Gorge, The Calculus of Variations and Optimal control. Plenum Press. NY& London, 1981

Appendix

Theorem 4. *If $B = 0$ and if A is invertible, then the matrix Riccati equation (19) can be turned into the second order matrix linear differential equation.*

$$V'' - (ACA^{-} + A'A^{-1})V + ADV = 0 \tag{15}$$

using the change of variable

$$Y = -A^{-1}V'V^{-1} \tag{16}$$

where V is invertible.

Proof.

$$Y' = -(A^{-1})'U'U^{-1} + A^{-1}U''U^{-1} + A^{-1}U'(U^{-1})'$$

Using (2) on U ,

$$Y' = -(A^{-1})'U'U^{-1} + A^{-1}U''U^{-1} + A^{-1}U'U^{-1}U'U^{-1}$$

Since

$$Y' = A^{-1}U'U^{-1}U'U^{-1} - CA^{-1}U'U^{-1} + D$$

then

$$-(A^{-1})'U'U^{-1}+A^{-1}U''U^{-1}+A^{-1}U'U^{-1}U'U^{-1} = A^{-1}U'U^{-1}U'U^{-1}-CA^{-1}U'U^{-1}+D$$

Using (2) on A and after simplification , we obtain

$$U^{''} - (ACA^{-1} + A^{'}A^{-1})U^{'} + ADU = 0$$

$\square$

Theorem 5. *The control function that minimizes the cost is given by:*

$$u(t) = -R^{-1}(t)B^T(t)P(t)x(t)$$

where $P(t)$ is a solution of the Riccati equation

$$\frac{dP(t)}{dt} = P(t)B(t)R^{-1}(t)B^T(t)P(t) - A^T(t)P(t) - P(t)A(t) - Q(t) \quad (17)$$

satisfying the initial condition $P(t_f) = 0$.

Proof. The Hamiltinian H of the problem is given by :

$$H(\hat{x}, u, \hat{\lambda}) = \hat{\lambda}^T \begin{pmatrix} A(t)x(t) + B(t)u(t) \\ 1 \end{pmatrix} - \frac{1}{2}(x^TQx + u^TRu) \quad (18)$$

$$H(\hat{x}, u, \hat{\lambda}) = \lambda^TAx + \lambda^TBu + \lambda_{m+1} - \frac{1}{2}x^TQx - \frac{1}{2}u^TRu \quad (19)$$

The adjoint equations are:

$$\frac{\partial\hat{\lambda}}{\partial t} = -\nabla_{\hat{x}}H$$

which implies

$$\frac{\partial\lambda}{\partial t} = -\nabla_{x}H$$

Since Q and R are symmetric then

$$\nabla_x(x^TQx) = 2Qx \qquad \nabla_x(x^TRx) = 2Rx$$

therefore

$$\frac{\partial\lambda}{\partial t} = -A^T(t)\lambda(t) + Q(t)x(t)$$

Moreover, since there is non constraint on u, therefore

$$\nabla_u H = 0 \quad (20)$$

so from (17) and (18)

$$B^T\lambda - Ru = 0 \tag{21}$$

therefore

$$u = R^{-1}(t)B^T(t)\lambda(t) \tag{22}$$

The goal is to express λ in term of $x(t)$. Let's replace $u(t)$ in the system of equations (3), we obtain

$$\frac{dx}{dt} = A(t)x(t) + B(t)R^{-1}(t)B^T(t)\lambda(t) \tag{23}$$

therefore, we get the following system with $2n$ variables

$$\left\{ \begin{array}{rcl} \frac{dx}{dt} & = & A(t)x(t) + B(t)R^{-1}(t)B^T(t)\lambda(t) \\ \frac{d\lambda}{dt} & = & Q(t)x(t) - A^T(t)\lambda(t) \end{array} \right. \tag{24}$$

that has a unique solution $(x(t), \lambda(t))$ given an initial condition.
Using the matrix representation,

$$\frac{d}{dt}\begin{pmatrix} x(t) \\ \lambda(t) \end{pmatrix} = H(t)\begin{pmatrix} x(t) \\ \lambda(t) \end{pmatrix} \tag{25}$$

where

$$H(t) = \begin{pmatrix} A(t) & B(t)R^{-1}B^T(t) \\ Q(t) & -A^T(t) \end{pmatrix} \tag{26}$$

therefore

$$\begin{pmatrix} x(t) \\ \lambda(t) \end{pmatrix} = M(t, t_0)\begin{pmatrix} x(t_0) \\ \lambda(t_0) \end{pmatrix}$$

where

$$M(t, t_0) = e^{\int_{t_0}^{t} H(\tau)d\tau}$$

In particular

$$\begin{pmatrix} x(t_f) \\ \lambda(t_f) \end{pmatrix} = M(t_f, t)\begin{pmatrix} x(t) \\ \lambda(t) \end{pmatrix}$$

for all $t \in [t_0, t_f]$
dividing $M(t_f, t)$ into blocks of $n \times n$ matrices

$$\begin{pmatrix} x(t_f) \\ \lambda(t_f) \end{pmatrix} = \begin{pmatrix} M_{11}(t_f, t) & M_{12}(t_f, t) \\ M_{21}(t_f, t) & M_{22}(t_f, t) \end{pmatrix}\begin{pmatrix} x(t) \\ \lambda(t) \end{pmatrix} \tag{27}$$

where M_{ij} are $n \times n$ matrices $1 \leq i, j \leq 2$.
Therefore

$$x(t_f) = M_{11}(t_f, t)x(t) + M_{12}(t_f, t)\lambda(t)$$
$$\lambda(t_f) = 0 = M_{21}(t_f, t)x(t) + M_{22}(t_f, t)\lambda(t).$$

Since $\lambda(t)$ is unique then $M_{22}(t_f, t)$ must be invertible, therefore

$$\lambda(t) = -M_{22}^{-1}(t_f, t)M_{21}(t_f, t)x(t)$$

Let $P(t) = M_{22}^{-1}(t_f, t)M_{21}(t_f, t)$
So

$$\lambda(t) = -P(t)x(t) \tag{28}$$

From (20) and (26), $u(t) = -R^{-1}(t)B^T(t)P(t)x(t)$.
Let's find $P(t)$.
Taking the derivative in both sides of the equation (26),and using (21) we obtain

$$\frac{d\lambda(t)}{dt} = -\frac{dP(t)}{dt}x(t) - P(t)A(t)x(t) + P(t)A(t)x(t) + P(t)B(t)R^{-1}(t)B^T(t)P(t)x(t). \tag{29}$$

Using (22) and (27), we get the following equation:

$$\frac{dP(t)}{dt} + P(t)A(t) - P(t)B(t)R^{-1}(t)B^T(t)P(t) + Q(t))u(t) = 0. \tag{30}$$

This equation holds for all $t_0 \leq t \leq t_f$.
So $P(t)$ is a solution of solution of the Matrix Riccati equation:

$$\frac{dP(t)}{dt} = P(t)B(t)R^{-1}(t)B^T(t)P(t) - A^T(t)P(t) - P(t)A(t) - Q(t) \tag{31}$$

satisfying the initial condition $P(t_f) = 0$. $\square$

Theorem 3. *The function $u^*(t) = -R^{-1}(t)B^T(t)P(t)x^*(t)$ is the optimal solution at x_0 and the minimum value of J is given by :*

$$J_{min} = \frac{1}{2}(x^*)^T(t_0)P(t_0)x^*(t_0)$$

where x^* is the corresponding optimal solution of (2).

Proof. First notice that $P(t)$ is symmetric that is $P(t)^T = P(t)$. Let's take the transpose in both side of the equation (29), we obtain:

$$\frac{dP^T(t)}{dt} = P^T(t)B(t)(R^{-1})^T(t)B^T(t)P^T - P^T(t)A(t) - A^T(t)P(t) - Q^T(t).$$

Since R and Q are symmetric then

$$\frac{dP^T(t)}{dt} = P^T(t)B(t)R^{-1}(t)B^T(t)P^T - P^T(t)A(t) - A^T(t)P(t) - Q(t)$$

which shows that $P^T(t)$ is also a solution of (29) satisfying $P^T(t_f) = 0$.
Since the solution of (29) along with initial condition $P^T(t_f) = 0$, is unique therefore
$P^T(t) = P(t)$.
Now to show that $J_{min} = \frac{1}{2}x^T(x_0)P(t_0)x(t_0)$, we first can show that

$$\frac{d}{dt}(x^TPx) = -(x^TQ(t)x + u^TR(t)u)$$

$$\frac{d}{dt}(x^TPx) = \frac{dx^T}{dt}Px + x^T\frac{dP}{dt}x + x^TP\frac{dx}{dt}.$$

Since P is symmetric then we can easily verify that $(\frac{dx}{dt})^TPx = x^TP\frac{dx}{dt}$.
Therefore

$$\frac{d}{dt}(x^TPx) = 2x^TP\frac{dx}{dt} + x^T\frac{dP}{dt}x$$

Using (22), (26) and (29)

$$\frac{d}{dt}(x^TPx) = 2x^TP(Ax - BR^{-1}B^TPx) + \frac{d}{dt}(x^TPx) - A^TP - PA - Q)x$$

Since $A^TP = (PA)^T$ then $x^T(A^TP + PA)x = 2x^TPAx$.
After cancellation,

$$\frac{d}{dt}(x^TPx) = -x^TPBR^{-1}B^TPx - x^TQx$$

Since $u = -R^{-1}(t)B(t)^TP(t)x$ then

$$\frac{d}{dt}(x^TPx) = -(u^TRu + x^TQx).$$

Taking the integral from t_0 to t_f in both side and multiplying by $\frac{1}{2}$, we obtain

$$\frac{1}{2}x^T(t_0)P(t_0)x(t_0) = \frac{1}{2}\int_{t_0}^{t_f}(u^T R(t)u + x^T Q(t)x)dt \tag{32}$$

which shows that $J_{min} = \frac{1}{2}x^{*T}(t_0)P(t_0)x^*(t_0)$.
To show that $u^* = -R^{-1}(t)B(t)^T P(t)x^*$ is the optimal solution, we can verify the three conditions of the theorem 4.
From (30), we can choose a test function defined in $\mathbb{R}^{n+1}$ as $J_1(x,t) = \frac{1}{2}x^T P(t)x$ where $(x,t) \in \{(x,t)/t < t_f\}$.
We can show that the test function satisfies the three conditions in theorem 4.
Since $P(t_f) = 0$ then $lim_{t->t_f} J_1(x,t) = lim_{t->t_f} J_1(x^*,t) = 0$ then condition (i) is satisfied.
For (ii), let

$$g(u) = \hat{L}(x,u) + grad^T J_1(x,t)\hat{f}(x,u)$$

so

$$g(u) = \frac{1}{2}(x^T Q(t)x + u^T Ru) + \frac{1}{2}grad^T(x^T Px)\begin{pmatrix} A(t)x + B(t)u \\ 1 \end{pmatrix}$$

$$= \frac{1}{2}(x^T Q(t)x + u^T Ru) + \frac{1}{2}(2(P(t)x)^T \quad x^T\frac{dP(t)}{dt}x)\begin{pmatrix} A(t)x + B(t)u \\ 1 \end{pmatrix}$$

$$= \frac{1}{2}x^T Q(t)x + \frac{1}{2}u^T Ru + x^T P(t)A(t)x + x^T P(t)B(t)u + \frac{1}{2}x^T\frac{dP(t)}{dt}x$$

Using (17)

$$g(u) = \frac{1}{2}x^T Q(t)x + \frac{1}{2}u^T Ru + x^T P(t)A(t)x + x^T P(t)B(t)u +$$

$$\frac{1}{2}(x^T P(t)B(t)R^{-1}(t)B^T(t)P(t)x - x^T A^T(t)Px - x^T P(t)A(t)x - x^T Q(t)x)$$

After simplification

$$g(u) = \frac{1}{2}u^T Ru + x^T P(t)B(t)u + \frac{1}{2}x^T P(t)B(t)R^{-1}(t)B^T(t)P(t)x$$

The gradient of $g(u)$, $\nabla_u g(u) = Ru + B^T(t)P(t)x$ so $\nabla_u g(u) = 0$ then $u = -R^{-1}B^T(t)P(t)x$

therefore $u = -R^{-1}B^T(t)P(t)x$ is a critical value for $g(u)$.
The Hessian $\nabla_u^2 = R^T$ which is positive definite, therefore $g(u)$ has a global minimum at $u^* = -R^{-1}B^T(t)P(t)x^*$. It can be shown easily that $g(u^*) = 0$. This shows (ii).
Since $g(u)$ has a global minimum at u^* then $g(u) \geq g(u^*)$ for all u and x therefore $g(u) \geq 0$ this shows (iii).

$\square$