Egy. J. Pure & Appl. Sci. 2024; 62(1):33-42

Egyptian Journal of Pure and Applied Science

Computing Exact Solution for Linear Integral Quadratic Control Problem

Mohammed Shehata

Department of Basic Science, Bilbeis Higher Institute for Engineering, Sharqia, Egypt.

ARTICLE INFO

Received 4 December 2023
Accepted 24 December 2023

Keywords
Quadratic optimal control problem,
Integral systems,
Necessary optimality conditions.

Correspondence
Mohammed Shehata

E-mail
mashehata_math@bhie.edu.eg

ABSTRACT

Quadratic optimal control problems have applications in various fields of science and engineering at the same time, they are relatively easy to solve, In addition, these problems are often taken as test examples that demonstrate the effectiveness of basic methods of optimal control theory. Many applications contain integrals, such as electrical circuits that contain capacitors, and it is not feasible in applications to convert integrals into differentials, especially if there are many integrals of many variables in the problem. In this paper, we first consider the general finite time horizon control problem for nonlinear linear integral systems, and then proceed to discuss the case for linear quadratic problem as well as we will construct an algorithm to find the exact solution of the quadratic optimal control problem for a system described by linear integral equations. As applications, this algorithm will be used to compute the optimal solution for single and coupled RC electrical circuits.

1. Introduction

Quadratic linear optimal control problems represent an important class of problems investigated in optimal control, [1-3]. They have applications in various fields as well as these problems are often taken as test examples that demonstrate the maximum Pontryagin's principle in calculus of variations. Many processes in science, biology, engineering, and economics can be modeled by a linear integral time invariant system, see [4-10]. Therefore, it is important to study the optimal control of these systems separately in its integral form, without converting them to the differential form, inorder to preserve its properties.

More precisely, this paper deals with a quadratic optimal control problem involving a dynamical system described by a linear integral equations. Such problems were previously considered in a general form, as the optimal control for Volterra integral equation e.g., [11-13].Optimal control problems with integral state constraint naturally arise in many engineering applications e.g ., [1-8], [12], [13]. We specifically mention [1], [2], [4] because the cost functional in them is quadratic. In [1], [4], existence theorems of the quadratic optimal control problem for an integro-differential Volterra equation has been proved.

In [2], Under some necessary convexity conditions, an optimal control exists for linear quadratic optimal control problems, and can be characterized via Fréchet derivative of the quadratic functional in a Hilbert space or via maximum principle type necessary conditions. In this paper, we construct an algorithm, gives the exact solution for the integralquadratic optimal control problem. We first consider the general finite time horizon control problem for nonlinear linear systems, and then proceed to discuss the case for linear quadratic problem.

2. Necessary conditions for nonlinear control problem

In this section, we formulate the control problem for nonlinear integral system. Consider the system described by the following nonlinear integral system:

$$\int_{t_0}^{t} x(\tau)d\tau = f(t, x(t), u(t)) \tag{1}$$

where $x(t)$ an n-vector function is determined by $u(t)$ an m-vector function, with $x \in \Re^n, u \in \Re^m$. Consider a performance index of the form

$$J(u) = \int_{t_0}^{t_f} L(t, x(t), u(t))dt \tag{2}$$

The problem is to find the functions $u(t), t \in [t_0, t_f]$ that minimize (or maximize)) $J(u)$. It is assumed that $f(t,x,u)$ and $L(t,x,u)$ are continuous for all $t \in [t_0, t_f], x \in \Re^n, u \in \Re^m$, and have continuous derivative up to the second order.

The necessary conditions of the constrained optimal control problem (1)-(2) are obtained by converting into an unconstrained optimal control problem using the Lagrange multiplier function $\lambda(t) \in \Re^n$:

$$\bar{J}(u) = \int_{t_0}^{t_f} L(t, x(t), u(t))dt + \int_{t_0}^{t_f} \lambda^T(t)\left[f(t, x(t), u(t)) - \int_{t_0}^{t} x(\tau)d\tau \right]dt$$

Let us define the Hamiltonian function

$$H(t, x(t), \lambda(t), u(t)) : \;= L(t, x(t), u(t)) + \lambda^T(t) f(t, x(t), u(t))$$

Thus,

$$\bar{J}(u) = \int_{t_0}^{t_f} \left[H(t, x(t), \lambda(t), u(t)) - \lambda^T(t)\int_{t_0}^{t} x(\tau)d\tau \right]dt$$

The necessary condition for optimality is that the variation $\delta\bar{J}$ of the modified cost with respect to all feasible variations $\delta x(t), \delta\lambda(t)$ and $\delta u(t)$ should vanish.

$$\delta\bar{J} = \int_{t_0}^{t_f} \left[\frac{\partial H}{\partial x}\delta x + \delta\lambda^T \frac{\partial H}{\partial \lambda} + \frac{\partial H}{\partial u}\delta u \right]dt$$
$$- \int_{t_0}^{t_f} \left[\lambda^T(t)\delta\left(\int_{t_0}^{t} x(\tau)d\tau \right) + \delta\lambda^T(t)\int_{t_0}^{t} x(\tau)d\tau \right]dt$$

By changing the order of integration, we get

$$\int_{t_0}^{t_f} \lambda^T(t)\int_{t_0}^{t} \delta x(\tau)d\tau dt = \int_{t_0}^{t_f}\left(\int_{t}^{t_f} \lambda^T(\tau)d\tau \right)\delta x(t)dt$$

Then

$$\delta\bar{J} = \int_{t_0}^{t_f}\left[\frac{\partial H}{\partial x} - \int_{t}^{t_f}\lambda^T(\tau)d\tau \right]\delta x dt + \int_{t_0}^{t_f}\delta\lambda^T\left[\frac{\partial H}{\partial \lambda} - \int_{t_0}^{t} x(\tau)d\tau \right]dt$$
$$+ \int_{t_0}^{t_f}\frac{\partial H}{\partial u}\delta u dt$$

Setting the terms that multiply variations to be zero yield:

$$\int_{t_0}^{t} x(\tau)d\tau = \frac{\partial H}{\partial \lambda} = f(t, x(t), u(t))$$

$$\int_{t}^{t_f} \lambda(\tau)d\tau = \left(\frac{\partial H}{\partial x} \right)^T$$

$$\frac{\partial H}{\partial u} = 0$$

Thus, we obtained the following theorem:

Theorem .1 *If $u(t)$ is a solution of the problem (1)-(2), then the following equations are satisfied:*

$$\int_{t_0}^{t} x(\tau)d\tau = f(t, x(t), u(t)) \tag{3}$$

$$\int_{t}^{t_f} \lambda(\tau)d\tau = \left(\frac{\partial L}{\partial x} \right)^T + \lambda(t)\left(\frac{\partial f}{\partial x} \right)^T \tag{4}$$

$$0 = \left(\frac{\partial L}{\partial u} \right)^T + \lambda(t)\left(\frac{\partial f}{\partial u} \right)^T \tag{5}$$

3. Second variations

Now consider the second variation $\delta^2 \overline{J}$ due to variations in the control vector $u(t)$.

$$\delta^2 J = \int_{t_0}^{t_f}\left[\delta x^T \frac{\partial^2 H}{\partial x^2}\delta x + \delta u^T \frac{\partial^2 H}{\partial x \partial u}\delta x + \delta\lambda^T \frac{\partial^2 H}{\partial x \partial \lambda}\delta x\right]dt + \int_{t_0}^{t_f}\left[\delta x^T \frac{\partial^2 H}{\partial u \partial x}\delta u + \delta u^T \frac{\partial^2 H}{\partial u^2}\delta u + \delta\lambda^T \frac{\partial^2 H}{\partial u \partial \lambda}\delta u - \delta\lambda^T \delta\left(\int_{t_0}^{t} x(\tau)d\tau\right)\right]dt$$

$$\delta^2 J = \int_{t_0}^{t_f}\left[\delta x^T \frac{\partial^2 H}{\partial x^2}\delta x + \delta u^T \frac{\partial^2 H}{\partial u \partial x}\delta x + \delta x^T \frac{\partial^2 H}{\partial x \partial u}\delta u + \delta u^T \frac{\partial^2 H}{\partial u^2}\delta u\right]dt + \int_{t_0}^{t_f}\left[\delta\lambda^T \frac{\partial^2 H}{\partial x \partial \lambda}\delta x + \delta\lambda^T \frac{\partial^2 H}{\partial u \partial \lambda}\delta u - \delta\lambda^T \delta\left(\int_{t_0}^{t} x(\tau)d\tau\right)\right]dt$$

By using $\dfrac{\partial H}{\partial \lambda} = f;$

$$\delta\lambda^T \frac{\partial^2 H}{\partial x \partial \lambda}\delta x + \delta\lambda^T \frac{\partial^2 H}{\partial u \partial \lambda}\delta u = \delta\lambda^T\left(f_x \delta x + f_u \delta u\right) = \delta\lambda^T \delta\left(\int_{t_0}^{t} x(\tau)d\tau\right)',$$

then

$$\delta^2 J = \int_{t_0}^{t_f}\left[\delta x^T \frac{\partial^2 H}{\partial x^2}\delta x + \delta u^T \frac{\partial^2 H}{\partial u \partial x}\delta x + \delta x^T \frac{\partial^2 H}{\partial x\ \partial u}\delta u + \delta u^T \frac{\partial^2 H}{\partial x^2}\delta u\right]dt$$

$$= \int_{t_0}^{t_f}[\delta x^T \quad \delta u^T]\begin{bmatrix}\dfrac{\partial^2 H}{\partial x^2} & \dfrac{\partial^2 H}{\partial x \partial u}\\ \dfrac{\partial^2 H}{\partial u \partial x} & \dfrac{\partial^2 H}{\partial u^2}\end{bmatrix}\begin{bmatrix}\delta x\\ \delta u\end{bmatrix} dt$$

In order that J be a local minimum, not only must we have $\delta J = 0$ but, in addition, the second-order expression for J holding $(\delta^2 J \geq 0),$ must be nonnegative for all values (infinitesimal) of $u;$ that is, we have

$$\int_{t_0}^{t_f}[\delta x^T \quad \delta u^T]\begin{bmatrix}\dfrac{\partial^2 H}{\partial x^2} & \dfrac{\partial^2 H}{\partial x \partial u}\\ \dfrac{\partial^2 H}{\partial u \partial x} & \dfrac{\partial^2 H}{\partial u^2}\end{bmatrix}\begin{bmatrix}\delta x\\ \delta u\end{bmatrix} dt \geq 0 \tag{6}$$

4. Linear quadratic problem

Consider the problem

$$J(u(.)) = \frac{1}{2}\int_{t_0}^{t_f}\left[x^T(t)S\,x(t) + u^T(t)T u(t)\right]dt \to \min \tag{7}$$

under the integral constraint

$$\int_{t_0}^{t} x(\tau)d\tau = M\,x(t) + g(t) + Nu(t), \qquad t \in [t_0, t_f] \tag{8}$$

where

$g(t)$ is non zero $n-$ vector differentiable function in $(t_0, t_f),$

M is $n \times n$ invertible matrix,

N is $n \times m$ matrix,

S is symmetric positive semi-definite $n \times n$ matrix,

T is a symmetric positive definite $m \times m$ matrix.

The control goal generally is to keep $x(t)$ close to 0, especially, at the final time t_f, using little control effort u.

$\frac{1}{2}x^T(t)Sx(t)$ penalizes the transient state deviation,

$\frac{1}{2}u^T(t)Tu(t)$ penalizes control effort.

Applying the necessary and sufficient conditions of the general optimal control given in the above sections to the linear quadratic problem(7)-(8), then the optimal control is given by:

$$u(t) = -T^{-1}N^T\lambda(t) \tag{9}$$

where $\lambda(t)$ and $x(t)$ satisfy the following equations:

$$\int_{t_0}^{t} x(\tau)d\tau = M\,x(t) - NT^{-1}N^T\lambda(t) + g(t) \tag{10}$$

$$\int_{t}^{t_f} \lambda(\tau)d\tau = S\,x(t) + M^T\lambda(t) \tag{11}$$

from (11), we have

$$\int_{t_0}^{t} \lambda(\tau)d\tau + \int_{t}^{t_f} \lambda(\tau)d\tau = \int_{t_0}^{t} \lambda(\tau)d\tau + S\,x(t) + M^T\lambda(t)$$

$$\Rightarrow S\,x(t_0) + M^T\lambda(t_0) = \int_{t_0}^{t} \lambda(\tau)d\tau + S\,x(t) + M^T\lambda(t)$$

then the systems (10)-(11) is equivalent to the following system:

$$\int_{t_0}^{t} x(\tau)d\tau = M\,x(t) - NT^{-1}N^T\lambda(t) + g(t) \tag{12}$$

$$\int_{t_0}^{t} \lambda(\tau)d\tau = -Sx(t) - M^T\lambda(t) + Sx(t_0) + M^T\lambda(t_0) \quad (13)$$

with the final condition

$$\lambda(t_f) = -(M^T)^{-1}Sx(t_f)$$

(12)-(13) can be rewrite in a matrix form:

$$\int_{t_0}^{t} \begin{bmatrix} x(\tau) \\ \lambda(\tau) \end{bmatrix} d\tau = \begin{bmatrix} M & -NT^{-1}N^T \\ -S & -M^T \end{bmatrix} \begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix} + \begin{bmatrix} g(t) \\ x(t_0) + M^T\lambda(t_0) \end{bmatrix}$$

which can by solved by differentiation, to obtain the following main theorem:

Theorem 2 (Solution of linear integral quadratic problem) *The optimal control of quadratic problem (7)-(8) is given by:*

$$u(t) = -T^{-1}N^T\lambda(t), \quad (14)$$

$$\begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix} = \Phi(t,t_0) \begin{bmatrix} M^{-1}\left[NT^{-1}N^T\lambda(t_0) - g(t_0)\right] \\ \lambda(t_0) \end{bmatrix} - \int_{t_0}^{t} \Phi(t,\tau)A^{-1}(\tau) \begin{bmatrix} g'(\tau) \\ 0 \end{bmatrix} d\tau, \quad (15)$$

$$\lambda(t_f) = -(M^T)^{-1}Sx(t_f), \quad (16)$$

where

$$\Phi(t,t_0) = e^{A^{-1}(t-t_0)}, \qquad A = \begin{bmatrix} M & -NT^{-1}N^T \\ -S & -M^T \end{bmatrix}$$

Theorem 2 can be transform to the following algorithm

Table 1. Exact solution algorithm for integral quadratic control problem

Step 1	Input t_0, t_f, M, N, S, T and g
Step 2	Evaluate A
Step 3	Calculate $\Phi(t,t_0)$
Step 4	Substitute in (15) to find $x(t)$ and $\lambda(t)$ with unknown $\lambda(t_0)$
Step 5	Substitute by $t = t_f$ into $x(t)$ and $\lambda(t)$ that we obtained in step 4 and then solving the algebraic equation $\lambda(t_f) = -(M^T)^{-1}Sx(t_f)$ to find $\lambda(t_0)$
Step 6	Substitute again by the value of $\lambda(t_0)$ that obtained in step 5 in (15) to find the exact solution of $x(t)$ and $\lambda(t)$
Step 7	Plot $x(t)$ and $u(t) = -T^{-1}N^T\lambda(t)$ in the interval $[t_0, t_f]$
Step 8	Substitute by the exact solution of $x(t)$ and $u(t)$ that obtained in step 7 to calculate $\min J$ from (7)

6. Application in circuits

RC-circuit

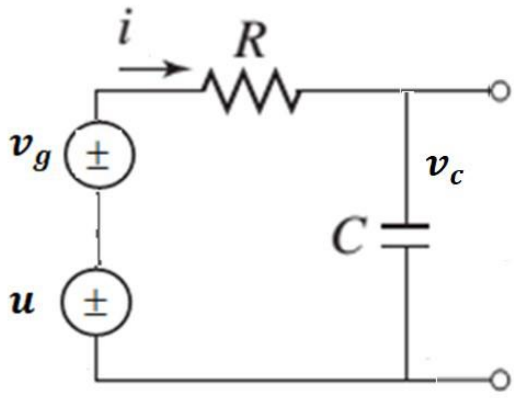


Fig. 1. RC circuit

In this application, we want to find the unknown supplied voltage $u(t)$ for the RC circuit in Fig. 1, which minimized the cost functional given by

$$J = \frac{C}{2}\int_0^{t_f} i^2(t)dt + \frac{1}{2}\int_0^{t_f} u^2(t)dt$$

where C is a capacitance value. By applying the Kirchhoff's voltage law, we get

$$u + v_g - Ri - v_c = 0 \quad \text{or} \int_0^t i(\tau)d\tau = -CRi + Cu + Cv_g$$

Step 1. Let $R = 2, C = 1, v_g = 5t$, $t_0 = 0$ and $t_f = 1$,
then $M = -2$, $N = 1$, $g = 5t$, $S = 1$, $T = 1$.

$$A = \begin{bmatrix} -2 & -1 \\ -1 & 2 \end{bmatrix} \Rightarrow A^{-1} = \frac{-1}{5}\begin{bmatrix} 2 & 1 \\ 1 & -2 \end{bmatrix}$$

Step 2.

Step 3.

$$\Phi(t,t_0) = e^{\frac{-1}{5}\begin{bmatrix} 2 & 1 \\ 1 & -2 \end{bmatrix}t}, \text{see for example [14]}$$

$$= \begin{bmatrix} \cosh(\frac{t}{\sqrt{5}}) - \frac{2}{\sqrt{5}}\sinh(\frac{t}{\sqrt{5}}) & -\frac{1}{\sqrt{5}}\sinh(\frac{t}{\sqrt{5}}) \\ -\frac{1}{\sqrt{5}}\sinh(\frac{t}{\sqrt{5}}) & \cosh(\frac{t}{\sqrt{5}}) + \frac{2}{\sqrt{5}}\sinh(\frac{t}{\sqrt{5}}) \end{bmatrix}$$

Step 4.

$$x(t) = \left(\frac{-1}{2}\lambda(0) - 5\right)\cosh(\frac{t}{\sqrt{5}}) - 2\sqrt{5}\sinh(\frac{t}{\sqrt{5}}) + 5$$

$$\lambda(t) = \lambda(0)\cosh(\frac{t}{\sqrt{5}}) + \left(\frac{\sqrt{5}}{2}\lambda(0) + \sqrt{5}\right)\sinh(\frac{t}{\sqrt{5}})$$

Step 5.

$$\lambda(0) = \frac{10\left(-1 + 2e^{\frac{1}{\sqrt{5}}} - e^{\frac{2}{\sqrt{5}}}\right)}{\left(5 + 2\sqrt{5}\right)e^{\frac{2}{\sqrt{5}}} + 5 - 2\sqrt{5}}$$

Step 6.

$$x(t) = \left(\frac{5\left(1 - 2e^{\frac{1}{\sqrt{5}}} + e^{\frac{2}{\sqrt{5}}}\right)}{\left(5 + 2\sqrt{5}\right)e^{\frac{2}{\sqrt{5}}} + 5 - 2\sqrt{5}} - 5\right)\cosh(\frac{t}{\sqrt{5}}) - 2\sqrt{5}\sinh(\frac{t}{\sqrt{5}}) + 5$$

$$\lambda(t) = \frac{10\left(-1 + 2e^{\frac{1}{\sqrt{5}}} - e^{\frac{2}{\sqrt{5}}}\right)}{\left(5 + 2\sqrt{5}\right)e^{\frac{2}{\sqrt{5}}} + 5 - 2\sqrt{5}}\cosh(\frac{t}{\sqrt{5}}) + \left(\frac{5\sqrt{5}\left(-1 + 2e^{\frac{1}{\sqrt{5}}} - e^{\frac{2}{\sqrt{5}}}\right)}{\left(5 + 2\sqrt{5}\right)e^{\frac{2}{\sqrt{5}}} + 5 - 2\sqrt{5}} + \sqrt{5}\right)\sinh(\frac{t}{\sqrt{5}})$$

Step 7. The optimal current and corresponding are depicted in Fig. 2.

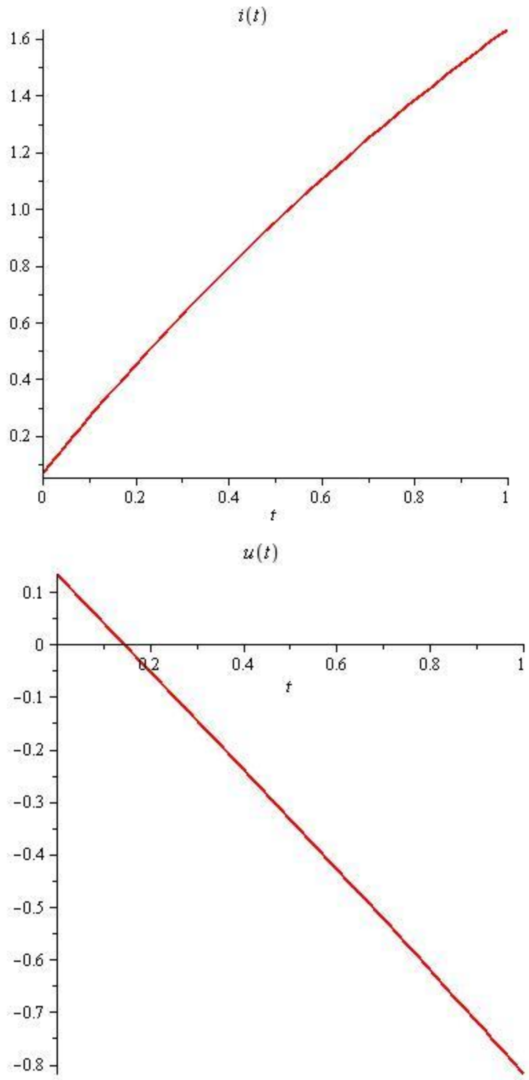


Fig. 2. optimal current $i(t)$ and corresponding voltage $u(t)$ of RC circuit

Step 8. $J_{\min} = 0.4774749710$

Although we calculate the optimal current $x(t)$ and voltage $u(t)$ and its corresponding minimum cost functional $J_{\min} = 0.4774749710$ by mathematical approach, we shall illustrate its efficiency by comparing with some other input voltage functions. Table 2 depicts the efficiency of series RC circuit in Fig. 1 with $R = 2, C = 1, v_g = 5t$ and various input voltage function $u(t).$ In this case the current is given by the formula

$$i(t) = e^{\frac{-1}{2}t}\left(\frac{u(0)}{2} + \int_0^t e^{\frac{1}{2}\tau}\left(\frac{\dot{u}(\tau)+5}{2}\right)d\tau\right)$$

Table 2. Known input voltages signals $u(t).$ and its corresponding for RC circuit.

Input voltage $u(t).$	Corresponding J
Unit step $u(t).=1$	37.10634097
Ramp $u(t).=t$	1.21504423
Sinusoidal $u(t).=\sin t$	1.149388549

RC-coupled circuit

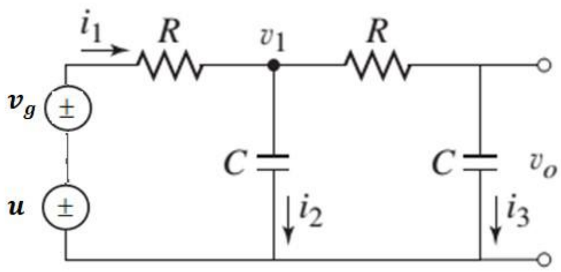


Fig 3. Coupled RC circuit

In this application, we want to find the unknown supplied voltage $u(t)$ for the coupled RC circuit in Fig. **3**, which minimized the cost functional given by

$$J = \frac{C}{2}\int_0^{t_f} i_3^2(t)dt + \frac{1}{2}\int_0^{t_f} u^2(t)dt$$

By applying the Kirchhoff's voltage law for the right and left-hand loops, we get

$$v_1 - Ri_3 - v_o = 0 \quad \text{or} \ \frac{1}{C}\int_0^t [i_2(\tau) - i_3(\tau)]d\tau = Ri_3 \tag{17}$$

$$u + v_g - Ri_1 - v_1 = 0 \quad \text{or} \ \frac{1}{C}\int_0^t i_2(\tau)d\tau = -Ri_2 - Ri_3 + u + v_g \tag{18}$$

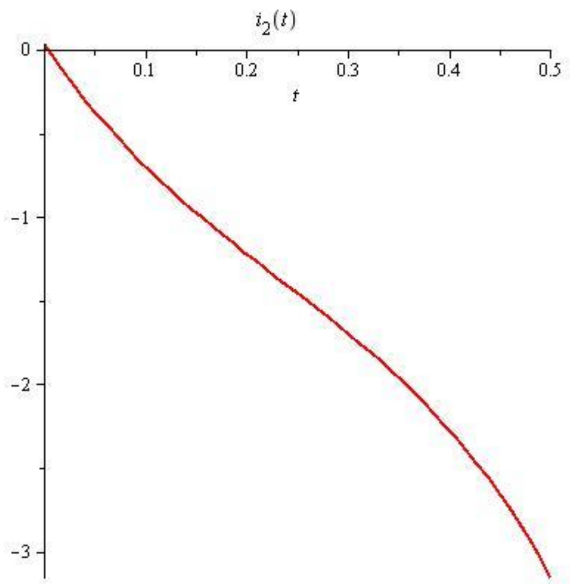
(17)- (18) can be rewritten in matrix form as

$$\int_0^t \begin{bmatrix} i_2(\tau) \\ i_3(\tau) \end{bmatrix} d\tau = RC \begin{bmatrix} -1 & -1 \\ -1 & -2 \end{bmatrix} \begin{bmatrix} i_2(t) \\ i_3(t) \end{bmatrix} + C \begin{bmatrix} 1 \\ 1 \end{bmatrix} [v_g(t) + u(t)]$$

If we take $R = \frac{1}{2}, C = 1, v_g = 5t,$ $t_0 = 0$ and $t_f = 1,$ then we have

$$M = \begin{bmatrix} \frac{-1}{2} & \frac{-1}{2} \\ \frac{-1}{2} & -1 \end{bmatrix}, \quad N = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, S = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \quad T = [1].$$

By applying the steps of the algorithm, we obtain the optimal current in fig. **4**, corresponding optimal voltage in fig. **5** and $J_{\min} = -0.0765439662$



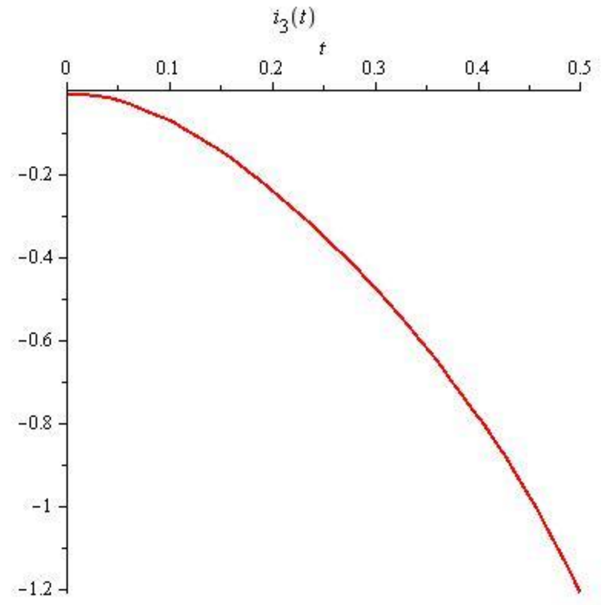


Fig. 4. Optimal currents $i_2(t)$ and $i_3(t)$ of coupled RC circuit

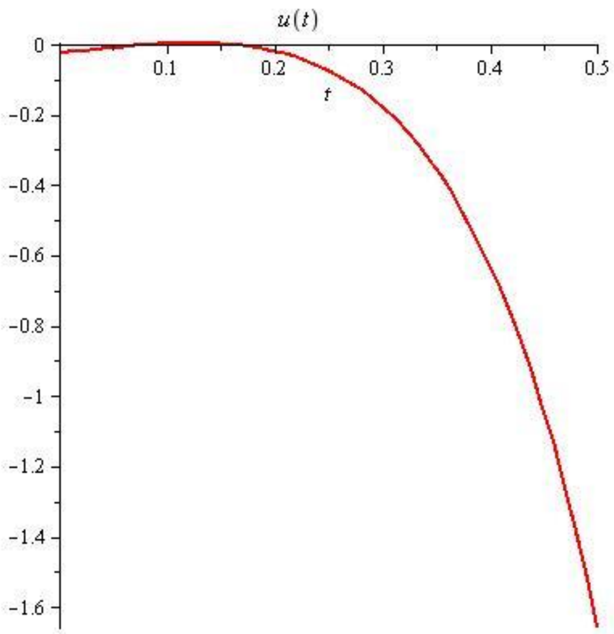


Fig. 5. optimal voltage $u(t)$ of coupled RC circuit

We calculate the optimal current $x(t)$ and voltage $u(t)$ and its corresponding minimum cost functional $J_{\min} = -0.07654396621$ by mathematical approach, we shall illustrate its efficiency by comparing with some other input voltage functions.

Table 3. Known input voltages signals $u(t).$ and its corresponding J for coupled RC circuts.

Input voltage $u(t).$	Corresponding J
Unit step $u(t).=1$	19.44940722
Ramp $u(t).=t$	28.51786709
Sinusoidal $u(t).=\sin t$	25.37690787

To perform calculations in the above applications easily and accurately, you can convert the algorithm into a general procedure using any mathematics program such as Matlab, Maple, or Mathematica. In the appendix, I constructed a Maple procedure call it **QISolu.**

7. Conclusion

In the paper, an algorithm has been constucted for computing the exact solutions for the quadratic optimal control problem with integral constraints. As applications, this algorithm has been used to find the optimal solution for single and coupled RC electrical circuits. In the future, we will develop this algorithm to include more difficult cases, such as those that contain both integral and differential constraints, so that we can apply it to electronic circuits of the type RLC.

8. References

1. **M., de Acutis. (1985).** On the quadratic optimal control problem for Volterra integro-differential Equations. Rend. Semin. Matem. Univ. Padova, **Vol. 73, 231-247.**
2. **S., Han., P., Lin., and J., Yong. (2021).** Causal state feedback representation for linear quadratic optimal control problems of singular Volterra integral equations. https://arxiv.org/abs/2109.07720.
3. **L., Pandolfi. (2018).** The quadratic regulator problem and the Riccati equation for a process governed by a linear Volterra integrodifferential equation. IEEE Tans. Autom. Control, **Vol. 63,. 5, 1517-1522.**

4. **T. S., Angell. (1976).** On the optimal control of systems governed by nonlinear Volterra equations. Journal of Optimization Theory and Applications, **19(1), 29-45.**
5. **Mohammed., Shehata. (2024).** From calculus to α calculus. Progr. Fract. Differ. Appl. Accepted for publication in Volume 10, 3 july https://www.naturalspublishing.com/FC.asp?JorID=48.
6. **V. L., Bakke. (1974).** A maximum principle for optimal control problem with integral Constraints. J. Optimiz. Th. Applic., **Vol. 13, 1, 32-55.**
7. **M., Connor. (1972).** Optimal control of linear systems represented by integral equations: An iterative method. IEEE Transactions on Automati,**17(5), 424-355.**
8. **N. G., Medhin. (1988).** Optimal processes governed by integral equation equations with unilateral constraints. Journal of Mathematical Analysis and Applications, **vol. 129, 269-283.**
9. **V.R., Vinokurov. (1969).** Optimal control of processes described by integral equations. I. SIAM Journal on Control, **7(2), 324-336.** https://doi.org/10. 1137/0307022
10. **M. I., Kamien., and E., Muller. (1976).** Optimal control with integral state equations. The Review of Economic Studies, **vol. 43(4), 469-473.**
11. **C.R., Wylie., and L.C., Barrett. (1982).** Advanced engineering mathematics. Mcgraw-Hill: New York.
12. **M., Ahmadinia., H., Afshariarjmand., and M., Salehi. (2023).** Numerical solution of Itˆo–Volterra integral equations by the QR factorization method. Journal of Applied Mathematics and Computing, **69(4), 3171-3188.**
13. **S.A., Belbas. (2008).** A reduction method for optimal control of Volterra integral equations. Applied Mathematics and Computation, **197(2), 880-890,** https://doi.org/10.1016/j.amc.2007.08.093
14. **C., Vega. (2006).** Necessary conditions for optimal terminal time control problems governed by a Volterra integral equation. Journal of Optimization Theory and Applications, **130(1), 79-93.**
15. **Mohammed, Shehata., and A. A., Khalil. (2022).** Algorithm for computing exact solution of the first order linear differential system.Sohag Journal of Sciences, **7(3),71-77.**

Appendix

Maple is a mathematical program that can solve complex mathematical problems by writing just a few lines of command. You can create your own procedures. The maple program contains important packages such as linear algebra package. In [14], we constructed a Maple procedure to get the exact solution of nonhomogeneous time-invariant continuous systems. Here, we will develop a Maple procedure for computing exact solution of the aforementioned quadratic control problem. Throughout, we will use some commands like:

> proc(.)...end proc: # Procedure.

> with(LinearAlgebra): # Linear algebra package.

> $seq(f(i), i = 0..m)\}$; # Sequence $f(1), f(2), \ldots, f(m)$

>$A := Vector([\,[a],[b]])$; # Defining vector $\begin{bmatrix} a \\ b \end{bmatrix}$

> $A := Matrix([\,[a,b,c],[d,e,f]])$; # Defining 2 by 3 matrices.

> Transpose(A); # Matrix transpose.

>MatrixInverse(A); # Matrix inverse.

> Plot($f(t), t = t_0..t_f$); # Plot function f in the interval [$[t_0, t_f$]

QISolu

Procedure QISolu computes and plots the optimal state and corresponding optimal control for linear integral quadratic control problem.

.

Output:

Computing and ploting solutions

Syntax:

$QISolu(\mathrm{t0}, \mathrm{tf}, M, N, S, T, g)$;

Input:

t0- a real number represents initial time;

tf- a real number represents final time;

M - an n- invertable square matrix represents system state coefficients;

N - an n x m matrix represents system control coefficients;

S - a symmetric positive semi-definite n- square matrix represents quadratic cost state;

T - a symmetric positive definite m square matrix represents quadratic cost control;

g - a vector of a given continous real functions on (t0,tf).

Definition:

% --------------**Step 1**.input t_0, t_f, M, N, S, T and g----------------

```
>
QISolu:=proc(t0,tf,M::'Matrix'(square),N::Matrix,S::'Matrix'(square),T::'Matrix'(square),g::Vector)
>local
M1,M2,N1,T1,G,C00,C0,C,C1,g1,h,z1,z2,eqs,csolu,z,zf,z1f,z1ff,z2f,z2ff,a,m,n,A,J1,P,P1,eJ,eJ1,yh,yp1,yp2,yp,i,Jm1,Jm2,Jm,xx,xT,v,vT;
```

% -------------------**Step 2**. Evaluate A, A^{-1} ---------------------

```
> n:=LinearAlgebra[RowDimension](M);
> m:=LinearAlgebra[RowDimension](T);
> M1:=LinearAlgebra[Transpose](M);
> N1:=LinearAlgebra[Transpose](N);
> M2:=LinearAlgebra[MatrixInverse](M);
> T1:=LinearAlgebra[MatrixInverse](T);G:=N.T1.N1;
> A:=LinearAlgebra[MatrixInverse](Matrix([[M,-G],[-S,-M1]]));
```

% --------------------------**Step 3** ----------------------------

```
> J1, P := LinearAlgebra[JordanForm](A, output = ['J','Q']);
> P1:=LinearAlgebra[MatrixInverse](P);
> eJ:=LinearAlgebra[MatrixExponential](J1,t-t0);
```

% -------------------------**Step 4** ---------------------------

```
> eJ1:=subs(t = -t, eJ);
> C00:=seq(C[i],i=1..n);C0:=Vector([C00]);
> C1:=Vector([M2.G.C0-M2.subs(t=t0,g),C0]);
> yh := P.eJ.P1.C1;g1:=map(diff,g,t);
> yh:=simplify(yh);h:=Vector([map(diff, g, t),seq(0,i=1..n)]);
> yp1 := eJ1.P1.A.h;yp2:=subs(t = x,yp1);
> yp:=P.eJ.map(int,yp2,x=t0..t);yp:=simplify(yp);
> z:=yh+yp; z:=convert(z,list); z:=simplify(z);
```

% ---------------------------**Step 5** --------------------------

```
> z1:=Vector([seq(z[i],i=1..n)]);
> z2:=Vector([seq(z[i],i=n+1..2*n)]);
> eqs:=S.subs(t = tf,z1)+M1.subs(t = tf,z2);
> eqs:=seq(eqs[i]=0,i=1..n);
> csolu:=solve({eqs},[C00]);csolu:=simplify(csolu);
> z1f:=convert(z1,list);
> z2f:=-T1.N1.z2;z2f:=convert(z2f,list);
```

% ----------------------------**Step 6** --------------------------

```
> for i from 1 to n do a[i]:= eval(C[i], csolu[1]);
> end do;
> z1ff:=subs(seq(C[i]=a[i],i=1..n),z1f);
> z2ff:=subs(seq(C[i]=a[i],i=1..n),z2f);
```

```
% ----------------------------Step 7 -------------------------
> for i from 1 to n do
> print(x[i](t));
> print(plot(z1ff[i],t=t0..tf,color=red,thickness=2));
> end do;
> for i from 1 to m do
> print(u[i](t));
> print(plot(z2ff[i],t=t0..tf,color=red,thickness=2));
> end do;
% ----------------------------Step 8 -------------------------
> xx:=Vector([z1ff]);v:=Vector([z2ff]);
> xT:=LinearAlgebra[Transpose](xx);
> vT:=LinearAlgebra[Transpose](v);
> Jm1:=(1)/(2)*map(int,xT.S.xx,t=t0..tf);
> Jm2:=(1)/(2)*map(int,vT.T.v,t=t0..tf);
> Jm:=evalf(Jm1+Jm2);
> print(J=Jm);
> end proc:
```

RC -circuit (Fig. 1)

$> \; S := Matrix([[1]]);\; g := Vector([5\,t]);\; M := Matrix([[-2]]);\; N := Matrix([[1]]);\; T := Matrix([[1]]);$

$$S := \begin{bmatrix} 1 \end{bmatrix} g := \begin{bmatrix} 5\,t \end{bmatrix} M := \begin{bmatrix} -2 \end{bmatrix} N := \begin{bmatrix} 1 \end{bmatrix} T := \begin{bmatrix} 1 \end{bmatrix}$$

$> \; QISolu(0, 1, M, N, S, T, g);$

$$J = 0.477474971($$

RC-coupled -circuit (Fig. 3)

$> \; S := Matrix([[0, 0], [0, 1]]);\; g := Vector([5\,t, 5\,t]);\; M := Matrix\left(\left[\left[-\frac{1}{2}, -\frac{1}{2}\right], \left[-\frac{1}{2}, -1\right]\right]\right);\; N := Matrix([[1], [1]]);\; T := Matrix([[1]]);$

$$S := \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} g := \begin{bmatrix} 5\,t \\ 5\,t \end{bmatrix} M := \begin{bmatrix} -\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & -1 \end{bmatrix} N := \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad T := \begin{bmatrix} 1 \end{bmatrix}$$

$> \; QISolu(0, 0.5, M, N, S, T, g);$

$$J = -0.0765439662$$