Journal of Mathematical Modeling
Vol. -, No. -, 2025, pp. -. Research Article

A gradient projection method for solving nonlinear optimal control problems with time-varying delays

Seyed Mojtaba Meshkani[†], **Sohrab Effati**[‡*], **Aghileh Heydari**[†]

[†]*Department of Mathematics, Payame Noor University, 19395-4697, Tehran, Iran*
[‡]*Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran*
Email(s): meshkani.m@pnurazavi.ac.ir, s-effati@um.ac.ir, a_heidari@pnu.ac.ir

Abstract. An effective numerical method using gradient projection is proposed for solving an optimal control problem that involves time-varying delays in control and state variables. First, a variational inequality is established as necessary conditions. The main idea in variational inequality is to compute the gradient of the objective functional, taking into account time-dependent delays in control and state variables. Then, an iterative scheme utilizing a projection operator is presented, followed by a convergence analysis of the method for a coercive objective functional. At the end, several examples are provided to illustrate that the theoretical finding is efficient.

Keywords: Nonlinear optimal control problems, time delay systems, variational inequality, time-varying delay, gradient projection method.
AMS Subject Classification 2010: 49J40, 49M05, 65K15.

1 Introduction

In real-life phenomena modeling, delays (if occur) depend on time. Moreover, to study optimal control problems involving time delays, when the delays in state or control variables are varying with time, the problems are converted to optimal control problems with time-varying delays (OCPTVD). Therefore, analyzing such problems sounds interesting to many researchers.

Here is a brief review of some articles on diverse categories of delayed optimal control problems. In the context of optimal control problems characterized by constant delays in control and/or state variables, the Pontryagin maximum principle has been proved, which can be found in [3, 6, 8, 10, 18, 20, 23]. In particular, Banks [2] obtained some necessary conditions of optimality while solving optimal control problems containing a time-dependent delay in the state variable. After that, in function spaces, the

*Corresponding author
Received: 5 April 2025 / Revised: 29 July 2025 / Accepted: 4 August 2025
DOI: 10.22124/jmm.2025.30121.2691

© 2025 University of Guilan

theory of necessary conditions for optimization problems was stated by Colonius and Hinrichsen [4]. They have also provided an approach for optimal control problems involving a delay in the state variable. For nonlinear delayed optimal control problems satisfying constraints of function-space state inequality, Angell and Kirsch [1] presented some optimality conditions. The inherent complexity of optimal control problems with time delays has led to the development of numerical methods for solving these problems. Several studies in this field can be found in [7, 12, 13, 16, 17, 19, 22, 24]. In recent years, A class of nonlinear systems with a time-varying state delay in the control function was studied by Liu, et al. [14]. They introduced a numerical strategy that yields an approximate optimization problem with finite dimension. Then, by developing a method grounded in variational argument, they generated approximate solutions for their optimal control problem. They also [15] proposed an alternative numerical method for solving an optimal control problem for nonlinear systems with fractional order that have multiple time-varying delays. Gong, et al. [9] presented a numerical method for solving optimal control of nonlinear systems that are fractional-order and have multiple pantograph time delays. Their approach is founded on a numerical integration method combined with a procedure for calculating gradients. In [21], the necessary optimality conditions of OCPTVD problems are given by the authors; however, when time-varying delays are present, this approach leads to complex boundary value problems involving delay differential equations (DDEs), which are often difficult to solve. It is important to note that this classical approach remains valuable when structural analysis and theoretical optimality characterization are of primary interest.

In the present study, we propose an effective gradient projection method for OCPTVD in state and control variables. First, we obtain a variational inequality as necessary optimality conditions by introducing the gradient of the objective functional with time-varying delays. The variational inequality has gained attention for developing an efficient and convergent numerical algorithm. Computation of the gradient is a critical issue in variational inequality. Then by defining a projection operator and finding a relation with variational inequality, we achieve an iterative scheme for solving OCPTVD. For coercive objective functional, the method's convergence is ensured.

The structure of the remainder of this paper is outlined as follows: We formulate the OCPTVD problem in Section 2. The variational inequality is given as necessary conditions in Section 3. The method of gradient projection is introduced in Section 4. Numerical examples are solved by using the proposed method to illustrate the theoretical findings in Section 5. Finally, Section 6 provides some concluding remarks.

2 OCPTVD statement

We consider an OCPTVD in which $\mathbf{k}(t)$ is the delay function in the state variable $\mathbf{x}(t)$ and $\mathbf{r}(t)$ is the delay function in the control variable $\mathbf{u}(t)$. This problem reads as

$$\min\ J = G(\mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} A(\mathbf{x}, \mathbf{x_k}, \mathbf{u}, \mathbf{u_r}, t)dt, \tag{1}$$

subject to

$$\dot{\mathbf{x}} = F(\mathbf{x}, \mathbf{x_k}, \mathbf{u}, \mathbf{u_r}, t), \qquad t \in [t_0, t_f], \tag{2}$$

$$\mathbf{x}(t) = \phi(t), \qquad t_0 - \mathbf{k}(t_0) \leq t \leq t_0, \tag{3}$$

$$\mathbf{u}(t) = \psi(t), \qquad t_0 - \mathbf{r}(t_0) \leq t \leq t_0, \tag{4}$$

in which $\mathbf{x} \in (H^1[t_0,t_f])^n$ and $\mathbf{u} \in U$ that U is a convex subset of $(L^2[t_0,t_f])^m$. Also, t_0 and t_f are fixed and $[t_0,t_f] \subset \mathbb{R}^+$. Furthermore, $\mathbf{u_r}(t)$ and $\mathbf{x_k}(t)$, the shifted control and state vectors, are as:

$$\mathbf{u_r}(t) = \mathbf{u}(t-\mathbf{r}(t)) = \begin{pmatrix} u_1(t-r_1(t)) \\ u_2(t-r_2(t)) \\ \vdots \\ u_m(t-r_m(t)) \end{pmatrix}$$

and

$$\mathbf{x_k}(t) = \mathbf{x}(t-\mathbf{k}(t)) = \begin{pmatrix} x_1(t-k_1(t)) \\ x_2(t-k_2(t)) \\ \vdots \\ x_n(t-k_n(t)) \end{pmatrix},$$

in which $k_i : \mathbb{R}^+ \longrightarrow \mathbb{R}^+$ for $i = 1,\ldots,n$ and $r_j : \mathbb{R}^+ \longrightarrow \mathbb{R}^+$ for $j = 1,\ldots,m$ and $0 \le \dot{\mathbf{k}}(t) < 1$ and $0 \le \dot{\mathbf{r}}(t) < 1$. Other relevant mappings, which will be used in this paper, are

$$\begin{aligned}
&G : \mathbb{R}^n \times \mathbb{R}^+ \longrightarrow \mathbb{R},\\
&A : (H^1[t_0,t_f])^n \times (H^1[t_0,t_f])^n \times (L^2[t_0,t_f])^m \times (L^2[t_0,t_f])^m \times \mathbb{R}^+ \longrightarrow L^2[t_0,t_f],\\
&F : (H^1[t_0,t_f])^n \times (H^1[t_0,t_f])^n \times (L^2[t_0,t_f])^m \times (L^2[t_0,t_f])^m \times \mathbb{R}^+ \longrightarrow (L^2[t_0,t_f])^n,
\end{aligned}$$

where all of them are twice continuously differentiable. We recall that $H^1[t_0,t_f]$ is a Sobolev space, and its n-fold product is denoted by $(H^1[t_0,t_f])^n$. Also, $L^2[t_0,t_f]$ is the space of square integrable functions, and its m-fold product is represented by $(L^2[t_0,t_f])^m$. Moreover in (3) and (4), $\psi(t)$ and $\phi(t)$ are specific functions.

3 Variational inequalities

We show that the necessary optimality conditions for the OCPTVD (1)–(4) can be formulated as a variational inequality. To begin with, let us state some definitions and theorems.

Definition 1. *Assume that $\mathscr{G}$ is a functional on a locally convex topological vector space X. Then the Gateaux derivative at $\omega \in X$ in the direction $\mu \in X$, also referred to as the G-derivative, is as*

$$\lim_{\theta \to 0} \frac{\mathscr{G}(\omega + \theta\mu) - \mathscr{G}(\omega)}{\theta} \equiv \delta\mathscr{G}(\omega,\mu).$$

The functional $\mathscr{G}$ is G-differentiable at $\omega \in X$, if $\delta\mathscr{G}(\omega,\mu)$ exists for all $\mu \in X$.

In a Banach space equipped with a well-defined inner product, by applying the G-derivative, one can generally characterize the gradient of a functional.

Definition 2. *Suppose that X is a Hilbert space with inner product $\langle \cdot,\cdot \rangle$ and $\mathscr{G}$ is a G-differentiable functional on X. The gradient of $\mathscr{G}$ at $\omega \in X$, denoted $\nabla\mathscr{G}(\omega)$, is defined as*

$$\delta\mathscr{G}(\omega,\nu) = \langle \nabla\mathscr{G}(\omega), \nu \rangle, \qquad \text{for all } \nu \in X.$$

Now, we recall the following key result (variational inequality necessary condition).

Theorem 1. *Assume that X is a topological vector space and that $V \subset X$ is an arbitrary convex set. Let functional $\mathscr{G}(w)$ on X be G-differentiable at $w^* \in V$. Then the following condition is necessary for $w^* \in V$ to be the minimum of $\mathscr{G}$ on V:*

$$\delta\mathscr{G}(w^*, w - w^*) \geq 0, \qquad \textit{for all } w \in V. \tag{5}$$

Proof. For the proof, see [5]. □

We mention that, for an arbitrary convex subset V of a Hilbert space, if the gradient of a functional $\mathscr{G}$ exists, then the variational inequality (5) becomes as

$$w^* \in V \quad \text{such that} \quad \langle \nabla\mathscr{G}(w^*), w - w^* \rangle \geq 0, \qquad \text{for all } w \in V. \tag{6}$$

In fact, when V is convex, $w - w^*$ gives rise to all direction vectors from w^* for all $w \in V$.

To determine the variational inequality for our problem, using (2) and (3), we obtain

$$\mathbf{x}(t) = \phi(t_0) + \int_{t_0}^{t} F[\mathbf{x}, \mathbf{x_k}, \mathbf{u}, \mathbf{u_r}, \xi] d\xi,$$

so

$$\delta\mathbf{x} = \int_{t_0}^{t} \left[\frac{\partial F}{\partial \mathbf{x}} \delta\mathbf{x} + \frac{\partial F}{\partial \mathbf{x_k}} \delta\mathbf{x_k} + \frac{\partial F}{\partial \mathbf{u}} \delta\mathbf{u} + \frac{\partial F}{\partial \mathbf{u_r}} \delta\mathbf{u_r} \right] d\xi.$$

From this equation, we get

$$\frac{d}{dt}(\delta\mathbf{x}) = \frac{\partial F}{\partial \mathbf{x}} \delta\mathbf{x}(t) + \frac{\partial F}{\partial \mathbf{x_k}} \delta\mathbf{x_k}(t) + \frac{\partial F}{\partial \mathbf{u}} \delta\mathbf{u}(t) + \frac{\partial F}{\partial \mathbf{u_r}} \delta\mathbf{u_r}(t); \quad \delta\mathbf{x}(t_0) = 0. \tag{7}$$

The variational of J obeys

$$\delta J = \left[\frac{\partial G(\mathbf{x}(t), t)}{\partial \mathbf{x}} \delta\mathbf{x}(t) \right]_{t=t_f} + \int_{t_0}^{t_f} \left(\frac{\partial A}{\partial \mathbf{x}} \delta\mathbf{x} + \frac{\partial A}{\partial \mathbf{x_k}} \delta\mathbf{x_k} + \frac{\partial A}{\partial \mathbf{u}} \delta\mathbf{u} + \frac{\partial A}{\partial \mathbf{u_r}} \delta\mathbf{u_r} \right) dt. \tag{8}$$

Let us consider terms in (8), which involve variations A with respect to both x and x_k. The change of variable $\mathfrak{z}_j = t - k_j(t)$ in these terms (details are in the Appendix) results in

$$\int_{t_0}^{t_f} \left(\frac{\partial A}{\partial \mathbf{x}} \delta\mathbf{x} + \frac{\partial A}{\partial \mathbf{x_k}} \delta\mathbf{x_k} \right) dt = \sum_{j=1}^{n} \int_{t_0}^{t_f - k(t_f)} \left(\frac{\partial A}{\partial x_j} + \left(\frac{\partial A}{\partial (x_k)_j} \times \frac{1}{1 - \dot{k}_j(t + k_j(t))} \right) \right) \delta x_j dt$$
$$+ \sum_{j=1}^{n} \int_{t_f - k(t_f)}^{t_f} \left(\frac{\partial A}{\partial x_j} \right) \delta x_j dt. \tag{9}$$

We introduce $\boldsymbol{\lambda}$ as the adjoint variable, which is a solution to a final-value problem as follows:

$$\begin{cases} (-1)\dfrac{d\boldsymbol{\lambda}}{dt} = \sum_{j=1}^{n}\left(\dfrac{\partial A}{\partial x_j} + \boldsymbol{\lambda}^T\dfrac{\partial F}{\partial x_j} + \left[\left(\dfrac{\partial A}{\partial (x_k)_j} + \boldsymbol{\lambda}^T\dfrac{\partial F}{\partial (x_k)_j}\right) \times \dfrac{1}{1-\dot{k}_j(t+k_j(t))}\right]\right), \\ \text{for} \quad t \in [t_0, t_f - k_j(t_f)] \quad \text{and} \\ (-1)\dfrac{d\boldsymbol{\lambda}}{dt} = \sum_{j=1}^{n}\left(\dfrac{\partial A}{\partial x_j} + \boldsymbol{\lambda}^T\dfrac{\partial F}{\partial x_j}\right), \quad \text{for} \quad t \in [t_f - k_j(t_f), t_f] \quad \text{and} \\ \boldsymbol{\lambda}^T(t_f) = \left[\dfrac{\partial G(\mathbf{x}(t),t)}{\partial \mathbf{x}}\right]_{t=t_f}. \end{cases} \tag{10}$$

By using (10), we have

$$\begin{aligned} &\int_{t_0}^{t_f-k(t_f)} \left[-\dot{\boldsymbol{\lambda}} - \boldsymbol{\lambda}^T \sum_{j=1}^{n}\left(\frac{\partial F}{\partial x_j} + \frac{\partial F}{\partial (x_k)_j} \times \frac{1}{1-\dot{k}_j(t+k_j(t))}\right)\right]\delta\mathbf{x}dt \\ &\qquad + \int_{t_f-k(t_f)}^{t_f}\left[-\dot{\boldsymbol{\lambda}} - \boldsymbol{\lambda}^T\sum_{j=1}^{n}(\frac{\partial F}{\partial x_j})\right]\delta\mathbf{x}dt \\ &\quad = \int_{t_0}^{t_f}(-\dot{\boldsymbol{\lambda}})\delta\mathbf{x}dt \\ &\qquad + \int_{t_0}^{t_f-k(t_f)}\left[-\boldsymbol{\lambda}^T\sum_{j=1}^{n}\left(\frac{\partial F}{\partial x_j} + \frac{\partial F}{\partial (x_k)_j} \times \frac{1}{1-\dot{k}_j(t+k_j(t))}\right)\right]\delta\mathbf{x}dt \\ &\qquad + \int_{t_f-k(t_f)}^{t_f}\left[-\boldsymbol{\lambda}^T\sum_{j=1}^{n}(\frac{\partial F}{\partial x_j})\right]\delta\mathbf{x}dt. \end{aligned} \tag{11}$$

Since $\delta\mathbf{x}(t_0) = 0$, we have

$$\left[\boldsymbol{\lambda}^T\delta\mathbf{x}(t)\right]_{t_0}^{t_f} = [\boldsymbol{\lambda}(t_f)]^T\delta\mathbf{x}(t_f) - [\boldsymbol{\lambda}(t_0)]^T\delta\mathbf{x}(t_0) = \left(\frac{\partial G(\mathbf{x}(t_f),t_f)}{\partial \mathbf{x}}\right)^T\delta\mathbf{x}(t_f),$$

and therefore

$$\int_{t_0}^{t_f}(-\dot{\boldsymbol{\lambda}})\delta\mathbf{x}dt = \int_{t_0}^{t_f}\boldsymbol{\lambda}^T\frac{d}{dt}(\delta\mathbf{x})dt - \left[\boldsymbol{\lambda}^T\delta\mathbf{x}(t)\right]_{t_0}^{t_f} = \int_{t_0}^{t_f}\boldsymbol{\lambda}^T\frac{d}{dt}(\delta\mathbf{x})dt - \frac{\partial G(\mathbf{x}(t_f),t_f)}{\partial \mathbf{x}}\delta\mathbf{x}(t_f).$$

It follows from (7) that

$$
\begin{aligned}
\int_{t_0}^{t_f}(-\dot{\boldsymbol{\lambda}})\delta\mathbf{x}dt = &\int_{t_0}^{t_f}\boldsymbol{\lambda}^T\left(\frac{\partial F}{\partial \mathbf{x}}\delta\mathbf{x}+\frac{\partial F}{\partial \mathbf{x_k}}\delta\mathbf{x_k}+\frac{\partial F}{\partial \mathbf{u}}\delta\mathbf{u}+\frac{\partial F}{\partial \mathbf{u_r}}\delta\mathbf{u_r}\right)dt\\
&-\left(\frac{\partial G(\mathbf{x}(t_f),t_f)}{\partial \mathbf{x}}\right)^T\delta\mathbf{x}(t_f)\\
=&\int_{t_0}^{t_f-k(t_f)}\boldsymbol{\lambda}^T\sum_{j=1}^{n}\left(\frac{\partial F}{\partial x_j}+\frac{\partial F}{\partial (x_k)_j}\times\frac{1}{1-\dot{k}_j(t+k_j(t))}\right)\delta\mathbf{x}dt\\
&+\int_{t_f-k(t_f)}^{t_f}\boldsymbol{\lambda}^T\sum_{j=1}^{n}(\frac{\partial F}{\partial x_j})\delta\mathbf{x}dt\\
&+\int_{t_0}^{t_f}\boldsymbol{\lambda}^T\left(\frac{\partial F}{\partial \mathbf{u}}\delta\mathbf{u}+\frac{\partial F}{\partial \mathbf{u_r}}\delta\mathbf{u_r}\right)dt-\left(\frac{\partial G(\mathbf{x}(t_f),t_f)}{\partial \mathbf{x}}\right)^T\delta\mathbf{x}(t_f).
\end{aligned}
$$

Hence

$$
\begin{aligned}
\delta J =&(\frac{\partial G(\mathbf{x}(t_f),t_f)}{\partial \mathbf{x}})^T\delta\mathbf{x}(t_f)\\
&+\int_{t_0}^{t_f-k(t_f)}\boldsymbol{\lambda}^T\sum_{j=1}^{n}\left(\frac{\partial F}{\partial x_j}+\frac{\partial F}{\partial (x_k)_j}\times\frac{1}{1-\dot{k}_j(t+k_j(t))}\right)\delta\mathbf{x}dt\\
&+\int_{t_f-k(t_f)}^{t_f}\boldsymbol{\lambda}^T\sum_{j=1}^{n}(\frac{\partial F}{\partial x_j})\delta\mathbf{x}dt\\
&+\int_{t_0}^{t_f}\boldsymbol{\lambda}^T\left(\frac{\partial F}{\partial \mathbf{u}}\delta\mathbf{u}+\frac{\partial F}{\partial \mathbf{u_r}}\delta\mathbf{u_r}\right)dt-\left(\frac{\partial G(\mathbf{x}(t_f),t_f)}{\partial \mathbf{x}}\right)^T\delta\mathbf{x}(t_f)\\
&+\int_{t_0}^{t_f-k(t_f)}\left[-\boldsymbol{\lambda}^T\sum_{j=1}^{n}\left(\frac{\partial F}{\partial x_j}+\frac{\partial F}{\partial (x_k)_j}\times\frac{1}{1-\dot{k}_j(t+k_j(t))}\right)\right]\delta\mathbf{x}dt\\
&+\int_{t_f-k(t_f)}^{t_f}\left[-\boldsymbol{\lambda}^T\sum_{j=1}^{n}(\frac{\partial F}{\partial x_j})\right]\delta\mathbf{x}dt+\int_{t_0}^{t_f}\left(\frac{\partial A}{\partial \mathbf{u}}\delta\mathbf{u}+\frac{\partial A}{\partial \mathbf{u_r}}\delta\mathbf{u_r}\right)dt,
\end{aligned}
$$

and therefore

$$
\delta J=\int_{t_0}^{t_f}\left[\frac{\partial A}{\partial \mathbf{u}}\delta\mathbf{u}+\frac{\partial A}{\partial \mathbf{u_r}}\delta\mathbf{u_r}+\boldsymbol{\lambda}^T\left(\frac{\partial F}{\partial \mathbf{u}}\delta\mathbf{u}+\frac{\partial F}{\partial \mathbf{u_r}}\delta\mathbf{u_r}\right)\right]dt. \tag{12}
$$

We define the Hamiltonian function H by

$$
H(\mathbf{x}(t),\mathbf{x_k}(t),\mathbf{u}(t),\mathbf{u_r}(t),\boldsymbol{\lambda},t)=A\left(\mathbf{x}(t),\mathbf{x_k}(t),\mathbf{u}(t),\mathbf{u_r}(t),t\right)+\boldsymbol{\lambda}^T F\left(\mathbf{x}(t),\mathbf{x_k}(t),\mathbf{u}(t),\mathbf{u_r}(t),t\right),
$$

and then we rewrite (12) as follows:

$$
\delta J=\int_{t_0}^{t_f}\left[\frac{\partial H}{\partial \mathbf{u}}\delta\mathbf{u}+\frac{\partial H}{\partial \mathbf{u_r}}\delta\mathbf{u_r}\right]dt. \tag{13}
$$

By change of variable $\eta_i = t-r_i(t)$, and noting that $\delta u_i = 0$ for any time $t<t_0$, (13) can be written as

$$
\delta J(\mathbf{u})=\sum_{i=1}^{m}\left\{\int_{t_0}^{t_f-r_i(t_f)}\left[\frac{\partial H}{\partial u_i}+\left(\frac{\partial H}{\partial (u_r)_i}\times\frac{1}{1-\dot{r}_i(t+r_i(t))}\right)\right]\delta u_i dt+\int_{t_f-r_i(t_f)}^{t_f}\frac{\partial H}{\partial u_i}\delta u_i dt\right\}. \tag{14}
$$

We define the gradient of the criterion (1) as follows:

$$\Big[\nabla J(\mathbf{u})\Big]_i = \begin{cases} \dfrac{\partial H}{\partial u_i} + \left(\dfrac{\partial H}{\partial (u_r)_i} \times \dfrac{1}{1-\dot{r}_i(t+r_i(t))}\right) & if \;\; t \in [t_0, t_f - r_i(t_f)] \\ \dfrac{\partial H}{\partial u_i} & if \;\; t \in [t_f - r_i(t_f), t_f], \end{cases}$$

for $i = 1,\ldots,m$. According to Theorem 1, we have the variational inequality optimality condition as follows:

$$\delta J(\mathbf{u}^*) = \sum_{i=1}^{m} \int_{t_0}^{t_f} \Big[\nabla J(\mathbf{u}^*)\Big]_i (u_i - u_i^*) dt \geq 0, \qquad \text{for all } \mathbf{u} \in U. \tag{15}$$

Finally

$$\forall \mathbf{u} \in U, \qquad \delta J(\mathbf{u}^*) = \langle \nabla J(\mathbf{u}^*), \mathbf{u} - \mathbf{u}^* \rangle \geq 0. \tag{16}$$

4 Gradient projection method

In this section, for solving the optimality condition (16), we provide a gradient projection method. We define the mathematical program

$$\min \quad \|u - w\| \qquad \text{such that} \quad u \in U \subset X,$$

in which $\|w\| = \left\langle w^T, w \right\rangle^{\frac{1}{2}}$ and X is a Hilbert space. We consider the projection operator $P_U : X \longrightarrow U$ as

$$P_U(w) = \underset{u \in U}{argmin} \|u - w\|. \tag{17}$$

We mention that (17) can be considered as the following inequality:

$$\langle P_U(w) - w, u - P_U(w) \rangle \geq 0 \qquad \text{for all } u \in U. \tag{18}$$

We recall from [5] that for any positive constant γ and all $\mathbf{u} \in U$, the inequality

$$\left\langle \mathbf{u}^* - \left(\mathbf{u}^* - \gamma \nabla J(\mathbf{u}^*)\right), \mathbf{u} - \mathbf{u}^* \right\rangle \geq 0, \tag{19}$$

is equivalent to the variational inequality (16). We compare the above inequality with (18), and for the optimal control $\mathbf{u}^*$, we rewrite it as follows:

$$\mathbf{u}^* = P_U\left(\mathbf{u}^* - \gamma \nabla J(\mathbf{u}^*)\right). \tag{20}$$

The optimal control $\mathbf{u}^*$ is the fixed point of $P_U\left(\mathbf{u}^* - \gamma \nabla J(\mathbf{u}^*)\right)$ on U. Now, we are ready to numerically approximate the control $\mathbf{u}^*$. For obtaining the approximated optimal control, we introduce the following fixed point iteration scheme:

$$\mathbf{u}^{N+1} = P_U\left(\mathbf{u}^N - \gamma \nabla J(\mathbf{u}^N)\right), \qquad N = 1, 2, \ldots. \tag{21}$$

The procedure (21) is a projected gradient method used to solve the variational inequality (16). Next, in Theorem 2, the convergence property of (21) is stated. We need to introduce coerciveness (α-convexity), which is important to state the convergence.

Definition 3. *Suppose that $\mathscr{G} : X \to \mathbb{R}^1$ is a functional on normed vector space X. Then $\mathscr{G}$ is called coercive (α-convex) if a real scalar $\alpha > 0$ exists such that*

$$\mathscr{G}[(1-\lambda)\omega + \lambda\nu] \leq (1-\lambda)\mathscr{G}(\omega) + \lambda\mathscr{G}(\nu) - \frac{\alpha}{2}\lambda(1-\lambda)\|\omega-\nu\|^2, \tag{22}$$

holds for all $\omega, \nu \in X$ and $\lambda \in (0,1)$.

If we put $\alpha = 0$, then (22) is the usual notion of convexity. The following result can now be stated.

Lemma 1. *Let $\mathscr{G} : X \to \mathbb{R}^1$ be G-differentiable at X. Then $\mathscr{G}$ is coercive if and only if*

$$\delta\mathscr{G}(\omega, \omega-\nu) - \delta\mathscr{G}(\nu, \omega-\nu) \geq \alpha\|\omega-\nu\|^2 \qquad \text{for all } \omega, \nu \in X. \tag{23}$$

Proof. For the proof, see [5]. □

Note that (23) can also be written as

$$\langle \nabla\mathscr{G}(\omega) - \nabla\mathscr{G}(\nu), \omega-\nu \rangle \geq \alpha\|\omega-\nu\|^2, \tag{24}$$

when the gradient of $\mathscr{G}$ is defined on X.

Theorem 2 demonstrates the convergence of our projection algorithm.

Theorem 2. *Suppose that X is a normed vector space and that $\mathscr{G} : U \subset X \to \mathbb{R}^1$ is a coercive functional for which $\alpha > 0$ exists such that $\nabla\mathscr{G}(\omega)$ is defined. If the inequality*

$$\|\nabla\mathscr{G}(\omega) - \nabla\mathscr{G}(\nu)\| \leq \beta\|\omega-\nu\| \qquad \text{for all } \omega, \nu \in U, \tag{25}$$

which is called the Lipschitz condition, holds, then by choosing γ as the fixed step size from $(0, \frac{2\alpha}{\beta^2})$, the projection method (21) converges to the minimum ω^* of $\mathscr{G}$ on U.

Proof. For the proof, see [5]. □

4.1 Structure of the algorithm

We state the algorithm of gradient projection method by the following rule:
Step 0: Set $N = 0$ and select an error tolerance ε and the initial control estimate $\mathbf{u}^0 \in U$.
Step 1: Solve the following problem by using $\mathbf{u}^N(t)$:

$$\begin{cases} \dot{\mathbf{x}} = F\big((\mathbf{x}(t), \mathbf{x_k}(t), \mathbf{u}^N(t), \mathbf{u}_r^N(t), t)\big), \\ \mathbf{x}(t) = \phi(t), \quad t_0 - \mathbf{k}(t_0) \leq t \leq t_0. \end{cases} \tag{26}$$

The solution is denoted as $\mathbf{x}^N(t)$.
Step 2: Solve the following problem using $\mathbf{u}^N(t)$ and $\mathbf{x}^N(t)$:

$$(-1)\frac{d\boldsymbol{\lambda}}{dt} = \sum_{j=1}^{n} \frac{\partial H(\mathbf{x}^N, \mathbf{x_k}^N, \mathbf{u}^N, \mathbf{u_r}^N, \boldsymbol{\lambda}, t)}{\partial x_j} + \left[\frac{\partial H(\mathbf{x}^N, \mathbf{x_k}^N, \mathbf{u}^N, \mathbf{u_r}^N, \boldsymbol{\lambda}, t)}{\partial (x_k)_j} \times \frac{1}{1-\dot{k}_j(t+k_j(t))}\right], \tag{27}$$

for $t \in [t_0, t_f - k_j(t_f)]$ and

$$(-1)\frac{d\boldsymbol{\lambda}}{dt} = \sum_{j=1}^{n} \frac{\partial H(\mathbf{x}^N, \mathbf{x_k}^N, \mathbf{u}^N, \mathbf{u_r}^N, \boldsymbol{\lambda}, t)}{\partial x_j}, \tag{28}$$

for $t \in [t_f - k_j(t_f), t_f]$ and $\boldsymbol{\lambda}(t_f) = \left[\frac{\partial G(\mathbf{x}(t), t)}{\partial \mathbf{x}}\right]_{t=t_f}$.

Step 3: Using $\mathbf{u}^N(t)$, $\mathbf{x}^N(t)$, and $\boldsymbol{\lambda}^N(t)$, for $i = 1, \ldots, m$, compute

$$\begin{aligned}\left[\nabla J(\mathbf{u}^N)\right]_i = &\frac{\partial H(\mathbf{x}^N, \mathbf{x_k}^N, \mathbf{u}^N, \mathbf{u_r}^N, \boldsymbol{\lambda}^N(t), t)}{\partial u_i} \\ &+ \left[\frac{\partial H(\mathbf{x}^N, \mathbf{x_k}^N, \mathbf{u}^N, \mathbf{u_r}^N, \boldsymbol{\lambda}^N(t), t)}{\partial (u_r)_i} \times \frac{1}{1 - \dot{r}_i(t + r_i(t))}\right],\end{aligned} \tag{29}$$

if $t \in [t_0, t_f - r_i(t_f)]$ and

$$\left[\nabla J(\mathbf{u}^N)\right]_i = \frac{\partial H(\mathbf{x}^N, \mathbf{x_k}^N, \mathbf{u}^N, \mathbf{u_r}^N, \boldsymbol{\lambda}^N(t), t)}{\partial u_i}, \tag{30}$$

if $t \in [t_f - r_i(t_f), t_f]$.

Step 4: Update $\mathbf{u}$, according to

$$\mathbf{u}^{N+1} = P_U\left(\mathbf{u}^N - \gamma \nabla J(\mathbf{u}^N)\right), \tag{31}$$

by a suitable step size γ.

Step 5: Repeat Steps 1–4 until the error $\|\mathbf{u}^{N+1} - \mathbf{u}^N\|$ reaches the tolerance ε.

For updating u in (31), we state how to compute the projection onto the convex set U.

Let us consider $U = \{u \in \mathbb{R}^m : a \le u \le b\}$, where $a, b \in \mathbb{R}^m$ in which $\mathbb{R} = U \cup \{+\infty, -\infty\}$ and also $a \le b$, $b \ne -\infty$, and $a \ne +\infty$. Then P_U is given by

$$P_U(u) = \begin{cases} b & if\ u > b, \\ u & if\ a \le u \le b, \\ a & if\ u < a. \end{cases}$$

Thus, for example, if $U = \mathbb{R}^m_+$, then $P_U(u) = \begin{cases} u & if\ 0 \le u, \\ 0 & if\ u < 0. \end{cases}$

5 Numerical results

The examples presented in this section demonstrate the effectiveness of the method discussed. We develop the codes using MATLAB 2020 on a machine with 8 GB of RAM and an Intel(R) Core(TM) i5 processor.

Example 1. The first example was studied in [7] and the analytical optimal solution is given. Consider the optimal control problem containing the time delay $r(t) = 2$ in the control and $k(t) = 1$ in the state as follows:

$$Min \qquad J = \int_0^3 \left(x^2(t) + u^2(t)\right) dt, \tag{32}$$

subject to

$$\frac{dx}{dt} = x(t-1)u(t-2), \quad t \in [0,3], \tag{33}$$

$$x(t) = 1, \quad -1 \le t \le 0, \tag{34}$$

$$u(t) = 0, \quad -2 \le t \le 0, \tag{35}$$

$$-1 \le u \le 1. \tag{36}$$

The algorithm is run setting $u^0 = 0$, $\gamma = 0.1$, and the tolerance as $\varepsilon = 10^{-3}$. The iterative procedure has the following structure:
Step 1:

$$\begin{cases} \frac{dx}{dt} = x(t-1)u(t-2), & \\ x(t) = 1, & -1 \le t \le 0, \end{cases}$$

Step 2:
We know from (27) and (28) that

$$-\frac{d\lambda}{dt} = \begin{cases} 2x(t) + \lambda u(t-2), & 0 \le t \le 2, \\ 2x(t), & 2 \le t \le 3, \end{cases}$$

and $\lambda(3) = 0$.
Step 3:
Also from (29) and (30), we have

$$\nabla J = \begin{cases} 2u(t) + \lambda x(t-1), & 0 \le t < 1, \\ 2u(t), & 1 \le t \le 3. \end{cases}$$

Step 4:
Update u as follows:

$$u^{N+1} = P_U(u^N - \gamma \nabla J(u^N)) = \begin{cases} 1 & if\ u > 1, \\ u^N - \gamma \nabla J(u^N) & if\ -1 \le u \le 1, \\ -1 & if\ u < -1. \end{cases}$$

Step 5:
Repeat this for subsequent iterations until the stopping test is satisfied.

After implementing the algorithm, the stopping condition is satisfied after 9 iterations in 67 seconds. In Table 1, some iterations are provided. The numerical convergence can be found in the right column.

Numerical results obtained by the method are shown in Figure 1. By using the optimal solution, we determine the optimal performance index explicitly as follows:

$$J = \int_0^3 \left(x^2(t) + u^2(t) \right) dt \approx 2.76159752.$$

Gollmann, Kern, and Maurer [7] proposed a numerical method to solve the problem given by (32)–(36). Also, the analytical solutions of this problem are presented in [7]. We compare analytical solutions with

Table 1: Control function in iterations for Example 1.

N	$u^N(t)$	t	$\|u^{N+1}-u^N\|$
1	$0.2t-0.6$, 0	$[0,1)$, $[1,3]$	-
2	-1, $0.36t-1.0667$, 0	$[0,0.19)$, $[0.19,1)$, $[1,3]$	0.3067
3	-1, $0.35t-1.1121$, 0	$[0,0.18)$, $[0.18,1)$, $[1,3]$	0.0554
⋮	⋮	⋮	⋮
8	-1, $0.29155t-1.01514$, 0	$[0,0.11)$, $[0.11,1)$, $[1,3]$	0.0022
9	-1, $0.28851t-1.01227$, 0	$[0,0.10)$, $[0.10,1)$, $[1,3]$	0.00017

Table 2: Cost functional values for Example 1.

Method	J
Presented method	2.76159752
Analytical solution	2.761594156
Gollmann, Kern, and Maurer [7]	2.761598

our suggested method results. The comparison results of control and state variables are shown in Figure 1. A comparison among the analytical solution, the value of J reported in [7], and the value of J obtained by our method, is shown in Table 2.

Example 2. This example is taken from [11]. Consider the following delayed problem:

$$\dot{x}(t) = \frac{t-1}{t} x_k(t)x(t) + u(t), \quad 1 \le t \le 3, \tag{37}$$

$$x(t) = 1, \quad 0 \le t \le 1, \tag{38}$$

where $k(t) = \ln(t) + 1$ is a continuous variable time-varying lag in state and the performance index that must be minimized is as follows:

$$J = \int_1^3 \Big(x^2(t) + u^2(t) \Big) dt. \tag{39}$$

In order to employ the algorithm, we set $u^0 = \ln(t) - 2$, $\gamma = 0.1$, and the tolerance as $\varepsilon = 10^{-3}$. Note that $\dot{k}(t) = \frac{1}{t}$. Therefore by using (27) and (28), the adjoint equation and its condition become

$$\begin{cases} \dfrac{d\lambda}{dt} = -2x(t) - \lambda(\frac{t-1}{t}) & 1 \le t \le 3 \\ \lambda(3) = 0. \end{cases}$$

Also from (29) and (30), we have

$$\nabla J = 2u(t) + \lambda.$$

Figure 2 shows the approximate values of our gradient projection method. The computational result of the cost functional is $J = 1.663243$. This was achieved in 132 seconds. Table 3 indicates a comparison

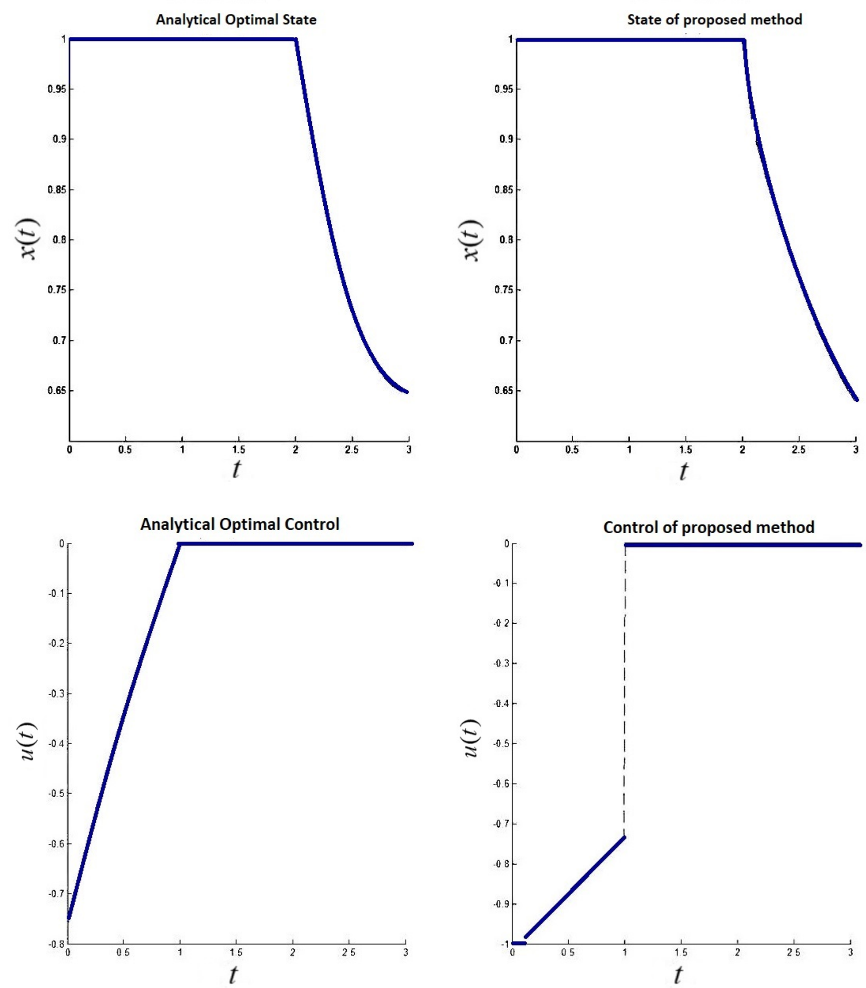

Figure 1: Results from the simulation of state and control for Example 1 and comparing with the optimal solution.

between our method results and the values of J in different cases reported by Hoseini and Marzban [11].

Example 3. Consider the OCPTVD

$$Min \quad J = \int_0^2 \left(x_k^2(t) + u_r^2(t) \right) dt, \tag{40}$$

subject to

$$\frac{d}{dt}x(t) = tx(t) + x_k(t) + u_r(t), \qquad 0 \le t \le 2, \tag{41}$$

$$x(t) = 1, \qquad t \le 0, \tag{42}$$

$$u(t) = 1.4t - 2.8, \qquad t \le 0, \tag{43}$$

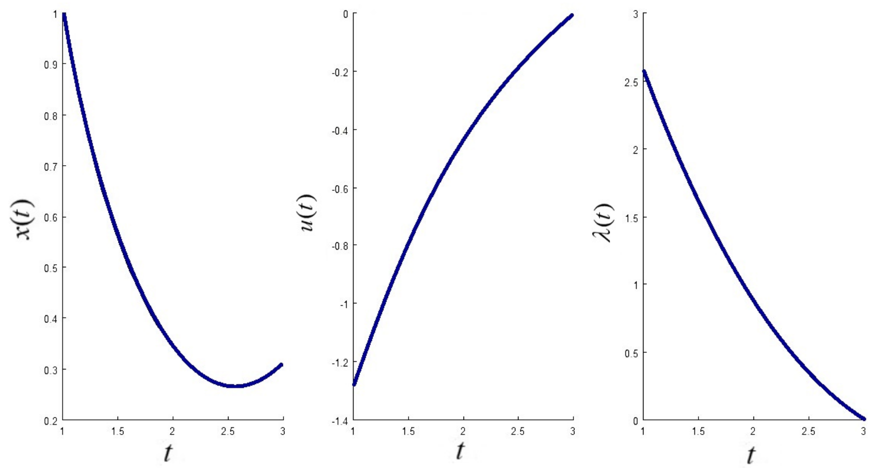

Figure 2: Approximation of state $x(t)$ and control $u(t)$ and costate $\lambda(t)$ for Example 2.

Table 3: Comparison results for Example 2.

Method	J
Hoseini and Marzban [11] for d=1	1.667931637055
Hoseini and Marzban [11] for d=2	1.667931637160
Proposed method	1.663243

where $k(t) = 1 + \dfrac{t^2}{2(1+t)}$ and $r(t) = 1 - \dfrac{1}{t-3}$ are the delay functions in control and state that satisfy $k(t) \geq 0$, $r(t) \geq 0$, and $0 \leq \dot{k}(t) < 1$ and $0 \leq \dot{r}(t) < 1$ for $0 \leq t \leq 2$.

We perform the algorithm with $u^0 = 1.4t - 2.8$, $\gamma = 0.01$, and the tolerance as $\varepsilon = 10^{-3}$. The results are shown in Figure 3. We calculate $J = 6.0745$ as the minimum cost functional value. The CPU time for this example is 148 seconds.

6 Conclusion

We proposed a method to solve the OCPTVD problem using the gradient projection method. First, by defining the gradient of the objective functional, we derived a variational inequality as an optimality condition. Then, we introduced a projection operator and achieved a convergent iterative method. To illustrate the effectiveness of our method, we applied it to several examples. The results demonstrated the applicability and validity of the presented method.

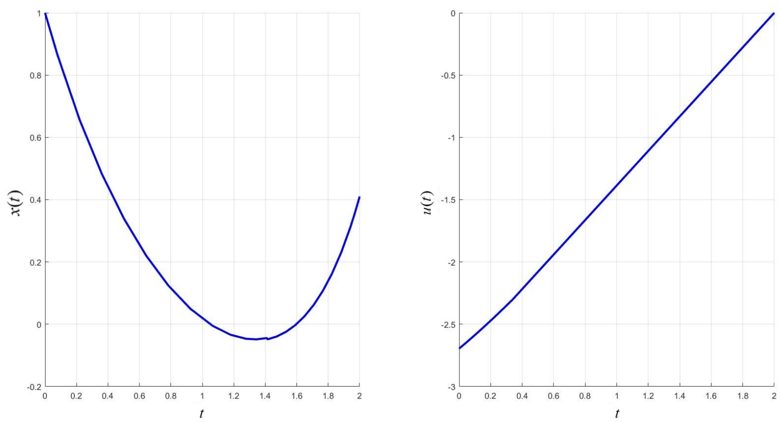

Figure 3: State $x(t)$ and control $u(t)$ for Example 3.

Appendix

The term of the form

$$\int_{t_0}^{t_f} \frac{\partial A}{\partial x_{k_j}} \left(\delta x_k\right)_j dt = \int_{t_0}^{t_f} \frac{\partial A}{\partial x_j(t-k_j(t))} \Big(\delta x_j(t-k_j(t))\Big) dt,$$

is re-expressed by change of variable

$$\mathfrak{z}_j = t - k_j(t). \tag{44}$$

Since the $k_j(t)$ is differentiable, we have

$$\frac{d\mathfrak{z}_j}{dt} = 1 - \dot{k}_j(t). \tag{45}$$

Therefore using the change of variable defined in (44) implies that

$$\int_{t_0}^{t_f} \frac{\partial A}{\partial x_j(t-k_j(t))} \delta x_j(t-k_j(t)) dt = \int_{t_0-k_j(t_0)}^{t_f-k_j(t_f)} \left[\frac{\partial A}{\partial (x_k)_j} \times \frac{1}{1-\dot{k}_j(t)}\right]_{t=\mathfrak{z}+k_j(\mathfrak{z})} \delta x_j d\mathfrak{z}_j. \tag{46}$$

By changing the apparent variable, (46) becomes the equivalent expression

$$\int_{t_0}^{t_f} \frac{\partial A}{\partial (x_k)_j} \delta (x_k)_j dt = \int_{t_0-k_j(t_0)}^{t_f-k_j(t_f)} \left(\frac{\partial A}{\partial (x_k)_j} \times \frac{1}{1-\dot{k}_j(t+k_j(t))}\right) \delta x_j dt. \tag{47}$$

Furthermore, the facts that $x(t) = \phi(t)$ and that $\phi(t)$ is a specified function for any time $t \leq t_0$, imply that the variational ϕ is zero and that $\delta x_j = 0$ for any time $t \leq t_0$. Therefore, we conclude from (47) that

$$\int_{t_0}^{t_f} \frac{\partial A}{\partial (x_k)_j} \delta (x_k)_j dt = \int_{t_0}^{t_f-k_j(t_f)} \left(\frac{\partial A}{\partial (x_k)_j} \times \frac{1}{1-\dot{k}_j(t+k_j(t))}\right) \delta x_j dt.$$

References

[1] T.S. Angell, A. Kirsch, *On the necessary conditions for optimal control of retarded systems,* Appl. Math. Optim. **22** (1990) 117–145.

[2] H.T. Banks, *Necessary condition for control problems with variable time lags,* SIAM J. Control. **8** (1968) 9–47.

[3] J.T. Betts, S.L. Campbell, K.C. Thompson, *Solving optimal control problems with control delays using direct transcription,* Appl. Numer. Math. **108** (2016) 185–203.

[4] F. Colonius, D. Hinrichsen, *Optimal control of functional differential systems,* SIAM J. Control Optim. **16** (1978) 861–879.

[5] T.L. Friesz, *Dynamic Optimization and Differential Games,* Springer: New York, 2010.

[6] T. Guinn, *Reduction of delayed optimal control problems to nondelayed problems,* J. Optim. Theory Appl. **18** (1976) 371–377.

[7] L. Gollmann, D. Kern, H. Maurer, *Optimal control problems with delays in state and control variables subject to mixed control-state constraints,* Optim. Control Appl. Meth. **30** (2009) 341–365.

[8] L. Gollmann, H. Maurer, *Theory and applications of optimal control problems with multiple time-delays,* J. Ind. Manag. Optim. **10** (2014) 413.

[9] Z. Gong, C. Liu, K. L. Teo, X. Yi, *Optimal control of nonlinear fractional systems with multiple pantograph-delays,* Appl. Math. Comput. **425** (2022) 127094.

[10] A. Halanay, *Optimal controls for systems with time lag,* SIAM J. Control **6** (1968) 215–234.

[11] S.M. Hoseini, H.R. Marzban, *Costate Computation by an Adaptive Pseudospectral Method for Solving Optimal control problems with piecewise constant Time Lag,* J. Optim. Theory Appl. **170** (2016) 735–755.

[12] A. Jajarmi, M. Hajipour, *An efficient finite difference method for the time-delay optimal control problems with time-varying delay,* Asian J. Control **19** (2017) 554–563.

[13] A. Jajarmi, M. Hajipour, D. Baleanu, *A new approach for the optimal control of time-varying delay systems with external persistent matched disturbances,* J. Vib. Control **24** (2018) 4505–4512, .

[14] C. Liu, R. Loxton, K.L. Teo, S. Wang, *Optimal state-delay control in nonlinear dynamic systems,* Automatica **135** (2022) 109981.

[15] C. Liu, Z. Gong, K.L. Teo, S. Wang, *Optimal control of nonlinear fractional-order systems with multiple time-varying delays,* J. Optim. Theory Appl. **193** (2022) 856–876.

[16] S.M. Mirhosseini-Alizamini, S. Effati, A. Heydari, *An iterative method for suboptimal control of linear time-delayed systems,* Systems Control Lett. **82** (2015) 40–50.

[17] H.R. Marzban, S.M. Hoseini, *Optimal control of linear multi-delay systems with piecewise constant delays,* IMA J. Math. Control Inform. **35** (2018) 183–212.

[18] H.R. Marzban, H. Pirmoradian, *A direct approach for the solution of nonlinear optimal control problems with multiple delays subject to mixed state-control constraints,* Appl. Math. Model. **53** (2018) 189–213.

[19] H.R. Marzban, H. Pirmoradian, *A novel approach for the numerical investigation of optimal control problems containing multiple delays,* Optimal Control Appl. Methods **39** (2018) 302–325.

[20] S.T. Malik, *On necessary optimality conditions for singular controls in dynamical systems with a delay in control,* Numer. Funct. Anal. Optim. **39** (2018) 1669–1689.

[21] S.M. Meshkani, S. Effati, A. Heydari, *Necessary optimality conditions for optimal control problems with time-varying delays*, J. Adv. Math. Model. **13** (2023) 269–283.

[22] S.M. Mirhosseini-Alizamini, S. Effati, *An iterative method for suboptimal control of a class of nonlinear time-delayed systems,* Internat. J. Control **92** (2019) 2869–2885.

[23] S.M. Mirhosseini-Alizamini, *Solving linear optimal control problems of the time delayed systems by Adomian decomposition method,* Iranian J. Numer. Anal. Optim. **9** (2019) 165–183.

[24] D. Wu, Y. Bai, *Time- scaling transformation for optimal control problem with time- varying delay,* Discrete Contin. Dyn. Syst. Ser. S **13** (2020) 1683–1695.