A UNIFIED PARAMETER IDENTIFICATION METHOD FOR NONLINEAR TIME-DELAY SYSTEMS

Qinqin Chai[a,b]
Ryan Loxton[b] and Kok Lay Teo[b]
Chunhua Yang[a]

[a]School of Information Science & Engineering, Central South University, Changsha, China
[b]Department of Mathematics & Statistics, Curtin University, Perth, Australia

(Communicated by Cheng-Chew Lim)

Abstract. This paper deals with the problem of identifying unknown time-delays and model parameters in a general nonlinear time-delay system. We propose a unified computational approach that involves solving a dynamic optimization problem, whose cost function measures the discrepancy between predicted and observed system output, to determine optimal values for the unknown quantities. Our main contribution is to show that the partial derivatives of this cost function can be computed by solving a set of auxiliary time-delay systems. On this basis, the parameter identification problem can be solved using existing gradient-based optimization techniques. We conclude the paper with two numerical simulations.

1. **Introduction.** Time-delay systems arise in many important applications, including medicine [21], chromatography [24], aerospace engineering [5], and chemical reactor control [9]. Time-delays are often the cause of unpredictable and unusual system behavior. For example, it is known that the introduction of time-delays can destabilize systems that would otherwise be uniformly asymptotically stable [4, 19].

If the time-delays in a system are fixed and known, then existing powerful optimal control algorithms (e.g. the control parameterization method; see [3, 25]) can be applied to compute an optimal control for the system. In many systems, however, the time-delays are unknown, which renders most of the existing optimal control algorithms unusable. In this case, the time-delays must first be estimated before an optimal control algorithm is applied. The problem of estimating the time-delays (and possibly other unknown system parameters) from a given set of experimental data is one of the key problems in the study of time-delay systems [20]. Such problems are known as *parameter identification problems*.

Parameter identification for time-delay systems has been the subject of vigorous research activity over the last two decades. An exact least squares algorithm for the estimation of a single input delay is reported in [8]. Algebraic techniques [2] and the steepest descent algorithm [6] have also been proposed for the identification of input delays. In [26], information theory is used to identify time-delays for systems in which each nonlinear term contains at most one unknown delay. Furthermore, in [18] a genetic algorithm is developed for identifying a single time-delay in a

---

linear discrete-time system. Lyapunov function methods have also been used to design delay estimators in [7]. One of the limitations of the existing identification methods in [2, 6, 7, 8, 18, 26] is that they are mainly designed for systems with a single input delay and no unknown parameters. Nonlinear systems with multiple unknown delays are rarely considered in the literature.

In this paper, we consider a general nonlinear delay-differential system with unknown time-delays and unknown system parameters. We formulate the problem of identifying these unknown quantities as a nonlinear optimization problem in which the cost function measures the least-squares error between predicted and observed system output. This type of parameter identification problem was previously considered in [14] for systems in which each nonlinear component contains at most one unknown delay and no unknown system parameters. However, in many real-world systems, such as the purification process of zinc sulphate solution [22], the nonlinear terms contain *both* delays and parameters that need to be identified. Our goal in this paper is to extend the approach pioneered in [14] to these more complicated systems. The key idea is to introduce a set of auxiliary delay-differential systems, and then express the gradient of the least-squares cost function in terms of the solution of these auxiliary systems. On this basis, numerical integration can be used to solve the auxiliary systems, and thereby obtain the gradient of the cost function, which is the main information needed to solve the parameter identification problem via numerical optimization techniques [16]. Based on this idea, we propose a computational algorithm for identifying the unknown time-delays and system parameters in a general nonlinear system. We then demonstrate the effectiveness of this algorithm on two nonlinear parameter identification problems, one being the parameter identification problem for the zinc sulphate purification process.

2. **Problem formulation.** Consider the following nonlinear time-delay system:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta}), \quad t \in [0, T], \tag{1}$$

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \leq 0, \tag{2}$$

where $T > 0$ is a given *terminal time*; $\boldsymbol{x}(t) = [x_1(t), \ldots, x_n(t)]^\top \in \mathbb{R}^n$ is the *state vector*; $\tilde{\boldsymbol{x}}(t) = [\boldsymbol{x}(t - \tau_1)^\top, \ldots, \boldsymbol{x}(t - \tau_m)^\top]^\top \in \mathbb{R}^{nm}$ is the *delayed state vector*; and $\boldsymbol{\zeta} = [\zeta_1, \ldots, \zeta_r]^\top \in \mathbb{R}^r$ is a vector of unknown *system parameters*. Furthermore, $\boldsymbol{f} : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{nm} \times \mathbb{R}^r \to \mathbb{R}^n$ and $\boldsymbol{\phi} : \mathbb{R} \to \mathbb{R}^n$ are given functions.

The time-delays in (1)-(2) are unknown quantities that need to be determined. We assume that the ith time-delay belongs to the interval $[a_i, b_i]$, where a_i and b_i are given constants such that $0 \leq a_i < b_i$. Hence, the unknown time-delays satisfy the following bound constraints:

$$a_i \leq \tau_i \leq b_i, \quad i = 1, \ldots, m. \tag{3}$$

Any vector $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_m]^\top \in \mathbb{R}^m$ that satisfies (3) is called a *candidate time-delay vector*. Let $\mathcal{T}$ denote the set of all such candidate time-delay vectors.

In addition to the time-delays, the system parameters in (1)-(2) are also unknown quantities that need to be determined. We suppose that

$$c_j \leq \zeta_j \leq d_j, \quad j = 1, \ldots, r, \tag{4}$$

where c_j and d_j are given real numbers such that $0 \leq c_j < d_j$. Note that there is no loss of generality in assuming that $c_j \geq 0$; if $c_j < 0$, then we may replace ζ_j with $\zeta_j + c_j$. Any vector $\boldsymbol{\zeta} = [\zeta_1, \ldots, \zeta_r]^\top \in \mathbb{R}^r$ that satisfies (4) is called a *candidate parameter vector*. Let $\mathcal{Z}$ denote the set of all such candidate parameter vectors.

The output of system (1)-(2) is given by

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{\zeta}), \quad t \in [0, T], \tag{5}$$

where $\boldsymbol{g} : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^p$ is a given function.

We assume that the following conditions are satisfied.

Assumption 1. *The given functions $\boldsymbol{f}$ and $\boldsymbol{g}$ are continuously differentiable, and $\boldsymbol{\phi}$ is twice continuously differentiable.*

Assumption 2. *There exists a real number $L_1 > 0$ such that*

$$|\boldsymbol{f}(t, \boldsymbol{x}, \tilde{\boldsymbol{x}}, \boldsymbol{\zeta})| \leq L_1(1 + |\boldsymbol{x}| + |\tilde{\boldsymbol{x}}| + |\boldsymbol{\zeta}|), \quad (t, \boldsymbol{x}, \tilde{\boldsymbol{x}}, \boldsymbol{\zeta}) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{nm} \times \mathbb{R}^r,$$

where $| \cdot |$ denotes the Euclidean norm.

On the basis of Assumptions 1 and 2, the dynamic system (1)-(2) admits a unique solution corresponding to each pair $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$ [1]. We denote this solution by $\boldsymbol{x}(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$. Substituting $\boldsymbol{x}(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$ into (5) gives $\boldsymbol{y}(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$, the predicted system output corresponding to $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$. More formally,

$$\boldsymbol{y}(t|\boldsymbol{\tau}, \boldsymbol{\zeta}) = \boldsymbol{g}(\boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta}), \boldsymbol{\zeta}), \quad t \in [0, T]. \tag{6}$$

Suppose that the output from system (1)-(2) has been measured experimentally at times $t = t_l$, $l = 1, \ldots, q$, where each $t_l \in [0, T]$. Let $\hat{\boldsymbol{y}}^l \in \mathbb{R}^p$ denote the measured output at time $t = t_l$. Then the problem of identifying the unknown time-delays and system parameters can be formulated mathematically as follows.

Problem (P). *Choose $\boldsymbol{\tau} \in \mathcal{T}$ and $\boldsymbol{\zeta} \in \mathcal{Z}$ to minimize the following cost function:*

$$J(\boldsymbol{\tau}, \boldsymbol{\zeta}) = \sum_{l=1}^{q} \left| \boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l \right|^2. \tag{7}$$

Problem (P) is a nonlinear dynamic optimization problem whose decision variables are the delays and model parameters in system (1)-(2). Our aim is to select optimal values for these delays and parameters so that the predicted system output best fits the experimental data. Almost all of the existing optimization techniques for time-delay systems are based on the assumption that the delays are fixed and known (see, for example, [3, 12, 25]). Problem (P) is unique in that the delays are not fixed, but are instead decision variables to be chosen optimally. The cost function in Problem (P) is also highly non-standard, as it depends on the system's state at a set of discrete time points, not just at the terminal time. Such cost functions have been considered in [13, 17] for non-delay systems, and in [22] for systems with fixed delays. However, the computational techniques developed in references [13, 17, 22] are not applicable to Problem (P) because the time-delays in system (1)-(2) are variable.

3. **Gradient computation.** Problem (P) involves choosing a finite number of decision variables to minimize the cost function (7). Thus, in principle, Problem (P) can be viewed as a nonlinear programming problem. Standard algorithms for solving nonlinear programming problems—for example, sequential quadratic programming or interior-point methods [16]—typically require the gradient of the cost function, which is difficult to determine in Problem (P) because the delays and parameters influence (7) *implicitly* through the dynamic system (1)-(2). The aim of this section is to develop an efficient computational method for computing the gradient of the cost function in Problem (P). This method, which is inspired by our earlier work in

[10, 11, 14, 15], can be integrated with a standard nonlinear programming algorithm to solve Problem (P).

3.1. **Preliminaries.** Throughout this subsection, we assume that $k \in \{1, \ldots, m\}$ and $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$ are arbitrary but fixed. For simplicity, we write $\boldsymbol{x}(t)$ instead of $\boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta})$, and $\boldsymbol{x}^\epsilon(t)$ instead of $\boldsymbol{x}(t|\boldsymbol{\tau} + \epsilon \boldsymbol{e}^k, \boldsymbol{\zeta})$, where $\boldsymbol{e}^k$ denotes the kth unit basis vector in $\mathbb{R}^m$.

Define

$$I = [a_k - \tau_k, b_k - \tau_k].$$

Note that $I \neq \emptyset$ and $0 \in I$. Clearly,

$$\epsilon \in I \quad \Longleftrightarrow \quad \boldsymbol{\tau} + \epsilon \boldsymbol{e}^k \in \mathcal{T}.$$

For each $\epsilon \in I$, define

$$\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{x}^\epsilon(t) - \boldsymbol{x}(t), \quad t \leq T,$$

and

$$\boldsymbol{\theta}^{\epsilon,i}(t) = \boldsymbol{x}^\epsilon(t - \tau_i - \epsilon \delta_{ki}) - \boldsymbol{x}(t - \tau_i), \quad t \leq T, \quad i = 1, \ldots, m,$$

where δ_{ki} denotes the Kronecker delta function. Furthermore, let

$$\boldsymbol{\theta}^\epsilon(t) = [(\boldsymbol{\theta}^{\epsilon,1}(t))^\top, \ldots, (\boldsymbol{\theta}^{\epsilon,m}(t))^\top]^\top \in \mathbb{R}^{nm}, \quad t \leq T.$$

Clearly,

$$\boldsymbol{\theta}^{\epsilon,i}(t) = \boldsymbol{\varphi}^\epsilon(t - \tau_i), \quad t \leq T, \quad i \neq k, \tag{8}$$

$$\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{0}, \quad t \leq 0. \tag{9}$$

In the sequel, we will use the notation $\frac{\partial}{\partial \tilde{\boldsymbol{x}}^i}$ to denote differentiation with respect to the ith delayed state in $\tilde{\boldsymbol{x}}(t)$ (i.e. differentiation with respect to $\boldsymbol{x}(t - \tau_i)$).

Now, define

$$\boldsymbol{\chi}(t) = \begin{cases} \dot{\boldsymbol{\phi}}(t), & \text{if } t \leq 0, \\ \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta}), & \text{if } t \in (0, T]. \end{cases}$$

We immediately see that, for almost all $t \in (-\infty, T]$,

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{\chi}(t).$$

Let $\bar{b} > 0$ be a fixed constant such that $\bar{b} \geq b_i$ for each $i = 1, \ldots, m$. By following similar arguments to those used in [14], it is possible to show that there exists a positive real number $L_2 > 0$ such that for each $\epsilon \in I$,

$$|\boldsymbol{x}^\epsilon(t)|, |\boldsymbol{\chi}(t)| \leq L_2, \quad t \in [-\bar{b}, T], \tag{10}$$

and

$$|\boldsymbol{\varphi}^\epsilon(t)|, \max_{i=1,\ldots,m} |\boldsymbol{\theta}^{\epsilon,i}(t)| \leq L_2|\epsilon|, \quad t \in [0, T]. \tag{11}$$

Furthermore, one can also show that for almost all $t \in [0, T]$,

$$\lim_{\epsilon \to 0} \frac{\boldsymbol{\theta}^{\epsilon,k}(t) - \boldsymbol{\varphi}^\epsilon(t - \tau_k)}{\epsilon} = -\boldsymbol{\chi}(t - \tau_k). \tag{12}$$

See Appendix B of [14] for more details.

3.2. **State variation with respect to time-delays.** The solution of system (1)-(2) is normally viewed as a function of time, with $\boldsymbol{\tau}$ and $\boldsymbol{\zeta}$ being fixed vectors. Now, by instead fixing $t \in (-\infty, T]$, while allowing the vectors $\boldsymbol{\tau}$ and $\boldsymbol{\zeta}$ to vary, we obtain a new function $\boldsymbol{x}(t|\cdot,\cdot) : \mathcal{T} \times \mathcal{Z} \to \mathbb{R}^n$ whose value at $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$ is $\boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta})$. In the following result, we show that $\boldsymbol{x}(t|\cdot,\cdot)$ is differentiable with respect to the time-delays. This result is central to the development of a computational procedure for solving Problem (P).

Theorem 3.1. *Let $t \in (0, T]$ be a fixed time point. Then $\boldsymbol{x}(t|\cdot,\cdot)$ is differentiable with respect to τ_k on $\mathcal{T} \times \mathcal{Z}$. In fact, for each $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$,*

$$\frac{\partial \boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \tau_k} = \boldsymbol{\Lambda}^k(t|\boldsymbol{\tau}, \boldsymbol{\zeta}), \quad k = 1, \ldots, m, \tag{13}$$

where $\boldsymbol{\Lambda}^k(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$ satisfies the auxiliary time-delay system

$$\dot{\boldsymbol{\Lambda}}^k(t) = \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \boldsymbol{x}} \boldsymbol{\Lambda}^k(t) + \sum_{i=1}^m \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \tilde{\boldsymbol{x}}^i} \boldsymbol{\Lambda}^k(t - \tau_i)$$
$$- \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \tilde{\boldsymbol{x}}^k} \boldsymbol{\chi}(t - \tau_k) \tag{14}$$

with initial condition

$$\boldsymbol{\Lambda}^k(t) = \boldsymbol{0}, \quad t \le 0. \tag{15}$$

Proof. Let $k \in \{1, \ldots, m\}$ and $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$ be arbitrary but fixed. As in Subsection 3.1, we write $\boldsymbol{x}^\epsilon(t)$ instead of $\boldsymbol{x}(t|\boldsymbol{\tau}+\epsilon\boldsymbol{e}^k, \boldsymbol{\zeta})$, and $\boldsymbol{x}(t)$ instead of $\boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta})$.

For each $\epsilon \in I \setminus \{0\}$, define

$$\rho(\epsilon) = \int_0^T \left| \epsilon^{-1} \boldsymbol{\theta}^{\epsilon, k}(s) - \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(s - \tau_k) + \boldsymbol{\chi}(s - \tau_k) \right| ds. \tag{16}$$

It follows from (9), (10), and (11) that for each $\epsilon \in I \setminus \{0\}$,

$$\left| \epsilon^{-1} \boldsymbol{\theta}^{\epsilon, k}(s) - \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(s - \tau_k) + \boldsymbol{\chi}(s - \tau_k) \right| \le 3L_2, \quad s \in [0, T].$$

Hence, the integrand in (16) is uniformly bounded with respect to $\epsilon \in I \setminus \{0\}$. Furthermore, it follows from (12) that the integrand in (16) converges to zero almost everywhere on $[0, T]$ as $\epsilon \to 0$. Thus, from the Lebesgue dominated convergence theorem,

$$\lim_{\epsilon \to 0} \rho(\epsilon) = \lim_{\epsilon \to 0} \int_0^T \left| \epsilon^{-1} \boldsymbol{\theta}^{\epsilon, k}(s) - \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(s - \tau_k) + \boldsymbol{\chi}(s - \tau_k) \right| ds = 0.$$

Now, keeping $\epsilon \in I \setminus \{0\}$ fixed for the time being, we define

$$\bar{\boldsymbol{f}}(s, \alpha) = \boldsymbol{f}(s, \boldsymbol{x}(s) + \alpha \boldsymbol{\varphi}^\epsilon(s), \tilde{\boldsymbol{x}}(s) + \alpha \boldsymbol{\theta}^\epsilon(s), \boldsymbol{\zeta}), \quad (s, \alpha) \in [0, T] \times [0, 1].$$

Then by the chain rule,

$$\frac{\partial \bar{\boldsymbol{f}}(s, \alpha)}{\partial \alpha} = \frac{\partial \bar{\boldsymbol{f}}(s, \alpha)}{\partial \boldsymbol{x}} \boldsymbol{\varphi}^\epsilon(s) + \sum_{i=1}^m \frac{\partial \bar{\boldsymbol{f}}(s, \alpha)}{\partial \tilde{\boldsymbol{x}}^i} \boldsymbol{\theta}^{\epsilon, i}(s), \tag{17}$$

where

$$\frac{\partial \bar{\boldsymbol{f}}(s, \alpha)}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{f}(s, \boldsymbol{x}(s) + \alpha \boldsymbol{\varphi}^\epsilon(s), \tilde{\boldsymbol{x}}(s) + \alpha \boldsymbol{\theta}^\epsilon(s), \boldsymbol{\zeta})}{\partial \boldsymbol{x}}, \tag{18}$$

$$\frac{\partial \bar{\boldsymbol{f}}(s, \alpha)}{\partial \tilde{\boldsymbol{x}}^i} = \frac{\partial \boldsymbol{f}(s, \boldsymbol{x}(s) + \alpha \boldsymbol{\varphi}^\epsilon(s), \tilde{\boldsymbol{x}}(s) + \alpha \boldsymbol{\theta}^\epsilon(s), \boldsymbol{\zeta})}{\partial \tilde{\boldsymbol{x}}^i}. \tag{19}$$

We can rewrite (17) as follows:

$$\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \alpha} = \Delta_1(s,\alpha) + \Delta_2(s,\alpha) + \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\boldsymbol{\varphi}^\epsilon(s) + \sum_{i=1}^m \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\varphi}^\epsilon(s - \tau_i)$$

$$+ \sum_{i=1}^m \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\big\{\boldsymbol{\theta}^{\epsilon,i}(s) - \boldsymbol{\varphi}^\epsilon(s - \tau_i)\big\},$$

where

$$\Delta_1(s,\alpha) = \Big\{\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \boldsymbol{x}} - \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\Big\}\boldsymbol{\varphi}^\epsilon(s), \tag{20}$$

$$\Delta_2(s,\alpha) = \sum_{i=1}^m \Big\{\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \tilde{\boldsymbol{x}}^i} - \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\Big\}\boldsymbol{\theta}^{\epsilon,i}(s). \tag{21}$$

Applying (8) gives

$$\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \alpha} = \Delta_1(s,\alpha) + \Delta_2(s,\alpha) + \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\boldsymbol{\varphi}^\epsilon(s)$$

$$+ \sum_{i=1}^m \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\varphi}^\epsilon(s - \tau_i) + \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^k}\big\{\boldsymbol{\theta}^{\epsilon,k}(s) - \boldsymbol{\varphi}^\epsilon(s - \tau_k)\big\}. \tag{22}$$

Now,

$$\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{x}^\epsilon(t) - \boldsymbol{x}(t) = \int_0^t \big\{\bar{\boldsymbol{f}}(s,1) - \bar{\boldsymbol{f}}(s,0)\big\}ds.$$

Thus, by the fundamental theorem of calculus,

$$\boldsymbol{\varphi}^\epsilon(t) = \int_0^t \big\{\bar{\boldsymbol{f}}(s,1) - \bar{\boldsymbol{f}}(s,0)\big\}ds = \int_0^t \int_0^1 \frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \alpha}d\alpha ds. \tag{23}$$

Substituting (22) into (23) yields

$$\boldsymbol{\varphi}^\epsilon(t) = \int_0^t \int_0^1 \big\{\Delta_1(s,\alpha) + \Delta_2(s,\alpha)\big\}d\alpha ds + \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\boldsymbol{\varphi}^\epsilon(s)ds$$

$$+ \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^k}\big\{\boldsymbol{\theta}^{\epsilon,k}(s) - \boldsymbol{\varphi}^\epsilon(s - \tau_k)\big\}ds \tag{24}$$

$$+ \sum_{i=1}^m \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\varphi}^\epsilon(s - \tau_i)ds.$$

Now, by using (18) and (19), we can write the auxiliary system (14) as follows:

$$\dot{\boldsymbol{\Lambda}}^k(s) = \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\boldsymbol{\Lambda}^k(s) + \sum_{i=1}^m \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\Lambda}^k(s - \tau_i) - \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^k}\boldsymbol{\chi}(s - \tau_k).$$

Hence,

$$\boldsymbol{\Lambda}^k(t) = \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\boldsymbol{\Lambda}^k(s)ds + \sum_{i=1}^m \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\Lambda}^k(s - \tau_i)ds$$

$$- \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^k}\boldsymbol{\chi}(s - \tau_k)ds. \tag{25}$$

Now, since $\boldsymbol{f}$ is continuously differentiable (recall Assumption 1), and $\boldsymbol{x}$ is bounded on $[-\bar{b}, T]$ (recall inequality (10)), there exists constants $M_1 > 0$ and $M_2 > 0$ such

that

$$\left|\frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\right| \le M_1, \quad \left|\frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\right| \le M_2, \quad s \in [0,T],$$

where $|\cdot|$ denotes the natural matrix norm on $\mathbb{R}^{n \times n}$. Thus, from (24) and (25),

$$\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Lambda}^k(t)\right| \le M_2\rho(\epsilon) + \int_0^t M_1\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\right|ds$$
$$+ \sum_{i=1}^m \int_0^t M_2\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\tau_i) - \boldsymbol{\Lambda}^k(s-\tau_i)\right|ds \tag{26}$$
$$+ |\epsilon|^{-1}\int_0^t\int_0^1 \left\{|\Delta_1(s,\alpha)| + |\Delta_2(s,\alpha)|\right\}d\alpha ds,$$

where $\rho(\epsilon)$ is as defined in (16). The second integral term on the right-hand side of (26) can be simplified as follows:

$$\sum_{i=1}^m \int_0^t M_2\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\tau_i) - \boldsymbol{\Lambda}^k(s-\tau_i)\right|ds = \sum_{i=1}^m \int_{-\tau_i}^{t-\tau_i} M_2\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\right|ds$$
$$\le \sum_{i=1}^m \int_0^t M_2\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\right|ds$$
$$= \int_0^t mM_2\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\right|ds.$$

Hence, (26) becomes

$$\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Lambda}^k(t)\right| \le M_2\rho(\epsilon) + \int_0^t \bar{M}\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\right|ds$$
$$+ |\epsilon|^{-1}\int_0^t\int_0^1 \left\{|\Delta_1(s,\alpha)| + |\Delta_2(s,\alpha)|\right\}d\alpha ds, \tag{27}$$

where $\bar{M} = M_1 + mM_2$. Since $\boldsymbol{f}$ is continuously differentiable and $\boldsymbol{x}^\epsilon$ is uniformly bounded with respect to ϵ, $\frac{\partial \bar{\boldsymbol{f}}}{\partial \boldsymbol{x}}$ and $\frac{\partial \bar{\boldsymbol{f}}}{\partial \tilde{\boldsymbol{x}}^i}$ are uniformly continuous on $[0,T] \times [0,1]$. Furthermore, by (11), $\boldsymbol{x}(s) + \alpha\boldsymbol{\varphi}^\epsilon(s) \to \boldsymbol{x}(s)$ and $\tilde{\boldsymbol{x}}(s) + \alpha\boldsymbol{\theta}^\epsilon(s) \to \tilde{\boldsymbol{x}}(s)$ uniformly on $[0,T] \times [0,1]$ as $\epsilon \to 0$. Thus, for each $\delta > 0$, there exists an $\epsilon' > 0$ such that for all ϵ satisfying $|\epsilon| < \epsilon'$,

$$\left|\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \boldsymbol{x}} - \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\right| < \delta, \quad (s,\alpha) \in [0,T] \times [0,1],$$
$$\left|\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \tilde{\boldsymbol{x}}^i} - \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\right| < \delta, \quad (s,\alpha) \in [0,T] \times [0,1].$$

By taking the norm of (20) and (21), and then using the above inequalities together with (11), we obtain

$$|\Delta_1(s,\alpha)| \le \delta L_2|\epsilon|, \qquad |\Delta_2(s,\alpha)| \le \delta mL_2|\epsilon|, \tag{28}$$

where $|\epsilon| < \epsilon'$. Substituting these inequalities into (27) yields,

$$\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Lambda}^k(t)\right| \le M_2\rho(\epsilon) + (L_2T + mL_2T)\delta + \int_0^t \bar{M}\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\right|ds.$$

Now, recall that $\rho(\epsilon) \to 0$ as $\epsilon \to 0$. Hence, there exists an $\epsilon'' > 0$ such that $\rho(\epsilon) < \delta$ whenever $|\epsilon| < \epsilon''$. Thus, for all ϵ such that $|\epsilon| < \min\{\epsilon', \epsilon''\}$,

$$\left|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(t) - \boldsymbol{\Lambda}^{k}(t)\right| \leq M_2\delta + (L_2T + mL_2T)\delta + \int_0^t \bar{M}\left|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(s) - \boldsymbol{\Lambda}^{k}(s)\right|ds.$$

Applying the Gronwall-Bellman Lemma [1] gives

$$\left|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(t) - \boldsymbol{\Lambda}^{k}(t)\right| \leq \delta(M_2 + L_2T + mL_2T)\exp\{\bar{M}T\},$$

where $|\epsilon| < \min\{\epsilon', \epsilon''\}$. Since δ is arbitrary, this shows that $\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(t) \to \boldsymbol{\Lambda}^{k}(t)$ as $\epsilon \to 0$. It follows that

$$\frac{\partial \boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \tau_k} = \lim_{\epsilon \to 0} \frac{\boldsymbol{x}^{\epsilon}(t) - \boldsymbol{x}(t)}{\epsilon} = \lim_{\epsilon \to 0} \epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(t) = \boldsymbol{\Lambda}^{k}(t),$$

as required.   □

In Theorem 3.1, we derived formulae for the gradient of the state with respect to the time-delays. In the next subsection, we turn our attention to the gradient of the state with respect to the system parameters.

3.3. **State variation with respect to system parameters.** Let w be a new state variable with dynamics

$$\dot{w}(t) = 1, \quad t \in [0, T], \tag{29}$$

$$w(t) = t, \quad t \leq 0. \tag{30}$$

Clearly, $w(t) = t$ for all $t \in (-\infty, T]$. Thus, we can express the system parameters in (1)-(2) in terms of the new state variable w as follows:

$$\zeta_j = t - w(t - \zeta_j), \quad j = 1, \ldots, r. \tag{31}$$

Substituting (31) into the original system (1)-(2) gives

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), t - w(t - \zeta_1), \ldots, t - w(t - \zeta_r)), \quad t \in [0, T], \tag{32}$$

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \leq 0. \tag{33}$$

The system parameters ζ_j, $j = 1, \ldots, r$, are now time-delays in the enlarged system consisting of (29)-(30) and (32)-(33). Thus, to determine the state variation with respect to the system parameters in system (1)-(2), we just need to apply Theorem 3.1 to the enlarged system consisting of (29)-(30) and (32)-(33). It is important to note that each system parameter is bounded below by zero (see the problem formulation in Section 2). Thus, the enlarged system considered here is a valid time-delay system with all time-delays non-negative.

Let $\boldsymbol{z}(t) \in \mathbb{R}^{n+1}$ and $\tilde{\boldsymbol{z}}(t) \in \mathbb{R}^{(n+1)(m+r)}$ denote, respectively, the state and delayed state vectors for the enlarged system, where

$$\boldsymbol{z}(t) = \left[x_1(t), \ldots, x_n(t), w(t)\right]^{\top}$$

and

$$\tilde{\boldsymbol{z}}(t) = \left[\boldsymbol{z}(t - \tau_1), \ldots, \boldsymbol{z}(t - \tau_m), \boldsymbol{z}(t - \zeta_1), \ldots, \boldsymbol{z}(t - \zeta_r)\right]^{\top}.$$

The enlarged system consisting of (29)-(30) and (32)-(33) can be written as follows:

$$\dot{\boldsymbol{z}}(t) = \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t)), \quad t \in [0, T], \tag{34}$$

$$\boldsymbol{z}(t) = \hat{\boldsymbol{\phi}}(t), \quad t \leq 0, \tag{35}$$

where

$$\hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t)) = \begin{bmatrix} \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), t - w(t - \zeta_1), \ldots, t - w(t - \zeta_r)) \\ 1 \end{bmatrix}$$

and

$$\hat{\boldsymbol{\phi}}(t) = \begin{bmatrix} \boldsymbol{\phi}(t) \\ t \end{bmatrix}.$$

Define

$$\hat{\boldsymbol{\chi}}(t) = \begin{cases} \dot{\hat{\boldsymbol{\phi}}}(t), & \text{if } t \leq 0, \\ \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t)), & \text{if } t \in (0, T]. \end{cases}$$

Let $j \in \{1, \ldots, r\}$ and $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$. Then the auxiliary system for (34)-(35) corresponding to the system parameter ζ_j is

$$\begin{bmatrix} \dot{\boldsymbol{\Gamma}}^j(t) \\ \dot{\gamma}^j(t) \end{bmatrix} = \frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \boldsymbol{z}} \begin{bmatrix} \boldsymbol{\Gamma}^j(t) \\ \gamma^j(t) \end{bmatrix} + \sum_{i=1}^{m} \frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \tilde{\boldsymbol{z}}^i} \begin{bmatrix} \boldsymbol{\Gamma}^j(t - \tau_i) \\ \gamma^j(t - \tau_i) \end{bmatrix}$$

$$+ \sum_{i=1}^{r} \frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \tilde{\boldsymbol{z}}^{m+i}} \begin{bmatrix} \boldsymbol{\Gamma}^j(t - \zeta_i) \\ \gamma^j(t - \zeta_i) \end{bmatrix} - \frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \tilde{\boldsymbol{z}}^{m+j}} \hat{\boldsymbol{\chi}}(t - \zeta_j) \tag{36}$$

with the initial conditions

$$\begin{bmatrix} \boldsymbol{\Gamma}^j(t) \\ \gamma^j(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ 0 \end{bmatrix}, \quad t \leq 0. \tag{37}$$

Here, $\boldsymbol{\Gamma}^j(t) : (-\infty, T] \to \mathbb{R}^n$ is the variation of the original state $\boldsymbol{x}$ with respect to ζ_j and $\gamma^j(t) : (-\infty, T] \to \mathbb{R}^n$ is the variation of the new state w with respect to ζ_j. Note that

$$\frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \boldsymbol{z}} = \begin{bmatrix} \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \boldsymbol{x}} & \boldsymbol{0} \\ \boldsymbol{0} & 0 \end{bmatrix},$$

$$\frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \tilde{\boldsymbol{z}}^i} = \begin{bmatrix} \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \tilde{\boldsymbol{x}}^i} & \boldsymbol{0} \\ \boldsymbol{0} & 0 \end{bmatrix}, \quad i = 1, \ldots, m,$$

$$\frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \tilde{\boldsymbol{z}}^{m+i}} = \begin{bmatrix} \boldsymbol{0} & -\frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \zeta_i} \\ \boldsymbol{0} & 0 \end{bmatrix}, \quad i = 1, \ldots, r.$$

Furthermore, it is clear that $\gamma^j(t) = 0$ for all $t \in (-\infty, T]$. Thus, the auxiliary system (36)-(37) becomes

$$\dot{\boldsymbol{\Gamma}}^j(t) = \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}^j(t) + \sum_{i=1}^{m} \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \tilde{\boldsymbol{x}}^i} \boldsymbol{\Gamma}^j(t - \tau_i)$$

$$+ \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \zeta_j} \tag{38}$$

with initial conditions

$$\boldsymbol{\Gamma}^j(t) = \boldsymbol{0}, \quad t \leq 0. \tag{39}$$

Applying Theorem 3.1 to the enlarged system consisting of (29)-(30) and (32)-(33) yields the following result.

Theorem 3.2. *Let $t \in (0, T]$ be a fixed time point. Then $\boldsymbol{x}(t|\cdot, \cdot)$ is differentiable with respect to ζ_j on $\mathcal{T} \times \mathcal{Z}$. In fact, for each $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$,*

$$\frac{\partial \boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \zeta_j} = \boldsymbol{\Gamma}^j(t|\boldsymbol{\tau}, \boldsymbol{\zeta}), \quad j = 1, \ldots, r,$$

where $\boldsymbol{\Gamma}^j(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$ satisfies the auxiliary time-delay system (38)-(39).

We are now ready to derive formulae for the gradient of the cost function in Problem (P). By using Theorems 3.1 and 3.2 and the chain rule of differentiation, we obtain

$$\frac{\partial J(\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \tau_k} = 2 \sum_{l=1}^{q} \left[\boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l \right]^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}), \boldsymbol{\zeta})}{\partial \boldsymbol{x}} \boldsymbol{\Lambda}^k(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}), \tag{40}$$

and

$$\begin{aligned}
\frac{\partial J(\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \zeta_j} &= 2 \sum_{l=1}^{q} \left[\boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l \right]^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}), \boldsymbol{\zeta})}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}^j(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}) \\
&\quad + 2 \sum_{l=1}^{q} \left[\boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l \right]^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}), \boldsymbol{\zeta})}{\partial \zeta_j}.
\end{aligned} \tag{41}$$

We now present the following algorithm for computing the cost function (7) and its gradient at a given pair $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$.

Step 1. Obtain $\boldsymbol{x}(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$, $\boldsymbol{\Lambda}^k(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$, $k = 1, \ldots, m$, and $\boldsymbol{\Gamma}^j(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$, $j = 1, \ldots, r$, by solving the enlarged time-delay system consisting of the original system (1)-(2) and the auxiliary systems (14)-(15) and (38)-(39).

Step 2. Use the state values $\boldsymbol{x}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta})$, $l = 1, \ldots, q$, to compute $\boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta})$ through equation (6).

Step 3. Use $\boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta})$, $l = 1, \ldots, q$, to compute $J(\boldsymbol{\tau}, \boldsymbol{\zeta})$ through equation (7).

Step 4. Use $\boldsymbol{x}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta})$, $\boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta})$, $\boldsymbol{\Lambda}^k(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta})$, and $\boldsymbol{\Gamma}^j(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta})$ to compute $\frac{\partial J(\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \tau_k}$, $k = 1, \ldots, m$, and $\frac{\partial J(\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \zeta_j}$, $j = 1, \ldots, r$, through equations (40) and (41).

This gradient computation algorithm can be integrated with a standard gradient-based optimization method (e.g. sequential quadratic programming (SQP)) to solve Problem (P) as a nonlinear programming problem. Note that gradient-based optimization methods such as SQP are only guaranteed to converge to a locally optimal solution. Thus, when applied to Problem (P), these methods could potentially generate parameter estimates that are not globally optimal. However, this is unlikely for two reasons. First, in a practical identification problem, we usually have reasonably tight bounds and good initial approximations for the unknown delays and parameters. Second, the number of unknown delays and parameters is usually small (models with more than 2 or 3 delays are rare). Thus, global optimization is usually not a major issue for identification problems.

In some applications, the governing dynamic system includes input-delays as well as state-delays. For example, consider the following system:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{u}(t), \tilde{\boldsymbol{u}}(t), \boldsymbol{\zeta}), \quad t \in [0, T], \tag{42}$$

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \leq 0, \tag{43}$$

where $\boldsymbol{u}(t) = [u_1(t), \ldots, u_v(t)]^\top \in \mathbb{R}^v$ is the *control input vector* for the system; $\tilde{\boldsymbol{u}}(t) = [\boldsymbol{u}(t - \lambda_1)^\top, \ldots, \boldsymbol{u}(t - \lambda_d)^\top]^\top \in \mathbb{R}^{vd}$ is the *delayed control vector*; and

λ_i, $i = 1, \ldots, d$, are unknown *control delays*. The other symbols are as defined in Section 2.

In (42)-(43), $\tilde{\boldsymbol{u}}$ is assumed to be a known input function. Thus, we can write (42)-(43) in the form of (1)-(2) as follows:

$$\dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{f}}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta}, \boldsymbol{\lambda}), \quad t \in [0, T], \tag{44}$$

where $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_d]^\top$ is a parameter vector containing the control delays. If the input function $\boldsymbol{u}$ is continuously differentiable, then $\bar{\boldsymbol{f}}$ is also continuously differentiable, and thus the approach outlined above for solving Problem (P) is applicable. Hence, our identification method can also be applied to systems with input delay (assuming that the input is smooth).

4. Numerical examples.

4.1. **Example 1.** We now apply the solution method developed in Section 3 to the industrial purification process described in [22, 23]. The purpose of this process is to remove harmful cobalt and cadmium ions from a zinc sulphate electrolyte by adding zinc powder to encourage deposition. This is a key step in the production of zinc.

The changes in concentrations of cobalt and cadmium ions in the electrolyte are described by the following differential equations:

$$V\dot{x}_1(t) = Qx_1^0 - Qx_1(t - \tau) - \alpha u(t)x_1(t - \tau) + cx_2(t - \tau), \tag{45}$$

$$V\dot{x}_2(t) = Qx_2^0 - Qx_2(t - \tau) - \beta v(t)x_2(t - \tau) + dx_1(t - \tau), \tag{46}$$

and

$$x_1(t) = 3.3 \times 10^{-4}, \quad x_2(t) = 4.0 \times 10^{-3} \quad t \le 0, \tag{47}$$

where x_1 is the concentration of cobalt ions; x_2 is the concentration of cadmium ions; and u and v are control variables representing the zinc powder reaction surface areas (which correspond to the level of zinc powder added to the reaction tank).

Furthermore, V is the volume of the reaction tank ($V = 400$); Q is the flux of solution ($Q = 200$); α, β, c, d, are model parameters; and x_1^0 and x_2^0 are the concentrations of cobalt and cadmium ions at the inlet of the reaction tank, respectively ($x_1^0 = 6 \times 10^{-4}$, $x_2^0 = 9 \times 10^{-3}$). Reference [23] considers the parameter identification problem for system (45)-(47) with a given time-delay of $\tau = 2$. Here, we consider the problem of identifying the time-delay. We assume that β, c, and d are equal to the optimal values reported in [23]:

$$\beta = 2.823 \times 10^{-4}, \quad c = 16.67, \quad d = 7.107 \times 10^2. \tag{48}$$

These values were obtained using data from a real zinc production factory in China. We assume that the terminal time is $T = 8$. We set the input variables u and v as equal to the optimal control functions obtained in [23]:

$$u(t) = \sum_{k=1}^{8} \sigma^k \psi_{[\bar{t}_{k-1}, \bar{t}_k)}(t), \quad t \in [0, 8], \tag{49}$$

$$v(t) = \sum_{k=1}^{8} \bar{\sigma}^k \psi_{[\bar{t}_{k-1}, \bar{t}_k)}(t), \quad t \in [0, 8], \tag{50}$$

TABLE 1. Control values and switching times for Example 1.

k	1	2	3	4	5	6	7	8
$\bar{t}_k$	1	2	3	4	5	6	7	8
$\sigma_k \times 10^{-5}$	1.08	1.57	1.24	1.56	1.59	1.43	1.25	1.25
$\bar{\sigma}_k \times 10^{-5}$	5.20	4.70	4.97	4.60	4.53	4.64	4.74	4.62

where the values of $\bar{t}_k$, σ^k, and $\bar{\sigma}^k$, $k = 1, \ldots, 8$, are listed in Table 1, and

$$\psi_{[\bar{t}_{k-1}, \bar{t}_k)}(t) = \begin{cases} 1, & \text{if } t \in [\bar{t}_{k-1}, \bar{t}_k), \\ 0, & \text{otherwise.} \end{cases}$$

The output of the system is the concentration of cadmium ions:

$$y(t) = x_2(t). \tag{51}$$

Given system (45)-(47) and (51), with data (48)-(50), our goal is to identify the model parameter α and the delay τ.

We simulate system (45)-(47) with $\tau = \hat{\tau} = 2$ and $\alpha = \hat{\alpha} = 7.828 \times 10^{-4}$ to generate the observed data in Problem (P). The sample times are $t_l = l/2$, $l = 1, \ldots, 16$, and the observed output is

$$\hat{y}^l = x_2(t_l | \hat{\tau}, \hat{\alpha}).$$

Our identification problem is: choose τ and α to minimize

$$J(\tau, \alpha) = \sum_{l=1}^{16} \left| y(t_l | \tau, \alpha) - \hat{y}^l \right|^2 = \sum_{l=1}^{16} \left| x_2(t_l | \tau, \alpha) - x_2(t_l | \hat{\tau}, \hat{\alpha}) \right|^2$$

subject to the dynamic system (45)-(47).

Note that this problem cannot be solved using the identification method in [14], as the third term on the right-hand side of (45) contains both an unknown parameter and an unknown delay. The method in [14] is only applicable when each nonlinear term contains a single delay and no unknown parameters. We instead solve this problem using a Matlab program that integrates the SQP optimization method with the gradient computation algorithm described in Section 3.

Computational results for different initial guesses are shown in Table 2. The convergence of the output trajectory for the initial guess $\tau = 3$ and $\alpha = 0$ is displayed in Figure 1. This figure shows the output trajectory at intermediate iterations of the optimization process, as well as the final (converged) trajectory. In Table 2 and Figure 1, τ^i and α^i are the values of τ and α at the ith iteration of the optimization process ($i = 0$ refers to the initial guess). We see from Table 2 and Figure 1 that the optimal trajectory converges quickly to the observed data, regardless of the initial guess. Thus, the algorithm easily recovers the true values of the delay and parameter for this problem.

For comparison, we also solve this problem using the genetic algorithm (GA) in [18]. The parameters of GA are: the size of population is 20, the crossover probability is 0.8, the selection rate is 0.9, the mutation probability is 0.01, the number of bits for each individual is 14, and the maximum number of iterations is 1000. It takes about 40 minutes for GA to solve this problem, which is more than 20 times longer than the computation time taken by our method. Moreover, the cost value obtained by GA is 1.3787×10^{-9} with corresponding parameter estimates

TABLE 2. Numerical convergence of the cost values in Example 1.

| No. | Initial guess | | Cost value at ith iteration | | | |
	τ^0	α^0	$i = 0$	$i = 5$	$i = 10$	$i = 70$
1	0.5	0.5	9.111×10^{33}	5.392×10^{-6}	5.157×10^{-9}	7.751×10^{-15}
2	1.0	1.0	4.558×10^{20}	5.106×10^{-6}	7.709×10^{-10}	1.088×10^{-13}
3	1.5	0.5	3.346×10^{10}	1.722×10^{-6}	1.496×10^{-6}	1.700×10^{-13}
4	3.0	0.0	7.094×10^{-5}	2.536×10^{-5}	2.209×10^{-5}	3.341×10^{-14}
5	3.0	1.0	8.533×10^{3}	2.589×10^{-5}	2.180×10^{-5}	2.050×10^{-14}

$\tau = 2.0026$ and $\alpha = 7.9351 \times 10^{-4}$. Clearly, the results obtained by our new method are better than those from GA. This is not surprising, as our method exploits the gradient of the cost function to achieve fast convergence.
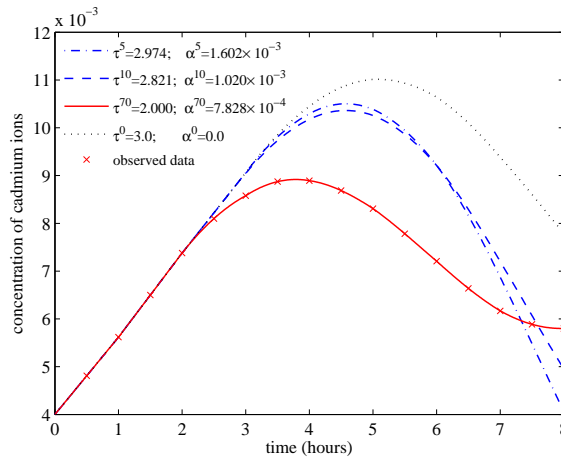

FIGURE 1. Numerical convergence of the output trajectory in Example 1 for initial guess No.4.

4.2. **Example 2.** We now demonstrate the applicability of our approach to systems with multiple delays. Consider the dynamic system given below:

$$\dot{x}_1(t) = -2x_1(t) + 0.1(1 - x_1(t - \tau_1)) \exp\left\{\frac{20x_2(t)}{20 + x_2(t)}\right\} \\ + 0.1x_1(t - \tau_1)x_2(t - \tau_2) + u(t - \tau_3), \tag{52}$$

$$\dot{x}_2(t) = -2.5x_2(t) + 0.8(1 - x_1(t - \tau_1)) \exp\left\{\frac{20x_2(t)}{20 + x_2(t)}\right\} \\ + 0.1x_2(t - \tau_1)x_2(t - \tau_2) + u(t - \tau_3), \tag{53}$$

with initial condition

$$x_1(t) = 1, \quad x_2(t) = 1, \quad t \leq 0. \tag{54}$$

Here, τ_1 and τ_2 are unknown state-delays, and τ_3 is an unknown input delay. Assume that the terminal time of this system is $T = 10$. The input function is defined as

follows:

$$u(t) = 0.1\sin(t), \quad t \le 10.$$

Furthermore, the output is

$$y(t) = x_2(t), \quad t \le 10.$$

We use the output trajectory of (52)-(54) with $[\hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_3] = [2.4, 1.8, 1.1]^\top$ to generate the observed data for Problem (P). We set

$$\hat{y}^l = x_2(t_l|\hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_3), \quad l = 1, \ldots, 20,$$

where $t_l = l/2$. Thus, our identification problem is: choose τ_1, τ_2, and τ_3 to minimize

$$J(\boldsymbol{\tau}) = \sum_{l=1}^{20} \left| y(t_l|\tau_1, \tau_2, \tau_3) - \hat{y}^l \right|^2 = \sum_{l=1}^{20} \left| x_2(t_l|\tau_1, \tau_2, \tau_3) - x_2(t_l|\hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_3) \right|^2$$

subject to the dynamics (52)-(54).

    We solved this problem using the same Matlab program that was used to solve Example 1. The convergence progress of our program is shown in Table 3 for four sets of initial guesses. The convergence of the output trajectory for the initial guess $\boldsymbol{\tau}^0 = [3.0, 3.0, 3.0]^\top$ is shown in Figure 2. In Table 3 and Figure 2, $\boldsymbol{\tau}^i = [\tau_1^i, \tau_2^i, \tau_3^i]^\top$ is the value of $\boldsymbol{\tau}$ at the ith iteration, while $i = 0$ refers to the initial guess. We also solve this problem using GA with the same parameters as in Example 1. The optimal cost obtained by GA is 1.3×10^{-4}. Moreover, the computation time is much longer than our new method. As with Example 1, we see that the optimization process converges from all initial guesses to the optimal solution.

TABLE 3. Numerical convergence of the cost values in Example 2.

No.	Initial guess			Cost value at ith iteration			
	τ_1^0	τ_2^0	τ_3^0	$i = 0$	$i = 5$	$i = 10$	$i = 30$
1	0.5	0.5	0.5	0.4922	0.0188	$5.667{\times}10^{-3}$	$6.661{\times}10^{-15}$
2	1.5	1.5	1.5	0.1386	0.0035	$3.357{\times}10^{-6}$	$6.618{\times}10^{-15}$
3	2.5	2.5	2.5	0.0747	0.0083	$4.405{\times}10^{-4}$	$1.534{\times}10^{-14}$
4	3.0	3.0	3.0	0.1710	0.0298	$2.780{\times}10^{-3}$	$6.656{\times}10^{-15}$

5. **Conclusion.** In this paper, we have developed a gradient-based computational method for determining unknown time-delays and unknown parameters in a general nonlinear system. This method is unified in the sense that the delays and parameters are determined simultaneously by solving a dynamic optimization problem. The gradient of the cost function in this problem is obtained by solving a set of auxiliary delay-differential systems from $t = 0$ to $t = T$. The numerical simulations in Section 4 demonstrate that this approach is highly effective. In particular, it converges quickly even when the initial estimates for the delays and parameters are far away from the optimal values. Our future research will involve extending this technique to adaptively identify delays and parameters as further information about the system becomes available.
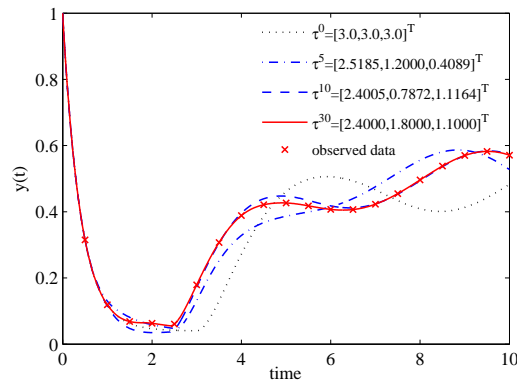
FIGURE 2. Numerical convergence of the output trajectory in Example 2 for initial guess No.4.

REFERENCES

[1] N. U. Ahmed, "Dynamic Systems and Control with Applications," World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2006.

[2] L. Belkoura, J.-P. Richard and M. Fliess, *Parameters estimation of systems with delayed and structured entries*, Automatica J. IFAC, **45** (2009), 1117–1125.

[3] Q. Q. Chai, C. H. Yang, K. L. Teo and W. H. Gui, *Time-delayed optimal control of an industrial-scale evaporation process sodium aluminate solution*, Control Engineering Practice, **20** (2012), 618–628.

[4] R. Datko, *Two examples of ill-posedness with respect to time delays revisited*, IEEE Transactions on Automatic Control, **42** (1997), 511–515.

[5] L. Denis-Vidal, C. Jauberthie and G. Joly-Blanchard, *Identifiability of a nonlinear delayed-differential aerospace model*, IEEE Transactions on Automatic Control, **51** (2006), 154–158.

[6] S. Diop, I. Kolmanovsky, P. E. Moraal and M. V. Nieuwstadt, *Preserving stability/performance when facing an unknown time-delay*, Control Engineering Practice, **9** (2001), 1319–1325.

[7] S. V. Drakunov, W. Perruquetti, J. P. Richard and L. Belkoura, *Delay identification in time-delay systems using variable structure observers*, Annual Reviews in Control, **30** (2006), 143–158.

[8] P. J. Gawthrop and M. T. Nihtilä, *Identification of time delays using a polynomial identification method*, Systems and Control Letters, **5** (1985), 267–271.

[9] L. Göllmann, D. Kern and H. Maurer, *Optimal control problems with delays in state and control variables subject to mixed control-state constraints*, Optimal Control Applications and Methods, **30** (2009), 341–365.

[10] Q. Lin, R. Loxton, K. L. Teo and Y. H. Wu, *A new computational method for optimizing nonlinear impulsive systems*, Dynamics of Continuous, Discrete and Impulsive Systems Series B: Applications and Algorithms, **18** (2011), 59–76.

[11] Q. Lin, R. Loxton, K. L. Teo and Y. H. Wu, *A new computational method for a class of free terminal time optimal control problems*, Pacific Journal of Optimization, **7** (2011), 63–81.

[12] X. Liu, *Constrained control of positive systems with delays*, IEEE Transactions on Automatic Control, **54** (2009), 1596–1600.

[13] R. C. Loxton, K. L. Teo and V. Rehbock, *Optimal control problems with multiple characteristic time points in the objective and constraints*, Automatica, **44** (2008), 2923–2929.

[14] R. Loxton, K. L. Teo and V. Rehbock, *An optimization approach to state-delay identification*, IEEE Transactions on Automatic Control, **55** (2010), 2113–2119.

[15] R. Loxton, K. L. Teo and V. Rehbock, *Robust suboptimal control of nonlinear systems*, Applied Mathematics and Computation, **217** (2011), 6566–6576.

[16] D. G. Luenberger and Y. Ye, "Linear and Nonlinear Programming," 3rd edition, International Series in Operations Research & Management Science, **116**, Springer, New York, 2008.

[17] R. B. Martin, *Optimal control drug scheduling of cancer chemotherapy*, Automatica J. IFAC, **28** (1992), 1113–1123.

[18] F. Pan, R. C. Han and D. M. Feng, "An identification method of time-varying delay based on genetic algorithm," in Proceedings of the 2003 International Conference on Machine Learning and Cybernetics, Xi'an, China, (2003), 781–783.

[19] C. Pignotti, *A note on stabilization of locally damped wave equations with time delay*, Systems and Control Letters, **61** (2012), 92–97.

[20] J.-P. Richard, *Time-delay systems: An overview of some recent advances and open problems*, Automatica J. IFAC, **39** (2003), 1667–1694.

[21] R. F. Stengel, R. Ghigliazza, N. Kulkarni and O. Laplace, *Optimal control of innate immune response*, Optimal Control Applications and Methods, **23** (2002), 91–104.

[22] L. Wang, W. Gui, K. L. Teo, R. Loxton and C. Yang, *Time delayed optimal control problems with multiple characteristic time points: Computation and industrial applications*, Journal of Industrial and Management Optimization, **5** (2009), 705–718.

[23] L. Y. Wang, W. H. Gui, K. L. Teo, R. Loxton and C. H. Yang, *Optimal control problems arising in the zinc sulphate electrolyte purification process*, Journal of Global Optimization, **54** (2012), 307–323.

[24] F. Y. Wang and Q. Yu, *Optimal protein separations with time lags in control functions*, Journal of Process Control, **4** (1994), 135–142.

[25] K. H. Wong, L. S. Jennings and F. Benyah, *The control parametrization enhancing transform for constrained time-delayed optimal control problems*, ANZIAM Journal, **43** (2001/02), E154–E185.

[26] L. Zunino, M. C. Soriano, I. Fischer, O. A. Rosso and C. R. Mirasso, *Permutation-information-theory approach to unveil delay dynamics from time-series analysis*, Physical Review E, **82** (2010), 046212, 9 pp.

E-mail address: kppqing@163.com
E-mail address: r.loxton@curtin.edu.au
E-mail address: k.l.teo@curtin.edu.au
E-mail address: ychh@csu.edu.cn