

Copyright © 2014 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

State-Delay Estimation for Nonlinear Systems Using Inexact Output Data

LIN Qun¹, Ryan LOXTON^{1,2}, XU Chao², Kok Lay TEO¹

1. Department of Mathematics and Statistics, Curtin University, Perth, Australia
E-mail: q.lin@curtin.edu.au; r.loxton@curtin.edu.au; k.l.teo@curtin.edu.au
2. Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou, China
E-mail: rcloxtan@gmail.com; cxu@zju.edu.cn

Abstract: This paper considers the problem of using inexact output data to estimate the values of unknown state-delays in a general nonlinear time-delay system. We formulate the problem as a nonlinear optimization problem in which the state-delays are decision parameters and the cost function penalizes a weighted sum of the mean and variance of the least-squares error between actual and predicted system output. Our main result shows that the gradient of the least-squares cost function can be computed by solving an auxiliary time-advance system backward in time. On this basis, the state-delay estimation problem can be solved efficiently using standard gradient-based optimization algorithms such as sequential quadratic programming. We conclude the paper by testing this approach on a dynamic model of a continuously-stirred tank reactor with recycle loop.

Key Words: Parameter estimation, Time-delay system, Nonlinear system, Gradient-based optimization

1 Introduction

Time-delay systems arise in a multitude of real-world applications, including purification processes [1], irrigation channels [2], aerospace engineering [3], and chromatography processes [4]. Control techniques for such systems often depend heavily on accurate knowledge of the time-delays. Hence, time-delay estimation is one of the key issues in the study of time-delay systems [5].

The time-delay estimation problem can be formulated as an *optimization problem* in which the time-delays are decision variables chosen to minimize a cost function that penalizes the deviation between the real system output (measured during experiments) and the predicted system output (generated by the mathematical model). Popular approaches for solving this problem include methods based on finite-dimensional approximations of the infinite-dimensional time-delay model [6], or swarm intelligence algorithms such as particle swarm optimization [7, 8]. Recently, a new gradient-based optimization approach, which combines nonlinear programming techniques with novel algorithms for computing the cost function's gradient, was proposed in references [9–11]. This approach was pioneered in [11] for systems in which each nonlinear term contains a single delay, and then extended in [9] to general nonlinear time-delay systems. In [10], this approach was applied to a more difficult time-delay estimation problem in which the dynamic system contains both state- and input-delays, and the input function is discontinuous.

The gradient-based optimization methods in [9–11] usually converge quickly to the optimal time-delay estimates, even for highly nonlinear systems. However, these methods are based on the assumption that the output data used in the cost function is *exact* (recall that the cost function penalizes the discrepancy between predicted and measured system output). Of course, this assumption is often violated in prac-

tice, as it is impossible to guarantee perfect precision when measuring the output of a real system.

In this paper, we address this limitation by developing a new gradient-based optimization method for time-delay estimation that explicitly caters for uncertainties in the system output data. We consider a general time-delay dynamic model consisting of a set of coupled delay-differential equations with state-delays (i.e., time-delays in the state variables). These state-delays are unknown and must be “tuned” by comparing the system output predicted by the model with the real system output. To allow for possible measurement errors in the system output data, we view the output data points as random variables, rather than fixed constants. Thus, the actual and measured system output could differ due to measurement inaccuracies. With the output data points as random variables, our state-delay estimation problem is formulated as a *stochastic optimization problem* in which the cost function penalizes both the expectation and the variance of the least-squares error between actual and predicted system output. We develop a computational algorithm for solving this problem and then examine its performance on a dynamic model of a continuously-stirred tank reactor with recycle loop. The results show that our approach can successfully determine optimal state-delay estimates in the presence of inexact output data.

2 Problem Statement

Consider the following nonlinear system with m state-delays:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{x}(t - \tau_1), \dots, \mathbf{x}(t - \tau_m)), t > 0, \quad (1)$$

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

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the *state vector*; $\tau_i, i = 1, \dots, m$, are the *state-delays*; and $\mathbf{f} : \mathbb{R}^{(m+1)n} \rightarrow \mathbb{R}^n$ and $\boldsymbol{\phi} : \mathbb{R} \rightarrow \mathbb{R}^n$ are given continuously differentiable functions.

The output $\mathbf{y}(t) \in \mathbb{R}^q$ of system (1)-(2) is given by the following equation:

$$\mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t)), \quad t \geq 0, \quad (3)$$

This work was supported by the National Natural Science Foundation of China (International Young Scientists Research Fund 11350110208) and the State Key Laboratory for Industrial Control Technology at Zhejiang University, China (Open Research Project ICT1301).

where $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^q$ is a given continuously differentiable function.

The state-delays $\tau_i, i = 1, \dots, m$, in equation (1) are unknown and need to be estimated using experimental data. Let a_i and b_i denote the lower and upper bounds of the i th state-delay, respectively. Then $\boldsymbol{\gamma} \in \mathbb{R}^m$ is called a *candidate state-delay vector* if its components satisfy the following bound constraints:

$$a_i \leq \gamma_i \leq b_i, \quad i = 1, \dots, m. \quad (4)$$

Let Γ denote the set of all such candidate state-delay vectors. Obviously, the real state-delay vector $\boldsymbol{\tau} = [\tau_1, \dots, \tau_m]^\top$ is an element of Γ .

For each candidate state-delay vector $\boldsymbol{\gamma} \in \Gamma$, consider the following parameterized system in which the components of $\boldsymbol{\gamma}$ are used as the state-delays in (1)-(2):

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{x}(t - \gamma_1), \dots, \mathbf{x}(t - \gamma_m)), \quad t > 0, \quad (5)$$

$$\mathbf{x}(t) = \boldsymbol{\phi}(t), \quad t \leq 0. \quad (6)$$

Let $\mathbf{x}(\cdot|\boldsymbol{\gamma})$ denote the solution of (5)-(6) corresponding to $\boldsymbol{\gamma} \in \Gamma$. Furthermore, let

$$\mathbf{y}(t|\boldsymbol{\gamma}) = \mathbf{g}(\mathbf{x}(t|\boldsymbol{\gamma})), \quad t \geq 0. \quad (7)$$

Our goal is to determine $\boldsymbol{\tau}$, the real state-delay vector, by comparing the predicted system output from (5)-(7) with the actual system output measured during experiments. Intuitively, if the predicted output $\mathbf{y}(\cdot|\boldsymbol{\gamma})$ is close to the measured system output, then $\boldsymbol{\gamma}$ will be a good approximation of $\boldsymbol{\tau}$.

Let $\{t_k\}_{k=1}^p$ be a set of sample times, where

$$0 < t_1 < t_2 < \dots < t_{p-1} < t_p.$$

Furthermore, let $\hat{\mathbf{y}}^k$ denote the actual system output at sample time $t = t_k$. In references [9–11], we assumed that $\hat{\mathbf{y}}^k, k = 1, \dots, p$, can be measured exactly. However, this assumption is idealistic; due to system noise and measurement errors, the true system output will often differ from the measured system output. Thus, in this paper, we assume that $\hat{\mathbf{y}}^k, k = 1, \dots, p$, are random vectors distributed according to the multivariate normal distribution, where the corresponding mean vector (of dimension pq) and covariance matrix (of dimension $pq \times pq$) are given.

To measure estimation accuracy, we use the following least-squares error function:

$$\begin{aligned} J(\boldsymbol{\gamma}) &= \sum_{k=1}^p [\mathbf{y}(t_k|\boldsymbol{\gamma}) - \hat{\mathbf{y}}^k]^\top [\mathbf{y}(t_k|\boldsymbol{\gamma}) - \hat{\mathbf{y}}^k] \\ &= \sum_{k=1}^p \|\mathbf{y}(t_k|\boldsymbol{\gamma}) - \hat{\mathbf{y}}^k\|^2. \end{aligned}$$

Our aim is to choose $\boldsymbol{\gamma} \in \Gamma$ so that the expected value of $J(\boldsymbol{\gamma})$ is minimized (i.e., the predicted system output should be close to the actual system output “on average”). At the same time, the variance of $J(\boldsymbol{\gamma})$ should also be minimized to ensure that the optimal estimate for $\boldsymbol{\gamma}$ is robust with respect to uncertainties in the output data. This motivates the following dynamic optimization problem.

Problem P. Choose a candidate state-delay vector $\boldsymbol{\gamma} \in \Gamma$ to minimize the following cost function:

$$G(\boldsymbol{\gamma}) = \mathbb{E}\{J(\boldsymbol{\gamma})\} + \alpha \text{Var}\{J(\boldsymbol{\gamma})\},$$

where $\mathbb{E}\{\cdot\}$ denotes expectation, $\text{Var}\{\cdot\}$ denotes variance, and $\alpha \geq 0$ is a given weight.

3 An Equivalent Problem

For simplicity, we write $\mathbf{y}(\cdot)$ instead of $\mathbf{y}(\cdot|\boldsymbol{\gamma})$. Then the least-squares error function $J(\boldsymbol{\gamma})$ can be written as follows:

$$\begin{aligned} J(\boldsymbol{\gamma}) &= \sum_{k=1}^p \left\{ \mathbf{y}(t_k)^\top \mathbf{y}(t_k) - 2(\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k) + (\hat{\mathbf{y}}^k)^\top \hat{\mathbf{y}}^k \right\} \\ &= \sum_{k=1}^p \mathbf{y}(t_k)^\top \mathbf{y}(t_k) - 2 \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k) + \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \hat{\mathbf{y}}^k. \end{aligned}$$

Hence,

$$\begin{aligned} \mathbb{E}\{J(\boldsymbol{\gamma})\} &= \sum_{k=1}^p \mathbf{y}(t_k)^\top \mathbf{y}(t_k) - 2 \sum_{k=1}^p \mathbb{E}\{\hat{\mathbf{y}}^k\}^\top \mathbf{y}(t_k) \\ &\quad + \sum_{k=1}^p \mathbb{E}\{(\hat{\mathbf{y}}^k)^\top \hat{\mathbf{y}}^k\} \end{aligned} \quad (8)$$

and

$$\begin{aligned} \text{Var}\{J(\boldsymbol{\gamma})\} &= 4 \text{Var}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k) \right\} \\ &\quad + \text{Var}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \hat{\mathbf{y}}^k \right\} \\ &\quad - 4 \text{Cov}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k), \sum_{l=1}^p (\hat{\mathbf{y}}^l)^\top \hat{\mathbf{y}}^l \right\}, \end{aligned} \quad (9)$$

where $\text{Cov}\{\cdot, \cdot\}$ denotes covariance.

Note that

$$\begin{aligned} \text{Var}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k) \right\} &= \text{Cov}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k), \sum_{l=1}^p (\hat{\mathbf{y}}^l)^\top \mathbf{y}(t_l) \right\} \\ &= \sum_{k=1}^p \sum_{l=1}^p \text{Cov}\{(\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k), (\hat{\mathbf{y}}^l)^\top \mathbf{y}(t_l)\} \\ &= \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top \boldsymbol{\Xi}^{k,l} \mathbf{y}(t_l), \end{aligned} \quad (10)$$

where $\boldsymbol{\Xi}^{k,l}$ is a $q \times q$ matrix whose (i, j) th element is defined by

$$[\boldsymbol{\Xi}^{k,l}]_{ij} = \text{Cov}\{\hat{y}_i^k, \hat{y}_j^l\}.$$

Furthermore,

$$\begin{aligned} \text{Cov}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k), \sum_{l=1}^p (\hat{\mathbf{y}}^l)^\top \hat{\mathbf{y}}^l \right\} &= \sum_{k=1}^p \sum_{l=1}^p \text{Cov}\{(\hat{\mathbf{y}}^k)^\top \mathbf{y}(t_k), (\hat{\mathbf{y}}^l)^\top \hat{\mathbf{y}}^l\} \\ &= \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top \mathbf{Y}^{k,l} \mathbf{1}^q, \end{aligned} \quad (11)$$

where $\mathbf{1}^q$ denotes a column vector of ones in \mathbb{R}^q and $\mathbf{\Upsilon}^{k,l}$ is a $q \times q$ matrix whose (i, j) th element is defined by

$$[\mathbf{\Upsilon}^{k,l}]_{ij} = \text{Cov}\{\hat{y}_i^k, (\hat{y}_j^l)^2\}.$$

Substituting (10) and (11) into (9) gives

$$\begin{aligned} \text{Var}\{J(\gamma)\} &= 4 \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top \Xi^{k,l} \mathbf{y}(t_l) \\ &+ \text{Var}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \hat{\mathbf{y}}^k \right\} - 4 \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top \mathbf{\Upsilon}^{k,l} \mathbf{1}^q. \end{aligned} \quad (12)$$

Now, the (i, j) th element of $\mathbf{\Upsilon}^{k,l}$ can be evaluated as follows:

$$\begin{aligned} [\mathbf{\Upsilon}^{k,l}]_{ij} &= \text{Cov}\{\hat{y}_i^k, (\hat{y}_j^l)^2\} \\ &= \text{E}\{\hat{y}_i^k (\hat{y}_j^l)^2\} - \text{E}\{\hat{y}_i^k\} \text{E}\{(\hat{y}_j^l)^2\}. \end{aligned} \quad (13)$$

Since \hat{y}_i^k and \hat{y}_j^l are jointly normal random variables, it follows from Theorem 3.3.1 in [12] that

$$\text{E}\{\hat{y}_i^k | \hat{y}_j^l\} = \text{E}\{\hat{y}_i^k\} + \frac{\text{Cov}\{\hat{y}_i^k, \hat{y}_j^l\}}{\text{Var}\{\hat{y}_j^l\}} [\hat{y}_j^l - \text{E}\{\hat{y}_j^l\}],$$

where $\text{E}\{\cdot | \hat{y}_j^l\}$ denotes conditional expectation given \hat{y}_j^l . Thus, by the law of total expectation,

$$\begin{aligned} \text{E}\{\hat{y}_i^k (\hat{y}_j^l)^2\} &= \text{E}\{\text{E}\{\hat{y}_i^k (\hat{y}_j^l)^2 | \hat{y}_j^l\}\} \\ &= \text{E}\{(\hat{y}_j^l)^2 \text{E}\{\hat{y}_i^k | \hat{y}_j^l\}\} \\ &= \text{E}\{\hat{y}_i^k\} \text{E}\{(\hat{y}_j^l)^2\} + \frac{\text{E}\{(\hat{y}_j^l)^3\} \text{Cov}\{\hat{y}_i^k, \hat{y}_j^l\}}{\text{Var}\{\hat{y}_j^l\}} \\ &\quad - \frac{\text{E}\{\hat{y}_j^l\} \text{E}\{(\hat{y}_j^l)^2\} \text{Cov}\{\hat{y}_i^k, \hat{y}_j^l\}}{\text{Var}\{\hat{y}_j^l\}}. \end{aligned}$$

Using the identities $\text{E}\{(\hat{y}_j^l)^2\} = \text{Var}\{\hat{y}_j^l\} + \text{E}\{\hat{y}_j^l\}^2$ and $\text{E}\{(\hat{y}_j^l)^3\} = \text{E}\{\hat{y}_j^l\}^3 + 3\text{E}\{\hat{y}_j^l\} \text{Var}\{\hat{y}_j^l\}$, this equation can be simplified to obtain

$$\begin{aligned} \text{E}\{\hat{y}_i^k (\hat{y}_j^l)^2\} &= \text{E}\{\hat{y}_i^k\} \text{E}\{(\hat{y}_j^l)^2\} + 2\text{E}\{\hat{y}_j^l\} \text{Cov}\{\hat{y}_i^k, \hat{y}_j^l\}. \end{aligned} \quad (14)$$

Substituting (14) into (13) yields

$$[\mathbf{\Upsilon}^{k,l}]_{ij} = 2\text{E}\{\hat{y}_j^l\} \text{Cov}\{\hat{y}_i^k, \hat{y}_j^l\} = 2\text{E}\{\hat{y}_j^l\} [\Xi^{k,l}]_{ij}. \quad (15)$$

Let $\mathbf{\Psi}^l$ be a $q \times q$ matrix defined as follows:

$$\mathbf{\Psi}^l = \left. \begin{array}{c} \text{E}\{\hat{\mathbf{y}}^l\}^\top \\ \text{E}\{\hat{\mathbf{y}}^l\}^\top \\ \vdots \\ \text{E}\{\hat{\mathbf{y}}^l\}^\top \end{array} \right\} q \text{ rows}$$

Then it follows from (15) that

$$\mathbf{\Upsilon}^{k,l} = 2\mathbf{\Psi}^l \circ \Xi^{k,l}, \quad (16)$$

where \circ denotes the Hadamard (element-wise) product. This equation shows that $\mathbf{\Upsilon}^{k,l}$ can be expressed in terms of the

expected values and covariances of the output measurements $\hat{\mathbf{y}}^k$, $k = 1, \dots, p$. Substituting (16) into (12) gives

$$\begin{aligned} \text{Var}\{J(\gamma)\} &= 4 \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top \Xi^{k,l} \mathbf{y}(t_l) \\ &+ \text{Var}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \hat{\mathbf{y}}^k \right\} \\ &- 8 \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top (\mathbf{\Psi}^l \circ \Xi^{k,l}) \mathbf{1}^q. \end{aligned} \quad (17)$$

Now, using (8) and (17), the cost function in Problem P can be written as

$$\begin{aligned} G(\gamma) &= \sum_{k=1}^p \mathbf{y}(t_k)^\top \mathbf{y}(t_k) - 2 \sum_{k=1}^p \text{E}\{\hat{\mathbf{y}}^k\}^\top \mathbf{y}(t_k) \\ &+ \sum_{k=1}^p \text{E}\{(\hat{\mathbf{y}}^k)^\top \hat{\mathbf{y}}^k\} + 4\alpha \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top \Xi^{k,l} \mathbf{y}(t_l) \\ &+ \alpha \text{Var}\left\{ \sum_{k=1}^p (\hat{\mathbf{y}}^k)^\top \hat{\mathbf{y}}^k \right\} - 8\alpha \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top (\mathbf{\Psi}^l \circ \Xi^{k,l}) \mathbf{1}^q. \end{aligned}$$

Thus, since the third and fifth terms on the right-hand side of this equation are independent of γ , Problem P is equivalent to the following optimization problem.

Problem Q. Choose a candidate state-delay vector $\gamma \in \Gamma$ to minimize the following cost function:

$$\begin{aligned} H(\gamma) &= \sum_{k=1}^p \mathbf{y}(t_k)^\top \mathbf{y}(t_k) - 2 \sum_{k=1}^p \text{E}\{\hat{\mathbf{y}}^k\}^\top \mathbf{y}(t_k) \\ &+ 4\alpha \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top \Xi^{k,l} \mathbf{y}(t_l) \\ &- 8\alpha \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_k)^\top (\mathbf{\Psi}^l \circ \Xi^{k,l}) \mathbf{1}^q, \end{aligned}$$

where $\mathbf{y}(\cdot) = \mathbf{y}(\cdot | \gamma)$ is the output of system (5)-(7).

4 Computational Algorithm

Problem Q is a dynamic optimization problem in which the decision vector γ must be chosen to minimize the cost function H subject to the time-delay system (5)-(7) and the constraints (4). In principle, this problem can be solved using the following gradient-based optimization strategy:

1. Choose an initial guess $\gamma \in \Gamma$.
2. Compute $H(\gamma)$.
3. Compute $\partial H(\gamma) / \partial \gamma_i$, $i = 1, \dots, m$.
4. Use the gradient information computed in Step 3 to perform an optimality test.
5. If γ is optimal, then stop. Otherwise, use the gradient information computed in Step 3 to determine an appropriate search direction.
6. Perform a line search along the search direction determined in Step 5 to obtain a new state-delay vector $\gamma' \in \Gamma$.
7. Set $\gamma' \rightarrow \gamma$ and return to Step 2.

If the partial derivatives of H can be computed successfully in Step 3, then Steps 4-6 can be implemented using

standard methods in numerical optimization [13]. However, since H is not an explicit function of γ , its partial derivatives cannot be derived analytically using standard differentiation rules. We will instead develop a numerical algorithm for computing the partial derivatives of H .

Let

$$\bar{\mathbf{f}}(t|\gamma) = \mathbf{f}(\mathbf{x}(t|\gamma), \mathbf{x}(t - \gamma_1|\gamma), \dots, \mathbf{x}(t - \gamma_m|\gamma))$$

and

$$\frac{\partial \bar{\mathbf{f}}(t|\gamma)}{\partial \tilde{\mathbf{x}}^\varsigma} = \frac{\partial \mathbf{f}(\mathbf{x}(t|\gamma), \mathbf{x}(t - \gamma_1|\gamma), \dots, \mathbf{x}(t - \gamma_m|\gamma))}{\partial \mathbf{x}(t - \gamma_\varsigma)},$$

$$\varsigma = 0, \dots, m,$$

where $\gamma_0 = 0$ (i.e., $\partial \tilde{\mathbf{x}}^0$ denotes differentiation with respect to \mathbf{x}). Furthermore, for each $k = 1, \dots, p$, define

$$\boldsymbol{\sigma}^k(\gamma) = \mathbf{y}(t_k|\gamma) - \mathbf{E}\{\hat{\mathbf{y}}^k\}$$

$$+ 4\alpha \sum_{l=1}^p \{ \Xi^{k,l} \mathbf{y}(t_l|\gamma) - (\Psi^l \circ \Xi^{k,l}) \mathbf{1}^q \}.$$

Consider the following auxiliary ‘‘time-advance’’ system:

$$\dot{\boldsymbol{\lambda}}(t) = - \sum_{\varsigma=0}^m \left[\frac{\partial \bar{\mathbf{f}}(t + \gamma_\varsigma|\gamma)}{\partial \tilde{\mathbf{x}}^\varsigma} \right]^\top \boldsymbol{\lambda}(t + \gamma_\varsigma), \quad t \leq t_p, \quad (18)$$

with the intermediate jump conditions

$$\boldsymbol{\lambda}(t_k^-) = \boldsymbol{\lambda}(t_k^+) + 2 \left[\frac{\partial \mathbf{g}(\mathbf{x}(t_k|\gamma))}{\partial \mathbf{x}} \right]^\top \boldsymbol{\sigma}^k(\gamma), \quad (19)$$

$$k = 1, \dots, p,$$

and the terminal condition

$$\boldsymbol{\lambda}(t) = \mathbf{0}, \quad t > t_p, \quad (20)$$

where $\gamma \in \Gamma$ is an arbitrary candidate state-delay vector. Let $\boldsymbol{\lambda}(\cdot|\gamma)$ denote the left-continuous solution of (18)-(20) corresponding to $\gamma \in \Gamma$. We now express the partial derivatives of H with respect to the state-delays in terms of $\boldsymbol{\lambda}(\cdot|\gamma)$.

Theorem 1. For each $i = 1, \dots, m$, the partial derivative of H with respect to γ_i is given by

$$\frac{\partial H(\gamma)}{\partial \gamma_i} = - \int_0^{t_p} \boldsymbol{\lambda}(t|\gamma)^\top \frac{\partial \bar{\mathbf{f}}(t|\gamma)}{\partial \tilde{\mathbf{x}}^i} \boldsymbol{\chi}(t - \gamma_i|\gamma) dt, \quad (21)$$

where

$$\boldsymbol{\chi}(t|\gamma) = \begin{cases} \dot{\boldsymbol{\phi}}(t), & \text{if } t \leq 0, \\ \bar{\mathbf{f}}(t|\gamma), & \text{if } t > 0. \end{cases}$$

Proof. For simplicity, we use the notation $\mathbf{x}(\cdot) = \mathbf{x}(\cdot|\gamma)$ and $\mathbf{y}(\cdot) = \mathbf{y}(\cdot|\gamma)$. Differentiating H with respect to γ_i gives

$$\frac{\partial H(\gamma)}{\partial \gamma_i} = 2 \sum_{k=1}^p \mathbf{y}(t_k)^\top \frac{\partial \mathbf{y}(t_k)}{\partial \gamma_i} - 2 \sum_{k=1}^p \mathbf{E}\{\hat{\mathbf{y}}^k\}^\top \frac{\partial \mathbf{y}(t_k)}{\partial \gamma_i}$$

$$+ 8\alpha \sum_{k=1}^p \sum_{l=1}^p \mathbf{y}(t_l)^\top \Xi^{l,k} \frac{\partial \mathbf{y}(t_k)}{\partial \gamma_i}$$

$$- 8\alpha \sum_{k=1}^p \sum_{l=1}^p (\mathbf{1}^q)^\top (\Psi^l \circ \Xi^{k,l})^\top \frac{\partial \mathbf{y}(t_k)}{\partial \gamma_i}.$$

Since $\Xi^{k,l} = (\Xi^{l,k})^\top$, this equation can be rearranged to obtain

$$\frac{\partial H(\gamma)}{\partial \gamma_i} = 2 \sum_{k=1}^p \boldsymbol{\sigma}^k(\gamma)^\top \frac{\partial \mathbf{y}(t_k)}{\partial \gamma_i}$$

$$= 2 \sum_{k=1}^p \boldsymbol{\sigma}^k(\gamma)^\top \frac{\partial \mathbf{g}(\mathbf{x}(t_k))}{\partial \mathbf{x}} \frac{\partial \mathbf{x}(t_k)}{\partial \gamma_i}. \quad (22)$$

Now, let $\mathbf{v}^k : [t_{k-1}, t_k] \rightarrow \mathbb{R}^n$, $k = 1, \dots, p$, be a set of arbitrary absolutely continuous functions. Multiplying equation (5) by \mathbf{v}^k and integrating over subintervals $[t_{k-1}, t_k]$, $k = 1, \dots, p$, gives

$$\sum_{k=1}^p \int_{t_{k-1}}^{t_k} \mathbf{v}^k(t)^\top \dot{\mathbf{x}}(t) dt = \sum_{k=1}^p \int_{t_{k-1}}^{t_k} \mathbf{v}^k(t)^\top \bar{\mathbf{f}}(t|\gamma) dt.$$

Thus, using integration-by-parts,

$$\sum_{k=1}^p \left\{ \mathbf{v}^k(t_k)^\top \mathbf{x}(t_k) - \mathbf{v}^k(t_{k-1})^\top \mathbf{x}(t_{k-1}) \right\}$$

$$- \sum_{k=1}^p \int_{t_{k-1}}^{t_k} \dot{\mathbf{v}}^k(t)^\top \mathbf{x}(t) dt = \sum_{k=1}^p \int_{t_{k-1}}^{t_k} \mathbf{v}^k(t)^\top \bar{\mathbf{f}}(t|\gamma) dt.$$

Rearranging this equation gives

$$\sum_{k=1}^{p-1} \left\{ \mathbf{v}^k(t_k)^\top - \mathbf{v}^{k+1}(t_k)^\top \right\} \mathbf{x}(t_k)$$

$$+ \mathbf{v}^p(t_p)^\top \mathbf{x}(t_p) - \mathbf{v}^1(0)^\top \boldsymbol{\phi}(0)$$

$$= \sum_{k=1}^p \int_{t_{k-1}}^{t_k} \left\{ \dot{\mathbf{v}}^k(t)^\top \mathbf{x}(t) + \mathbf{v}^k(t)^\top \bar{\mathbf{f}}(t|\gamma) \right\} dt.$$

Now, by differentiating this equation with respect to γ_i , and noting that $\dot{\mathbf{x}}(t) = \boldsymbol{\chi}(t|\gamma)$ almost everywhere, we obtain

$$\sum_{k=1}^{p-1} \left\{ \mathbf{v}^k(t_k)^\top - \mathbf{v}^{k+1}(t_k)^\top \right\} \frac{\partial \mathbf{x}(t_k)}{\partial \gamma_i} + \mathbf{v}^p(t_p)^\top \frac{\partial \mathbf{x}(t_p)}{\partial \gamma_i}$$

$$= \sum_{k=1}^p \int_{t_{k-1}}^{t_k} \dot{\mathbf{v}}^k(t)^\top \frac{\partial \mathbf{x}(t)}{\partial \gamma_i} dt$$

$$+ \sum_{\varsigma=0}^m \sum_{k=1}^p \int_{t_{k-1}}^{t_k} \mathbf{v}^k(t)^\top \frac{\partial \bar{\mathbf{f}}(t|\gamma)}{\partial \tilde{\mathbf{x}}^\varsigma} \frac{\partial \mathbf{x}(t - \gamma_\varsigma)}{\partial \gamma_i} dt$$

$$- \sum_{k=1}^p \int_{t_{k-1}}^{t_k} \mathbf{v}^k(t)^\top \frac{\partial \bar{\mathbf{f}}(t|\gamma)}{\partial \tilde{\mathbf{x}}^i} \boldsymbol{\chi}(t - \gamma_i|\gamma) dt. \quad (23)$$

Recall that $\mathbf{v}^k : [t_{k-1}, t_k] \rightarrow \mathbb{R}^n$, $k = 1, \dots, p$, were chosen arbitrarily. Thus, we can define

$$\mathbf{v}^k(t) = \begin{cases} \boldsymbol{\lambda}(t_{k-1}^+), & \text{if } t = t_{k-1}, \\ \boldsymbol{\lambda}(t), & \text{if } t \in (t_{k-1}, t_k), \\ \boldsymbol{\lambda}(t_k^-), & \text{if } t = t_k, \end{cases} \quad (24)$$

where $\boldsymbol{\lambda}(\cdot) = \boldsymbol{\lambda}(\cdot|\gamma)$ is the left-continuous solution of the auxiliary time-advance system (18)-(20). Then by using

equations (19)-(22), equation (23) becomes

$$\begin{aligned} \frac{\partial H(\boldsymbol{\gamma})}{\partial \gamma_i} &= \int_0^{t_p} \dot{\boldsymbol{\lambda}}(t)^\top \frac{\partial \boldsymbol{x}(t)}{\partial \gamma_i} dt \\ &+ \sum_{\varsigma=0}^m \int_0^{t_p} \boldsymbol{\lambda}(t)^\top \frac{\partial \bar{\boldsymbol{f}}(t|\boldsymbol{\gamma})}{\partial \tilde{\boldsymbol{x}}^\varsigma} \frac{\partial \boldsymbol{x}(t-\gamma_\varsigma)}{\partial \gamma_i} dt \\ &- \int_0^{t_p} \boldsymbol{\lambda}(t)^\top \frac{\partial \bar{\boldsymbol{f}}(t|\boldsymbol{\gamma})}{\partial \tilde{\boldsymbol{x}}^i} \boldsymbol{\chi}(t-\gamma_i|\boldsymbol{\gamma}) dt. \end{aligned}$$

Hence, by applying a change of variable in the second integral and using equations (6) and (20),

$$\begin{aligned} \frac{\partial H(\boldsymbol{\gamma})}{\partial \gamma_i} &= \int_0^{t_p} \dot{\boldsymbol{\lambda}}(t)^\top \frac{\partial \boldsymbol{x}(t)}{\partial \gamma_i} dt \\ &+ \sum_{\varsigma=0}^m \int_0^{t_p} \boldsymbol{\lambda}(t+\gamma_\varsigma)^\top \frac{\partial \bar{\boldsymbol{f}}(t+\gamma_\varsigma|\boldsymbol{\gamma})}{\partial \tilde{\boldsymbol{x}}^\varsigma} \frac{\partial \boldsymbol{x}(t)}{\partial \gamma_i} dt \\ &- \int_0^{t_p} \boldsymbol{\lambda}(t)^\top \frac{\partial \bar{\boldsymbol{f}}(t|\boldsymbol{\gamma})}{\partial \tilde{\boldsymbol{x}}^i} \boldsymbol{\chi}(t-\gamma_i|\boldsymbol{\gamma}) dt. \end{aligned}$$

Substituting (18) into this equation completes the proof. \square

Recall from our discussion at the beginning of this section that computing the partial derivatives of H is the key step to solving Problem Q. By virtue of Theorem 1, these partial derivatives can be obtained by solving the state system (5)-(6) and the auxiliary system (18)-(20), and then substituting $\boldsymbol{x}(\cdot|\boldsymbol{\gamma})$ and $\boldsymbol{\lambda}(\cdot|\boldsymbol{\gamma})$ into equation (21). This procedure can be readily incorporated into standard gradient-based optimization methods such as sequential quadratic programming [13] to solve Problem Q effectively.

5 Numerical Example

Consider a continuously-stirred tank reactor in which the reaction $A \rightarrow B$ occurs. The reaction dynamics can be described by the following delay-differential equations [11]:

$$\begin{aligned} \dot{x}_1(t) &= \frac{1}{10}(1-x_1(t)) \exp \left[\frac{20x_2(t)}{x_2(t)+20} \right] \\ &- 2x_1(t) + x_1(t-\tau), \quad t \in (0, 10], \quad (25) \end{aligned}$$

$$\begin{aligned} \dot{x}_2(t) &= \frac{4}{5}(1-x_1(t)) \exp \left[\frac{20x_2(t)}{x_2(t)+20} \right] \\ &- \frac{5}{2}x_2(t) + x_2(t-\tau), \quad t \in (0, 10], \quad (26) \end{aligned}$$

with initial conditions

$$x_1(t) = 1, \quad x_2(t) = 1, \quad t \leq 0, \quad (27)$$

where x_1 is the (dimensionless) concentration of A , x_2 is the (dimensionless) temperature of the reactor, and τ is an unknown state-delay that needs to be identified.

We assume that both state variables can be measured. Thus, the state and output variables are the same:

$$y_1(t) = x_1(t), \quad y_2(t) = x_2(t), \quad t \in [0, 10]. \quad (28)$$

Let γ denote the state-delay estimate. Furthermore, let $y_1(\cdot|\boldsymbol{\gamma})$ and $y_2(\cdot|\boldsymbol{\gamma})$ denote the output trajectories from system (25)-(28) corresponding to the state-delay estimate $\boldsymbol{\gamma}$. Then the least-squares error function is

$$J(\boldsymbol{\gamma}) = \sum_{k=1}^{20} \left\{ (y_1(t_k|\boldsymbol{\gamma}) - \hat{y}_1^k)^2 + (y_2(t_k|\boldsymbol{\gamma}) - \hat{y}_2^k)^2 \right\},$$

where the sample times are $t_k = \frac{1}{2}k$, $k = 1, \dots, 20$. The problem is to choose $\boldsymbol{\gamma}$ to minimize

$$G(\boldsymbol{\gamma}) = \mathbb{E}\{J(\boldsymbol{\gamma})\} + \alpha \text{Var}\{J(\boldsymbol{\gamma})\}$$

subject to the delay-differential equations (25)-(26) and the initial conditions (27), where $\alpha > 0$ is a given weight. Here, we choose $\alpha = 1$.

We consider two scenarios for the inexact output data.

- *Scenario 1.* The output values \hat{y}_1^k and \hat{y}_2^k are such that

$$\begin{aligned} \mathbb{E}\{\hat{\boldsymbol{y}}^k\} &= \begin{bmatrix} \mathbb{E}\{\hat{y}_1^k\} \\ \mathbb{E}\{\hat{y}_2^k\} \end{bmatrix} = \begin{bmatrix} y_1(t_k|2) \\ y_2(t_k|2) \end{bmatrix}, \\ \Xi^{k,l} &= \begin{cases} \frac{1}{10}\boldsymbol{I}, & \text{if } k=l, \\ \mathbf{0}, & \text{otherwise,} \end{cases} \end{aligned}$$

where $y_1(\cdot|2)$ and $y_2(\cdot|2)$ denote the output trajectories of (25)-(28) corresponding to $\tau = 2$, \boldsymbol{I} denotes the 2×2 identity matrix, and $\mathbf{0}$ denotes the 2×2 zero matrix. Here, the output measurements are mutually independent.

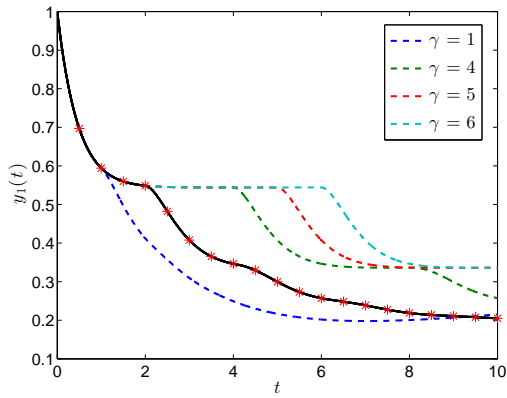
- *Scenario 2.* The output values \hat{y}_1^k and \hat{y}_2^k are such that

$$\begin{aligned} \mathbb{E}\{\hat{\boldsymbol{y}}^k\} &= \begin{bmatrix} \mathbb{E}\{\hat{y}_1^k\} \\ \mathbb{E}\{\hat{y}_2^k\} \end{bmatrix} = \begin{bmatrix} y_1(t_k|3) \\ y_2(t_k|3) \end{bmatrix}, \\ \Xi^{k,k-1} &= \frac{1}{10} \begin{bmatrix} \zeta_1^k & \zeta_2^k \\ \zeta_3^k & \zeta_4^k \end{bmatrix}, \\ \Xi^{k,k} &= \frac{1}{10} \begin{bmatrix} 1 & \zeta_5^k \\ \zeta_6^k & 1 \end{bmatrix}, \\ \Xi^{k,k+1} &= \frac{1}{10} \begin{bmatrix} \zeta_7^k & \zeta_8^k \\ \zeta_9^k & \zeta_{10}^k \end{bmatrix}, \\ \Xi^{k,l} &= \mathbf{0}, \quad l \neq k-1, k, k+1, \end{aligned}$$

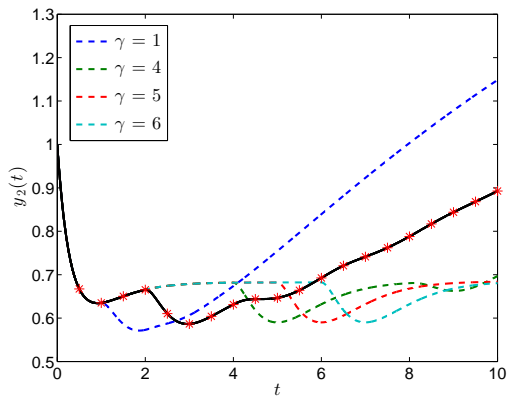
where $y_1(\cdot|3)$ and $y_2(\cdot|3)$ denote the output trajectories of (25)-(28) corresponding to $\tau = 3$, and ζ_j^k , $j = 1, \dots, 10$, are random numbers from $[0, \frac{4}{5}]$. Here, the output measurements are dependent; each output sample depends on its neighbouring output samples.

To implement the gradient-based optimization approach proposed in Section 4, we wrote a Fortran program that invokes the optimization code NLPQLP [14]. This program solves the state and auxiliary systems using the Runge-Kutta method of order 6, with the delay/advance values calculated using Hermite interpolation.

For each scenario, we applied our Fortran program starting from four initial guesses: $\gamma = 1$, $\gamma = 4$, $\gamma = 5$, and $\gamma = 6$. The program converged quickly from all initial guesses. The optimal state-delay estimates obtained are: $\gamma = 2$ for Scenario 1 and $\gamma = 3.12335$ for Scenario 2. As expected, the optimal estimates are close to the original state-delays used to compute the artificial output data ($\tau = 2$ for Scenario 1 and $\tau = 3$ for Scenario 2). In Scenario 1, when the observation measurements are independent, $\gamma = 2$ is actually the optimal solution. In Scenario 2, when the observation measurements are dependent random variables, the optimal estimate differs slightly from $\gamma = 3$. For each scenario, NLPQLP converged successfully in around 10-15 iterations. The numerical convergence of the output trajectories is shown in Figures 1 and 2. In each figure, the dashed lines show the initial output trajectories and the solid black line



(a) Output trajectory $y_1(t) = x_1(t)$.



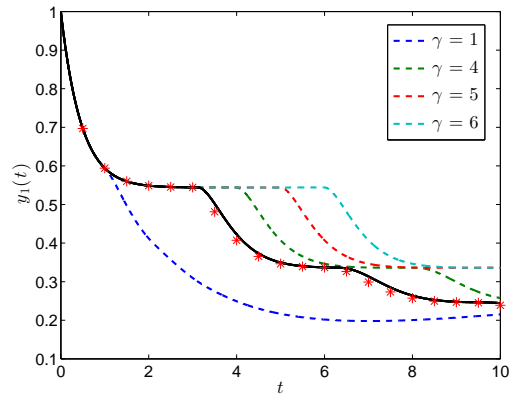
(b) Output trajectory $y_2(t) = x_2(t)$.

Fig. 1: Numerical convergence of the output trajectory in Scenario 1.

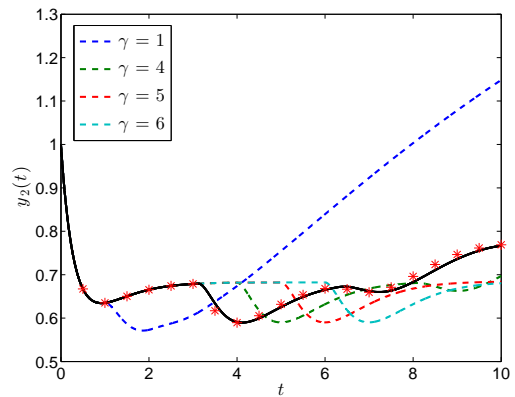
shows the final (converged) trajectory. The red crosses show the mean values of the observed data (which correspond to $\tau = 2$ in Scenario 1 and $\tau = 3$ in Scenario 2).

References

- [1] 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(2): 307–323, 2012.
- [2] Z. Benayache, G. Besançon, and D. Georges, A new nonlinear control methodology for irrigation canals based on a delayed input model, in *Proceedings of the 17th World Congress of the International Federation of Automatic Control*, Seoul, July 2008: 2544–2549.
- [3] L. Denis-Vidal, C. Jaubertie, and G. Joly-Blanchard, Identifiability of a nonlinear delayed-differential aerospace model, *IEEE Transactions on Automatic Control*, 51(1): 154–158, 2006.
- [4] F. Y. Wang and Q. Yu, Optimal protein separations with time lags in control functions, *Journal of Process Control*, 4(3): 135–142, 1994.
- [5] J. P. Richard, Time-delay systems: An overview of some recent advances and open problems, *Automatica*, 39(10): 1667–1694, 2003.
- [6] H. T. Banks, K. Rehm, and K. Sutton, Inverse problems for nonlinear delay systems, *Methods and Applications of Analysis*, 17(4): 331–356, 2010.
- [7] F. Gao, Y. Qi, Q. Yin, and J. Xiao, An artificial bee colony



(a) Output trajectory $y_1(t) = x_1(t)$.



(b) Output trajectory $y_2(t) = x_2(t)$.

Fig. 2: Numerical convergence of the output trajectory in Scenario 2.

algorithm for unknown parameters and time-delays identification of chaotic systems, in *Proceedings of the 5th International Conference on Computer Sciences and Convergence Information Technology*, Seoul, December 2010: 659–664.

- [8] Y. Tang and X. Guan, Parameter estimation for time-delay chaotic system by particle swarm optimization, *Chaos, Solitons and Fractals*, 40(3): 1391–1398, 2009.
- [9] Q. Chai, R. Loxton, K. L. Teo, and C. Yang, A unified parameter identification method for nonlinear time-delay systems, *Journal of Industrial and Management Optimization*, 9(2): 471–486, 2013.
- [10] Q. Chai, R. Loxton, K. L. Teo, and C. Yang, Time-delay estimation for nonlinear systems with piecewise-constant input, *Applied Mathematics and Computation*, 219(17): 9543–9560, 2013.
- [11] R. Loxton, K. L. Teo, and V. Rehbock, An optimization approach to state-delay identification, *IEEE Transactions on Automatic Control*, 55(9): 2113–2119, 2010.
- [12] B. Flury, *A First Course in Multivariate Statistics*. New York: Springer, 1997.
- [13] D. G. Luenberger and Y. Ye, *Linear and Nonlinear Programming*, 3rd Edition. New York: Springer, 2008.
- [14] K. Schittkowski, *NLPQLP: A Fortran Implementation of a Sequential Quadratic Programming Algorithm with Distributed and Non-monotone Line Search – User’s Guide*. Bayreuth: University of Bayreuth, 2007.