Department of Mathematics and Statistics

Computational Methods for Solving Optimal Industrial Process Control Problems

Qinqin Chai

This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University

March 2013

Declaration

To the best of my knowledge and belief, this thesis contains no material previously published by any other person except where due acknowledgment has been made.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

.....
Qinqin Chai
March 2013

Abstract

In this thesis, we develop new computational methods for three classes of dynamic optimization problems: (i) A parameter identification problem for a general nonlinear time-delay system; (ii) an optimal control problem involving systems with both input and output delays, and subject to continuous inequality state constraints; and (iii) a max-min optimal control problem arising in gradient elution chromatography.

In the first problem, we consider a parameter identification problem involving a general nonlinear time-delay system, where the unknown time delays and system parameters are to be identified. This problem is posed as a dynamic optimization problem, where its cost function is to measure the discrepancy between predicted output and observed system output. The aim is to find unknown time-delays and system parameters such that the cost function is minimized. We develop a gradient-based computational method for solving this dynamic optimization problem. We show that the gradients of the cost function with respect to these unknown parameters can be obtained via solving a set of auxiliary time-delay differential systems from $t = 0$ to $t = T$. On this basis, the parameter identification problem can be solved as a nonlinear optimization problem and existing optimization techniques can be used. Two numerical examples are solved using the proposed computational method. Simulation results show that the proposed computational method is highly effective. In particular, the convergence is very fast even when the initial guess of the parameter values is far away from the optimal values.

Unlike the first problem, in the second problem, we consider a time delay identification problem, where the input function for the nonlinear time-delay system is piecewise-constant. We assume that the time-delays—one involving the state variables and the other involving the input variables—are unknown and need to be estimated using experimental data. We also formulate the problem of estimating the unknown delays as a nonlinear optimization problem in which the cost function measures the least-squares error between predicted output and measured system output. This estimation problem can be viewed as a switched system optimal control problem with time-delays. We show that the gradient of the cost function with respect to the unknown state delay can be obtained via solving a auxiliary time-delay differential system. Furthermore, the gradient of the cost function with respect to the unknown input delay can be obtained via solving an auxiliary time-delay differential system with jump conditions at the delayed control switching time points. On this basis, we develop a heuristic computational algorithm for

solving this problem using gradient based optimization algorithms. Time-delays in two industrial processes are estimated using the proposed computational method. Simulation results show that the proposed computational method is highly effective.

For the third problem, we consider a general optimal control problem governed by a system with input and output delays, and subject to continuous inequality constraints on the state and control. We focus on developing an effective computational method for solving this constrained time delay optimal control problem. For this, the control parameterization technique is used to approximate the time planning horizon $[0, T]$ into N subintervals. Then, the control is approximated by a piecewise constant function with possible discontinuities at the pre-assigned partition points, which are also called the switching time points. The heights of the piecewise constant function are decision variables which are to be chosen such that a given cost function is minimized. For the continuous inequality constraints on the state, we construct approximating smooth functions in integral form. Then, the summation of these approximating smooth functions in integral form, which is called the constraint violation, is appended to the cost function to form a new augmented cost function. In this way, we obtain a sequence of approximate optimization problems subject to only boundedness constraints on the decision variables. Then, the gradient of the augmented cost function is derived. On this basis, we develop an effective computational method for solving the time-delay optimal control problem with continuous inequality constraints on the state and control via solving a sequence of approximate optimization problems, each of which can be solved as a nonlinear optimization problem by using existing gradient-based optimization techniques. This proposed method is then used to solve a practical optimal control problem arising in the study of a real evaporation process. The results obtained are highly satisfactory, showing that the proposed method is highly effective.

The fourth problem that we consider is a max-min optimal control problem arising in the study of gradient elution chromatography, where the manipulative variables in the chromatographic process are to be chosen such that the separation efficiency is maximized. This problem has three non-standard characteristics: (i) The objective function is non-smooth; (ii) each state variable is defined over a different time horizon; and (iii) the order of the final times for the state variable, the so-called retention times, are not fixed. To solve this problem, we first introduce a set of auxiliary decision variables to govern the ordering of the retention times. The integer constraints on these auxiliary decision variables are approximated by continuous boundedness constraints. Then, we approximate the control by a piecewise constant function, and apply a novel time-scaling transformation to map the retention times and control switching times to fixed points in a new time horizon. The retention times and control switching times become decision variables in the new time horizon. In addition, the max-min objective function is approximated by a minimization problem subject to an additional constraint. On this basis, the optimal control problem is

reduced to an approximate nonlinear optimization problem subject to smooth constraints, which is then solved using a recently developed exact penalty function method. Numerical results obtained show that this approach is highly effective.

Finally, some concluding remarks and suggestions for further study are made in the conclusion chapter.

List of publications

The following papers were published or accepted for publication during the PhD candidature:

- Q. Chai, R. Loxton, K. L. Teo, and C. H. Yang, "A class of optimal state-delay control problems," *Nonlinear Analysis: Real World Applications*, vol. 14, no. 3, pp. 1536-1550, 2013.
- Q. Chai, R. Loxton, K. L. Teo, and C. H. Yang, "A unified parameter identification method for nonlinear time-delay systems," *Journal of Industrial and Management Optimization*, vol. 9, no. 2, pp. 471-486, 2013.
- Q. Chai, R. Loxton, K. L. Teo, and C. H. Yang, "Time-delay estimation for nonlinear systems with piecewise-constant input," *Applied Mathematics and Computation*, vol. 219, pp. 9543-9560, 2013.
- Q. Chai, R. Loxton, K. L. Teo, and C. H. Yang, "A max-min control problem arising in gradient elution chromatography," *Industrial and Engineering Chemistry Research*, vol. 51, no. 17, pp. 6137-6144, 2012.
- Q. Chai, C. H. Yang, K. L. Teo, and W. H. Gui, "Optimal control of an industrial-scale evaporation process: sodium aluminate solution," *Control Engineering Practice*, vol. 20, no. 6, pp. 618-628, 2012.
- Y. G. Li, W. H. Gui, K. L. Teo, H. Q. Zhu, and Q. Q. Chai, "Optimal control for zinc solution purification based on interaction CSTR models," *Journal of Process Control*, vol. 22, no. 10, pp. 1878-1889, 2012.
- R. Loxton, Q. Chai, and K. L. Teo, "A class of max-min optimal control problems with applications to chromatography," *Proceedings of the 5th International Conference on Optimization and Control with Applications*, Bejing, China, December 4-8, 2012.

Acknowledgements

The research reported in this thesis was carried out from September 2008 to December 2012. During this period, I was a PhD student in the Department of Mathematics and Statistics, Curtin University and the School of Information Science and Engineering, Central South University. I wish to express my appreciation for all kinds of help I received from my supervisors, families, friends, and colleagues during this period of time.

I would like to express my heartfelt thanks to my supervisor, Prof. Kok Lay Teo, and his wife, Mrs. Lye-Hen Teo. Prof. Teo has guided my research during the past two years with remarkable patience and enthusiasm. During the two years of my stay in Australia, he was not only a great supervisor of my research, but also a gracious mentor of my life.

I would like to thank Dr. Ryan Loxton, my co-supervisor in the Department of Mathematics and Statistics, Curtin University. He is an excellent mathematician, and he is always willing to help whenever I have difficulties in my research. He has shared his experience with me on the skill of writing papers. I really appreciate all the unselfish help I received from him.

I would like to thank Prof. Chunhua Yang, my co-supervisor in the School of Information Science and Engineering, Central South University. I have known Prof. Yang since I became a post-graduate student in the School of Information Science and Engineering, Central South University in 2006. It was her who led me to the road of research and helped me to apply for a scholarship from China Scholarship Council, so that I could come to Australia and start a new stage of my research.

I would like to give thanks to Prof. Weihua Gui, in the School of Information Science and Engineering, Central South University. He is a great engineer. He has shared with me many practical experiences, which have inspired me in my research.

I also thank Prof. Yonghong Wu, Postgraduate Coordinator in the Department of Mathematics and Statistics, Curtin University. He is the Chair of my thesis committee and he has helped me on many occasions.

I would like to give thanks to the people and friends that I have worked with in Australia, especially, Dr. Qun Lin, Dr. Bin Li, Dr. Jingyang Zhou and his wife Yujing Wang, Dr. Changjun Yu, Prof. Honglei Xu, Yufei Sun, Yanyan Yin, Qian Sun, Mingliang Xue, Yanli Zhou and the department's academic visitors that I had the opportunity to meet, Dr. Canghua Jiang, A/Prof. Chuanjiang Li, A/Prof. Tieqiao Tang, Prof. Xuegang Hu, A/Prof. Lingling Xu, and A/Prof. Xiangyu Gao.

In addition, I thank each of my friends and teachers in Changsha, Prof. Yalin Wang, Prof. Yongfang Xie, Prof. Jianjun He, A/Prof. Yonggang Li, Dr. Hongqiu Zhu, Dr. Can Huang, Dr. Huifeng Ren, Dr. Canhui Xu, Dr. Cao Shi, Jianhua Liu, and Jingjing Qin. I really enjoyed the time that I spent with them.

I thank all of the staff in the Department of Mathematics and Statistics for contributing to a friendly working environment. The administrative staff—Joyce Yang, Shuie Liu, Lisa Holling, Jeannie Darmago, and Carey Ryken Rapp—deserve special thanks for providing kind and professional help on numerous occasions.

Finally, on a more personal note, I sincerely thank my husband Xinmei Lin and everyone in my family for their love, understanding and support during my PhD candidature in Australia.

Contents

1 Introduction **1**

1.1 Motivation and background 1
1.2 Nonlinear programming problems 2
1.3 Numerical techniques for optimal control problems 4
1.4 Parameter identification for time-delay systems 10
1.5 Control methods for time-delay systems 12
1.5.1 Optimal control for time-delay systems 12
1.5.2 Model predictive control for time-delay systems 12
1.6 Max-Min optimal control problems 15
1.7 Overview of the thesis 17

2 Identification of time-delays and parameters for nonlinear systems **21**

2.1 Introduction 21
2.2 Problem formulation 22
2.3 Preliminaries 23
2.4 Gradient computation 30
2.4.1 State variation with respect to time-delays 30
2.4.2 State variation with respect to system parameters 34
2.4.3 Gradient computation algorithm 37
2.5 Numerical examples 38
2.5.1 Example 2.1 38
2.5.2 Example 2.2 40
2.6 Conclusion 42

3 Identification of time-delays for nonlinear systems with piecewise-constant input **44**

3.1 Introduction 44
3.2 Problem formulation 45
3.3 State variation 48
3.3.1 State variation with respect to the state-delay 48
3.3.2 State variation with respect to the input-delay 49
3.4 Computation algorithm 62

3.5 Numerical examples . . . 64
3.5.1 Example 1: Zinc sulphate purification . . . 64
3.5.2 Example 2: Sodium aluminate evaporation . . . 67
3.6 Conclusions . . . 70

4 Time-delay optimal control application problem: an industrial evaporation process 71
4.1 Introduction . . . 71
4.2 Problem statement . . . 72
4.3 Solution method . . . 74
4.3.1 Control parameterization . . . 74
4.3.2 Constraints transformation . . . 76
4.3.3 Convergence analysis . . . 78
4.3.4 Computational method . . . 85
4.4 Application: optimal control of an evaporation process . . . 91
4.4.1 The evaporation system . . . 92
4.4.2 Mathematical model of the evaporation system . . . 94
4.4.3 Optimal control problem formulation . . . 99
4.4.4 Numerical results . . . 102
4.5 Conclusion . . . 107

5 A max-min control problem arising in gradient elution chromatography 108
5.1 Introduction . . . 108
5.2 Problem statement . . . 111
5.3 Problem transformation . . . 112
5.4 A computational method . . . 118
5.5 Numerical examples . . . 125
5.5.1 Example 5.1 . . . 125
5.5.2 Example 5.2 . . . 127
5.6 Conclusions . . . 128

6 Summary and future research directions 130
6.1 Main contributions of the thesis . . . 130
6.2 Future research directions . . . 132

Bibliography 133

CHAPTER 1

Introduction

1.1 Motivation and background

In an optimal control problem, there is a governing dynamic system whose trajectory, called the state, is influenced by an adjustable variable, called the control. Its aim is to find a control such that a performance index is optimized subject to some specified constraints. The performance index, which is also called the cost function, could represent energy consumption, wastage of consumable materials, or the time taken to achieve a given task, just to name a few examples. The specified constraints could arise due to design specifications, safety operation conditions or engineering requirements. Optimal control has many successful real world applications in areas ranging from engineering to economics. Many of these real world problems are too complicated to admit analytical solutions. Thus, it is unavoidable to rely on numerical methods to deal with these problems. There are now many computational methods available in the literature for solving various classes of optimal control problems. Most of these methods are for control problems in which the governing dynamic system does not involve time-delay. However, time-delays arise in many real world applications, such as chemical tank reactors [86], aerospace engineering [143], chromatography [107], and power converters [153]. The effects of time-delays must not be ignored, because it is known [28, 123] that time-delays in a dynamic system could cause instability in the system concerned. However, techniques, theory and methods for optimal control problems without time-delay are often not applicable to optimal control problems with time-delays. Consequently, it has attracted a considerable interest amongst mathematicians and engineers, especially process engineers, to develop effective computational methods for solving optimal control problems involving nonlinear time-delay systems [25, 74, 77, 131].

In this thesis, we will formulate several optimal control problems arising in practical industrial processes. Then, we will develop effective computational methods for solving these real world optimal control problems.

1.2 Nonlinear programming problems

Nonlinear programming (NLP) problem is an optimization problem with nonlinear objective function and/or nonlinear constraints. NLP problems arise in many real world applications. A general NLP problem can be stated as follows:

Problem ($\tilde{P}_1$).

$$\begin{aligned} Minimize \quad & f(\boldsymbol{x}) \\ s.t. \quad & g_i(\boldsymbol{x}) = 0, \quad i = 1, \ldots, m_1, \\ & g_i(\boldsymbol{x}) \leq 0, \quad i = m_1 + 1, \ldots, m_1 + m_2. \end{aligned}$$

where $\boldsymbol{x} \in \mathbb{R}^r$ is the decision vector; $f(\boldsymbol{x})$ is the objective function; $g_i(\boldsymbol{x})$, $i = 1, \ldots, m_1$, are given equality constraint functions; and $g_i(\boldsymbol{x})$, $i = m_1 + 1, \ldots, m_1 + m_2$, are given inequality constraint functions. Assume that the objective function and the constraint functions are twice continuously differentiable.

For NLP problem, a vector $\boldsymbol{x}$ is called a feasible solution if it satisfies all the constraint functions of Problem ($\tilde{P}_1$). The set containing all the feasible solutions is called the feasible region. It is denoted by $\mathcal{F}$. Furthermore, the jth inequality constraint is said to be active at the point $\boldsymbol{x}$ if $g_j(\boldsymbol{x}) = 0$.

Definition 1.1. *(Active set.) The active set $\mathcal{A}(\boldsymbol{x})$ at point $\boldsymbol{x}$ is the set of indices defined by*

$$\mathcal{A}(\boldsymbol{x}) = \{j \in \{m_1 + 1, \ldots, m_1 + m_2\} | g_j(\boldsymbol{x}) = 0\}.$$

Let $\bar{\mathcal{A}}(\boldsymbol{x}) = \mathcal{A}(\boldsymbol{x}) \cup \{1, \ldots, m_1\}$ denote the set of indices of the active constraints and the equality constraints at $\boldsymbol{x}$.

A feasible point $\boldsymbol{x}^* \in \mathcal{F}$ is called a global minimum if $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$, $\forall \boldsymbol{x} \in \mathcal{F}$. Moreover, a feasible point $\boldsymbol{x}^* \in \mathcal{F}$ is called a local minimum if there exists an $\varepsilon > 0$ such that

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{N}_\varepsilon(\boldsymbol{x}^*) = \{\boldsymbol{x}^* \in \mathcal{F} | \parallel \boldsymbol{x}^* - \boldsymbol{x} \parallel \leq \varepsilon\}.$$

We say that the linearly independent constraint qualification (LICQ) holds at a point $\hat{\boldsymbol{x}} \in \mathcal{F}$ if the following set at point $\hat{\boldsymbol{x}}$ is linearly independent:

$$\{\nabla g_j(\hat{\boldsymbol{x}}) | j \in \bar{\mathcal{A}}(\hat{\boldsymbol{x}})\},$$

where $\nabla \boldsymbol{g}_j(\hat{\boldsymbol{x}})$ denotes the partial derivative (i.e., gradient) of $g_j(\boldsymbol{x})$ evaluated at $\boldsymbol{x} = \hat{\boldsymbol{x}}$. In addition, a point at which the LICQ holds is called a regular point.

Consider Problem ($\tilde{P}_1$). The Lagrangian is:

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i=1}^{m_1+m_2} \lambda_i g_i(\boldsymbol{x}),$$

where $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_{m_1+m_2}]^\top$ is the vector of Lagrange multipliers.

Let $\nabla \boldsymbol{h}(\boldsymbol{x})$ be the partial derivatives of function h with respect to $\boldsymbol{x}$ and let $\nabla^2 \boldsymbol{h}(\boldsymbol{x})$ be the second partial derivatives (Hessian matrix) of h with respect to $\boldsymbol{x}$, i.e.,

$$\nabla \boldsymbol{f}(\boldsymbol{x}) = [\frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \ldots, \frac{\partial f(\boldsymbol{x})}{\partial x_n}]^\top,$$

$$\nabla^2 \boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1^2} & \cdots & \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\boldsymbol{x})}{\partial x_2 \partial x_1} & \ddots & \frac{\partial^2 f(\boldsymbol{x})}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \vdots \\ \frac{\partial^2 f(\boldsymbol{x})}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f(\boldsymbol{x})}{\partial x_n \partial x_n} \end{bmatrix}.$$

Similarly, $\nabla \boldsymbol{L}(\boldsymbol{x})$ and $\nabla^2 \boldsymbol{L}(\boldsymbol{x}, \boldsymbol{\lambda})$ denote, respectively, the partial derivatives and the second partial derivatives (Hessian matrix) of L with respect to $\boldsymbol{x}$.

The first-order optimality conditions, called the Karush-Kuhn-Tucker (KKT) conditions, for local optimal solutions are given below.

Theorem 1.1. *(KKT conditions). Suppose that $\boldsymbol{x}^* \in \mathcal{F}$ is a local optimal solution of Problem ($\tilde{P}_1$) and that the LICQ holds at $\boldsymbol{x}^*$. Then, there exists a non trivial vector $\boldsymbol{\lambda}^*$ such that the following conditions are satisfied*

$$\begin{aligned} \nabla \boldsymbol{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) &= \boldsymbol{0}, \\ g_i(\boldsymbol{x}^*) &= 0, \quad i = 1, \ldots, m_1, \\ g_i(\boldsymbol{x}^*) &\leq 0, \quad i = m_1 + 1, \ldots, m_1 + m_2, \\ \lambda_i^* &\geq 0, \quad i = m_1 + 1, \ldots, m_1 + m_2, \\ \lambda_i^* g_i(\boldsymbol{x}^*) &= 0, \quad i = 1, \ldots, m_1 + m_2. \end{aligned}$$

For NLP problem, the global optimal solution is difficult to find, and hence the focus is on finding a local optimal solution. There are many methods available for solving various NLP problems. Examples include convex programming [71], separable programming [49], fractional programming [22], quadratic programming [23], and sequential quadratic programming [54]. In real word applications, the optimization problems are usually nonlinear, where both the objective function and constraint functions are nonlinear. For such problems, sequential quadratic programming (SQP) method is known to be effective. For more details on theory and computational algorithms, see, for example, [73, 104].

The main idea of SQP method is to solve a sequence of quadratic programming (QP)

subproblems, each of which is a quadratic model with quadratic objective function and linearized constraints. A QP subproblem is solved in each iteration step, giving rise to a search direction for the NLP for the current iterate $\boldsymbol{x}(k)$, where k denotes the kth iteration. More specifically, the objective function f is approximated by its local quadratic approximation

$$f(\boldsymbol{x}(k)+d(\boldsymbol{x}))\approx f(\boldsymbol{x}(k))+\nabla f(\boldsymbol{x}(k))d(\boldsymbol{x})+\frac{1}{2}d(\boldsymbol{x})^{\top}\nabla^2 L(\boldsymbol{x}(k),\boldsymbol{\lambda}(k))d(\boldsymbol{x}(k)),$$

where $d(\boldsymbol{x})=\boldsymbol{x}-\boldsymbol{x}(k)$, and the constraint functions g_i, $i=1,\ldots,m_1+m_2$, are approximated by their local affine approximations

$$g_i(\boldsymbol{x}(k)+d(\boldsymbol{x}))\approx g_i(\boldsymbol{x}(k))+\nabla g_i(\boldsymbol{x}(k))^{\top}d(\boldsymbol{x}),\quad i=1,\ldots,m_1+m_2.$$

Let $\boldsymbol{B}(k)=\nabla^2 L(\boldsymbol{x}(k),\boldsymbol{\lambda}(k))$. The QP subproblem is:

Problem ($\tilde{P}_2$).

$$\begin{aligned}
\min\quad & f(\boldsymbol{x}(k))+(\nabla f(\boldsymbol{x}(k),\boldsymbol{\lambda}(k)))^{\top}d(\boldsymbol{x})+\frac{1}{2}d(\boldsymbol{x})^{\top}\boldsymbol{B}(k)d(\boldsymbol{x}),\\
s.t.\quad & g_i(\boldsymbol{x}(k))+(\nabla g(\boldsymbol{x}(k)))^{\top}d(\boldsymbol{x})=0,\quad i=1,\ldots,m_1,\\
& g_i(\boldsymbol{x}(k))+(\nabla g(\boldsymbol{x}(k)))^{\top}d(\boldsymbol{x})\leq 0,\quad i=m_1+1,\ldots,m_1+m_2.
\end{aligned}$$

Problem ($\tilde{P}_2$) is solved as a quadratic programming problem with the active set strategy. For further details, see [62, 80].

To summarize, the SQP method is an iterative method for solving nonlinear optimization problems. It has been implemented in many software packages, such as NPSOL, NLPQL, OPSYC, OPTIMA, and SQP in MATLAB. It is important to note that the gradients of the cost function and constraint functions are essential information needed in the optimization process. Thus, this thesis pertains to the development of theory and methods for computing gradients of the cost function and the constraint functions. These gradients are then used in conjunction with the SQP iterative method for solving the optimization problems under condiserations.

1.3 Numerical techniques for optimal control problems

In engineering, a mathematical model is commonly used to describe the behavior of a dynamic process. This mathematical model is often expressed in terms of a system of

ordinary differential equations as given below:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t, \boldsymbol{x}(t), \boldsymbol{u}(t)), \quad t \in [0, T], \tag{1.1}$$

$$\boldsymbol{x}(0) = \boldsymbol{x}^0, \tag{1.2}$$

where $T > 0$ is the given terminal time of the time planning horizon $[0, T]$; $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector; $\boldsymbol{u} \in \mathbb{R}^r$ is the control vector; $\boldsymbol{f} \in \mathbb{R}^n$ is a given function describing the evolution of the states; and $\boldsymbol{x}^0$ is the initial state vector at the initial time $t = 0$. Note that here, the value of a function ω at time t is denoted by $\omega(t)$. The control can change its values from $t = 0$ to $t = T$. For a given control, the state evolves according to the system of ordinary differential equations (1.1) with initial condition (1.2) over the time planning horizon $[0, T]$.

In practice, the control strategy for (1.1)-(1.2) cannot be completely arbitrary, because the control strategy is limited by the capacity of the equipment used. For example, the feed flow rate into a 30 m^3 reactor tank must be less than or equal to $\frac{30}{T}$ m^3. In other words, the control is subject to physical limitations, often expressed mathematically as the following control restraint set:

$$\boldsymbol{U} = \{\boldsymbol{v} = [v_1, \ldots, v_r]^\top : a_i \leq v_i \leq b_i, i = 1, \ldots, r\},$$

where a_i and b_i, $i = 1, \ldots, r$ are given constants; and the superscript $\top$ denotes the transpose. A measurable function $\boldsymbol{u}$ such that $\boldsymbol{u}(t) \in \boldsymbol{U}$ for almost all $t \in [0, T]$ is called an admissible control. Let $\mathcal{U}$ be the set which consists of all such admissible controls. It is called the set of admissible controls.

Note that the state is influenced by the control through system (1.1). For an optimal control problem, it is required to choose an admissible control such that a performance measure, which could be energy consumption, wastage of consumable materials, etc., is optimized. This performance measure is also called an objective function or a cost function. In general, there are two terms in a cost function—a terminal cost and an integral cost. For example, in an evaporation process, the energy usage should be minimized while the solution level in each of the evaporators must be as close as possible to specific given values at the terminal time. Let x_j, $j = 1, \ldots, 7$, be the states describing the levels; and let $\hat{x}_j$, $j = 1, \ldots, 7$, be the target levels at terminal time. Furthermore, let u be the control representing the flow rate of the high temperature steam, and let W be a function of u and $\boldsymbol{x} = [x_1, \ldots, x_7]^\top$ representing the water evaporated from the process. A typical cost function of the form is given below:

$$J = \sum_{j=1}^{7} (x_j(T) - \hat{x}_j)^2 + \int_0^T \frac{u(t)^2}{W(u(t), \boldsymbol{x}(t))^2} dt. \tag{1.3}$$

The first term on the right hand side of (1.3) measures the differences between the real levels and the target levels at the terminal time. It is the terminal cost. The second term on the right hand side of (1.3) evaluates the energy usage during the whole period of the time horizon.

The cost function given by (1.3) is a special case of a general cost function given as follows:

$$J(\boldsymbol{u}) = \Phi(\boldsymbol{x}(T)) + \int_0^T \mathcal{L}(t, \boldsymbol{x}(t), \boldsymbol{u}(t))dt, \tag{1.4}$$

where $\Phi : \mathbb{R}^n \to \mathbb{R}$ is given function measuring the terminal cost, and $\mathcal{L} : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}$ measures the cost during the whole time horizon. An admissible control which minimizes the cost function (1.4) is called an optimal control. Note that the cost function J depends entirely on $\boldsymbol{u}$, as $\boldsymbol{x}$ is implicitly determined by $\boldsymbol{u}$ through the system of ordinary differential equations (1.1)-(1.2).

We may now state formally a simple optimal control problem in the following.

Problem ($\tilde{P}_3$). *Given the dynamic system* (1.1)-(1.2)*, find a control* $\boldsymbol{u} \in \mathcal{U}$ *such that the cost function* (1.4) *is minimized.*

Problem ($\tilde{P}_3$) is called a Bolza problem. It can, in principle, be solved by using Pontryagin Minimum Principle or Bellman's Principle of Optimality.

Let us look at how the Pontryagin Minimum Principle is used to solve Problem ($\tilde{P}_3$). For this, we introduce the Hamiltonian function for Problem ($\tilde{P}_3$) given below:

$$H(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\lambda}(t)) = \mathcal{L}(t, \boldsymbol{x}(t), \boldsymbol{u}(t)) + \boldsymbol{\lambda}^\top(t)\boldsymbol{f}(t, \boldsymbol{x}(t), \boldsymbol{u}(t)), \tag{1.5}$$

where $\boldsymbol{\lambda}$ is called the co-state, which satisfies the following system of differential equations:

$$\dot{\boldsymbol{\lambda}}(t) = -\Big[\frac{\partial H(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\lambda}(t))}{\partial \boldsymbol{x}}\Big]^\top, \quad t \in [0, T], \tag{1.6}$$

with boundary condition

$$\boldsymbol{\lambda}(T) = \Big[\frac{\partial \Phi(\boldsymbol{x}(T))}{\partial \boldsymbol{x}}\Big]^\top. \tag{1.7}$$

System (1.6)-(1.7) is called the co-state system. If $\boldsymbol{u}^*$ is an optimal control, and $\boldsymbol{x}^*$ and $\boldsymbol{\lambda}^*$ are the corresponding state and co-state, respectively, then it can be shown [110] that

$$H(t, \boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{\lambda}^*(t)) = \min_{\boldsymbol{v} \in U} H(t, \boldsymbol{x}^*(t), \boldsymbol{v}, \boldsymbol{\lambda}^*(t)), \tag{1.8}$$

for all $t \in [0, T]$, except possibly on a finite subset of $[0, T]$. This condition is known as the Pontryagin Minimum Principle. By solving the Pontryagin Minimum Principle, the optimal control can, in principle, be obtained as a function of time, state, and co-state. If

such a control is obtained, we can substitute it into the state system (1.1)-(1.2) and the co-state system (1.6)-(1.7), yielding a two-point boundary-value (TPBV) problem. The optimal control can be obtained through solving this TPBV problem. This is, however, a very difficult task (even solving it numerically is difficult, let alone analytically).

We now look at Bellman's Principle of Optimality. Let the system (1.1) be evolved starting at time point $t \in [0,T]$ from a given state $\boldsymbol{x}$, and let the corresponding solution of the system be denoted as $\boldsymbol{y}(s|t,\boldsymbol{x})$, where $s \in [t,T]$. Then, define the value function $V : [0,T] \times \mathbb{R}^n \to \mathbb{R}$ as follows:

$$V(t,\boldsymbol{x}) = \inf_{\boldsymbol{u}\in\mathcal{U}} \Big\{ \Phi(\boldsymbol{y}(T|t,\boldsymbol{x})) + \int_t^T \mathcal{L}(s,\boldsymbol{y}(s|t,\boldsymbol{x}),\boldsymbol{u}(s))ds \Big\}, \quad (t,\boldsymbol{x}) \in [0,T] \times \mathbb{R}^n.$$

By Bellman's Principle of Optimality, it can be shown that the following partial differential equation is satisfied:

$$\frac{\partial V(t,\boldsymbol{x})}{\partial t} + \inf_{\boldsymbol{v}\in U} \Big\{ \frac{\partial V(t,\boldsymbol{x})}{\partial \boldsymbol{x}} \boldsymbol{f}(t,\boldsymbol{x}(t),\boldsymbol{v}) + \mathcal{L}(t,\boldsymbol{x}(t),\boldsymbol{v}) \Big\} = 0, \quad (t,\boldsymbol{x}) \in [0,T] \times \mathbb{R}^n, \quad (1.9)$$

with the boundary condition

$$V(T,\boldsymbol{x}) = \Phi(\boldsymbol{x}(T)), \quad \boldsymbol{x} \in \mathbb{R}^n. \quad (1.10)$$

Equation (1.9) is known as the Hamilton-Jacobi-Bellman (HJB) equation. The solution of the HJB equation (1.9) with boundary condition (1.10) can be used to construct an optimal feedback control to Problem ($\tilde{P}_3$). However, the HJB equation can only be solved analytically for problems involving linear dynamics and quadratic cost function. Furthermore, to solve it numerically, the dimension of the problem must be small, because the numerical solution of HJB equation is computationally very demanding.

In view of the difficulties involved in the direct application of the Pontryagin Minimum Principle and Bellman's Principle of Optimality, even for a simple class of optimal control problems, it is inevitable to rely on computational algorithms to solve optimal control problems, especially for real world problems which are often very complicated and subject to various constraints arising from engineering limitations and design specifications. Consequently, numerous families of computational approaches have been developed, such as the direct collocation approach [30, 75], the iterative dynamic programming technique [130], the leap-frog algorithm [36], the switching time computation method [35], the sequential gradient-restoration methods [135], the multiple shooting methods [44,59], and the control parameterization methods [80].

The direct collocation approach is to approximate a constrained optimal control problem by a finite dimensional nonlinearly constrained optimization problem (NLP), where the entire time horizon is divided into a finite number of subintervals. In each subinter-

val, the controls are approximated by a continuous and piecewise linear function, and the state variables are approximated by a continuously differentiable piecewise cubic function. Clearly, the dimension of the discretised problem is directly related to the number of partition points of the time horizon. A sequence of refinement steps is applied resulting in a sequence of NLPs of increasing size. Each of these NLPs is solved by standard sequential quadratic programming (SQP) methods (see, for example, [29, 48]). This approach (see, for example, [114] and [113]) is applicable to solve optimal control problems with nonlinear dynamical equations subject to nonlinear constraints. Some convergence properties are derived in [138], which are further used to obtain reliable estimates of the co-state variables. Various applications of the method are demonstrated in [56, 102]. Although it can readily solve small scale problems, the discretization of both control and state variables in the direct collocation approach can lead to excessive computation cost for large scale optimal control problems, especially if a reasonably accuracy is required to be met.

The iterative dynamic programming (IDP) technique is derived in [130] based on the Dynamic Programming Principle. It is refined in [128] and [96] to improve the efficiency of the computational procedure. This technique constructs a grid structure for the discretization of both the state and the control. The grid of the state defines accessible points in the state trajectory, while the grid of the control defines admissible control values. The grids are refined iteratively until a satisfactory control policy is obtained. Initial development employed piecewise-constant controls and this was later extended to piecewise linear control policies [129]. Constraints are handled by using a penalty function approach to incorporate them into the objective function. The IDP technique has been successfully applied to a wide range of optimal control problems in chemical engineering, see, for example, [12, 154]. However, as mentioned in [129], there exist many algorithmic parameters, which include the region contraction factor, the number of allowable values for each control variable, the number of grid points, the initial region size and the restoration factor. Proper determination of these parameters is not an easy task.

The leap-frog algorithm is initially developed in [36], where it is used to solve a special type of Two Point Boundary Value Problem arising in geodesics. In [108] and [37], the algorithm is further developed and implemented to handle general nonlinear systems with unbounded and bounded controls. A description of the algorithm is presented in [34], while some theoretical analysis of the algorithm is presented in [37] for a class of optimal control problems with bounded controls in the plane.

The switching time computation (STC) method, proposed in [35], is a computational procedure to find optimal locations of switching points for single-input nonlinear systems. A concatenation of constant-input arcs is used to take the system from a given initial point to the target. In [137] and [39], the STC method is used in the development of a time optimal bang-bang control algorithm. However, this approach is rather restrictive. It is not directly applicable to many types of constraints which appear in practice.

The sequential gradient-restoration algorithms [5, 6] are applicable to optimal control problems involving differentiable constraints, non-differentiable constraints, and terminal constraints. This family of algorithms involves a sequence of two-phase cycles, where each cycle includes a gradient phase and a restoration phase. There is a function, called the augmented function, that consists of the original cost function and the constraints violations. In the gradient phase, the value of the augmented function is decreased; in the restoration phase, the constraint error is decreased, while avoiding excessive change in the value of the cost function. In the complete gradient-restoration cycle, the value of the cost function is decreased, while the constraints are satisfied to a predetermined accuracy. Hence a succession of suboptimal solutions is obtained. It is further enhanced by the dual version [7] of the algorithm.

The multiple shooting approach is proposed in [59]. It divides the time horizon into many subintervals. Then, at each subinterval, the shooting method is used to solve the co-state dynamic system based on an initial guess of the solution of the co-state system. Re-estimating the co-states is continued based on the mismatches until the conditions of the minimum principle are satisfied. This approach is rather sensitive [142] to the initial guess of the co-states at the initial time point.

For the control parameterization method, it is done by partitioning the time horizon of an optimal control problem into several subintervals such that each control can be approximated by a piecewise-constant function (or piecewise linear function or piecewise smooth function) which is consistent with the corresponding partition. The partition points are often referred to as control switching times. The heights of the approximating piecewise-constant function are decision variables, known as control parameters. For the continuous inequality constraints on the state and/or control, the constraint transcription method was first proposed in [82] to approximate the continuous inequality constraints on the state by constraints in integral form, called constraints in canonical form. The constraint transcription method is later extended in [80] to handle continuous inequality constraints on the state and control. Thus, by using the control parameterization method together with the constraint transcription technique, an optimal control problem subject to continuous inequality constraints on the state and/or control is approximated by a sequence of optimal parameter selection problems subject to canonical constraints, where the cost function is to be minimized with respect to the control parameters subject to the constraints being satisfied. Each of the resulting optimal parameter selection problems can be regarded as a mathematical programming problem solvable by gradient-based optimization techniques, and hence many existing optimization software packages can be readily used. The control parameterization technique is used extensively in the literature (see [45, 93, 149]). In [146], a survey on developments of the technique is presented. It is observed that the technique is applicable to a wide range of optimal control problems. In particular, several computational algorithms supported by sound theoretical convergence

analysis are presented in [127] and [16] for dealing with a variety of different classes of optimal control problems.

It is intuitively clear that the switching times of the control should also be taken as decision variables. However, it is numerically sensitive if the gradients of the cost function and the canonical constraints with respect to these switching times are used in the optimization process (see Chapter 5 of [80]). Thus, a time-scaling transform is introduced in [66,81], where it is called the control parameterization enhancing transform (CPET), to map these switching times to fixed knot points in a new time horizon via introducing a new control variable, called the time-scaling control, and an additional differential equation describing the relationship between the original time variable and the new one. Thus, under the time-scaling transform, the optimal control problem subject to continuous inequality constraints on the state and/or control is also approximated by a sequence of optimal parameter selection problems subject to canonical constraints. Each of these problems can be solved as a nonlinear mathematical programming problem.

The optimal control software package MISER [91] is an implementation of algorithms based on the control parameterization technique [80] and the constraint transcription method [89]. It can be used in conjunction with the time-scaling transform [81]. NUDOC-CCS [19] is another optimal control software based on the control parameterization approach. It is used for both the simulation and the optimization of dynamical systems. While it lacks some of the flexibility of the MISER 3.3 package, it has some additional features such as an adaptive grid refinement strategy and efficient posterior sensitivity analysis.

1.4 Parameter identification for time-delay systems

For a given practical process, it is required to construct a mathematical model to describe the interactions between various factors so as to behave as a whole unit. In practice, there are delay effects on the process of the system. For example, in an imperfectly mixed system [63,64], it takes some time for the changes of mass and energy in particular parts of the vessel to reach the rest of the vessel. Time-delays do arise in many real world situations, including chemical tank reactors [86], aerospace engineering [143], chromatography [107], and power converters and batteries [153]. For such systems, the mathematical model is often expressed as a system of ordinary differential equations involving time-delay arguments.

A typical nonlinear time-delay system is given below:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}\big(t, \boldsymbol{x}(t), \boldsymbol{x}(t-\alpha_1), \ldots, \boldsymbol{x}(t-\alpha_m), \boldsymbol{u}(t), \boldsymbol{u}(t-\beta_1), \ldots, \boldsymbol{u}(t-\beta_p), \boldsymbol{\zeta}\big), \quad t \in (0, T], \tag{1.11}$$

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \leq 0, \tag{1.12}$$

where $T > 0$ is the terminal time; $\boldsymbol{\zeta} \in \mathbb{R}^q$ is a vector of system parameters; $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state of the system; $\boldsymbol{x}(t-\alpha_i) \in \mathbb{R}^n$, $i = 1, \ldots, m$, are delayed states, meaning that if $\boldsymbol{x}(t)$ is the value of $\boldsymbol{x}$ at the time point t, then $\boldsymbol{x}(t-\alpha_i)$ denotes the value of $\boldsymbol{x}$ at the time point $t - \alpha_i$; $\boldsymbol{u}(t) \in \mathbb{R}^r$ is the control; $\boldsymbol{u}(t-\beta_k) \in \mathbb{R}^r$, $k = 1, \ldots, p$, are the delayed controls. Let all the state time-delays and all the control time-delays be referred to collectively as $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_m]^\top$ and $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_p]^\top$, respectively. Furthermore, $\boldsymbol{f} : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{nm} \times \mathbb{R}^r \times \mathbb{R}^{pr} \times \mathbb{R}^q \to \mathbb{R}^n$ is a given function, and $\boldsymbol{\phi} : \mathbb{R} \to \mathbb{R}^n$ is a given function representing the state before the time $t = 0$. In practice, the state of the system may not be available for observation. Instead, what we can observe is some function of the state variables given by

$$\boldsymbol{y}(t) = \Psi(\boldsymbol{x}(t)), \tag{1.13}$$

where $\boldsymbol{y}(t) \in \mathbb{R}^m$ with $m < n$.

In a practical process, the system parameters and the time-delays are often not known. However, once suitable values for the system parameters and time-delays are chosen, the time-delay system (1.11) can be solved and hence the corresponding output can be computed from the output system (1.13). On the other hand, the output of the real process can be measured at certain time points over the time horizon. The sum of the differences between the computed output and real output at these time points is called the *mismatch*. To identify the system parameters and time-delays, it amounts to minimize the mismatch with respect to the system parameters and time-delays. This is, in fact, an optimal parameter selection problem.

We now return to the time-delay system (1.11)-(1.12). Once the system parameters and the time-delays are identified, then the mathematical model is completely specified. It is then possible to construct optimal control algorithms based on the control parameterization method to synthesize an optimal control for the system (e.g. [79, 94]). The problem of identifying the system parameters and the time-delays based on a given time series data is a key problem in the study of time-delay systems [74]. Such problems are known as *parameter identification problems*.

There are many results available in the literature pertaining to parameter identification problems. An exact least squares algorithm for single time-delay estimation is studied in [116]. Algebraic techniques [85] and the steepest descent algorithm [134] are proposed to determine the input delays. In [95], information theory is used to identify multiple time-

delays from a time series. In [95], it is assumed that each nonlinear term in the dynamical system contains at most one unknown delay. In [57], a genetic algorithm is utilized for the identification of multiple time-delays. However, this method depends critically on the initial guess of the parameters that are to be identified. Lyapunov functions are used to design delay estimators in [141], where it is shown that for an appropriately chosen guess for each of the delays, the approximate delays will converge to the real ones after a finite number of iterations. In [125], a class of systems is considered, where the system dynamics are expressed as the sum of a finite number of nonlinear terms and each nonlinear term involves only one unknown state delay. There is no unknown system parameters involved. The gradient formula of the objective function with respect to the time-delays is derived. This gradient formula is expressed in terms of the solutions of the dynamical system and an auxiliary delay-differential system, both of which are to be solved forward in time over the time horizon. With this gradient formula, existing gradient-based optimization techniques can be incorporated to solve the parameter identification problem for this time-delay system.

1.5 Control methods for time-delay systems

1.5.1 Optimal control for time-delay systems

Consider the process evolving according to system (1.11)-(1.12) over the time horizon $[0, T]$, where system parameters $\boldsymbol{\zeta}$ and time-delays $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are assumed to be given. Let $\boldsymbol{U}$ be a given compact and convex subset of $\mathbb{R}^r$, and let $\boldsymbol{\gamma} : [-\beta, 0) \to \mathbb{R}^r$. A function $\boldsymbol{u} : [-\beta, T] \to \mathbb{R}^r$ such that $\boldsymbol{u}(t) = \boldsymbol{\gamma}(t)$ on $[-\beta, 0)$ and $\boldsymbol{u}(t) \in \boldsymbol{U}$ for almost all $t \in [0, T]$ is called an admissible control. Let $\mathcal{U}$ be the class of all such admissible controls. A simple optimal control problem for time-delay systems may now be stated as follows:

Problem ($\tilde{P}_4$). *Consider system* (1.11)-(1.12)*, find a control* $\boldsymbol{u} \in \mathcal{U}$ *such that the cost function* (1.4) *is minimized.*

1.5.2 Model predictive control for time-delay systems

Model predictive control (MPC) is also called the receding horizon control. It was first introduced in 1960s [50], and has since become popular in areas such as chemical processes and paper industries. This is because the MPC algorithm is simple and intuitive. The main structure of MPC is show in Figure 1.1. In Figure 1.1, T_s is the sample time; t_k is the present time; $n_c T_s$ and $n_p T_s$ are, respectively, the length of the control horizon and the prediction horizon, where $n_c \leq n_p$ are positive integers. We can see that, in each sample time preriod, MPC involves the prediction of the system output over a finite prediction period $[t_k, t_k + n_p T_s]$ by using process model based on empirical data fitting

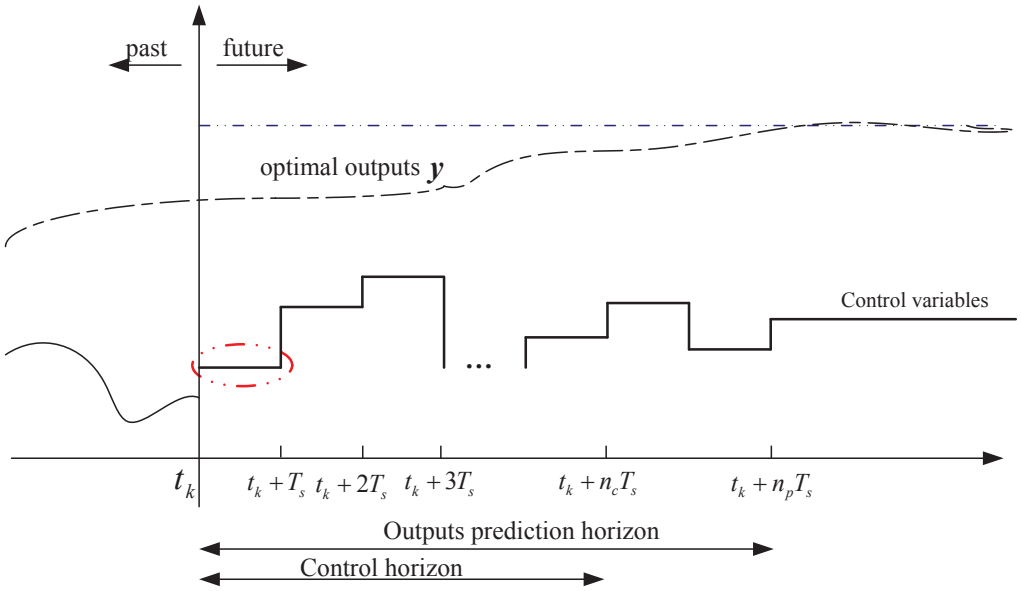

Figure 1.1: Structure of MPC

or dynamic model based on fundamental mass and energy balances. The optimal control is obtained through solving the following open-loop constrained optimal control problem over the prediction period $[t_k, t_k + n_p T_s]$.

Problem ($\tilde{P}_5$). *Consider system* (1.11) *over the time horizon* $[t_k, t_k + n_p T_s]$ *with given system parameters* $\boldsymbol{\zeta}$ *and time-delays* $\boldsymbol{\alpha}$ *and* $\boldsymbol{\beta}$*, the following optimal control problem is minimized*

$$\begin{aligned}
\min \quad & J_{\boldsymbol{u}(t_k),\ldots,\boldsymbol{u}(t_k+n_c T_s)} = \sum_{i=1}^{n_p} \left\{ \boldsymbol{y}(t_k + iT_s) - \boldsymbol{y}^* \right\}^2 + \sum_{i=1}^{n_c} \Delta \boldsymbol{u}(t_k + iT_s)^\top \boldsymbol{R} \Delta \boldsymbol{u}(t_k + iT_s) \\
s.t. \quad & \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t, \boldsymbol{x}(t), \boldsymbol{u}(t)), \quad t \in [t_k, t_k + n_p T_s], \\
& \underline{\boldsymbol{\alpha}} \leq \Delta \boldsymbol{u}(t) \leq \bar{\boldsymbol{\alpha}}, \\
& \underline{\boldsymbol{\beta}} \leq \boldsymbol{u}(t) \leq \bar{\boldsymbol{\beta}}, \\
& \boldsymbol{g}(t, \boldsymbol{x}, \boldsymbol{u}) \leqq 0,
\end{aligned}$$

with initial conditions

$$\boldsymbol{u}(t) = \boldsymbol{\gamma}(t), \qquad \boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \leq t_k,$$

and the output function

$$\boldsymbol{y}(t) = \boldsymbol{\Psi}(\boldsymbol{x}(t)), \quad t \in [t_k, t_k + n_p T_s],$$

where $\boldsymbol{x} = [x_1, \ldots, x_n]^\top \in \mathbb{R}^n$ is the state vector and $\boldsymbol{\phi} = [\phi_1, \ldots, \phi_n]^\top \in \mathbb{R}^n$ is its

initial condition; $\boldsymbol{u} = [u_1, \ldots, u_r]^\top \in \mathbb{R}^r$ is the control vector and $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_r]^\top \in \mathbb{R}^r$ is its initial condition; $\boldsymbol{f} = [f_1, \ldots, f_n]^\top \in \mathbb{R}^n$ is the dynamic system function; $\Delta\boldsymbol{u} = [\Delta u_1, \ldots, \Delta u_r]^\top \in \mathbb{R}^r$ is the change rate of the control at time $t_k + iT_s$, $i = 1, \ldots, n_c$, with $\Delta\boldsymbol{u}(t_k + iT_s) = \boldsymbol{u}(t_k + iT_s) - \boldsymbol{u}(t_k + (i-1)T_s)$; $\boldsymbol{R} \in \mathbb{R}^{rr}$ is a given weighting matrix; $\boldsymbol{g} = [g_1, \ldots, g_{n_g}]^\top \in \mathbb{R}^{n_g}$ denotes the constraint function; $\underline{\boldsymbol{\beta}} = [\underline{\beta}_1, \ldots, \underline{\beta}_r]^\top \in \mathbb{R}^r$ and $\bar{\boldsymbol{\beta}} = [\bar{\beta}_1, \ldots, \bar{\beta}_r]^\top \in \mathbb{R}^r$ are the lower and upper bounds of the control, respectively; $\underline{\boldsymbol{\alpha}} = [\underline{\alpha}_1, \ldots, \underline{\alpha}_r]^\top \in \mathbb{R}^r$ and $\bar{\boldsymbol{\alpha}} = [\bar{\alpha}_1, \ldots, \bar{\alpha}_r]^\top \in \mathbb{R}^r$ are the lower and upper bounds of the change rate of the control, respectively; $\boldsymbol{y} = [y_1, \ldots, y_m]^\top \in \mathbb{R}^m$ is the system output; $\boldsymbol{\Psi} = [\Psi_1, \ldots, \Psi_m]^\top \in \mathbb{R}^m$ is the given output function; $\boldsymbol{y}^* = [y_1^*, \ldots, y_m^*]^\top \in \mathbb{R}^m$ is the given reference trajectory. Moreover, $\boldsymbol{u}(t) = \boldsymbol{u}(t_k + n_c T_s)$ if $t > t_k + n_c T_s$.

Problem $(\tilde{P}_5)$ is solvable by gradient based algorithms. Note that for MPC, only the control adjustments for the next instant $t_k + T_s$ of the optimal control over $[t_k, t_k + T_s]$ is used. Then, the time is moved forwards to $t_k := t_k + T_s$ and a new optimal control problem in the form of Problem $(\tilde{P}_5)$ is solved for subsequent sampling periods. This process is repeated.

Amongst the existing literature, the MPC can be divided into two categories: (i) Linear MPC; and (ii) nonlinear MPC. For the linear MPC, it involves solving a sequence of open-loop optimal control problems, each of which contains a linear system dynamics, a quadratic cost function and linear constraints over a future time horizon. It has been widely used in practice (see, for example, [1, 68]). In the linear MPC, the resulting optimal control problems can be solved as quadratic programming problems involving linear dynamical systems. Since the control to be applied over the next instant is to be calculated online, the MPC can only be used for systems with slow dynamics and long sampling time. The linear MPC is extended to nonlinear systems in [15], where the process model is approximately linear over a small operating range. Then, the linearization of the nonlinear process model in a small operating range can be carried out. However, it achieves poor performance when the process model could not be approximated accurately by linear model.

This problem is overcome in the nonlinear model predictive control (NMPC) [136]. The NMPC is characterized by nonlinear system model, nonlinear constraints, and non-quadratic cost function. The NMPC techniques have been successfully implemented in many large scale industrial processes since 1980s (see, for example, [31, 61, 139] for time-delay industrial processes). However, it may not yet be suitable for on-line implementation. To be more specific, let us mention an example, where the NMPC is applied to an evaporation process in [61]. In this example, it takes nearly one hour of computational time to compute the control by using the NMPC on a workstation with dual Pentium III Xeon processors for one hour of simulation of the evaporation process using the control obtained. The computational time grows as the prediction horizon is increased. Since many practical processes have large time-delays [32, 61], a short prediction horizon may

not be able to capture the effect on the changes of the state for such processes. Thus, a long prediction horizon is required, but then the consequent computational burden will be much increased so that it will become impractical for on-line implementation. Other studies also show that for optimal control problems involving nonlinear dynamics with high dimension and state constraints, especially continuous state constraints, NMPC requires a large computation time in each sampling period. Indeed, the computation time needed for NMPC is much more than that of the linear MPC. The high computational burden has limited its application to optimal control problems involving nonlinear dynamics and subject to constraints on the state and control [144].

To overcome the drawback of requiring a large computation time, many improved methods have been proposed. The most important one is the linear matrix inequalities based MPC [106,158]. As the linear matrix inequalities based optimization can be solved in polynomial time, the computation time for MPC is reduced. However, this approach is only applicable to problems involving linear dynamics and linear constraints, while the cost function is quadratic. A decreasing horizon MPC is introduced in [87]. As the control horizon decreases from one iteration to the next, the computation time will clearly be decreased. In [17], the off-time calculation is carried on a discrete set of constraints and terminal costs. The results are then utilized in the implementation of the MPC. However, this method can only be used for systems with lower dimension. Some on-line optimization methods, such as swarm-staring method and grid method, have been proposed to speed-up the computation time for MPC [156]. However, they are yet to be realized in practice.

1.6 Max-Min optimal control problems

A max-min optimal control problem involves finding a control such that the minimum of a set of objectives is maximized. Thus, its cost function is a max-min function which is non-smooth and non-differentiable. This problem has extensive real world applications in engineering optimality design [115], electronic circuit design [21], robust control design [4], economics [18], and the staffing problems in call centers [9]. Let us look at an example arising in the study of a separation process, where several different kinds of products are drained from a single outlet. It is required to ensure the purification of each product. For this, it is required that one product is totally drained at one time. Thus, the duration time between two successive outlet times should be maximized. However, large duration times will increase the total operation time and decrease the productivity. In practice, the total operation time should be as short as possible. Thus, the optimal control problem for this separation process can be formulated as: find a control such that the minimum ratio of the duration time between each pair of two successive outlet times to the total operation time is maximized subject to some constraints due to engineering specifications.

As solving a max-min optimal control problem basically involves solving a sequence of approximate max-min optimization problems, we will mention some approaches proposed for solving max-min optimization problems. One way of solving a max-min optimization problem is to view the max-min objective as multiple objectives. Then, the aim is to find an efficient solution over a set of possible solutions, which is called the Pareto set. Thus, it is required to first find the Pareto solution set. Then, a sorting procedure is applied to evaluate the solutions. The Pareto solutions could, in principle, be found by Pareto-set based optimization methods, such as genetic algorithm [3], evolution algorithm [4, 10] and particle warm optimization algorithm [155]. However, these methods are heuristic methods.

It is known that max-min problems and min-max problems are equivalent. For closely related minmax problems, smoothing techniques are proposed to convert min-max problems into simple, smooth, unconstrained or constrained optimization problems. Then, standard unconstrained or constrained minimization techniques can be used to solve these converted problems. More specifically, the smoothing technique [53] is used to approximate a max function by a smooth function, while the maximum entropy function is used in [72] to approximate the inner maximum objective function by a continuous smooth function. However, if the accuracy requirement on the approximation is high, then the smooth approximating problems will become ill-conditioned. Hence, when applied to these problems, the unconstrained optimization techniques may experience numerical difficulties, leading to slow convergence—and in some cases no convergence at all. In addition to this, two distinct search directions based algorithms are proposed in [21] to solve min-max problems directly, while interior-point method is used in [43, 52].

For max-min optimal control problems, we wish to mention the following two methods. For the first method reported in [8], the max-min optimal control problem is first converted to min-max optimal control problem, then a smoothing technique is used to transform the min-max optimal control problem into a standard optimal control problem in the form of Bolza with additional inequality constraints on the state/or control variables. This standard optimal control problem can be solved by existing methods and theories. This idea has been widely adopted, see, for example, [16, 88, 97]. For another method reported in [90], the parametrization technique is used to approximate the max-min optimal control problem by a sequence of max-min optimal parameter selection problems. Then, each of these approximate problems is shown to be equivalent to a standard min-max optimization problem. Hence, it is solvable by existing optimization software for solving minmax optimization problems, such as FFSQP 3.7 [76] and CONSOL-OPTCAD [99].

1.7 Overview of the thesis

In previous sections, we presented brief surveys on computational methods for solving optimal control problems, time-delay optimal control problems and maxmin optimal control problems. Furthermore, a brief introduction to MPC for time-delay optimal control problems is also given.

The purpose of this thesis is to present new computational methods for several classes of practical optimal control problems. They are briefly mentioned below.

In Chapter 2, we consider a general nonlinear time-delay system described by (1.11)-(1.12), where not only the unknown system parameters, but also the unknown time-delays need to be identified. Furthermore, the input is smooth function. Let $\boldsymbol{\alpha} \in \mathcal{T}$, $\boldsymbol{\beta} \in \mathcal{B}$, and $\boldsymbol{\zeta} \in \mathcal{Z}$ denote, respectively, the state time-delays, input time-delays and system parameters, where $\mathcal{T}$, $\mathcal{B}$ and $\mathcal{Z}$ are the set of candidate parameter vectors for the corresponding variables. Our parameter identification problem for time-delay system can be described as follows:

Problem ($\tilde{P}_6$). *Choose $\boldsymbol{\alpha} \in \mathcal{T}$, $\boldsymbol{\beta} \in \mathcal{B}$, and $\boldsymbol{\zeta} \in \mathcal{Z}$ such that the cost function*

$$J(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\zeta}) = \sum_{l=1}^{M} \left| \boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\beta}, \boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l \right|^2. \tag{1.14}$$

is minimized, where $|\cdot|$ denotes the usual Euclidean norm. Here, $\hat{\boldsymbol{y}}^l$ denotes the value of the output function (1.13) *at the sample time t_l, $l = 1, \ldots, M$. The cost function measures the discrepancy between predicted and observed system output.*

We develop a unified gradient-based computational approach that involves solving Problem ($\tilde{P}_6$). Since the delays and parameters influence the cost function *implicitly* through the dynamic system, in this computational method, the gradients of the cost function with respect to the delays and system parameters are derived. They are obtained by solving a set of auxiliary delay-differential systems from $t = 0$ to $t = T$. Then, the delays and parameters are determined simultaneously through solving a dynamic optimization problem using existing optimization techniques. Two nonlinear parameter identification problems involving time-delay systems are solved by using the method proposed. From numerical simulations, it is clearly indicated that this algorithm is effective.

In Chapter 3, we consider a more difficult time-delay identification problem, where the input function of the nonlinear time-delay system is piecewise-constant. The main difficulties with this problem are: i) Since the input function is discontinuous, the dynamics are clearly discontinuous with respect to the input delay. Thus, the results obtained in Chapter 2 cannot be used to determine the state variation with respect to input time-delay; and ii) Problem ($\tilde{P}_6$) in this case can be restated as a switched system optimal control problem. Unfortunately, the well-known *time-scaling transform* technique for solving optimal

control problems involving switched systems (see [118, 124, 132, 151]) is not applicable to time-delay systems such as system (1.11)-(1.12) defined above (see the discussion in [33]). Thus, we propose a new computational approach, which is based on a novel derivation of the cost function's gradient. We then apply this approach to estimate the time-delays in two industrial chemical processes—a zinc sulphate purification process and a sodium aluminate evaporation process. Numerical simulations demonstrate the effectiveness of this algorithm.

In Chapter 4, we consider a general class of optimal control problems for time-delay systems subject to continuous inequality constraints on states and controls. In other words, these constraints must be satisfied for each $t \in [0, T]$. This problem can be stated as follows.

Problem ($\tilde{P}_7$). *Given system* (1.11)-(1.12) *with given* $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\zeta}$, *find a control* $\boldsymbol{u} \in \mathcal{U}$ *such that the cost function* (1.4) *is minimized subject to the following constraints:*

$$g_i(t, \boldsymbol{x}(t), \boldsymbol{u}(t)) \leq 0, \quad i = 1, \ldots, N_c, \quad \forall t \in [0, T].$$

An efficient gradient-based computational method is devised for solving this optimal control problem. In this method, the control parameterization technique is used to approximate the control by a piecewise-constant function (it could also be approximated by a piecewise linear or piecewise smooth function) with possible discontinuities at the $N - 1$ pre-assigned partition points. The heights of the piecewise-constant function are regarded as decision variables, which are referred to collectively as the control parameter vector. Then, the constrained time-delay optimal control problem is approximated by a sequence of optimal parameter selection problems involving time-delay dynamical system and subject to continuous inequality constraints on the state and boundedness constraints on the control parameter vector. For the continuous inequality constraints, they are transformed, by using the constraint transcription method, into equivalent equality constraints in integral form. However, the integrands of these equality constraints are nonlinear and nonsmooth. Thus, a local smoothing technique is used to approximate these nonsmooth integrands by smooth functions. Then, the equality constraints in integral form are approximated by inequality constraints in integral form, where their integrands are approximating smooth functions. There are two parameters involved in the inequality constraints in integral form—one controls the accuracy of the approximation and the other controls the feasibility of the original constraints satisfaction. Now, by using penalty function ideas, the summation of these inequality constraints in integral form is appended to the cost function to form an augmented cost function. In this way, the constrained time-delay optimal control problem is approximated by a sequence of optimal parameter selection problems involving time-delay dynamical system and subject to only boundedness constraints on the control parameter vector. The gradient of the

augmented cost function with respect to the control parameter vector is derived. On this basis, an effective gradient-based optimization method is developed for solving each of these optimal parameter selection problems with simple boundedness constraints on the control parameter vector. The optimal control parameter vector obtained can then be used to construct a piecewise-constant control for the original constrained time-delay optimal control problem. Supporting convergence results are established. In particular, it is shown that when the penalty factor is sufficiently large, then the optimal piecewise-constant control obtained will satisfy the continuous inequality constraints of the original problem. Furthermore, when the number of partition points is increased, the cost corresponding to the optimal piecewise-constant control will converge to the true optimal cost. The computational method proposed is applied to an optimal control problem for an industrial-scale evaporation process with long time-delays, where the mass units of live steam consumption used for evaporating one unit of water is minimized subject to the requirements for the product concentrations and the levels in the evaporators being satisfied.

In Chapter 5, we consider the following set of differential equations:

$$\begin{aligned} \dot{x}_1(t) &= f_1(t, \boldsymbol{x}(t), \boldsymbol{u}(t)), \quad t \in [0, t_1], \\ &\vdots \\ \dot{x}_n(t) &= f_n(t, \boldsymbol{x}(t), \boldsymbol{u}(t)), \quad t \in [0, t_n]. \end{aligned}$$

where $t_i > 0$, $i = 1, \ldots, n$, are the unknown final times for the ith state dynamic function, respectively; and if $i \neq j$, then $t_i \neq t_j$.

This problem arises in the study of gradient elution chromatography, which is used to separate different kinds of components in a solution. A typical chromatography system consists of a column containing an absorbent (called the stationary phase) and a liquid that flows through the column (called the mobile phase). The mixture to be separated is injected into the mobile phase and flows through the column. Since different components are attracted to the adsorbent at different grades, they exit the gradient elution column at different times. The main concern of this process is to maximize separation efficiency so as to ensure the purification of each component. Thus, our goal is to find a control such that the minimum duration between successive final times is maximized. This max-min problem has three non-standard characteristics: (i) The objective function is non-smooth; (ii) each state variable is defined over a different time horizon; and (iii) the ordering of the final times is unknown. In this max-min optimal control problem, there are multiple characteristics times, max-min objective function, and binary decision variables. We will propose an efficient gradient-based computational method to solve this complicated optimal control problem. To demonstrate the effectiveness of the computational method proposed, two numerical examples are solved. One of these two examples is an optimal

control problem involving a real chromatography process. The results obtained are highly promising.

Finally, in Chapter 6, we summarize the main contributions of the thesis and discuss some interesting directions for future research.

CHAPTER 2

Identification of time-delays and parameters for nonlinear systems

2.1 Introduction

In this chapter, 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-square error between predicted output and observed system output. This type of parameter identification problem is previously considered in [125] for a simple class of systems in which each nonlinear component contains at most one unknown delay with no unknown system parameters. However, in many real-world systems, such as the purification process of zinc sulphate solution [93], the nonlinear terms contain *both* delays and parameters that need to be identified. Our goal in this chapter is to develop an efficient method to identify the unknown delays and the unknown parameters in a complicated time-delay dynamical system. We will introduce a set of auxiliary delay-differential systems. Then, we will show that the gradient of the least-square cost function can be expressed in terms of the solutions of these auxiliary systems. A numerical integration is used to solve the auxiliary systems, and thereby we obtain the gradient of the cost function, which is the main information needed to solve the parameter identification problem as a nonlinear optimization problem by using numerical optimization techniques. Based on this idea, a computational algorithm is developed for identifying the unknown time-delays and system parameters in a general nonlinear system. We demonstrate the effectiveness of the proposed algorithm on two nonlinear parameter identification problems, one of which is the parameter identification problem for the zinc sulphate purification process.

2.2 Problem formulation

Consider the following nonlinear time-delay system:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta}), \quad t \in [0, T], \tag{2.1}$$

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \leq 0, \tag{2.2}$$

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*; $\tilde{\boldsymbol{x}}(t) = [\boldsymbol{x}(t-\tau_1)^\top, \ldots, \boldsymbol{x}(t-\tau_m)^\top]^\top \in \mathbb{R}^{nm}$ is the *delayed state*; 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 (2.1)-(2.2) are unknown quantities that need to be determined. We assume that the ith time-delay is in 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:

$$a_i \leq \tau_i \leq b_i, \quad i = 1, \ldots, m. \tag{2.3}$$

Any vector $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_m]^\top \in \mathbb{R}^m$ that satisfies (2.3) 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 (2.1)-(2.2) are also unknown quantities that need to be determined. We suppose that

$$c_j \leq \zeta_j \leq d_j, \quad j = 1, \ldots, r, \tag{2.4}$$

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 (2.4) is called a *candidate parameter vector*. Let $\mathcal{Z}$ denote the set of all such candidate parameter vectors.

The output of system (2.1)-(2.2) is given by

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{\zeta}), \quad t \in [0, T], \tag{2.5}$$

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.

(2.A.1). The given functions $\boldsymbol{f}$ and $\boldsymbol{g}$ are continuously differentiable, and $\boldsymbol{\phi}$ is twice continuously differentiable.

(2.A.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 (2.A.1) and (2.A.2), the dynamic system (2.1)-(2.2) admits a unique solution corresponding to each pair $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$ [111]. We denote this solution by $\boldsymbol{x}(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$. Substituting $\boldsymbol{x}(\cdot|\boldsymbol{\tau}, \boldsymbol{\zeta})$ into (2.5) 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,

$$\boldsymbol{y}(t|\boldsymbol{\tau}, \boldsymbol{\zeta}) = \boldsymbol{g}(\boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta}), \boldsymbol{\zeta}), \quad t \leq T. \tag{2.6}$$

Suppose that the output from system (2.1)-(2.2) 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 $\boldsymbol{\zeta} \in \mathcal{Z}$ such that the following cost function:*

$$J(\boldsymbol{\tau}, \boldsymbol{\zeta}) = \sum_{l=1}^{q} \left| \boldsymbol{y}(t_l|\boldsymbol{\tau}, \boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l \right|^2 \tag{2.7}$$

is minimized, where $|\cdot|$ denotes the usual Euclidean norm.

Problem (P) is a nonlinear dynamic optimization problem whose decision variables are the delays and system parameters in system (2.1)-(2.2). We need to select optimal values for these delays and parameters so that the predicted system output best fits the experimental data. There are very few optimization techniques available in the literature for time-delay systems. In the existing literature, the delays are often assumed to be fixed and known (see, for example, [79, 119, 150]). Problem (P) is unique in that the delays are not fixed, but are 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 [120, 121] for non-delay systems, and in [93] for systems with fixed delays. However, the computational techniques developed in these references are not applicable to Problem (P) because the time-delays in system (2.1)-(2.2) are decision variables to be identified.

2.3 Preliminaries

Throughout this subsection, let $k \in \{1, \ldots, m\}$ and $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$ be 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,

$$\boldsymbol{\theta}^{\epsilon,i}(t) = \boldsymbol{\varphi}^\epsilon(t - \tau_i), \quad t \leq T, \quad i \neq k, \tag{2.8}$$

$$\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{0}, \quad t \leq 0. \tag{2.9}$$

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

Now, define

$$\boldsymbol{\chi}^\epsilon(t) = \begin{cases} \dot{\boldsymbol{\phi}}(t), & \text{if } t \leq 0, \\ \boldsymbol{f}(t, \boldsymbol{x}^\epsilon(t), \boldsymbol{x}^\epsilon(t - \tau_1 - \epsilon\delta_{k1}), \ldots, \boldsymbol{x}^\epsilon(t - \tau_m - \epsilon\delta_{km}), \boldsymbol{\zeta}), & \text{if } t \in (0, T]. \end{cases} \tag{2.10}$$

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

$$\dot{\boldsymbol{x}}^\epsilon(t) = \boldsymbol{\chi}^\epsilon(t). \tag{2.11}$$

Let $\bar{b} > 0$ be a fixed constant such that

$$\bar{b} = \max_{i=1,\ldots,m} \{b_i\}.$$

We have the following lemma.

Lemma 2.1. *There exists a a positive real number $L_2 > 0$ such that for each $\epsilon \in I$,*

$$|\boldsymbol{x}^\epsilon(t)|, |\boldsymbol{\chi}^\epsilon(t)| \leq L_2, \quad t \in [-\bar{b}, T], \tag{2.12}$$

Proof. Recall that $\boldsymbol{x}(s) = \boldsymbol{\phi}(s)$ is given for all $s \leq 0$. By (2.A.1), $\boldsymbol{\phi}(s)$ is twice differentiable. Clearly, there exist positive numbers α_1 and α_2 such that

$$|\boldsymbol{\phi}(s)| \leq \alpha_1, \quad s \in [-\bar{b}, 0], \tag{2.13}$$

$$|\dot{\boldsymbol{\phi}}(s)| \leq \alpha_2, \quad s \in [-\bar{b}, 0]. \tag{2.14}$$

For brevity, we denote

$$\tilde{\boldsymbol{x}}^\epsilon(t) = [\boldsymbol{x}^\epsilon(t - \tau_1 - \epsilon\delta_{k1})^\top, \ldots, \boldsymbol{x}^\epsilon(t - \tau_m - \epsilon\delta_{km})^\top]^\top.$$

For each $s \in [0, T]$, we have

$$\boldsymbol{x}^\epsilon(t) = \boldsymbol{x}(0) + \int_0^t \boldsymbol{f}(s, \boldsymbol{x}^\epsilon(s), \tilde{\boldsymbol{x}}^\epsilon(s), \boldsymbol{\zeta})ds, \quad t \in [0, T]. \tag{2.15}$$

Since $\boldsymbol{\zeta}$ is bounded in $\mathcal{Z}$, applying (2.A.2) to (2.15) it follows that

$$\begin{aligned}
|\boldsymbol{x}^\epsilon(t)| &\leq \alpha_1 + \int_0^t L_1(1 + |\boldsymbol{x}^\epsilon(s)| + |\tilde{\boldsymbol{x}}^\epsilon(s)| + |\boldsymbol{\zeta}|)ds \\
&\leq \alpha_1 + L_1 T + \int_0^t L_1(|\boldsymbol{x}^\epsilon(s)| + |\tilde{\boldsymbol{x}}^\epsilon(s)| + |\boldsymbol{\zeta}|)ds \\
&\leq \alpha_1 + L_1 T + \int_0^t L_1(|\boldsymbol{x}^\epsilon(s)| + |\boldsymbol{\zeta}|)ds + \sum_{i=1}^m \int_{-\tau_i}^{t-\tau_i} L_1 |\boldsymbol{x}^\epsilon(s)| ds \\
&\leq L_0 + \int_0^t L_1 |\boldsymbol{x}^\epsilon(s)| ds + \sum_{i=1}^m \int_{-\bar{b}}^t L_1 |\boldsymbol{x}^\epsilon(s)| ds, \quad t \in [0, T],
\end{aligned}$$

where $L_0 = \alpha_1 + L_1 T + r L_1 \bar{d} T$ and $\bar{d} = \max_{j=1,\ldots,r}\{\zeta_j\}$. Therefore, by using (2.13),

$$|\boldsymbol{x}^\epsilon(t)| \leq L_0 + m\alpha_1 \bar{b} + (m+1)L_1 \int_0^t |\boldsymbol{x}^\epsilon(s)| ds, \quad t \in [0, T].$$

Then, by using (2.10) and Gronwall-Bellman's lemma [111], we obtain

$$|\boldsymbol{x}^\epsilon(t)| \leq \rho_1 \exp\big((m+1)L_1 T\big), \quad t \in [0, T], \tag{2.16}$$

where $\rho_1 = L_0 + m\alpha_1 \bar{b}$.

In addition, for each $s \in [0, T]$, consider (2.10)-(2.11), then by using (2.A.2),

$$|\boldsymbol{\chi}^\epsilon(t)| \leq L_1(1 + |\boldsymbol{x}^\epsilon(t)| + |\tilde{\boldsymbol{x}}^\epsilon(t)| + |\boldsymbol{\zeta}|), \quad t \in [0, T],$$

Clearly,

$$|\boldsymbol{\chi}^\epsilon(t)| \le L_1 + rL_1\bar{d} + L_1|\boldsymbol{x}^\epsilon(t)| + L_1\sum_{i=1}^{m}|\boldsymbol{x}^\epsilon(s-\tau_i-\epsilon\delta_{ki})|. \tag{2.17}$$

Thus, by applying (2.16) to (2.17), we obtain

$$|\boldsymbol{\chi}^\epsilon(t)| \le L_2, \tag{2.18}$$

where $L_2 = L_1 + rL_1\bar{d} + (m+1)\rho_1 L_1 \exp\big((m+1)L_1T\big)$. Combining (2.13), (2.14), (2.16) and (2.18), the conclusion of the lemma follows readily. □

Define

$$\boldsymbol{\Xi} = \{\boldsymbol{\omega} \in \mathbb{R}^n : |\boldsymbol{\omega}| \le L_2\}. \tag{2.19}$$

Then, it follows form Lemma 2.1 that $\boldsymbol{x}^\epsilon(t) \in \boldsymbol{\Xi}$ for all $t \in [-\bar{b}, T]$ and $\epsilon \in I$. We have the following lemma.

Lemma 2.2. *There exists a positive real number $L_3 > 0$ such that for all $\epsilon \in I$,*

$$|\boldsymbol{\varphi}^\epsilon(t)|, |\boldsymbol{\chi}^\epsilon(t) - \boldsymbol{\chi}^0(t)|, \max_{i=1,\dots,m}|\boldsymbol{\theta}^{\epsilon,i}(t)| \le L_3|\epsilon|, \quad t \in [0,T]. \tag{2.20}$$

Proof. For each $t \in [0,T]$, we have

$$|\boldsymbol{\varphi}^\epsilon(t)| = |\boldsymbol{x}^\epsilon(t) - \boldsymbol{x}(t)| \le \int_0^t |\boldsymbol{\chi}^\epsilon(s) - \boldsymbol{\chi}^0(s)|ds, \quad t \in [0,T]. \tag{2.21}$$

By using (2.A.1), the function $\boldsymbol{f}$ is Lipschitz continuous on $\boldsymbol{\Xi} \times \boldsymbol{\Xi}$. Hence, there exists a real number $\rho > 0$ such that

$$|\boldsymbol{\chi}^\epsilon(s) - \boldsymbol{\chi}^0(s)| \le \rho|\boldsymbol{\varphi}^\epsilon(s)| + \rho\sum_{i=1}^{m}|\boldsymbol{\theta}^{\epsilon,i}(s)|, \quad s \in [0,T]. \tag{2.22}$$

Let $\epsilon \in I$ be arbitrary but fixed. For each $s \in [0,T]$,

$$\begin{aligned}|\boldsymbol{\theta}^{\epsilon,k}(s)| &= |\boldsymbol{x}^\epsilon(s-\tau_k-\epsilon) - \boldsymbol{x}(s-\tau_k)| \\ &\le |\boldsymbol{x}^\epsilon(s-\tau_k-\epsilon) - \boldsymbol{x}^\epsilon(s-\tau_k)| + |\boldsymbol{x}^\epsilon(s-\tau_k) - \boldsymbol{x}(s-\tau_k)|.\end{aligned}$$

Hence, by (2.11),

$$|\boldsymbol{\theta}^{\epsilon,k}(s)| \le \int_{\alpha(s)}^{\beta(s)}|\boldsymbol{\chi}^\epsilon(\eta)|d\eta + |\boldsymbol{\varphi}^\epsilon(t-\tau_k)|, \quad s \in [0,T], \tag{2.23}$$

where

$$\alpha(s) = \min\{s - \tau_k, s - \tau_k - \epsilon\}, \qquad \beta(s) = \max\{s - \tau_k, s - \tau_k - \epsilon\}.$$

Clearly,

$$|\beta(s) - \alpha(s)| = \epsilon, \quad s \in [0, T], \tag{2.24}$$

and

$$[\alpha(s), \beta(s)] \subset [-\bar{b}, T], \quad s \in [0, T]. \tag{2.25}$$

Substituting (2.24)-(2.25) into (2.23), and using Lemma 2.1, it yields

$$|\boldsymbol{\theta}^{\epsilon,k}(s)| \leq L_2|\epsilon| + |\boldsymbol{\varphi}^{\epsilon}(s - \tau_k)|, \quad s \in [0, T]. \tag{2.26}$$

Substituting (2.26) into (2.22) gives

$$|\boldsymbol{\chi}^{\epsilon}(s) - \boldsymbol{\chi}^{0}(s)| \leq \rho|\boldsymbol{\varphi}^{\epsilon}(s)| + m\rho L_2|\epsilon| + \rho\sum_{i=1}^{m}|\boldsymbol{\varphi}^{\epsilon}(s - \tau_i)|, \quad s \in [0, T]. \tag{2.27}$$

Substituting (2.27) into (2.21), it follows that

$$\begin{aligned}
|\boldsymbol{\varphi}^{\epsilon}(t)| &\leq m\rho L_2|\epsilon|T + \int_0^t \rho|\boldsymbol{\varphi}^{\epsilon}(s)|ds + \rho\sum_{i=1}^{m}\int_0^t |\boldsymbol{\varphi}^{\epsilon}(s - \tau_i)|ds \\
&\leq m\rho L_2|\epsilon|T + \int_0^t \rho|\boldsymbol{\varphi}^{\epsilon}(s)|ds + \rho\sum_{i=1}^{m}\int_{-\tau_i}^{t-\tau_i} |\boldsymbol{\varphi}^{\epsilon}(s)|ds \\
&\leq m\rho L_2|\epsilon|T + \rho\int_0^t |\boldsymbol{\varphi}^{\epsilon}(s)|ds + \rho m\int_{-\bar{b}}^{t} |\boldsymbol{\varphi}^{\epsilon}(s)|ds, \quad t \in [0, T].
\end{aligned}$$

Recall that $\boldsymbol{\varphi}^{\epsilon}(s) = 0$ for all $s \leq 0$. Therefore,

$$|\boldsymbol{\varphi}^{\epsilon}(t)| \leq m\rho L_2|\epsilon|T + (m+1)\rho\int_0^t |\boldsymbol{\varphi}^{\epsilon}(s)|ds, \quad t \in [0, T].$$

Thus, by Gronwall-Bellman's lemma,

$$|\boldsymbol{\varphi}^{\epsilon}(t)| \leq \rho_2|\epsilon|, \quad t \in [0, T],$$

where $\rho_2 = m\rho L_2 T\exp\{(m+1)\rho T\}$. Since $\boldsymbol{\varphi}^{\epsilon}(s) = 0$ for all $s \leq 0$, it follows that

$$|\boldsymbol{\varphi}^{\epsilon}(t)| \leq \rho_2|\epsilon|, \quad t \leq T. \tag{2.28}$$

Substituting (2.28) into (2.26) yields

$$|\boldsymbol{\theta}^{\epsilon,k}(s)| \leq (L_2 + \rho_2)|\epsilon|, \quad s \in [0, T].$$

Substituting (2.28) into (2.27), we obtain

$$|\boldsymbol{\chi}^{\epsilon}(s) - \boldsymbol{\chi}^{0}(s)| \leq m\rho L_2|\epsilon| + \rho\rho_2|\epsilon| + \rho\sum_{i=1}^{m}\rho_2|\epsilon| = (m\rho L_2 + (m+1)\rho\rho_2)|\epsilon|, \quad s \in [0, T].$$

Choose $L_3 = \max\{m\rho L_2 + (m+1)\rho\rho_2, L_2 + \rho_2, \rho_2\}$. The proof is completed. □

To proceed further, we need the following lemma.

Lemma 2.3. *For almost all $t \in [0, T]$, it holds that*

$$\lim_{\epsilon\to 0}\frac{\boldsymbol{\theta}^{\epsilon,k}(t) - \boldsymbol{\varphi}^{\epsilon}(t-\tau_k)}{\epsilon} = -\boldsymbol{\chi}^{0}(t-\tau_k). \tag{2.29}$$

Proof. Let $t \in [0, T] \setminus \tau_k$ be arbitrary but fixed. Then, for each $\epsilon \in I \setminus 0$,

$$\boldsymbol{\theta}^{\epsilon,k}(t) - \boldsymbol{\varphi}^{\epsilon}(t-\tau_k) = \boldsymbol{x}^{\epsilon}(t-\tau_k-\epsilon) - \boldsymbol{x}^{\epsilon}(t-\tau_k).$$

Hence, by (2.11),

$$\frac{\boldsymbol{\theta}^{\epsilon,k}(t) - \boldsymbol{\varphi}^{\epsilon}(t-\tau_k)}{\epsilon} = \frac{1}{\epsilon}\int_{t-\tau_k}^{t-\tau_k-\epsilon}\boldsymbol{\chi}^{\epsilon}(s)ds.$$

We can write this equation as follows:

$$\begin{aligned}\frac{\boldsymbol{\theta}^{\epsilon,k}(t) - \boldsymbol{\varphi}^{\epsilon}(t-\tau_k)}{\epsilon} &= -\boldsymbol{\chi}^{0}(t-\tau_k) + \boldsymbol{\chi}^{0}(t-\tau_k) + \frac{1}{\epsilon}\int_{t-\tau_k}^{t-\tau_k-\epsilon}\boldsymbol{\chi}^{\epsilon}(s)ds \\ &= -\boldsymbol{\chi}^{0}(t-\tau_k) + \rho(\epsilon),\end{aligned} \tag{2.30}$$

where

$$\rho(\epsilon) = \frac{1}{\epsilon}\int_{t-\tau_k}^{t-\tau_k-\epsilon}\big\{\boldsymbol{\chi}^{\epsilon}(s) - \boldsymbol{\chi}^{0}(t-\tau_k)\big\}ds.$$

Then, by using triangle inequality,

$$|\rho(\epsilon)| \leq \frac{1}{|\epsilon|}\int_{\alpha_1}^{\beta_1}|\boldsymbol{\chi}^{\epsilon}(s) - \boldsymbol{\chi}^{0}(s)|ds + \frac{1}{|\epsilon|}\int_{\alpha_1}^{\beta_1}|\boldsymbol{\chi}^{0}(s) - \boldsymbol{\chi}^{0}(t-\tau_k)|ds, \tag{2.31}$$

where

$$\alpha_1 = \min\{t-\tau_k, t-\tau_k-\epsilon\}, \qquad \beta_1 = \max\{t-\tau_k, t-\tau_k-\epsilon\}.$$

Clearly, $\beta_1 - \alpha_1 = |\epsilon|$. Thus, by Lemma 2.2 and (2.31), it follows that, for each $\epsilon \in I$,

$$|\rho(\epsilon)| \leq L_3|\epsilon| + \frac{1}{|\epsilon|}\int_{\alpha_1}^{\beta_1} |\boldsymbol{\chi}^0(s) - \boldsymbol{\chi}^0(t-\tau_k)|ds. \tag{2.32}$$

Since $t \neq \tau_k$, we need to consider the following two cases. *Case 1:* $t < \tau_k$ and *Case 2:* $t > \tau_k$.

Case 1: $t < \tau_k$. Clearly,

$$\epsilon \in I, \quad |\epsilon| < \tau_k - t \quad \Rightarrow \quad [\alpha_1, \beta_1] \subset [-\bar{b}, 0]. \tag{2.33}$$

Now, since $\boldsymbol{f}$ is continuously differentiable (recall (2.A.1)), and $\boldsymbol{\zeta}$ and $\boldsymbol{x}$ are bounded on $[-\bar{b}, T]$ (recall (2.4) and (2.12)), we can show that $\boldsymbol{\chi}^0$ is Lipschitz continuous on $[-\bar{b}, 0]$. Hence, there exists a real number $\eta_1 > 0$ such that

$$|\boldsymbol{\chi}^0(s) - \boldsymbol{\chi}^0(t-\tau_k)| \leq \eta_1|s - t + \tau_k|, \quad s \in [-\bar{b}, 0] \tag{2.34}$$

It follows from (2.33) and (2.34) that

$$|\boldsymbol{\chi}^0(s) - \boldsymbol{\chi}^0(t-\tau_k)| \leq \eta_1|s - t + \tau_k| \leq \eta_1(\beta_1 - \alpha_1) = \eta_1|\epsilon|, \quad s \in [\alpha_1, \beta_1],$$

when $\epsilon \in I$ is sufficiently small. Substituting this inequality into (2.32) gives

$$|\rho(\epsilon)| \leq (L_3 + \eta_1)|\epsilon|.$$

This shows that

$$\lim_{\epsilon \to 0} \rho(\epsilon) = 0, \quad \epsilon \in I \setminus 0, \quad s \in [-\bar{b}, 0] \tag{2.35}$$

Case 2: $t > \tau_k$

Suppose $t > \tau_k$. Clearly,

$$\epsilon \in I, \quad |\epsilon| < t - \tau_k \quad \Rightarrow \quad [\alpha_1, \beta_1] \subset (0, T]. \tag{2.36}$$

Similarly, since $\boldsymbol{f}$ is continuously differentiable (recall (2.A.1)), and $\boldsymbol{\zeta}$ and $\boldsymbol{x}$ are bounded on $[-\bar{b}, T]$ (recall (2.4) and (2.12)), we can show that $\boldsymbol{\chi}^0$ is Lipschitz continuous on $(0, T]$. Hence, there exists a real number $\eta_2 > 0$ such that

$$|\boldsymbol{\chi}^0(s) - \boldsymbol{\chi}^0(t-\tau_k)| \leq \eta_2|s - t + \tau_k|, \quad s \in (0, T] \tag{2.37}$$

It follows form (2.36) and (2.37) that

$$|\boldsymbol{\chi}^0(s) - \boldsymbol{\chi}^0(t-\tau_k)| \leq \eta_2|s-t+\tau_k| \leq \eta_2(\beta_1-\alpha_1) = \eta_2|\epsilon|, \quad s \in [\alpha_1, \beta_1],$$

when $\epsilon \in I$ is sufficiently small.

Substituting this inequality into (2.32) gives

$$|\rho(\epsilon)| \leq (L_3+\eta_2)|\epsilon|, \quad s \in (0,T].$$

This shows that

$$\lim_{\epsilon\to 0}\rho(\epsilon) = 0, \quad \epsilon \in I \setminus 0, \quad s \in (0,T] \tag{2.38}$$

Applying (2.35) and (2.38) to (2.30) completes the proof. □

2.4 Gradient computation

Problem (P) involves choosing a finite number of decision variables to minimize the cost function (2.7). 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 [79]—typically require the gradient of the cost function, which is difficult to determine in Problem (P) because the delays and parameters influence (2.7) *implicitly* through the dynamic system (2.1)-(2.2). 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 earlier works in [117, 125, 126], can be integrated with a standard nonlinear programming algorithm to solve Problem (P).

2.4.1 State variation with respect to time-delays

The solution of system (2.1)-(2.2) is normally viewed as a function of time, with $\boldsymbol{\tau}$ and $\boldsymbol{\zeta}$ being fixed vectors. By fixing $t \in (-\infty, T]$ while allowing $\boldsymbol{\tau}$ 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 theorem, 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 2.1. *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 $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$,*

$$\frac{\partial \boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \tau_k} = \boldsymbol{\Lambda}^k(t|\boldsymbol{\tau}, \boldsymbol{\zeta}), \quad k = 1, \ldots, m, \tag{2.39}$$

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

$$\begin{aligned}\dot{\boldsymbol{\Lambda}}^k(t) = \frac{\partial \boldsymbol{f}(t,\boldsymbol{x}(t),\tilde{\boldsymbol{x}}(t),\boldsymbol{\zeta})}{\partial \boldsymbol{x}}\boldsymbol{\Lambda}^k(t) + \sum_{i=1}^{m}\frac{\partial \boldsymbol{f}(t,\boldsymbol{x}(t),\tilde{\boldsymbol{x}}(t),\boldsymbol{\zeta})}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\Lambda}^k(t-\tau_i)\\ - \frac{\partial \boldsymbol{f}(t,\boldsymbol{x}(t),\tilde{\boldsymbol{x}}(t),\boldsymbol{\zeta})}{\partial \tilde{\boldsymbol{x}}^k}\boldsymbol{\chi}(t-\tau_k)\end{aligned} \tag{2.40}$$

with initial condition

$$\boldsymbol{\Lambda}^k(t) = \boldsymbol{0}, \quad t \le 0. \tag{2.41}$$

Proof. Let $k \in \{1,\dots,m\}$ and $(\boldsymbol{\tau},\boldsymbol{\zeta}) \in \mathcal{T}\times\mathcal{Z}$ be arbitrary but fixed. As in Section 2.3, 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

$$\rho(\epsilon) = \int_0^T \big|\epsilon^{-1}\boldsymbol{\theta}^{\epsilon,k}(s) - \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\tau_k) + \boldsymbol{\chi}(s-\tau_k)\big|ds. \tag{2.42}$$

It follows from (2.9), (2.12), and (2.20) that for each $\epsilon \in I\setminus\{0\}$,

$$\big|\epsilon^{-1}\boldsymbol{\theta}^{\epsilon,k}(s) - \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\tau_k) + \boldsymbol{\chi}(s-\tau_k)\big| \le L_2 + 2L_3, \quad s\in[0,T].$$

Hence, the integrand in (2.42) is uniformly bounded with respect to $\epsilon \in I\setminus\{0\}$. Furthermore, it follows from (2.29) that $\epsilon^{-1}\boldsymbol{\theta}^{\epsilon,k}(s) - \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\tau_k) + \boldsymbol{\chi}(s-\tau_k)$ converges to zero almost everywhere on $[0,T]$ as $\epsilon\to 0$. Thus, from the Lebesgue dominated convergence theorem,

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

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

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

Then, by the chain rule,

$$\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial\alpha} = \frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \boldsymbol{x}}\boldsymbol{\varphi}^\epsilon(s) + \sum_{i=1}^{m}\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\theta}^{\epsilon,i}(s), \tag{2.43}$$

where

$$\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{f}\big(s,\boldsymbol{x}(s)+\alpha\boldsymbol{\varphi}^\epsilon(s),\tilde{\boldsymbol{x}}(s)+\alpha\boldsymbol{\theta}^\epsilon(s),\boldsymbol{\zeta}\big)}{\partial \boldsymbol{x}}, \tag{2.44}$$

$$\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \tilde{\boldsymbol{x}}^i} = \frac{\partial \boldsymbol{f}\big(s,\boldsymbol{x}(s)+\alpha\boldsymbol{\varphi}^\epsilon(s),\tilde{\boldsymbol{x}}(s)+\alpha\boldsymbol{\theta}^\epsilon(s),\boldsymbol{\zeta}\big)}{\partial \tilde{\boldsymbol{x}}^i}. \tag{2.45}$$

We can rewrite (2.43) 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

$$\Delta_1(s,\alpha) = \Big\{\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \boldsymbol{x}} - \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\Big\}\boldsymbol{\varphi}^{\epsilon}(s), \tag{2.46}$$

$$\Delta_2(s,\alpha) = \sum_{i=1}^{m}\Big\{\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \tilde{\boldsymbol{x}}^i} - \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\Big\}\boldsymbol{\theta}^{\epsilon,i}(s). \tag{2.47}$$

Applying (2.8) gives

$$\begin{aligned}\frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \alpha} &= \Delta_1(s,\alpha) + \Delta_2(s,\alpha) + \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\boldsymbol{\varphi}^{\epsilon}(s) \\ &\quad + \sum_{i=1}^{m} \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\varphi}^{\epsilon}(s-\tau_i) + \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^k}\big\{\boldsymbol{\theta}^{\epsilon,k}(s) - \boldsymbol{\varphi}^{\epsilon}(s-\tau_k)\big\}.\end{aligned} \tag{2.48}$$

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,

$$\boldsymbol{\varphi}^{\epsilon}(t) = \int_0^t \big\{\bar{\boldsymbol{f}}(s,1) - \bar{\boldsymbol{f}}(s,0)\big\}ds = \int_0^t \int_0^1 \frac{\partial \bar{\boldsymbol{f}}(s,\alpha)}{\partial \alpha} d\alpha ds. \tag{2.49}$$

Substituting (2.48) into (2.49) yields

$$\begin{aligned}\boldsymbol{\varphi}^{\epsilon}(t) &= \int_0^t \int_0^1 \big\{\Delta_1(s,\alpha) + \Delta_2(s,\alpha)\big\}d\alpha ds + \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}}\boldsymbol{\varphi}^{\epsilon}(s)ds \\ &\quad + \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^k}\big\{\boldsymbol{\theta}^{\epsilon,k}(s) - \boldsymbol{\varphi}^{\epsilon}(s-\tau_k)\big\}ds + \sum_{i=1}^{m}\int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i}\boldsymbol{\varphi}^{\epsilon}(s-\tau_i)ds.\end{aligned} \tag{2.50}$$

Now, by using (2.44) and (2.45), we can write the auxiliary system (2.40) 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,

$$\begin{aligned}\boldsymbol{\Lambda}^k(t) = \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \boldsymbol{x}} \boldsymbol{\Lambda}^k(s) ds + \sum_{i=1}^m \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^i} \boldsymbol{\Lambda}^k(s-\tau_i) ds \\ - \int_0^t \frac{\partial \bar{\boldsymbol{f}}(s,0)}{\partial \tilde{\boldsymbol{x}}^k} \boldsymbol{\chi}(s-\tau_k) ds.\end{aligned} \tag{2.51}$$

Now, since $\boldsymbol{f}$ is continuously differentiable (recall (2.A.1)), and $\boldsymbol{x}$ is bounded on $[-\bar{b}, T]$ (recall (2.12)), there exists constants $M_1 > 0$ and $M_2 > 0$ such that

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

where $|\cdot|$ denotes the usual Euclidean norm on $\mathbb{R}^{n\times n}$. Thus, by multiplying (2.50) by ϵ^{-1}, subtracting (2.51), and then finally taking the norm of both sides, we obtain

$$\begin{aligned}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Lambda}^k(t)\big| \le M_2\rho(\epsilon) + \int_0^t M_1\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\big| ds \\ + \sum_{i=1}^m \int_0^t M_2\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\tau_i) - \boldsymbol{\Lambda}^k(s-\tau_i)\big| ds \\ + |\epsilon|^{-1}\int_0^t\int_0^1 \big\{|\Delta_1(s,\alpha)| + |\Delta_2(s,\alpha)|\big\} d\alpha ds,\end{aligned} \tag{2.52}$$

where $\rho(\epsilon)$ is as defined in (2.42). The second integral term on the right-hand side of (2.52) can be simplified as follows:

$$\begin{aligned}\sum_{i=1}^m \int_0^t M_2\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\tau_i) - \boldsymbol{\Lambda}^k(s-\tau_i)\big| ds &= \sum_{i=1}^m \int_{-\tau_i}^{t-\tau_i} M_2\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\big| ds \\ &\le \sum_{i=1}^m \int_0^t M_2\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\big| ds \\ &= \int_0^t m M_2\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\big| ds.\end{aligned}$$

Hence, (2.52) becomes

$$\begin{aligned}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Lambda}^k(t)\big| \le M_2\rho(\epsilon) + \int_0^t \bar{M}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\big| ds \\ + |\epsilon|^{-1}\int_0^t\int_0^1 \big\{|\Delta_1(s,\alpha)| + |\Delta_2(s,\alpha)|\big\} d\alpha ds,\end{aligned} \tag{2.53}$$

where $\bar{M} = M_1 + mM_2$. Since $\boldsymbol{f}$ is continuously differentiable and $\boldsymbol{x}^\epsilon$ is uniformly bounded with respect to ϵ, both $\frac{\partial \bar{\boldsymbol{f}}}{\partial \boldsymbol{x}}$ and $\frac{\partial \bar{\boldsymbol{f}}}{\partial \tilde{\boldsymbol{x}}^i}$ are uniformly continuous on $[0,T]\times[0,1]$. Furthermore, by (2.20), $\boldsymbol{x}(s) + \alpha\boldsymbol{\varphi}^\epsilon(s) \to \boldsymbol{x}(s)$ 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 $|\epsilon| < \epsilon'$,

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

By taking the norm of (2.46) and (2.47), and then using these inequalities together with (2.20), we obtain

$$|\Delta_1(s,\alpha)| \leq \delta L_3 |\epsilon|, \qquad |\Delta_2(s,\alpha)| \leq \delta m L_3 |\epsilon|,$$

where $|\epsilon| < \epsilon'$. Substituting these inequalities into (2.53) yields,

$$\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Lambda}^k(t)\big| \leq M_2\rho(\epsilon) + (L_3T + mL_3T)\delta + \int_0^t \bar{M}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\big|ds.$$

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 $|\epsilon| < \min\{\epsilon', \epsilon''\}$,

$$\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Lambda}^k(t)\big| \leq M_2\delta + (L_3T + mL_3T)\delta + \int_0^t \bar{M}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Lambda}^k(s)\big|ds.$$

Applying the Gronwall-Bellman Lemma [111] gives

$$\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Lambda}^k(t)\big| \leq \delta(M_2 + L_3T + mL_3T)\exp\{\bar{M}T\},$$

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. □

2.4.2 State variation with respect to system parameters

In Theorem 2.1, we derive formulae for the gradient of the state with respect to the time-delays. We now turn our attention to the gradient of the state with respect to the system parameters.

Let w be a new state variable with dynamics

$$\dot{w}(t) = 1, \quad t \in [0,T], \tag{2.54}$$
$$w(t) = t, \quad t \leq 0. \tag{2.55}$$

Clearly, $w(t) = t$ for all $t \in (-\infty, T]$. Thus, we can express the system parameters in (2.1)-(2.2) in terms of the new state w as follows:

$$\zeta_j = t - w(t - \zeta_j), \quad j = 1, \ldots, r. \tag{2.56}$$

Substituting (2.56) into the original system (2.1)-(2.2) gives

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), t - w(t - \zeta_1), \ldots, t - w(t - \zeta_r)), \quad t \in [0, T], \tag{2.57}$$

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \leq 0. \tag{2.58}$$

The system parameters ζ_j, $j = 1, \ldots, r$, are now time-delays in the enlarged system consisting of (2.54)-(2.55) and (2.57)-(2.58). Thus, to determine the state variation with respect to the system parameters in system (2.1)-(2.2), we just need to apply Theorem 2.1 to the enlarged system consisting of (2.54)-(2.55) and (2.57)-(2.58). It is important to note that each system parameter is bounded below by zero (see the problem formulation in Section 2.2). Thus, the enlarged system considered here is a valid time-delay system with all time-delays being 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

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

and

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

The enlarged system consisting of (2.54)-(2.55) and (2.57)-(2.58) can be written as follows:

$$\dot{\boldsymbol{z}}(t) = \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t)), \quad t \in [0, T], \tag{2.59}$$

$$\boldsymbol{z}(t) = \hat{\boldsymbol{\phi}}(t), \quad t \leq 0, \tag{2.60}$$

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 (2.59)-(2.60) corresponding to the system parameter ζ_j is

$$\begin{aligned}\begin{bmatrix}\dot{\boldsymbol{\Gamma}}^j(t)\\ \dot{\gamma}^j(t)\end{bmatrix} &= \frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \boldsymbol{z}} \begin{bmatrix}\boldsymbol{\Gamma}^j(t)\\ \gamma^j(t)\end{bmatrix} + \sum_{i=1}^{m} \frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \tilde{\boldsymbol{z}}^i} \begin{bmatrix}\boldsymbol{\Gamma}^j(t-\tau_i)\\ \gamma^j(t-\tau_i)\end{bmatrix} \\ &\quad + \sum_{i=1}^{r} \frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \tilde{\boldsymbol{z}}^{m+i}} \begin{bmatrix}\boldsymbol{\Gamma}^j(t-\zeta_i)\\ \gamma^j(t-\zeta_i)\end{bmatrix} - \frac{\partial \hat{\boldsymbol{f}}(t, \boldsymbol{z}(t), \tilde{\boldsymbol{z}}(t))}{\partial \tilde{\boldsymbol{z}}^{m+j}} \hat{\boldsymbol{\chi}}(t-\zeta_j)\end{aligned} \tag{2.61}$$

with the initial conditions

$$\begin{bmatrix}\boldsymbol{\Gamma}^j(t)\\ \gamma^j(t)\end{bmatrix} = \begin{bmatrix}\mathbf{0}\\ 0\end{bmatrix}, \quad t \le 0. \tag{2.62}$$

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

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

Furthermore, it is clear that $\gamma^j(t) = 0$ for all $t \in (-\infty, T]$. Thus, the auxiliary system (2.61)-(2.62) becomes

$$\begin{aligned}\dot{\boldsymbol{\Gamma}}^j(t) &= \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}^j(t) + \sum_{i=1}^{m} \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \tilde{\boldsymbol{x}}^i} \boldsymbol{\Gamma}^j(t-\tau_i) \\ &\quad + \frac{\partial \boldsymbol{f}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta})}{\partial \zeta_j}\end{aligned} \tag{2.63}$$

with initial conditions

$$\boldsymbol{\Gamma}^j(t) = \mathbf{0}, \quad t \le 0. \tag{2.64}$$

Applying Theorem 2.1 to the enlarged system consisting of (2.54)-(2.55) and (2.57)-(2.58) yields the following result.

Theorem 2.2. *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 $(\boldsymbol{\tau}, \boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$,*

$$\frac{\partial \boldsymbol{x}(t|\boldsymbol{\tau}, \boldsymbol{\zeta})}{\partial \zeta_j} = \boldsymbol{\Gamma}^j(t|\boldsymbol{\tau}, \boldsymbol{\zeta}), \quad j = 1, \ldots, r, \tag{2.65}$$

where $\mathbf{\Gamma}^j(\cdot|\boldsymbol{\tau},\boldsymbol{\zeta})$ *satisfies the auxiliary time-delay system* (2.63)-(2.64).

2.4.3 Gradient computation algorithm

We are now ready to derive formulae for the gradients of the cost function in Problem (P). By using Theorems 2.1 and 2.2 and the chain rule of differentiation, we obtain

$$\frac{\partial J(\boldsymbol{\tau},\boldsymbol{\zeta})}{\partial \tau_k} = 2\sum_{l=1}^{q} \big(\boldsymbol{y}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l\big)^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta}),\boldsymbol{\zeta})}{\partial \boldsymbol{x}} \mathbf{\Lambda}^k(t_l|\boldsymbol{\tau},\boldsymbol{\zeta}), \tag{2.66}$$

$$\begin{aligned}\frac{\partial J(\boldsymbol{\tau},\boldsymbol{\zeta})}{\partial \zeta_j} &= 2\sum_{l=1}^{q} \big(\boldsymbol{y}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l\big)^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta}),\boldsymbol{\zeta})}{\partial \boldsymbol{x}} \mathbf{\Gamma}^j(t_l|\boldsymbol{\tau},\boldsymbol{\zeta}) \\ &\quad + 2\sum_{l=1}^{q} \big(\boldsymbol{y}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta}) - \hat{\boldsymbol{y}}^l\big)^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta}),\boldsymbol{\zeta})}{\partial \zeta_j}.\end{aligned} \tag{2.67}$$

We now present the following algorithm for computing the cost function (2.7) and its gradient at a given pair $(\boldsymbol{\tau},\boldsymbol{\zeta}) \in \mathcal{T} \times \mathcal{Z}$.

Algorithm 2.1.

Step 1. Obtain $\boldsymbol{x}(\cdot|\boldsymbol{\tau},\boldsymbol{\zeta})$, $\mathbf{\Lambda}^k(\cdot|\boldsymbol{\tau},\boldsymbol{\zeta})$, $k = 1,\ldots,m$, and $\mathbf{\Gamma}^j(\cdot|\boldsymbol{\tau},\boldsymbol{\zeta})$, $j = 1,\ldots,r$, by solving the enlarged time-delay system consisting of the original system (2.1)-(2.2) and the auxiliary systems (2.40)-(2.41) and (2.63)-(2.64).

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 (2.6).

Step 3. Use $\boldsymbol{y}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta})$, $l = 1,\ldots,q$, to compute $J(\boldsymbol{\tau},\boldsymbol{\zeta})$ through equation (2.7).

Step 4. Use $\boldsymbol{x}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta})$, $\boldsymbol{y}(t_l|\boldsymbol{\tau},\boldsymbol{\zeta})$, $\mathbf{\Lambda}^k(t_l|\boldsymbol{\tau},\boldsymbol{\zeta})$, and $\mathbf{\Gamma}^j(t_l|\boldsymbol{\tau},\boldsymbol{\zeta})$, $l = 1,\ldots,q$, 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 (2.66) and (2.67).

This gradient computation algorithm can be integrated with a standard gradient-based optimization method (e.g. sequential quadratic programming) to solve Problem (P) as a nonlinear programming problem.

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

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t,\boldsymbol{x}(t),\tilde{\boldsymbol{x}}(t),\boldsymbol{u}(t),\tilde{\boldsymbol{u}}(t),\boldsymbol{\zeta}), \quad t \in [0,T], \tag{2.68}$$

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \le 0, \tag{2.69}$$

where $\boldsymbol{u}(t) = [u_1(t), \ldots, u_v(t)]^\top \in \mathbb{R}^v$ is the *control input* of system (2.68)-(2.69); $\tilde{\boldsymbol{u}}(t) = [\boldsymbol{u}(t-\lambda_1)^\top, \ldots, \boldsymbol{u}(t-\lambda_d)^\top]^\top \in \mathbb{R}^{vd}$ is the *delayed control*; and λ_i, $i = 1, \ldots, d$ are unknown *control delays*. The other symbols are as defined in Section 2.2.

In (2.68)-(2.69), $\tilde{\boldsymbol{u}}$ is assumed to be a known input function. Thus, we can write (2.68)-(2.69) in the form of (2.1)-(2.2) as follows:

$$\dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{f}}(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{\zeta}, \boldsymbol{\lambda}), \quad t \in [0, T], \tag{2.70}$$

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).

2.5 Numerical examples

2.5.1 Example 2.1

We now apply the solution method developed in Section 2.4 to the industrial purification process described in [93,94]. The purpose of this process is to remove harmful cobalt and cadmium ions from a zinc sulphate electrolyte by adding zinc powder to induce 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:

$$V\dot{x}_1(t) = Qx_1^0 - Qx_1(t-\tau) - \alpha u(t)x_1(t-\tau) + cx_2(t-\tau), \tag{2.71}$$

$$V\dot{x}_2(t) = Qx_2^0 - Qx_2(t-\tau) - \beta v(t)x_2(t-\tau) + dx_1(t-\tau), \tag{2.72}$$

and

$$x_1(t) = 3.3 \times 10^{-4}, \quad x_2(t) = 4.0 \times 10^{-3} \quad t \leq 0, \tag{2.73}$$

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 for two metallic impurities ions which depend on the amount 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 system 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 [94] considers the parameter identification problem for system

(2.71)-(2.73) 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 [94]:

$$\beta = 2.823 \times 10^{-4}, \quad c = 16.67, \quad d = 7.107 \times 10^{2}. \tag{2.74}$$

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 [94].

$$u(t) = \sum_{k=1}^{8} \sigma^k \psi_{[\bar{t}_{k-1}, \bar{t}_k)}(t), \quad t \in [0, 8], \tag{2.75}$$

$$v(t) = \sum_{k=1}^{8} \bar{\sigma}^k \psi_{[\bar{t}_{k-1}, \bar{t}_k)}(t), \quad t \in [0, 8], \tag{2.76}$$

where the values of $\bar{t}_k$, σ^k, and $\bar{\sigma}^k$, $k = 1, \ldots, 8$, are listed in Table 2.1, 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, $y(t) = x_2(t)$.

Given system (2.71)-(2.73), with data (2.74)-(2.76), our goal is to identify the system parameter α and the delay τ.

Table 2.1: Control values for Example 2.1.

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

We simulate system (2.71)-(2.73) 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

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

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 (2.71)-(2.73).

Note that this problem cannot be solved using the identification method in [125], as the third term on the right-hand side of (2.71) is a nonlinear term containing both an unknown parameter and an unknown delay. The identification method in [125] 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 2.4.3.

Computational results for different initial guesses are shown in Table 2.2. The convergence of the output trajectory for the initial guess $\tau = 3$ and $\alpha = 0$ is displayed in Figure 2.1. This figure shows the output trajectory at intermediate iterations of the algorithm, as well as the final (converged) trajectory. In Table 2.2 and Figure 2.1, τ^i and α^i are the values of τ and α at the ith iteration during the optimization process ($i = 0$ signifies the initial guess). We can see from Table 2.2 and Figure 2.1 that the optimal trajectory converges to the observed data well, regardless of the initial guess. Thus, the algorithm easily recovers the true values of the delay and parameter for this problem.

Table 2.2: Convergence of the cost values in Example 2.1.

	Initial guess		Cost value at ith iteration			
No.	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}

For comparison, we also solve this problem using the genetic algorithm (GA) in [57]. 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 $\tau = 2.0026$ 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.

2.5.2 Example 2.2

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

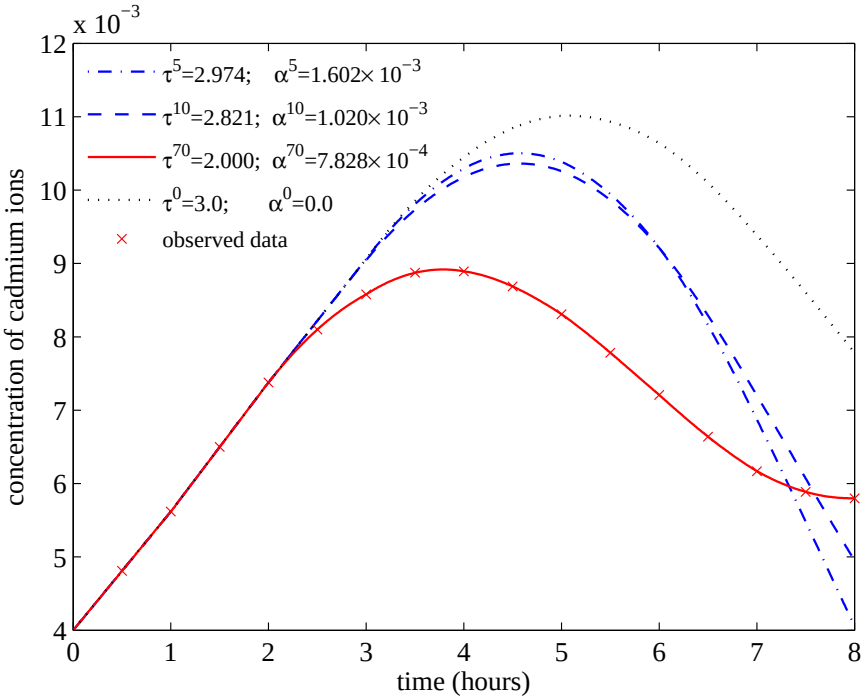

Figure 2.1: Convergence of the output trajectory in Example 2.1 for initial guess No.4.

$$\dot{x}_1(t) = -2x_1(t) + 0.1(1 - x_1(t-\tau_1)) \exp\left\{\frac{20x_2(t)}{20 + x_2(t)}\right\} + 0.1x_1(t-\tau_1)x_2(t-\tau_2) + u(t-\tau_3), \tag{2.77}$$

$$\dot{x}_2(t) = -2.5x_2(t) + 0.8(1 - x_1(t-\tau_1)) \exp\left\{\frac{20x_2(t)}{20 + x_2(t)}\right\} + 0.1x_2(t-\tau_1)x_2(t-\tau_2) + u(t-\tau_3), \tag{2.78}$$

with initial condition

$$x_1(t) = 1, \quad x_2(t) = 1, \quad t \leq 0. \tag{2.79}$$

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 given by

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

Furthermore, the output is

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

We use the output trajectory of (2.77)-(2.79) 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 $t_l = l/2$, $l = 1, \ldots, 20$. Thus, our identification problem is: choose τ_1, τ_2, and τ_3 to minimize

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

subject to the dynamics (2.77)-(2.79).

We solved this problem using the same Matlab program that was used to solve Example 2.1. The convergence process of the program is shown in Table 2.3 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 2.2. In Table 2.3 and Figure 2.2, $\boldsymbol{\tau}^i = [\tau_1^i, \tau_2^i, \tau_3^i]^\top$ is the values of $\boldsymbol{\tau}$ at the ith iteration, while $i = 0$ signifies the initial guess. We also solve this problem using GA with the same parameters as in Example 2.1. 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 2.1, we see that the optimization results converge from all initial guesses to the optimal solution.

Table 2.3: Convergence of the cost values in Example 2.2.

	Initial guess			Cost value at ith iteration			
No.	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	5.667×10^{-3}	6.661×10^{-15}
2	1.5	1.5	1.5	0.1386	0.0035	3.357×10^{-6}	6.618×10^{-15}
3	2.5	2.5	2.5	0.0747	0.0083	4.405×10^{-4}	1.534×10^{-14}
4	3.0	3.0	3.0	0.1710	0.0298	2.780×10^{-3}	6.656×10^{-15}

2.6 Conclusion

In this chapter, 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 2.5 demonstrate that this approach

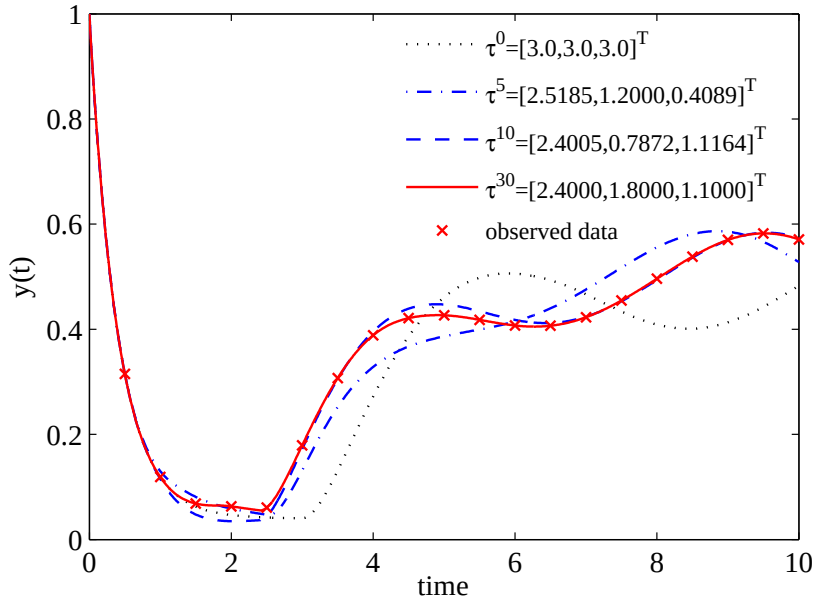

Figure 2.2: Convergence of the output trajectory in Example 2.2 for initial guess No.4.

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.

CHAPTER 3

Identification of time-delays for nonlinear systems with piecewise-constant input

3.1 Introduction

The optimization-based approach developed in Chapter 2 is designed for nonlinear delayed systems with smooth inputs. For systems with input-delays, if the input function is smooth, then the system dynamics will be continuously differentiable with respect to the input-delays, and thus the approach proposed in Chapter 2 can be easily modified to estimate the input-delays in this case. Unfortunately, the input function is often non-smooth in practical applications. Examples include biodiesel production [58], evaporation process [119], chromatography process [133], quadruple-tank process [55], batch reactor [105], and distillation column [60]. Since flow rate transmission, sensors, transfer delays of sensor-to-controller that are involved in control loops unavoidably introduce input-delays. As such, the estimation method in Chapter 2 and [125] is not applicable in such situations.

Time-delays identification for nonlinear delay systems with input-delays has been an interest research topic. However, the vast majority of delay estimation methods for delayed systems with piecewise inputs are only applicable to simple systems with linear dynamics and a single delay, see for example, step input and system parameters identification method [103], annihilation and integration based identification method [103]. In this chapter, we consider the time-delay estimation problem for nonlinear systems in which the input function is piecewise-constant. Such estimation problems arise, for example, in evaporation and purification processes [94, 119]. We assume that the governing system contains one state-delay and one input-delay, both of which are unknown and need to be estimated using experimental data. As with Chapter 2, we formulate the delay estimation problem as a dynamic optimization problem in which the cost function measures the least-squares error between predicted output and observed system output. The main difficulties in solving this problem are: i) The delays are decision variables to be optimized, rather than fixed values. Thus, conventional optimization techniques are

not directly applicable; and ii) since the input function is discontinuous, thus the dynamics are clearly discontinuous with respect to input delays. Hence, the results obtained in Chapter 2 can not be used to determine this state variation with respect to input time-delays. In this chapter, we focus on the derivation of a computational procedure for determining the gradient of the cost function for this problem. This procedure, which involves integrating an auxiliary impulsive system with instantaneous jumps forward in time, is far more complex than the procedure given in Chapter 2, which does not involve any jumps. Moreover, because of the discontinuous nature of the input function, the cost function's gradient does not exist at certain points. We propose a heuristic strategy for dealing this complication. Subsequently, this heuristic strategy can be combined with our gradient computation procedure to solve the estimation problem using standard nonlinear programming algorithms. We then apply this approach to estimate the time-delays in two large-scale industrial engineering systems. The purpose of this chapter is to develop a new method for estimating the time-delays.

3.2 Problem formulation

Consider the following nonlinear time-delay system:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t), \boldsymbol{u}(t-\beta)), \quad t \in [0, T], \tag{3.1}$$

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \leq 0, \tag{3.2}$$

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*; $\boldsymbol{u}(t) = [u_1(t), \ldots, u_r(t)]^\top \in \mathbb{R}^r$ is the *input vector*; α and β are unknown time-delays that need to be determined; and $\boldsymbol{f} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R}^r \to \mathbb{R}^n$ and $\boldsymbol{\phi} : \mathbb{R} \to \mathbb{R}^n$ are given functions. Many dynamic processes in chemical engineering—for example, the distillation process described in [60]—can be modeled by equations (3.1) and (3.2). We assume that $\boldsymbol{f}$, $\boldsymbol{g}$, and $\boldsymbol{\phi}$ are continuously differentiable. We also assume that there exists a positive real number $L_1 > 0$ such that for all $\boldsymbol{x}', \boldsymbol{x}'' \in \mathbb{R}^n$ and $\boldsymbol{u}', \boldsymbol{u}'' \in \mathbb{R}^r$,

$$|\boldsymbol{f}(\boldsymbol{x}', \boldsymbol{x}'', \boldsymbol{u}', \boldsymbol{u}'')| \leq L_1(1 + |\boldsymbol{x}'| + |\boldsymbol{x}''| + |\boldsymbol{u}'| + |\boldsymbol{u}''|), \tag{3.3}$$

where $|\cdot|$ denotes the Euclidean norm. This assumption is standard in the control systems literature [93, 111, 118, 122, 124].

The output $\boldsymbol{y}(t)$ of system (3.1)-(3.2) is defined by

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t)), \quad t \in [0, T], \tag{3.4}$$

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

We refer to α as the *state-delay* and β as the *input-delay*. The exact values of these

delays are unknown; the only information we are given is that α lies within the interval $[\alpha_{\min}, \alpha_{\max}]$ and β lies within the interval $[\beta_{\min}, \beta_{\max}]$, where $\alpha_{\min} \geq 0$ and $\beta_{\min} > 0$. Thus, we have the following bound constraints:

$$\alpha_{\min} \leq \alpha \leq \alpha_{\max}, \tag{3.5}$$

$$\beta_{\min} \leq \beta \leq \beta_{\max}. \tag{3.6}$$

We assume that the input signal $\boldsymbol{u}$ is a given piecewise-constant function (this is the case in many engineering systems). Hence, $\boldsymbol{u}$ can be expressed as follows:

$$\boldsymbol{u}(t) = \boldsymbol{\sigma}^i, \quad t \in [t_{i-1}, t_i), \quad i = 1, \ldots, p, \tag{3.7}$$

where $\boldsymbol{\sigma}^i \in \mathbb{R}^r$, $i = 1, \ldots, p$, are given vectors and t_i, $i = 0, \ldots, p$, are given time points such that $-\beta_{\max} = t_0 < t_1 < \cdots < t_p = T$. Equation (3.7) can be rewritten as

$$\boldsymbol{u}(t) = \sum_{i=1}^{p} \boldsymbol{\sigma}^i \chi_{[t_{i-1}, t_i)}(t), \quad t \in [-\beta_{\max}, T], \tag{3.8}$$

where the characteristic function $\chi_{[t_{i-1}, t_i)} : \mathbb{R} \to \mathbb{R}$ is defined by

$$\chi_{[t_{i-1}, t_i)}(t) = \begin{cases} 1, & \text{if } t \in [t_{i-1}, t_i), \\ 0, & \text{otherwise.} \end{cases}$$

For each pair $(\alpha, \beta) \in [\alpha_{\min}, \alpha_{\max}] \times [\beta_{\min}, \beta_{\max}]$, let $\boldsymbol{x}(\cdot|\alpha, \beta)$ denote the corresponding solution of system (3.1)-(3.2). Substituting $\boldsymbol{x}(\cdot|\alpha, \beta)$ into equation (3.4) gives $\boldsymbol{y}(\cdot|\alpha, \beta)$, the predicted system output corresponding to (α, β). Mathematically,

$$\boldsymbol{y}(t|\alpha, \beta) = \boldsymbol{g}(\boldsymbol{x}(t|\alpha, \beta)), \quad t \leq T. \tag{3.9}$$

Suppose that the output from system (3.1)-(3.2) has been measured experimentally at a set of sample times $\{\tau_j\}_{j=1}^m \subset [0, T]$. Let $\hat{\boldsymbol{y}}^j \in \mathbb{R}^q$ denote the measured output at the jth sample time. Our goal is to use the experimental data $\{(\tau_j, \hat{\boldsymbol{y}}^j)\}_{j=1}^m$ to identify the time-delays α and β. We formulate this problem mathematically as follows.

Problem (P). *Choose the state-delay α and the input-delay β to minimize the cost function*

$$J(\alpha, \beta) = \sum_{j=1}^{m} \left| \boldsymbol{y}(\tau_j|\alpha, \beta) - \hat{\boldsymbol{y}}^j \right|^2 \tag{3.10}$$

subject to the dynamic system (3.1)-(3.2) *and the bound constraints* (3.5)-(3.6).

Problem (P) is a dynamic optimization problem governed by the time-delay system (3.1)-(3.2). The most interesting aspect of Problem (P) is that the time-delays in (3.1)-

(3.2) are actually decision variables to be chosen optimally. This is highly unusual; in most optimization problems involving time-delay systems, the delays are fixed and known, and the control input function is the decision variable to be chosen optimally [74,93,119]. In Problem (P), the input function is known, and the delays are the variables that need to be optimized.

We now conclude this section by showing that Problem (P) can be transformed into a *switched system optimal control problem*.

First, from (3.8),

$$\boldsymbol{u}(t-\beta) = \sum_{i=1}^{p} \boldsymbol{\sigma}^i \chi_{[t_{i-1},t_i)}(t-\beta) = \sum_{i=1}^{p} \boldsymbol{\sigma}^i \chi_{[t_{i-1}+\beta,t_i+\beta)}(t) = \sum_{i=1}^{p} \boldsymbol{\sigma}^i \chi_{[v_{i-1},v_i)}(t), \tag{3.11}$$

where v_i, $i = 0, \ldots, p$ are new decision variables defined by

$$v_i = t_i + \beta, \quad i = 0, \ldots, p. \tag{3.12}$$

It follows from (3.12) that

$$v_i - t_i = v_{i-1} - t_{i-1}, \quad i = 1, \ldots, p. \tag{3.13}$$

Substituting (3.11) into (3.1) gives

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}^i(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t)), \quad t \in [v_{i-1}, v_i) \cap [0, T], \quad i = 1, \ldots, p, \tag{3.14}$$

where

$$\boldsymbol{f}^i(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t)) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t), \boldsymbol{\sigma}^i).$$

System (3.14) is a *switched system* in which the dynamics change instantaneously at the *switching times* v_i, $i = 1, \ldots, p$.

Problem (P) can now be restated as follows: Choose the state-delay α and the switching times v_i, $i = 1, \ldots, p$ to minimize (3.10) subject to the switched system (3.14), the initial condition (3.2), and the constraints (3.5)-(3.6) and (3.13). This is an example of a switched system optimal control problem. Such problems have been the subject of active research over the last decade (see, for example, [24, 26, 69, 100] and the references cited therein). In particular, the well-known *time-scaling transformation* is a powerful tool for solving switched system optimal control problems (see [118,124,132,151]). Unfortunately, the time-scaling transformation is not applicable to time-delay systems such as system (3.14) defined above. Thus, a new method is needed to solve Problem (P).

3.3 State variation

Our goal is to solve Problem (P) using nonlinear optimization techniques. To do this, we need the partial derivatives of J with respect to the decision variables α and β. However, since J is not an explicit function of α and β, these partial derivatives cannot be determined using standard differentiation rules. To derive formulae for the partial derivatives of J, we first need to consider the *state variation* with respect to α and β.

3.3.1 State variation with respect to the state-delay

Define

$$\boldsymbol{\psi}(t) = \begin{cases} \dot{\boldsymbol{\phi}}(t), & \text{if } t \leq 0, \\ \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t), \boldsymbol{u}(t-\beta)), & \text{if } t \in (0, T]. \end{cases}$$

Furthermore, let $\frac{\partial}{\partial \tilde{\boldsymbol{x}}}$ denote differentiation with respect to the delayed state argument. We will use this notation frequently throughout this chapter.

The solution $\boldsymbol{x}(\cdot|\alpha, \beta)$ of system (3.1)-(3.2) is normally viewed as a function of time, with α and β being fixed values. By instead of fixing $t \in (-\infty, T]$, while allowing α and β to vary, we obtain function $\boldsymbol{x}(t|\cdot, \cdot) : [\alpha_{\min}, \alpha_{\max}] \times [\beta_{\min}, \beta_{\max}] \to \mathbb{R}^n$ whose value at (α, β) is $\boldsymbol{x}(t|\alpha, \beta)$. The partial derivative of $\boldsymbol{x}(t|\cdot, \cdot)$ with respect to α is called the *state variation* with respect to α. The following result, which can be proved in a similar manner to the proof of Theorem 2.1 in Chapter 2, gives a method for determining this state variation.

Theorem 3.1. *Let $t \in (0, T]$ be a fixed time point. Then $\boldsymbol{x}(t|\cdot, \cdot)$ is differentiable with respect to the state-delay α. In fact, for each $(\alpha, \beta) \in [\alpha_{\min}, \alpha_{\max}] \times [\beta_{\min}, \beta_{\max}]$,*

$$\frac{\partial \boldsymbol{x}(t|\alpha, \beta)}{\partial \alpha} = \boldsymbol{\Lambda}(t|\alpha, \beta), \tag{3.15}$$

where $\boldsymbol{\Lambda}(\cdot|\alpha, \beta)$ satisfies the auxiliary time-delay system

$$\begin{aligned} \dot{\boldsymbol{\Lambda}}(t) = {} & \frac{\partial \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t), \boldsymbol{u}(t-\beta))}{\partial \boldsymbol{x}} \boldsymbol{\Lambda}(t) \\ & + \frac{\partial \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t), \boldsymbol{u}(t-\beta))}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\Lambda}(t-\alpha) \\ & - \frac{\partial \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t), \boldsymbol{u}(t-\beta))}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\psi}(t-\alpha) \end{aligned} \tag{3.16}$$

with initial condition

$$\boldsymbol{\Lambda}(t) = \mathbf{0}, \quad t \leq 0. \tag{3.17}$$

According to Theorem 3.1, the state variation with respect to α can be computed by solving the auxiliary time-delay system (3.16)-(3.17). This result is a simple extension of

the main result in Chapter 2, which pertains to systems with multiple state-delays but no input delays. To solve Problem (P), we also need the state variation with respect to β. Unfortunately, the results in Chapter 2, which are based on the assumption that the system dynamics are continuous with respect to the time-delays, cannot be used to determine this state variation. Indeed, since the input function $\boldsymbol{u}$ is discontinuous, the dynamics (3.1) are clearly discontinuous with respect to β. In the next subsection, we describe a new method for computing the state variation with respect to β.

3.3.2 State variation with respect to the input-delay

A Preliminaries

Before deriving the state variation with respect to β, we first need to derive several preliminary results. Let $(\alpha, \beta) \in [\alpha_{\min}, \alpha_{\max}] \times [\beta_{\min}, \beta_{\max}]$ be a fixed pair. Define

$$\mathcal{S} = [\beta_{\min} - \beta, \beta_{\max} - \beta].$$

Note that $\mathcal{S}$ is a non-empty closed interval of positive measure. Clearly,

$$\epsilon \in \mathcal{S} \quad \iff \quad \beta + \epsilon \in [\beta_{\min}, \beta_{\max}].$$

Now, for each $\epsilon \in \mathcal{S}$, define

$$\begin{aligned} \boldsymbol{\varphi}^{\epsilon}(t) &= \boldsymbol{x}(t|\alpha, \beta + \epsilon) - \boldsymbol{x}(t|\alpha, \beta), \quad t \leq T, \\ \boldsymbol{x}^{\epsilon} &= \boldsymbol{x}(t|\alpha, \beta + \epsilon). \end{aligned} \tag{3.18}$$

By (3.2),

$$\boldsymbol{\varphi}^{\epsilon}(t) = \mathbf{0}, \quad t \leq 0. \tag{3.19}$$

Since the system dynamics satisfy the linear growth condition (3.3), it can be shown (see [80]) that there exists a positive real number $L_2 > 0$ such that

$$\left|\boldsymbol{x}(t|\alpha, \beta + \epsilon)\right| \leq L_2, \quad t \in [-\alpha_{\max}, T], \quad \epsilon \in \mathcal{S}. \tag{3.20}$$

Our first preliminary result is stated and proved below.

Lemma 3.1. *There exists a positive real number $L_3 > 0$ such that for all $\epsilon \in \mathcal{S}$ of sufficiently small magnitude,*

$$|\boldsymbol{\varphi}^{\epsilon}(t)| \leq L_3|\epsilon|, \quad t \in (-\infty, T]. \tag{3.21}$$

Proof. Let $\epsilon \in \mathcal{S}$ be such that

$$|\epsilon| < \tfrac{1}{2} \min \left\{ t_i - t_{i-1} \right\}_{i=1}^{p}.$$

For each $i = 1, \ldots, p$, define $I_i = (t_{i-1} + |\epsilon|, t_i - |\epsilon|)$. Furthermore, for each $i = 0, \ldots, p$, define

$$J_i = \begin{cases} [t_0, t_0 + |\epsilon|], & i = 0, \\ [t_i - |\epsilon|, t_i + |\epsilon|], & i = 1, \ldots, p-1, \\ [t_p - |\epsilon|, t_p], & i = p. \end{cases}$$

Note that $\{I_i\}_{i=1}^{p}$ and $\{J_i\}_{i=0}^{p}$ form a partition of $[-\beta_{\max}, T]$. Also, $|J_i| \leq 2|\epsilon|$, $i = 0, \ldots, p$, and

$$\boldsymbol{u}(s) = \boldsymbol{u}(s - \epsilon) = \boldsymbol{\sigma}^i, \quad s \in I_i, \quad i = 1, \ldots, p. \tag{3.22}$$

Now, if $t \leq 0$, then $\boldsymbol{\varphi}^\epsilon(t) = \mathbf{0}$ and the proof is complete. Thus, assume that $t > 0$. Then

$$\begin{aligned} |\boldsymbol{\varphi}^\epsilon(t)| &\leq |\boldsymbol{x}^\epsilon(t) - \boldsymbol{x}(t)| \\ &\leq \int_0^t \Big| \boldsymbol{f}(\boldsymbol{x}^\epsilon(s), \boldsymbol{x}^\epsilon(s-\alpha), \boldsymbol{u}(s), \boldsymbol{u}(s-\beta-\epsilon)) - \boldsymbol{f}(\boldsymbol{x}(s), \boldsymbol{x}(s-\alpha), \boldsymbol{u}(s), \boldsymbol{u}(s-\beta)) \Big| ds, \end{aligned}$$

where $\boldsymbol{x}^\epsilon(s) = \boldsymbol{x}(s|\alpha, \beta + \epsilon)$ and $\boldsymbol{x}(s) = \boldsymbol{x}(s|\alpha, \beta)$.

Thus, since $\boldsymbol{x}^\epsilon$ is uniformly bounded with respect to $\epsilon \in \mathcal{S}$ (recall (3.20)) and $\boldsymbol{f}$ is continuously differentiable, there exists a constant $M_1 > 0$ such that

$$|\boldsymbol{\varphi}^\epsilon(t)| \leq M_1 \int_0^t |\boldsymbol{\varphi}^\epsilon(s)| ds + M_1 \int_0^t |\boldsymbol{\varphi}^\epsilon(s-\alpha)| ds + M_1 \int_0^t |\boldsymbol{u}(s-\beta-\epsilon) - \boldsymbol{u}(s-\beta)| ds.$$

By shifting the time variable in the second and third integrals and then using (3.19), we obtain

$$\begin{aligned} |\boldsymbol{\varphi}^\epsilon(t)| &\leq M_1 \int_0^t |\boldsymbol{\varphi}^\epsilon(s)| ds + M_1 \int_{-\alpha}^{t-\alpha} |\boldsymbol{\varphi}^\epsilon(s)| ds + M_1 \int_{-\beta}^{t-\beta} |\boldsymbol{u}(s-\epsilon) - \boldsymbol{u}(s)| ds \\ &\leq 2M_1 \int_0^t |\boldsymbol{\varphi}^\epsilon(s)| ds + M_1 \int_{-\beta}^{t-\beta} |\boldsymbol{u}(s-\epsilon) - \boldsymbol{u}(s)| ds \\ &= 2M_1 \int_0^t |\boldsymbol{\varphi}^\epsilon(s)| ds + M_1 \sum_{i=1}^{p} \int_{I_i \cap (-\beta, t-\beta)} |\boldsymbol{u}(s-\epsilon) - \boldsymbol{u}(s)| ds \\ &\qquad + M_1 \sum_{i=0}^{p} \int_{J_i \cap (-\beta, t-\beta)} |\boldsymbol{u}(s-\epsilon) - \boldsymbol{u}(s)| ds. \end{aligned}$$

Hence, by (3.22),

$$|\boldsymbol{\varphi}^\epsilon(t)| \le 2M_1 \int_0^t |\boldsymbol{\varphi}^\epsilon(s)|ds + M_1 M_2 \sum_{i=0}^{p} |J_i|,$$

where $M_2 = \max_{j\neq k} |\boldsymbol{\sigma}^j - \boldsymbol{\sigma}^k|$. Since $|J_i| \le 2|\epsilon|$, we have

$$|\boldsymbol{\varphi}^\epsilon(t)| \le 2M_1 \int_0^t |\boldsymbol{\varphi}^\epsilon(s)|ds + 2(p+1)M_1 M_2 |\epsilon|.$$

Finally, applying the Gronwall-Bellman Lemma [111] yields

$$|\boldsymbol{\varphi}^\epsilon(t)| \le 2(p+1)M_1 M_2 \exp\{2M_1 T\}|\epsilon|.$$

This completes the proof. □

For each $\epsilon \in \mathcal{S}$, define

$$\bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma}) = \boldsymbol{f}\big(\boldsymbol{x}(s) + \eta\boldsymbol{\varphi}^\epsilon(s), \boldsymbol{x}(s-\alpha) + \eta\boldsymbol{\varphi}^\epsilon(s-\alpha), \boldsymbol{u}(s), \boldsymbol{\sigma}\big),$$

where, as in the proof of Lemma 3.1, let $\boldsymbol{x}(t) = \boldsymbol{x}(t|\alpha,\beta)$. Then by the chain rule,

$$\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \eta} = \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \boldsymbol{x}}\boldsymbol{\varphi}^\epsilon(s) + \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \tilde{\boldsymbol{x}}}\boldsymbol{\varphi}^\epsilon(s-\alpha), \tag{3.23}$$

where

$$\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{f}\big(\boldsymbol{x}(s) + \eta\boldsymbol{\varphi}^\epsilon(s), \boldsymbol{x}(s-\alpha) + \eta\boldsymbol{\varphi}^\epsilon(s-\alpha), \boldsymbol{u}(s), \boldsymbol{\sigma}\big)}{\partial \boldsymbol{x}}, \tag{3.24}$$

$$\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \tilde{\boldsymbol{x}}} = \frac{\partial \boldsymbol{f}\big(\boldsymbol{x}(s) + \eta\boldsymbol{\varphi}^\epsilon(s), \boldsymbol{x}(s-\alpha) + \eta\boldsymbol{\varphi}^\epsilon(s-\alpha), \boldsymbol{u}(s), \boldsymbol{\sigma}\big)}{\partial \tilde{\boldsymbol{x}}}. \tag{3.25}$$

We can rewrite (3.23) as follows:

$$\begin{aligned}\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \eta} = {} & \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})}{\partial \boldsymbol{x}}\boldsymbol{\varphi}^\epsilon(s) + \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})}{\partial \tilde{\boldsymbol{x}}}\boldsymbol{\varphi}^\epsilon(s-\alpha) \\ & + \Delta_1(s,\eta,\boldsymbol{\sigma}) + \Delta_2(s,\eta,\boldsymbol{\sigma}),\end{aligned} \tag{3.26}$$

where

$$\Delta_1(s,\eta,\boldsymbol{\sigma}) = \Big\{\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \boldsymbol{x}} - \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})}{\partial \boldsymbol{x}}\Big\}\boldsymbol{\varphi}^\epsilon(s), \tag{3.27}$$

$$\Delta_2(s,\eta,\boldsymbol{\sigma}) = \Big\{\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \tilde{\boldsymbol{x}}} - \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})}{\partial \tilde{\boldsymbol{x}}}\Big\}\boldsymbol{\varphi}^\epsilon(s-\alpha). \tag{3.28}$$

Since $\boldsymbol{f}$ is continuously differentiable and $\boldsymbol{x}$ and $\boldsymbol{u}$ are bounded, the following result is easily established.

Lemma 3.2. *For each $\boldsymbol{\sigma} \in \mathbb{R}^r$, there exists a corresponding $L_4 > 0$ such that*

$$\left|\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})}{\partial \boldsymbol{x}}\right| \le L_4, \quad \left|\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})}{\partial \tilde{\boldsymbol{x}}}\right| \le L_4, \quad s \in [0,T], \tag{3.29}$$

where $|\cdot|$ denotes the Euclidean norm on $\mathbb{R}^{n\times n}$.

We now show that the functions Δ_1 (defined by (3.27)) and Δ_2 (defined by (3.28)) are of order ϵ.

Lemma 3.3. *Let $\delta > 0$ and $\boldsymbol{\sigma} \in \mathbb{R}^r$ be arbitrary. Then for any $\epsilon \in \mathcal{S}$ of sufficiently small magnitude,*

$$|\Delta_1(s,\eta,\boldsymbol{\sigma})| \le L_3\delta|\epsilon|, \quad |\Delta_2(s,\eta,\boldsymbol{\sigma})| \le L_3\delta|\epsilon|,$$

where $L_3 > 0$ is as defined in Lemma 3.1.

Proof. By (3.21), $\boldsymbol{x}(s)+\eta\boldsymbol{\varphi}^\epsilon(s) \to \boldsymbol{x}(s)$ and $\boldsymbol{x}(s-\alpha)+\eta\boldsymbol{\varphi}^\epsilon(s-\alpha) \to \boldsymbol{x}(s-\alpha)$ uniformly on $[0,T]$ as $\epsilon \to 0$. Hence, since $\boldsymbol{f}$ is continuously differentiable and $\boldsymbol{x}^\epsilon$ is uniformly bounded with respect to ϵ, there exists an $\epsilon' > 0$ such that for any $\epsilon \in \mathcal{S}$ satisfying $|\epsilon| < \epsilon'$,

$$\left|\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \boldsymbol{x}} - \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})}{\partial \boldsymbol{x}}\right| < \delta,$$
$$\left|\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \tilde{\boldsymbol{x}}} - \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})}{\partial \tilde{\boldsymbol{x}}}\right| < \delta.$$

By taking the norm of (3.27)-(3.28), and then using the above inequalities together with (3.21), we obtain the desired result. □

Let a and b be given constants such that $a, b \in [0,T]$. Define

$$\rho_\epsilon(a,b,\boldsymbol{\sigma}) = \int_a^b \left\{\bar{\boldsymbol{f}}^\epsilon(s,1,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(a,0,\boldsymbol{\sigma})\right\} ds. \tag{3.30}$$

Our final preliminary result is stated and proved below.

Lemma 3.4. *For each $\boldsymbol{\sigma} \in \mathbb{R}^r$, there exists a corresponding $L_5 > 0$ such that for all $\epsilon \in \mathcal{S}$ of sufficiently small magnitude,*

$$\left|\rho_\epsilon(a,b,\boldsymbol{\sigma})\right| \le L_5|b-a|\cdot|\epsilon| + L_5(b-a)^2 + L_5\int_{\min\{a,b\}}^{\max\{a,b\}} \left|\boldsymbol{u}(s)-\boldsymbol{u}(a)\right| ds.$$

Proof. From (3.30),

$$\rho_\epsilon(a,b,\boldsymbol{\sigma}) = \int_a^b \left\{\bar{\boldsymbol{f}}^\epsilon(s,1,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})\right\} ds + \int_a^b \left\{\bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(a,0,\boldsymbol{\sigma})\right\} ds.$$

Thus,

$$\begin{aligned}\big|\rho_\epsilon(a,b,\boldsymbol{\sigma})\big| \leq \int_{\min\{a,b\}}^{\max\{a,b\}} \big|\bar{\boldsymbol{f}}^\epsilon(s,1,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})\big| ds \\ + \int_{\min\{a,b\}}^{\max\{a,b\}} \big|\bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(a,0,\boldsymbol{\sigma})\big| ds.\end{aligned} \tag{3.31}$$

Consider the first integrand on the right-hand side of (3.31). Using (3.26) and (3.29) yields

$$\begin{aligned}\big|\bar{\boldsymbol{f}}^\epsilon(s,1,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})\big| &\leq \int_0^1 \left|\frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,\eta,\boldsymbol{\sigma})}{\partial \eta}\right| d\eta \\ &\leq \int_0^1 \big\{|\Delta_1(s,\eta,\boldsymbol{\sigma})| + |\Delta_2(s,\eta,\boldsymbol{\sigma})|\big\} d\eta \\ &\quad + L_4|\boldsymbol{\varphi}^\epsilon(s)| + L_4|\boldsymbol{\varphi}^\epsilon(s-\alpha)|.\end{aligned}$$

By Lemma 3.1 and Lemma 3.3 with $\delta = 1$, we see that for any $\epsilon \in \mathcal{S}$ of sufficiently small magnitude,

$$\big|\bar{\boldsymbol{f}}^\epsilon(s,1,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma})\big| \leq 2L_3|\epsilon| + 2L_3L_4|\epsilon|. \tag{3.32}$$

Now, consider the second integrand on the right-hand side of (3.31). Since $\boldsymbol{f}$ is continuously differentiable and $\boldsymbol{x}^\epsilon$ is uniformly bounded with respect to ϵ (recall (3.20)), there exists a constant $M_3 > 0$ such that

$$\big|\bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(a,0,\boldsymbol{\sigma})\big| \leq M_3|\boldsymbol{x}(s) - \boldsymbol{x}(a)| + M_3|\boldsymbol{x}(s-\alpha) - \boldsymbol{x}(a-\alpha)| + M_3|\boldsymbol{u}(s) - \boldsymbol{u}(a)|.$$

Note that $\dot{\boldsymbol{x}}(s) = \boldsymbol{\psi}(s)$ for almost all $s \in (-\infty, T]$, where $\boldsymbol{\psi}$ is as defined in Subsection 3.3.1. Thus,

$$\begin{aligned}\big|\bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma}) - \bar{\boldsymbol{f}}^\epsilon(a,0,\boldsymbol{\sigma})\big| &\leq M_3 \int_{\min\{a,s\}}^{\max\{a,s\}} |\boldsymbol{\psi}(\eta)| d\eta + M_3 \int_{\min\{a,s\}-\alpha}^{\max\{a,s\}-\alpha} |\boldsymbol{\psi}(\eta)| d\eta \\ &\quad + M_3|\boldsymbol{u}(s) - \boldsymbol{u}(a)| \\ &\leq M_3M_4|s-a| + M_3M_4|s-a| + M_3|\boldsymbol{u}(s) - \boldsymbol{u}(a)|,\end{aligned} \tag{3.33}$$

where $M_4 = \max_{\eta \in [-\alpha_{\max}, T]} |\boldsymbol{\psi}(\eta)|$. Substituting (3.32) and (3.33) into (3.31) gives

$$|\rho_\epsilon(a,b,\boldsymbol{\sigma})| \leq (2L_3 + 2L_3L_4)|b-a| \cdot |\epsilon| + 2M_3M_4(b-a)^2 + M_3 \int_{\min\{a,b\}}^{\max\{a,b\}} |\boldsymbol{u}(s) - \boldsymbol{u}(a)| ds.$$

Taking $L_5 = \max\{2L_3 + 2L_3L_4, 2M_3M_4, M_3\}$ completes the proof. □

B Main result

Equipped with Lemmas 3.1-3.4, we are now ready to derive the state variation with respect to the input-delay β. First, define

$$\mathcal{I} = \{t_i + \beta,\ i = 0, \dots, p\}.$$

Consider the following auxiliary system:

$$\begin{aligned}\dot{\boldsymbol{\Gamma}}(t) = &\frac{\partial \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t), \boldsymbol{u}(t-\beta))}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}(t) \\ &+ \frac{\partial \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-\alpha), \boldsymbol{u}(t), \boldsymbol{u}(t-\beta))}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\Gamma}(t-\alpha),\end{aligned} \tag{3.34}$$

where, for each $t \in \mathcal{I} \cap (0, T]$,

$$\lim_{t \to (t_i+\beta)+} \boldsymbol{\Gamma}(t) = \lim_{t \to (t_i+\beta)-} \boldsymbol{\Gamma}(t) + \bar{\boldsymbol{f}}^{\epsilon}(t_i + \beta, 0, \boldsymbol{\sigma}^i) - \bar{\boldsymbol{f}}^{\epsilon}(t_i + \beta, 0, \boldsymbol{\sigma}^{i+1}), \tag{3.35}$$

and

$$\boldsymbol{\Gamma}(t) = \mathbf{0}, \quad t \leq 0. \tag{3.36}$$

Let $\boldsymbol{\Gamma}(\cdot|\alpha, \beta)$ denote the unique right continuous solution of (3.34)-(3.36). We have the following important result.

Theorem 3.2. *Let $(\alpha, \beta) \in [\alpha_{\min}, \alpha_{\max}] \times [\beta_{\min}, \beta_{\max})$ be a fixed pair such that*

$$t_i + \beta \notin \{0\} \cup \{t_j, j = 0, \dots, p\}, \quad i = 0, \dots, p.$$

Furthermore, consider a fixed time point $t \in (t_{i-1} + \beta, t_i + \beta) \cap (0, T]$, where $i \in \{1, \dots, p\}$. Then

$$\lim_{\epsilon \to 0+} \epsilon^{-1} \boldsymbol{\varphi}^{\epsilon}(t) = \boldsymbol{\Gamma}(t|\alpha, \beta), \tag{3.37}$$

where $\boldsymbol{\varphi}^{\epsilon}$ is as defined in (3.18).

Proof. Let

$$a_i = \max\{t_{i-1} + \beta, 0\}.$$

Then

$$\boldsymbol{x}(t) = \boldsymbol{x}(a_i) + \int_{a_i}^{t} \bar{\boldsymbol{f}}^{\epsilon}(s, 0, \boldsymbol{\sigma}^i) ds. \tag{3.38}$$

Let $\epsilon \in \mathcal{S}$ be sufficiently small so that $0 < \epsilon < \min\{t_j - t_{j-1}\}_{j=1}^{p}$ and $t > t_{i-1} + \beta + \epsilon$.

Define

$$a_i^\epsilon = \max\{t_{i-1} + \beta + \epsilon, 0\}.$$

Then

$$\boldsymbol{x}^\epsilon(t) = \boldsymbol{x}^\epsilon(a_i^\epsilon) + \int_{a_i^\epsilon}^t \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^i)ds. \tag{3.39}$$

We can write (3.39) as follows:

$$\begin{aligned}\boldsymbol{x}^\epsilon(t) &= \boldsymbol{x}^\epsilon(a_i) + \int_{a_i}^{a_i^\epsilon} \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^{i-1})ds + \int_{a_i^\epsilon}^t \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^i)ds \\ &= \boldsymbol{x}^\epsilon(a_i) + \int_{a_i}^{a_i^\epsilon} \big\{\bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^i)\big\}ds + \int_{a_i}^t \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^i)ds, \end{aligned} \tag{3.40}$$

where $\boldsymbol{\sigma}^{i-1}$ is arbitrary if $i = 1$ (in this case, we must have $a_i^\epsilon = a_i = 0$ when ϵ is sufficiently small, because $\beta < \beta_{\max}$). From (3.38) and (3.40), we have

$$\begin{aligned}\boldsymbol{\varphi}^\epsilon(t) &= \boldsymbol{x}^\epsilon(t) - \boldsymbol{x}(t) \\ &= \boldsymbol{\varphi}^\epsilon(a_i) + \int_{a_i}^t \big\{\bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^i) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^i)\big\}ds + \int_{a_i}^{a_i^\epsilon} \big\{\bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^i)\big\}ds.\end{aligned}$$

Thus,

$$\begin{aligned}\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{\varphi}^\epsilon(a_i) &+ \int_{a_i}^t \big\{\bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^i) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^i)\big\}ds - \rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^i) + \rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^{i-1}) \\ &+ (a_i^\epsilon - a_i)\big\{\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^i)\big\},\end{aligned}$$

where ρ_ϵ is as defined in (3.30). By the fundamental theorem of calculus,

$$\begin{aligned}\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{\varphi}^\epsilon(a_i) &+ \int_{a_i}^t \int_0^1 \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, \eta, \boldsymbol{\sigma}^i)}{\partial \eta} d\eta ds - \rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^i) + \rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^{i-1}) \\ &+ (a_i^\epsilon - a_i)\big\{\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^i)\big\}.\end{aligned}$$

Using (3.26),

$$\begin{aligned}\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{\varphi}^\epsilon(a_i) &+ \int_{a_i}^t \int_0^1 \big\{\Delta_1(s, \eta, \boldsymbol{\sigma}^i) + \Delta_2(s, \eta, \boldsymbol{\sigma}^i)\big\}d\eta ds \\ &+ \int_{a_i}^t \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^i)}{\partial \boldsymbol{x}} \boldsymbol{\varphi}^\epsilon(s)ds + \int_{a_i}^t \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^i)}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\varphi}^\epsilon(s - \alpha)ds - \rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^i) \\ &+ \rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^{i-1}) + (a_i^\epsilon - a_i)\big\{\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^i)\big\}.\end{aligned} \tag{3.41}$$

We can express the solution of the auxiliary system as follows:

$$\boldsymbol{\Gamma}(t) = \boldsymbol{\Gamma}(a_i^+) + \int_{a_i}^t \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma}^i)}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}(s) ds + \int_{a_i}^t \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s,0,\boldsymbol{\sigma}^i)}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\Gamma}(s-\alpha) ds. \tag{3.42}$$

Thus, from Lemma 3.2 and equations (3.41) and (3.42),

$$\begin{aligned}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Gamma}(t)\big| \le \big|\boldsymbol{\gamma}_i(\epsilon)\big| &+ \epsilon^{-1}\int_{a_i}^t \int_0^1 \big\{|\Delta_1(s,\eta,\boldsymbol{\sigma}^i)| + |\Delta_2(s,\eta,\boldsymbol{\sigma}^i)|\big\} d\eta ds \\ &+ \int_{a_i}^t L_4 \big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\big| ds + \int_{a_i}^t L_4 \big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\alpha) - \boldsymbol{\Gamma}(s-\alpha)\big| ds \\ &+ \epsilon^{-1}\big|\rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^i)\big| + \epsilon^{-1}\big|\rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^{i-1})\big|,\end{aligned}$$

where L_4 is the constant defined in Lemma 3.2 and

$$\boldsymbol{\gamma}_i(\epsilon) = \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(a_i) - \boldsymbol{\Gamma}(a_i^+) + \epsilon^{-1}(a_i^\epsilon - a_i)\big\{\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^i)\big\}. \tag{3.43}$$

Recall that $a_i^\epsilon - a_i \le \epsilon$ and $t_{i-1} + \beta \ne t_j$ for all j. Thus, we may assume that ϵ is sufficiently small so that $\boldsymbol{u}(s) = \boldsymbol{u}(a_i)$ for all $s \in [a_i, a_i^\epsilon]$. It then follows from Lemma 3.4 that

$$\big|\rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^{i-1})\big| \le 2L_5'\epsilon^2, \quad \big|\rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^i)\big| \le 2L_5''\epsilon^2,$$

where L_5' and L_5'' are the constants in Lemma 3.4 corresponding to $\boldsymbol{\sigma}^{i-1}$ and $\boldsymbol{\sigma}^i$, respectively. By the above inequalities and Lemma 3.3, assuming that ϵ is sufficiently small,

$$\begin{aligned}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Gamma}(t)\big| \le 2TL_3\delta + 2L_5'\epsilon + 2L_5''\epsilon + \big|\boldsymbol{\gamma}_i(\epsilon)\big| &+ \int_{a_i}^t L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\big| ds \\ &+ \int_{a_i}^t L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\alpha) - \boldsymbol{\Gamma}(s-\alpha)\big| ds,\end{aligned} \tag{3.44}$$

where $\delta > 0$ is arbitrary and L_3 is the constant defined in Lemma 3.1. Performing a change of variable in the second integral on the right-hand side of (3.44) yields

$$\begin{aligned}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Gamma}(t)\big| &\le 2TL_3\delta + 4L_5\epsilon + \big|\boldsymbol{\gamma}_i(\epsilon)\big| + \int_{a_i}^t L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\big| ds \\ &\quad + \int_{a_i-\alpha}^{t-\alpha} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\big| ds \\ &\le 2TL_3\delta + 4L_5\epsilon + \big|\boldsymbol{\gamma}_i(\epsilon)\big| + \mu_i(\epsilon) + \int_{a_i}^t 2L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\big| ds,\end{aligned}$$

where $L_5 = \max\{L_5', L_5''\}$ and

$$\mu_i(\epsilon) = \int_{a_i-\alpha}^{a_i} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\big| ds.$$

Assuming that δ is sufficiently small so that $a_i + \delta \le t$,

$$\begin{aligned}\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Gamma}(t)\right| \le 2TL_3\delta + 4L_5\epsilon + \int_{a_i}^{a_i+\delta} 2L_4\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\right|ds \\ + \left|\boldsymbol{\gamma}_i(\epsilon)\right| + \mu_i(\epsilon) + \int_{a_i+\delta}^{t} 2L_4\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\right|ds.\end{aligned} \tag{3.45}$$

Now, since $\boldsymbol{\Gamma}$ is a piecewise continuous function, there exists a constant $M_1 > 0$ such that

$$|\boldsymbol{\Gamma}(s)| \le M_1, \quad s \in (-\infty, T].$$

Therefore, it follows from Lemma 3.1 that for all sufficiently small $\epsilon > 0$,

$$\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\right| \le L_3 + M_1, \quad s \in (-\infty, T]. \tag{3.46}$$

Substituting (3.46) into (3.45) gives

$$\begin{aligned}\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Gamma}(t)\right| \le 2TL_3\delta + 4L_5\epsilon + \left|\boldsymbol{\gamma}_i(\epsilon)\right| + \mu_i(\epsilon) + 2L_4(L_3 + M_1)\delta \\ + \int_{a_i+\delta}^{t} 2L_4\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\right|ds.\end{aligned} \tag{3.47}$$

Note that this inequality holds for all $t \in [a_i + \delta, t_i + \beta)$ and $t = (t_i + \beta)^-$, uniformly with respect to $\epsilon \le \delta$. Thus, by the Gronwall-Bellman Lemma [111],

$$\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Gamma}(t)\right| \le \left(2TL_3\delta + 4L_5\epsilon + |\boldsymbol{\gamma}_i(\epsilon)| + \mu_i(\epsilon) + 2L_4(L_3 + M_1)\delta\right)\exp\{2L_4T\}. \tag{3.48}$$

This inequality holds for all ϵ of sufficiently small magnitude.

Now, suppose that $t \in (t_{i-1} + \beta, t_i + \beta) \cap (0, T]$ for $i = \min\{j : t_j + \beta > 0\}$. Then $a_i = 0$, and thus by (3.2) and (3.36),

$$\mu_i(\epsilon) = \int_{-\alpha}^{0} L_4\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\right|ds = 0.$$

Since by assumption $t_{i-1} + \beta < 0$, $a_i^\epsilon = a_i = 0$ for all sufficiently small ϵ. Thus,

$$\boldsymbol{\gamma}_i(\epsilon) = \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(0) - \boldsymbol{\Gamma}(0^+) = \mathbf{0}.$$

Substituting $\mu_i(\epsilon) = 0$ and $\boldsymbol{\gamma}_i(\epsilon) = \mathbf{0}$ into (3.48) gives

$$\left|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) - \boldsymbol{\Gamma}(t)\right| \le (2TL_3\delta + 4L_5\epsilon + 2L_4(L_3 + M_1)\delta)\exp\{2L_4T\}. \tag{3.49}$$

Since $\delta > 0$ was chosen arbitrarily and ϵ can be made arbitrarily small, this shows that (3.37) holds for $i = \min\{j : t_j + \beta > 0\}$. Moreover, the derivation leading to (3.49) shows

that (3.37) also holds for $t = (t_i+\beta)^-$. It is also clear that (3.37) holds for all $t \in (-\infty, 0]$.

Now, suppose that (3.37) holds for all $t \in (-\infty, t_k+\beta) \setminus \{t_j+\beta\}_{j=0}^k$ and $t = (t_k+\beta)^-$, where

$$\min\{j : t_j + \beta > 0\} \le k \le p-1. \tag{3.50}$$

We will show that (3.37) holds for all $t \in (t_k+\beta, t_{k+1}+\beta)$ and $t = (t_{k+1}+\beta)^-$. The result will then follow by induction.

Let $t \in (t_k+\beta, t_{k+1}+\beta)$, where k satisfies (3.50). By our inductive hypothesis, for almost all $s \in (-\infty, t_k+\beta)$,

$$\lim_{\epsilon\to 0+} \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) = \boldsymbol{\Gamma}(s). \tag{3.51}$$

In view of (3.46) and (3.51), applying Lebesgue's dominated convergence theorem gives

$$\lim_{\epsilon\to 0+} \mu_{k+1}(\epsilon) = \lim_{\epsilon\to 0+} \int_{t_k+\beta-\alpha}^{t_k+\beta} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s)\big| ds = 0. \tag{3.52}$$

Furthermore,

$$\begin{aligned}\boldsymbol{\gamma}_{k+1}(\epsilon) &= \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t_k+\beta) - \boldsymbol{\Gamma}((t_k+\beta)^+) + \bar{\boldsymbol{f}}^\epsilon(t_k+\beta, 0, \boldsymbol{\sigma}^k) - \bar{\boldsymbol{f}}^\epsilon(t_k+\beta, 0, \boldsymbol{\sigma}^{k+1}) \\ &= \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t_k+\beta) - \boldsymbol{\Gamma}((t_k+\beta)^-).\end{aligned}$$

Thus, by our inductive hypothesis,

$$\lim_{\epsilon\to 0+} \boldsymbol{\gamma}_{k+1}(\epsilon) = \mathbf{0}. \tag{3.53}$$

By combining equations (3.52) and (3.53) with (3.48) for $i = k+1$, we see that (3.37) holds for $t \in (t_k+\beta, t_{k+1}+\beta)$. Similar arguments show that (3.37) also holds for $t = (t_{k+1}+\beta)^-$. The proof then follows by induction. □

Theorem 3.2 shows that $\epsilon^{-1}\boldsymbol{\varphi}^\epsilon \to \boldsymbol{\Gamma}(\cdot|\alpha,\beta)$ as $\epsilon \to 0+$. We now derive the analogous result for $\epsilon \to 0-$.

Theorem 3.3. *Let $(\alpha,\beta) \in [\alpha_{\min}, \alpha_{\max}] \times (\beta_{\min}, \beta_{\max}]$ be a fixed pair such that*

$$t_i + \beta \notin \{0\} \cup \{t_j, j = 0, \dots, p\}, \quad i = 0, \dots, p.$$

Furthermore, consider a fixed time point $t \in (t_{i-1}+\beta, t_i+\beta) \cap (0, T]$, where $i \in \{1, \dots, p\}$. Then

$$\lim_{\epsilon\to 0-} \epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{\Gamma}(t|\alpha,\beta). \tag{3.54}$$

Proof. Let a_i and a_i^ϵ be as defined in the proof of Theorem 3.2. Furthermore, let $\epsilon \in \mathcal{S}$

be such that $\min\{t_{j-1}-t_j\}_{j=1}^{p} < \epsilon < 0$ and $t < t_i + \beta + \epsilon$. Then

$$\boldsymbol{x}(t) = \boldsymbol{x}(a_i^\epsilon) + \int_{a_i^\epsilon}^{a_i} \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i-1})ds + \int_{a_i}^{t} \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i})ds,$$

where $\boldsymbol{\sigma}^{i-1}$ is arbitrary if $i = 1$ (in this case, we must have $a_i = a_i^\epsilon = 0$). Moreover,

$$\boldsymbol{x}^\epsilon(t) = \boldsymbol{x}^\epsilon(a_i^\epsilon) + \int_{a_i^\epsilon}^{a_i} \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^{i})ds + \int_{a_i}^{t} \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^{i})ds.$$

Thus,

$$\begin{aligned}\boldsymbol{\varphi}^\epsilon(t) &= \boldsymbol{x}^\epsilon(t) - \boldsymbol{x}(t)\\ &= \boldsymbol{\varphi}^\epsilon(a_i^\epsilon) + \int_{a_i^\epsilon}^{a_i} \big\{\bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^{i}) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i-1})\big\}ds + \int_{a_i}^{t} \big\{\bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^{i}) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i})\big\}ds.\end{aligned}$$

This equation can be rewritten as follows:

$$\begin{aligned}\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{\varphi}^\epsilon(a_i^\epsilon) &- \rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^i) + (a_i - a_i^\epsilon)\big\{\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i}) - \bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1})\big\}\\ &+ \int_{a_i^\epsilon}^{a_i} \big\{\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i-1})\big\}ds + \int_{a_i}^{t} \big\{\bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^{i}) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i})\big\}ds,\end{aligned}$$

where ρ_ϵ is as defined in (3.30). Using the fundamental theorem of calculus and (3.26),

$$\begin{aligned}\boldsymbol{\varphi}^\epsilon(t) = \boldsymbol{\varphi}^\epsilon(a_i^\epsilon) &- \rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^i) + \int_{a_i^\epsilon}^{a_i} \big\{\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i-1})\big\}ds\\ &+ (a_i - a_i^\epsilon)\big\{\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i}) - \bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1})\big\} + \int_{a_i}^{t} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i})}{\partial \boldsymbol{x}} \boldsymbol{\varphi}^\epsilon(s)ds\\ &+ \int_{a_i}^{t} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i})}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\varphi}^\epsilon(s-\alpha)ds + \int_{a_i}^{t}\int_{0}^{1} \big\{\Delta_1(s, \eta, \boldsymbol{\sigma}^{i}) + \Delta_2(s, \eta, \boldsymbol{\sigma}^{i})\big\}d\eta ds.\end{aligned}$$

Note that $t_{i-1} + \beta \neq t_j$ for all j. Thus, we may assume that ϵ is sufficiently small so that $\boldsymbol{u}(s) = \boldsymbol{u}(a_i)$ for all $s \in [a_i^\epsilon, a_i]$. It then follows from Lemma 3.4 that

$$\big|\rho_\epsilon(a_i, a_i^\epsilon, \boldsymbol{\sigma}^{i-1})\big| \leq 2L_5\epsilon^2.$$

Furthermore, assuming that ϵ is sufficiently small, by using a similar arguments to those in the proof of Lemma 3.4, one can show that there exists a constant $M_5 > 0$ such that

$$\int_{a_i^\epsilon}^{a_i} \big|\bar{\boldsymbol{f}}^\epsilon(a_i, 0, \boldsymbol{\sigma}^{i-1}) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^{i-1})\big|ds \leq M_5\epsilon^2.$$

Hence, as in the proof of Theorem 3.2,

$$\begin{aligned}\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t)-\boldsymbol{\Gamma}(t)\big| \le M_5|\epsilon|+2L_5|\epsilon|+2L_3T\delta+\int_{a_i}^{t} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s)-\boldsymbol{\Gamma}(s)\big|ds \\ +\big|\boldsymbol{\gamma}_i(\epsilon)\big|+\int_{a_i}^{t} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s-\alpha)-\boldsymbol{\Gamma}(s-\alpha)\big|ds,\end{aligned} \tag{3.55}$$

where $\delta>0$ is arbitrary and

$$\boldsymbol{\gamma}_i(\epsilon)=\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(a_i^\epsilon)-\boldsymbol{\Gamma}(a_i^+)+\epsilon^{-1}(a_i-a_i^\epsilon)\big\{\bar{\boldsymbol{f}}^\epsilon(a_i,0,\boldsymbol{\sigma}^i)-\bar{\boldsymbol{f}}^\epsilon(a_i,0,\boldsymbol{\sigma}^{i-1})\big\}.$$

Simplifying (3.55) gives

$$\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t)-\boldsymbol{\Gamma}(t)\big| \le M_5|\epsilon|+2L_5|\epsilon|+2L_3T\delta+\big|\boldsymbol{\gamma}_i(\epsilon)\big|+\mu_i(\epsilon)+\int_{a_i}^{t} 2L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s)-\boldsymbol{\Gamma}(s)\big|ds,$$

where

$$\mu_i(\epsilon)=\int_{a_i-\alpha}^{a_i} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s)-\boldsymbol{\Gamma}(s)\big|ds.$$

Finally, by applying Gronwall's Lemma [111] yields

$$\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t)-\boldsymbol{\Gamma}(t)\big| \le \big(M_5|\epsilon|+2L_5|\epsilon|+2L_3T\delta+\big|\boldsymbol{\gamma}_i(\epsilon)\big|+\mu_i(\epsilon)\big)\exp\{2L_4T\}. \tag{3.56}$$

In particular, this inequality also holds for all $t=a_i^+$, assuming that ϵ is of sufficiently small magnitude.

Now, suppose $t\in(t_{i-1}+\beta,t_i+\beta)\cap(0,T]$ for $i=\min\{j:t_j+\beta>0\}$. Then $a_i=0$, and thus by (3.2) and (3.36),

$$\mu_i(\epsilon)=\int_{-\alpha}^{0} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(s)-\boldsymbol{\Gamma}(s)\big|ds=0.$$

Also, since $a_i^\epsilon=a_i=0$, $\boldsymbol{\gamma}_i(\epsilon)=\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(0)-\boldsymbol{\Gamma}(0^+)=\mathbf{0}$. Substituting $\mu_i(\epsilon)=0$ and $\boldsymbol{\gamma}_i(\epsilon)=\mathbf{0}$ into (3.56) gives

$$\big|\epsilon^{-1}\boldsymbol{\varphi}^\epsilon(t)-\boldsymbol{\Gamma}(t)\big| \le (M_5|\epsilon|+2L_5|\epsilon|+2L_3T\delta)\exp\{2L_4T\}.$$

Since $\delta>0$ was chosen arbitrarily and ϵ can be made arbitrarily small, it follows that (3.54) holds for $i=\min\{j:t_j+\beta>0\}$. It is clear that (3.54) also holds for all $t\in(-\infty,0]$, and for $t=a_i^+$.

Now, suppose that (3.54) holds for all $t\in(-\infty,t_k+\beta)\setminus\{t_j+\beta\}_{j=0}^k$ and $t=a_k^+$, where

$$\min\{j:t_j+\beta>0\}\le k\le p-1. \tag{3.57}$$

We will show that (3.54) holds for all $t \in (t_k + \beta, t_{k+1} + \beta)$ and $t = a_{k+1}^+$. The result will then follow by induction.

Let $t \in (t_k + \beta, t_{k+1} + \beta)$, where k satisfies (3.57) above. By our inductive hypothesis, for almost all $s \in (-\infty, t_k + \beta)$,

$$\lim_{\epsilon \to 0-} \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(s) = \boldsymbol{\Gamma}(s). \tag{3.58}$$

Thus, as in the proof of Theorem 3.2, we can apply Lebesgue's dominated convergence theorem to obtain

$$\lim_{\epsilon \to 0+} \mu_{k+1}(\epsilon) = \int_{t_k+\beta-\alpha}^{t_k+\beta} L_4 \big| \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s) \big| ds = 0. \tag{3.59}$$

We have

$$\begin{aligned} \boldsymbol{\gamma}_{k+1}(\epsilon) &= \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(t_k + \beta + \epsilon) - \boldsymbol{\Gamma}((t_k + \beta)^+) + \bar{\boldsymbol{f}}^\epsilon(t_k + \beta, 0, \boldsymbol{\sigma}^k) - \bar{\boldsymbol{f}}^\epsilon(t_k + \beta, 0, \boldsymbol{\sigma}^{k+1}) \\ &= \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(t_k + \beta + \epsilon) - \boldsymbol{\Gamma}((t_k + \beta)^-). \end{aligned}$$

Hence,

$$\begin{aligned} \boldsymbol{\gamma}_{k+1}(\epsilon) &= \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(a_k) - \boldsymbol{\Gamma}(a_k^+) + \epsilon^{-1} \int_{a_k}^{t_k+\beta+\epsilon} \big\{ \bar{\boldsymbol{f}}^\epsilon(s, 1, \boldsymbol{\sigma}^k) - \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k) \big\} ds \\ &\quad - \int_{a_k}^{t_k+\beta} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k)}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}(s) ds - \int_{a_k}^{t_k+\beta} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k)}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\Gamma}(s - \alpha) ds \\ &= \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(a_k) - \boldsymbol{\Gamma}(a_k^+) + \epsilon^{-1} \int_{a_k}^{t_k+\beta+\epsilon} \int_0^1 \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, \eta, \boldsymbol{\sigma}^k)}{\partial \eta} d\eta ds \\ &\quad - \int_{a_k}^{t_k+\beta} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k)}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}(s) ds - \int_{a_k}^{t_k+\beta} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k)}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\Gamma}(s - \alpha) ds. \end{aligned}$$

Using (3.26), we obtain

$$\begin{aligned} \boldsymbol{\gamma}_{k+1}(\epsilon) &= \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(a_k) - \boldsymbol{\Gamma}(a_k^+) + \epsilon^{-1} \int_{a_k}^{t_k+\beta+\epsilon} \int_0^1 \big\{ \Delta_1(s, \eta, \boldsymbol{\sigma}^k) + \Delta_2(s, \eta, \boldsymbol{\sigma}^k) \big\} d\eta ds \\ &\quad + \int_{a_k}^{t_k+\beta+\epsilon} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k)}{\partial \boldsymbol{x}} \big\{ \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(s) - \boldsymbol{\Gamma}(s) \big\} ds \\ &\quad + \int_{a_k}^{t_k+\beta+\epsilon} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k)}{\partial \tilde{\boldsymbol{x}}} \big\{ \epsilon^{-1} \boldsymbol{\varphi}^\epsilon(s - \alpha) - \boldsymbol{\Gamma}(s - \alpha) \big\} ds \\ &\quad - \int_{t_k+\beta+\epsilon}^{t_k+\beta} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k)}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}(s) ds - \int_{t_k+\beta+\epsilon}^{t_k+\beta} \frac{\partial \bar{\boldsymbol{f}}^\epsilon(s, 0, \boldsymbol{\sigma}^k)}{\partial \tilde{\boldsymbol{x}}} \boldsymbol{\Gamma}(s - \alpha) ds. \end{aligned}$$

Thus, using Lemma 3.2 and Lemma 3.3,

$$\begin{aligned}\big|\boldsymbol{\gamma}_{k+1}(\epsilon)\big| &= \big|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(a_k) - \boldsymbol{\Gamma}(a_k^+)\big| + 2L_3T\delta + \int_{a_k}^{t_k+\beta+\epsilon} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(s) - \boldsymbol{\Gamma}(s)\big|ds \\ &\quad + \int_{a_k}^{t_k+\beta+\epsilon} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(s-\alpha) - \boldsymbol{\Gamma}(s-\alpha)\big|ds + L_4M_1|\epsilon| + L_4M_1|\epsilon| \\ &\leq \big|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(a_k) - \boldsymbol{\Gamma}(a_k^+)\big| + 2L_3T\delta + 2L_4M_1|\epsilon| \\ &\quad + \int_{a_k}^{t_k+\beta} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(s) - \boldsymbol{\Gamma}(s)\big|ds + \int_{a_k}^{t_k+\beta} L_4\big|\epsilon^{-1}\boldsymbol{\varphi}^{\epsilon}(s-\alpha) - \boldsymbol{\Gamma}(s-\alpha)\big|ds,\end{aligned}$$

where M_1 is as defined in the proof of Theorem 3.2. Using the Lebesgue's dominated convergence theorem and the induction hypothesis, the two integrals converge to zero as $\epsilon \to 0-$. It follows also from the induction hypothesis that the first term converges to zero as $\epsilon \to 0-$. Thus,

$$\lim_{\epsilon\to 0-} \big|\boldsymbol{\gamma}_{k+1}(\epsilon)\big| = 0. \tag{3.60}$$

Combining equations (3.59) and (3.60) with (3.56), for $i = k+1$, we see that (3.54) holds for all $t \in (t_k+\beta, t_{k+1}+\beta)$ and $t = (t_k+\beta)^+$. The proof then follows by induction. □

Together, Theorems 3.2 and 3.3 show that the state variation with respect to β is given by $\boldsymbol{\Gamma}(\cdot|\alpha,\beta)$. This is stated formally in the following theorem.

Theorem 3.4. *Let $(\alpha,\beta) \in [\alpha_{\min},\alpha_{\max}] \times [\beta_{\min},\beta_{\max}]$ be a fixed pair such that*

$$t_i + \beta \notin \{0\} \cup \{t_j, j = 0,\ldots,p\}, \quad i = 0,\ldots,p.$$

Furthermore, consider a fixed time point $t \in (t_{i-1}+\beta, t_i+\beta) \cap (0,T]$, where $i \in \{1,\ldots,p\}$. Then

$$\frac{\partial \boldsymbol{x}(t|\alpha,\beta)}{\partial\beta} = \boldsymbol{\Gamma}(t|\alpha,\beta). \tag{3.61}$$

3.4 Computation algorithm

In this section, based on the results in Section 3.3, we develop a computational algorithm for solving Problem (P). Our approach is to view Problem (P) as a nonlinear programming problem in which α and β are decision variables to be chosen optimally. On this basis, Problem (P) can, in principle, be solved using standard nonlinear programming algorithms such as the SQP method, which relies on the partial derivatives of the cost function to compute search directions leading to profitable areas of the search space. Thus, to solve Problem (P) as a nonlinear programming problem, we need to derive the partial derivatives of J with respect to both α and β.

By using the state variation formulae in Theorems 3.1 and 3.4, we can differentiate J using the chain rule. However, the state variation with respect to β does not exist for all values of β (recall that Theorem 3.4 is only valid when $t_i + \beta \notin \{0\} \cup \{t_j, j = 0, \ldots, p\}$ for each $i = 0, \ldots, p$). Thus, at each stage of the optimization process, we need to check the condition $t_i + \beta \notin \{0\} \cup \{t_j, j = 0, \ldots, p\}$, and if this condition is not satisfied, then we perturb β by a small amount ϵ. More precisely, we first check to see whether (α, β) is an optimal pair of delay estimates for Problem (P) (i.e. if the value of J, which measures the discrepancy between predicted output and observed system output, is below a desired tolerance). If it is, then we stop. Otherwise, we calculate the following modified value of β:

$$\bar{\beta} = \begin{cases} \beta, & \text{if } t_i + \beta \notin \{0\} \cup \{t_j, j = 0, \ldots, p\} \text{ for each } i = 0, \ldots, p, \\ \beta + \epsilon, & \text{otherwise,} \end{cases} \tag{3.62}$$

where ϵ is a small number chosen to ensure that $t_i + \beta + \epsilon \notin \{0\} \cup \{t_j, j = 0, \ldots, p\}$ for each $i = 0, \ldots, p$, and $\beta + \epsilon \in [\beta_{\min}, \beta_{\max}]$.

Note that the state variation formula in Theorem 3.4 is not applicable at the time points $t = t_i + \bar{\beta}$, $i = 0, \ldots, p$. Thus, if $\tau_j \in \{t_i + \bar{\beta}\}_{i=0}^p$ for some j, where τ_j is the jth sample time, then we will not be able to compute the state variation of $\boldsymbol{x}(\tau_j|\alpha, \bar{\beta})$ with respect to the input-delay. In this case, we need to consider a modified cost function in which the experimental data are slightly perturbed. The perturbation procedure is designed to ensure that none of the new sample times coincide with points in $\{t_i + \bar{\beta}\}_{i=0}^p$. Details are given below.

After arriving at a new delay pair $(\alpha, \bar{\beta})$ at some point during the optimization process, we define the jth perturbed sample time as follows:

$$\bar{\tau}_j = \begin{cases} \tau_j, & \text{if } \tau_j \notin \{t_i + \bar{\beta}\}_{i=0}^p, \\ \tau_j + \delta_j, & \text{if } \tau_j \in \{t_i + \bar{\beta}\}_{i=0}^p, \end{cases} \tag{3.63}$$

where δ_j is a small number chosen such that $\tau_j + \delta_j \in [0, T] \setminus \{t_i + \bar{\beta}\}_{i=0}^p$. The corresponding output vector is defined as follows:

$$\bar{\boldsymbol{y}}^j = \begin{cases} \hat{\boldsymbol{y}}^j, & \text{if } \tau_j \notin \{t_i + \bar{\beta}\}_{i=0}^p, \\ \hat{\boldsymbol{y}}^j + \boldsymbol{\gamma}^j, & \text{if } \tau_j \in \{t_i + \bar{\beta}\}_{i=0}^p, \end{cases} \tag{3.64}$$

where $\boldsymbol{\gamma}^j$ is computed using the original experimental data together with an appropriate interpolation technique. Our new objective function is

$$\bar{J}(\alpha, \bar{\beta}) = \sum_{j=1}^m \left| \boldsymbol{y}(\bar{\tau}_j|\alpha, \bar{\beta}) - \bar{\boldsymbol{y}}^j \right|^2 \approx \sum_{j=1}^m \left| \boldsymbol{y}(\tau_j|\alpha, \bar{\beta}) - \hat{\boldsymbol{y}}^j \right|^2 = J(\alpha, \bar{\beta}). \tag{3.65}$$

Using Theorem 3.1, the partial derivative of $\bar{J}$ with respect to the state delay is given by

$$\begin{aligned}\frac{\partial \bar{J}(\alpha,\bar{\beta})}{\partial \alpha} &= 2\sum_{j=1}^{m}\left(\boldsymbol{g}(\boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta})) - \bar{\boldsymbol{y}}^j\right)^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta}))}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta})}{\partial \alpha} \\ &= 2\sum_{j=1}^{m}\left(\boldsymbol{g}(\boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta})) - \bar{\boldsymbol{y}}^j\right)^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta}))}{\partial \boldsymbol{x}} \boldsymbol{\Lambda}(\bar{\tau}_j|\alpha,\bar{\beta}). \end{aligned} \tag{3.66}$$

The partial derivative of $\bar{J}$ with respect to the input delay can be determined in a similar manner to the derivation of $\frac{\partial \bar{J}}{\partial \alpha}$ given above:

$$\frac{\partial \bar{J}(\alpha,\bar{\beta})}{\partial \beta} = 2\sum_{j=1}^{m}\left(\boldsymbol{g}(\boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta})) - \bar{\boldsymbol{y}}^j\right)^\top \frac{\partial \boldsymbol{g}(\boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta}))}{\partial \boldsymbol{x}} \boldsymbol{\Gamma}(\bar{\tau}_j|\alpha,\bar{\beta}). \tag{3.67}$$

Since it is unlikely that many of the sample times will lie in the set $\{t_i + \bar{\beta}\}_{i=0}^{p}$, there should be no noticeable difference between minimizing $\bar{J}$ and minimizing J. Indeed, our numerical results in the next section indicate that this is precisely the case. Also, the input function can be chosen judiciously during experimentation to minimize the chance of one of the sample times lying in the set $\{t_i+\beta\}_{i=0}^{p}$. Our heuristic optimization strategy for descending from a point (α,β) is described below.

Step 1. Compute the modified input-delay $\bar{\beta}$ according to (3.62).

Step 2. Compute the new experimental data $\{(\bar{\tau}_j, \bar{\boldsymbol{y}}^j)\}_{j=1}^{m}$ using (3.63) and (3.64).

Step 3. Obtain $\boldsymbol{x}(\cdot|\alpha,\bar{\beta})$, $\boldsymbol{\Lambda}(\cdot|\alpha,\bar{\beta})$, and $\boldsymbol{\Gamma}(\cdot|\alpha,\bar{\beta})$ by solving the enlarged time-delay system consisting of the original system (3.1)-(3.2) and the auxiliary systems (3.16)-(3.17) and (3.34)-(3.36).

Step 4. Use $\boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta})$, $j = 1,\ldots,m$ to compute $\boldsymbol{y}(\bar{\tau}_j|\alpha,\bar{\beta})$ through equation (3.9).

Step 5. Use $\boldsymbol{y}(\bar{\tau}_j|\alpha,\bar{\beta})$, $j = 1,\ldots,m$ to compute $\bar{J}(\alpha,\bar{\beta})$ through equation (3.65).

Step 6. Use $\boldsymbol{x}(\bar{\tau}_j|\alpha,\bar{\beta})$, $\boldsymbol{y}(\bar{\tau}_j|\alpha,\bar{\beta})$, $\boldsymbol{\Lambda}(\bar{\tau}_j|\alpha,\bar{\beta})$ and $\boldsymbol{\Gamma}(\bar{\tau}_j|\alpha,\bar{\beta})$, $j = 1,\ldots,m$ to compute $\frac{\partial \bar{J}(\alpha,\bar{\beta})}{\partial \alpha}$ and $\frac{\partial \bar{J}(\alpha,\bar{\beta})}{\partial \beta}$ through equations (3.66) and (3.67).

This procedure can be combined with a standard nonlinear programming software to solve Problem (P) and determine optimal estimates for the time-delays.

3.5 Numerical examples

3.5.1 Example 1: Zinc sulphate purification

For our first example, we consider the industrial zinc sulphate purification process described in [94]. In this process, zinc powder is added to a zinc sulphate electrolyte to

induce deposition of harmful cobalt and cadmium ions. This is a key step in the production of zinc.

The rates of change of cobalt and cadmium ion concentrations in the electrolyte are described by the following differential equations:

$$V\dot{x}_1(t) = Qx_1^0 - Qx_1(t-\alpha) - c_1u(t-\beta)x_1(t-\alpha) + c_2x_2(t-\alpha), \tag{3.68}$$

$$V\dot{x}_2(t) = Qx_2^0 - Qx_2(t-\alpha) - c_3v(t)x_2(t-\alpha) + c_4x_1(t-\alpha), \tag{3.69}$$

and

$$x_1(t) = 3.3 \times 10^{-4}, \quad x_2(t) = 4.0 \times 10^{-3}, \quad t \leq 0, \tag{3.70}$$

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

Furthermore, V is the volume of the reaction tank ($V = 400$ m^3); Q is the flux of solution ($Q = 200$ m^3/h); c_1, c_2, c_3, c_4, 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 ($x_1^0 = 6\times10^{-4}$g/L, $x_2^0 = 9\times10^{-3}$g/L). Reference [94] considers the parameter identification problem for system (3.68)-(3.70) with a given state-delay of $\alpha = 2$ and no input delay (i.e. $\beta = 0$). Here, we assume that there is a non-negligible delay in the addition of zinc powder to the tank. We also assume that the model parameters are equal to the optimal values reported in [94]:

$$c_1 = 7.828 \times 10^{-4}, \quad c_2 = 16.67, \quad c_3 = 2.823 \times 10^{-4}, \quad c_4 = 7.107 \times 10^2. \tag{3.71}$$

These values were obtained using data from a real zinc production factory in China. We assume that the terminal time is $T = 8$, and we set the input variables u and v as equal to the optimal control functions obtained in [94]:

$$u(t) = \sigma^i, \quad t \in [t_{i-1}, t_i), \quad i = 1, \ldots, 8, \tag{3.72}$$

$$v(t) = \bar{\sigma}^i, \quad t \in [t_{i-1}, t_i), \quad i = 1, \ldots, 8, \tag{3.73}$$

where the values of t_i, σ^i, and $\bar{\sigma}^i$, $i = 1, \ldots, 8$, are given in Table 2.1. The output of the system is the concentration of cadmium ions:

$$y(t) = x_2(t). \tag{3.74}$$

Given system (3.68)-(3.70) with data (3.71) and piecewise-constant inputs (3.72)-(3.73), our goal is to identify the delays α and β. We simulate system (3.68)-(3.70) with $[\hat{\alpha}, \hat{\beta}]^\top =$

$[2, 0.25]$ to generate the observed data in Problem (P). The observed data $\hat{y}^j = x_2(\tau_j|\hat{\alpha}, \hat{\beta})$ is sampled at $\tau_j = j/5$, $j = 1, \ldots, 40$. Thus, our identification problem is: choose α and β to minimize

$$J(\alpha, \beta) = \sum_{j=1}^{40} \left| y(\tau_j|\alpha, \beta) - \hat{y}^j \right|^2 = \sum_{j=1}^{40} \left| x_2(\tau_j|\alpha, \beta) - x_2(\tau_j|\hat{\alpha}, \hat{\beta}) \right|^2 \tag{3.75}$$

subject to the dynamic system (3.68)-(3.70).

To solve this problem, we wrote a Matlab program that integrates the SQP optimization method with the gradient computation algorithm described in Section 3.4.

Computational results for different initial guesses are shown in Table 3.1. The output trajectory for the initial guess $(\alpha, \beta) = (3, 3)$ is displayed in Figure 3.1. In Table 3.1 and

Table 3.1: Numerical convergence of the cost values in Example 3.1.

	Initial guess		Cost value at the kth iteration					
No.	α^0	β^0	$k = 0$	$k = 5$	$k = 10$	$k = 20$	N_1	N_2
1	0.5	0.5	5.865×10^{-5}	9.001×10^{-8}	8.396×10^{-25}	1.151×10^{-27}	0	0
2	1.0	1.0	4.171×10^{-5}	6.265×10^{-8}	2.218×10^{-21}	2.287×10^{-34}	3	0
3	3.0	2.0	9.169×10^{-5}	2.007×10^{-5}	6.624×10^{-7}	8.209×10^{-27}	3	0
4	3.0	3.0	7.828×10^{-5}	2.122×10^{-6}	2.318×10^{-8}	1.141×10^{-26}	6	0

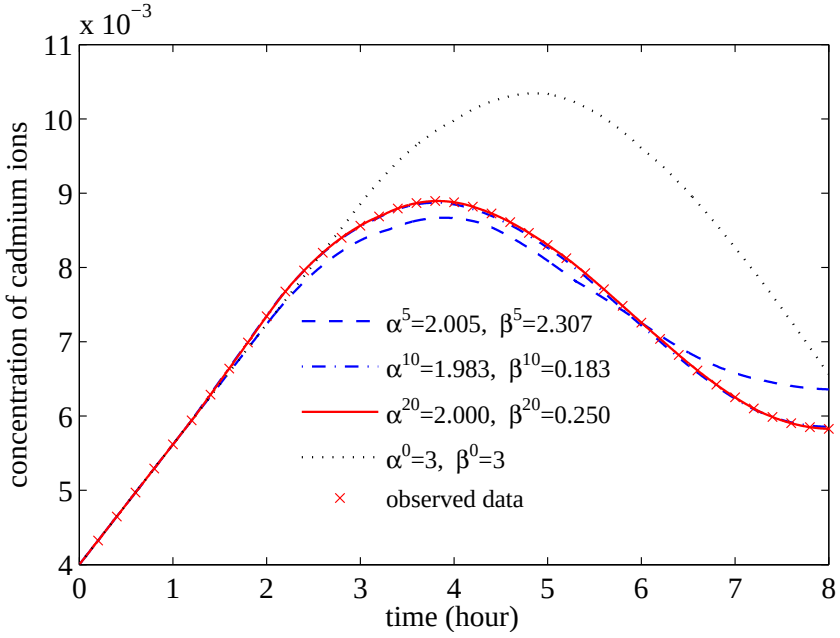

Figure 3.1: Numerical convergence of the output trajectory in Example 3.1 for initial guess No.4.

Figure 3.1, α^k and β^k are the values of α and β at the kth iteration during the optimization process, where α^0 and β^0 (i.e. $k = 0$) denote the initial guesses for the values of α and β. Furthermore, N_1 denotes the number of optimization iterations in which the condition $t_i + \beta \in \{0\} \cup \{t_j, j = 0, \ldots, 8\}$ occurs, and N_2 denotes the number of optimization iterations in which one of the sample times lies in the set $\mathcal{I} = \{t_j + \beta, j = 0, \ldots, 8\}$. We can see from Table 3.1 and Figure 3.1 that the optimal trajectory converges to the observed data for any initial guess. Note that, at each iteration, $\tau_j \in \mathcal{I}$ occurs at most m times (the worst case scenario is when every sample time is in $\mathcal{I}$). However, as expected, N_1 and N_2 are small, and thus the conditions for Theorem 3.4 are satisfied most of the time.

3.5.2 Example 2: Sodium aluminate evaporation

We now consider another industrial chemical process—specifically, the evaporation process described in [119]. This process, which takes place in a series of evaporators, is used to process a mother liquor consisting of sodium carbonate, sodium hydroxide, and alumina. The purpose of this process is to improve the concentration of the mother liquor to reach a specific concentration requirement, so that the sodium hydroxide and alumina components can be re-used. Since it takes time for the solution to flow from one reaction vessel to another, changes in the input variables do not cause changes in the evaporation vessel instantaneously—there are delays in the process. For simplicity, we just consider the case where there are two evaporators. The variables that are of interest are the sodium hydroxide concentration, temperature, and level of solution in each evaporation vessel. The dynamics in these two evaporation vessels can be described by the following differential equations:

$$\frac{dh_1(t)}{dt} = \frac{F_{01}(t-\beta)\rho_0(t-\alpha) + F_2(t-\beta)\rho_2(t-\alpha) - F_1(t)\rho_1(t) + V_0}{A_1\rho_1(t)}, \tag{3.76}$$

$$\frac{dh_2(t)}{dt} = \frac{F_0(t-\beta)\rho_0(t-\alpha) - F_2(t)\rho_2(t)}{A_1\rho_2(t)}, \tag{3.77}$$

$$\frac{dC_1(t)}{dt} = \frac{F_{01}(t-\beta)C_0(t-\alpha) + F_2(t-\beta)C_2(t-\alpha) - F_1(t)C_1(t)}{A_1h_1(t)} - \frac{dh_1(t)}{dt}\frac{C_1(t)}{h_1(t)}, \tag{3.78}$$

$$\frac{dC_2(t)}{dt} = \frac{F_0(t-\beta)C_0(t-\alpha) - F_2(t)C_2(t)}{A_2h_2(t)} - \frac{dh_2(t)}{dt}\frac{C_2(t)}{h_2(t)}, \tag{3.79}$$

$$\frac{dT_i(t)}{dt} = \frac{\Delta Q_i(t)}{A_ih_i(t)c_i^p(t)} - \frac{T_i(t)}{c_i^p(t)}\frac{dc_i^p}{dt} - \frac{T_i(t)\rho_i(t)}{h_i(t)}\frac{dh_i(t)}{dt}, \quad i = 1, 2, \tag{3.80}$$

where i, $i = 1, 2$, refers to the ith evaporator; h_i, T_i, and C_i are the state variables representing the level, temperature, and concentration of the solution in the ith evaporation vessel, respectively; A_i is the cross-sectional area of the ith evaporation vessel; F_i is the flow rate of the solution output from the ith evaporator; V_0 is the amount of vapor from

other heat sources mixed with the solution; F_0 is the flow rate of the feed; C_0 is the concentration of the feed; ΔQ_i is the heat change in the ith evaporation vessel (depends on the live steam flow rate); c_i^p and ρ_i are the specific heat capacity and the density of the solution in the ith evaporation vessel, respectively, and depend on the concentration and temperature. Note that c^p, ρ, and ΔQ are calculated by using the formulae given in [119].

The state vector for this system is $\boldsymbol{x}(t) = [h_1(t), h_2(t), C_1(t), C_2(t), T_1(t), T_2(t)]^\top$. The initial condition is

$$\boldsymbol{x}(t) = [1.91, 110.6, 73.0, 1.91, 97.2, 54.5]^\top, \quad t \leq 0. \tag{3.81}$$

Here, the inputs are $\boldsymbol{u} = [u_1, u_2, u_3, u_4, u_5]^\top = [F_1, F_2, F_0, F_{01}, V]^\top$, where V denotes the live steam flow rate. Also, α is an unknown state-delay, and β is an unknown input-delay. Assume that the terminal time of this system is $T = 240$ minutes. The input functions are

$$u_l(t) = \sigma_l^i, \quad t \in [t_{i-1}, t_i), \quad l = 1, 2, 3, 5, \quad i = 1, \ldots, 24, \tag{3.82}$$

and

$$u_4(t) = 0.165, \quad t \in [0, 240], \tag{3.83}$$

where σ_l^i , $i = 1, \ldots, 24$, are given control heights shown in Figure 3.2. The output is $\boldsymbol{y}(t) = [C_1(t), C_2(t)]^\top$. We use the output trajectory of (3.76)-(3.81) with $[\hat{\alpha}, \hat{\beta}] = [15, 6]^\top$ to generate the observed data for Problem (P). We set

$$\hat{\boldsymbol{y}}^j = [x_3(\tau_j|\hat{\alpha}, \hat{\beta}), x_4(\tau_j|\hat{\alpha}, \hat{\beta})]^\top, \quad j = 1, \ldots, 24,$$

where $\tau_{j+1} - \tau_j = 10$, Thus, our identification problem is: choose α and β to minimize

$$\begin{aligned} J(\alpha, \beta) &= \sum_{j=1}^{24} |\boldsymbol{y}(\tau_j|\alpha, \beta) - \hat{\boldsymbol{y}}^j|^2 \\ &= \sum_{j=1}^{24} |x_3(\tau_j|\alpha, \beta) - x_3(\tau_j|\hat{\alpha}, \hat{\beta})|^2 + \sum_{j=1}^{24} |x_4(\tau_j|\alpha, \beta) - x_4(\tau_j|\hat{\alpha}, \hat{\beta})|^2 \end{aligned} \tag{3.84}$$

subject to the dynamics (3.76)-(3.81).

As with Example 1, we solve this problem using a Matlab program that integrates the SQP optimization method with the gradient computation algorithm described in Section 3.4. The convergence process of the program is shown in Table 3.2 for four sets of initial guesses. The convergence process corresponding to the initial guess of

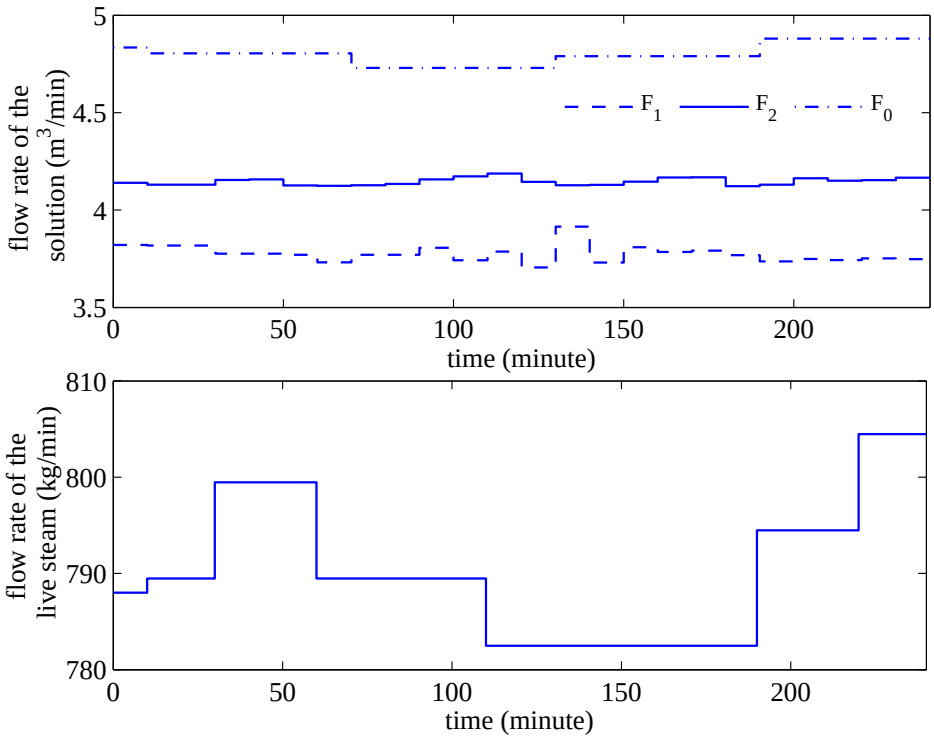

Figure 3.2: Control input variables for Example 3.2

$(\alpha, \beta) = (36, 36)$ is shown in Figure 3.3. In Table 3.2 and Figure 3.3, α^k and β^k are the values of α and β at the kth iteration, where $k = 0$ indicates the initial guess of (α, β). Furthermore, N_1 denotes the number of optimization iterations in which the condition $t_i + \beta \in \{0\} \cup \{t_j, j = 0, \ldots, 24\}$ occurs, and N_2 denotes the number of optimization iterations in which one of the sample times lies in the set $\{t_j + \beta, j = 0, \ldots, 24\}$. We observe that excellent convergence results are achieved for all the initial guesses.

Table 3.2: Numerical convergence of the cost values in Example 3.2.

	Initial guess		Cost value at the kth iteration					
No.	α^0	β^0	$k = 0$	$k = 5$	$k = 10$	$k = 20$	N_1	N_2
1	12	12	0.077	4.462×10^{-3}	2.782×10^{-8}	2.446×10^{-8}	1	0
2	24	24	0.559	4.496×10^{-4}	3.259×10^{-7}	4.684×10^{-8}	2	0
3	30	30	0.939	5.300×10^{-3}	3.259×10^{-7}	4.524×10^{-8}	1	0
4	36	36	1.170	8.751×10^{-1}	1.728×10^{-4}	8.455×10^{-8}	0	0
5	48	48	1.628	8.862×10^{-4}	6.432×10^{-7}	4.802×10^{-8}	0	0

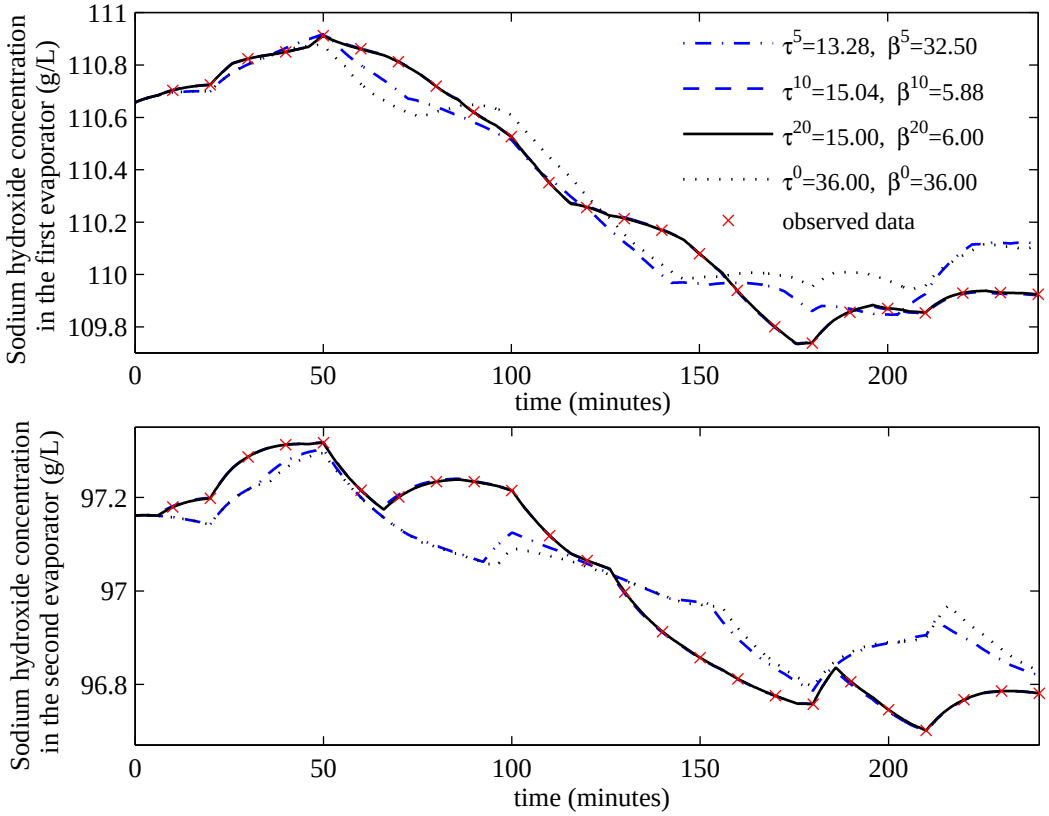

Figure 3.3: Numerical convergence of the output trajectory in Example 3.2 for initial guess No.4.

3.6 Conclusions

In this chapter, we have developed a gradient-based computational method for solving a time-delay identification problem for nonlinear systems in which the input function is piecewise-constant. We assume that there are two unknown time-delays in the system—a state-delay and an input-delay. The problem of determining optimal delay estimates is formulated as a dynamic optimization problem. The gradient of the cost function in this problem is obtained by solving two auxiliary delay-differential systems from $t = 0$ to $t = T$. The auxiliary system corresponding to the input-delay is more complicated and involves jump conditions at the delayed control switching time points. The industrial examples demonstrate that our approach is highly effective. In particular, it converges to optimal delay estimates even when the initial estimates are far away from the optimal values.

CHAPTER 4

Time-delay optimal control application problem: an industrial evaporation process

4.1 Introduction

Time-delay dynamics are encountered in many real-world systems ranging from engineering to economics, such as those reported in [93, 143, 153]. As time-delays will influence the interaction between various system components, control theory and methods developed for dynamical systems without time-delays are not applicable. Thus, new theory and methods have been active research areas over the years and some fundamental and interesting results are now available in the literature (see, for example, [51, 74]). For optimal control problems involving time-delay systems, they have also been extensively studied in the literature such as in [70, 84, 86, 140]. However, most of the results obtained (see, for example, [47, 83, 101]) are for linear dynamical systems. On the computational issues, computational methods are proposed in [78] for optimal control problems with single time-delay. Note that there is no constraint on the state variables in the problem formulation considered in [78], while constraints on the state variables are included in the problem formulation in [119].

In this chapter, we consider a class of optimal control problems involving dynamical systems with multiple time-delays and subject to constraints on the state and/or control variables. Some of these constraints, which are expressed in the form of inequalities, are to be satisfied for all time point over the time planning horizon $[0, t_f]$. These constraints are called the continuous inequality constraints. The objective is to find a control such that a cost function is minimized subject to the given constrains. The focus of this chapter is to develop an effective computational method for solving this difficult constrained time-delay optimal control problem. To begin, the control paremeterization technique is used to subdivide the time planning horizon $[0, t_f]$ into N subintervals. Then, the control is approximated by a piecewise-constant function with possible discontinuities at these partition points. The time-delay optimal control problem is thus approximated by a sequence of time-delay optimal parameter selection problems subject to constraints on the

state and control variables. Amongst these various constraints, the continuous inequality constraints, often involving only state variables, are very difficult to handle directly. Thus, the constraint transcription technique introduced in [80] is used to convert each of them into an equivalent equality constraint in integral form. However, the integrand of each of these equality constraints is non-smooth. Thus, a local smoothing method is used to approximate the non-smooth functions by smooth functions. Consequently, each of these continuous inequality constraints is approximated by a sequence of inequality constraints in integral form, where the integrands are smooth approximating functions. These inequality constraints are known as the inequality constraints in canonical form, as they appear in the same form as the cost function. Then, by using the idea of the penalty function, these inequality constraints are appended to the cost function, forming an augmented cost function. Thus, the constrained time-delay optimal control problem is approximated by a sequence of time-delay optimal parameter selection problems subject to simple bounds on the control parameter vector. Each of them is to be solved as a nonlinear programming problem by using a gradient-based optimization technique, such as the sequential quadratic programming approximation scheme with active set strategy (see, for example, [80,147]). For this, the gradient formula of the augmented cost function with respect to the control parameter vector is derived. On this basis, an effective computational algorithm is developed for the time-delay constrained optimal control problem through solving a sequence of optimal parameter selection problems subject to simple bounds, each of which is regarded as a nonlinear programming problem.

The rest of the chapter is organized as follows. The problem formulation is given in Section 4.2. For the solution method, which consists of constraint transformation, problem approximation, convergence analysis, and computation method, is presented in Section 4.3. For real world application, we consider an optimal control problem of practical alumina evaporation process in Section 4.4, where the objective is to find a control such that the specific requirements of the industrial sodium aluminate solution are met and the solution level for each effect is maintained to within its operation limits with the least energy consumption. In Section 4.5, some concluding remarks are made.

4.2 Problem statement

Consider a process that evolves over the time horizon $[0, t_f]$ as described below:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}\big(t, \boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t), \boldsymbol{u}(t), \tilde{\boldsymbol{u}}(t)\big), \quad t \in [0, t_f], \tag{4.1}$$

where $t_f > 0$ is the *terminal time*; $\boldsymbol{x}(t) = [x_1(t), \ldots, x_n(t)]^\top \in \mathbb{R}^n$ is called the *state*; $\tilde{\boldsymbol{x}}(t) = [(\boldsymbol{x}(t-\alpha_1))^\top, \ldots, (\boldsymbol{x}(t-\alpha_m))^\top]^\top \in \mathbb{R}^{nm}$ is called the *delayed state*. $\boldsymbol{u}(t) = [u_1(t), \ldots, u_r(t)]^\top \in \mathbb{R}^r$ is the *control*; and $\tilde{\boldsymbol{u}}(t) = [(\boldsymbol{u}(t-\beta_1))^\top, \ldots, (\boldsymbol{u}(t-\beta_p))^\top]^\top \in \mathbb{R}^{pr}$

is the *delayed control*. Furthermore, $\boldsymbol{f} : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{nm} \times \mathbb{R}^r \times \mathbb{R}^{pr} \to \mathbb{R}^n$ is a given function. α_i, $i = 1, \ldots, m$, are given state-delays satisfying $0 < \alpha_1 < \cdots < \alpha_m < t_f$; β_j, $j = 1, \ldots, p$, are given control-delays satisfying $0 < \beta_1 < \cdots < \beta_p < t_f$. For brevity, let these time-delays, which are sorted in ascending order, be referred to as τ_k, $k = 1, \ldots, M$. Note that τ_k, $k = 1, \ldots, M$, are not necessarily equal.

The initial functions for the time-delayed differential equations (4.1) are:

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \in [-\tau_M, 0), \tag{4.2}$$

$$\boldsymbol{x}(0) = \boldsymbol{x}^0, \tag{4.3}$$

where $\boldsymbol{\phi}(t) = [\phi_1(t), \ldots, \phi_n(t)]^\top$ is a given continuously differentiable function from $[-\tau_M, 0)$ into $\mathbb{R}^n$; and $\boldsymbol{x}^0 \in \mathbb{R}^n$ is a given vector. In addition, the initial condition for the control is:

$$\boldsymbol{u}(t) = \boldsymbol{\psi}(t), \quad t \in [-\tau_M, 0), \tag{4.4}$$

where $\boldsymbol{\psi}(t) = [\psi_1(t), \ldots, \psi_r(t)]^\top$ is a given piecewise continuous function from $[-\tau_M, 0)$ into $\mathbb{R}^r$.

Define

$$\boldsymbol{U} = \{\boldsymbol{u} = [u_1, \ldots, u_r]^\top \in \mathbb{R}^r : a_l \le u_l \le b_l, \quad l = 1, \ldots, r\}, \tag{4.5}$$

where a_l and b_l, $l = 1, \ldots, r$, are given constants. Clearly, $\boldsymbol{U}$ is a compact and convex subset of $\mathbb{R}^r$. Any measurable function $\boldsymbol{u} = [u_1, \ldots, u_r]^\top : [-\tau_M, t_f] \to \mathbb{R}^r$ such that $\boldsymbol{u}(t) = \boldsymbol{\psi}(t), \forall t \in [-\tau_M, 0)$ and $\boldsymbol{u}(t) \in \boldsymbol{U}$ for almost all $t \in [0, t_f]$, is called an admissible control. Let $\mathcal{U}$ be the set which consists of all such admissible controls.

We assume that the following conditions are satisfied.

(4.A.1). The function $\boldsymbol{f}$ is continuously differentiable with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$ for each $t \in [0, t_f]$, while it is piecewise differentiable with respect to t for each $(\boldsymbol{x}, \boldsymbol{u}) \in \mathbb{R}^n \times \mathbb{R}^r$.

(4.A.2). The function $\boldsymbol{\phi}$ is twice continuously differentiable.

(4.A.3). There exists a real number $L_1 > 0$ such that

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

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

On the basis of assumptions (4.A.1)-(4.A.3), the dynamic system (4.1)-(4.4) admits a unique solution corresponding to each control $\boldsymbol{u} \in \mathcal{U}$ [111]. Let $\boldsymbol{x}(\cdot|\boldsymbol{u})$ denote the corresponding solution of system (4.1) with initial conditions (4.2)-(4.4).

Assume that the following continuous state inequality constraints are to be satisfied:

$$h_i(t, \boldsymbol{x}(t|\boldsymbol{u})) \leq 0, \quad i = 1, \ldots, N_c, \quad \forall t \in [0, t_f]. \tag{4.6}$$

Now, a general class of time-delay optimal control problems with continuous state inequality constraints may be described as follows:

Problem (Q). *Given system* (4.1) *with initial conditions* (4.2)-(4.4)*, find a control* $\boldsymbol{u} \in \mathcal{U}$ *such that the cost function*

$$J_0(\boldsymbol{u}) = \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{u})\big) + \int_0^{t_f} \mathcal{L}_0\big(t, \boldsymbol{x}(t|\boldsymbol{u}), \tilde{\boldsymbol{x}}(t|\boldsymbol{u}), \boldsymbol{u}(t), \tilde{\boldsymbol{u}}(t)\big)dt, \tag{4.7}$$

is minimized subject to the continuous state inequality constraints (4.6)*, where* Φ_0 *is the terminal cost.*

We assume that the following conditions are satisfied.

(4.A.4). For each $i = 1, \ldots, N_c$, the function h_i is continuously differentiable with respect to $(t, \boldsymbol{x}) \in [0, t_f] \times \mathbb{R}^n$.

(4.A.5). The function Φ_0 satisfies the assumption (4.A.2), while the function $\mathcal{L}_0$ satisfies the assumption (4.A.1).

4.3 Solution method

4.3.1 Control parameterization

To solve Problem (Q), we apply the control parametrization scheme to approximate the control $\boldsymbol{u} \in \mathcal{U}$ by a piecewise-constant function with possible discontinuities at the partition points called the switching times. The heights of the piecewise-constant function are decision variables. More specifically, for each $l = 1, \ldots, r$,

$$u_l^N(t) = \sum_{q=1}^{N} \sigma_l^{N,q} \chi_{I_q}(t), \quad t \in [0, t_f], \tag{4.8}$$

where

$$\chi_{I_q}(t) = \begin{cases} 1, & \text{if } t \in I_q, \\ 0, & \text{otherwise,} \end{cases} \tag{4.9}$$

and

$$I_q = \begin{cases} [t_{q-1}, t_q), & \text{if } q \in \{1, 2, \ldots, N-1\}, \\ [t_{q-1}, t_q], & \text{if } q = N. \end{cases} \tag{4.10}$$

Here, t_q, $q = 1, \ldots, N$, are pre-assigned such that $t_{q-1} < t_q$ with $t_0 = 0$ and $t_N = t_f$. For each $l = 1, \ldots, r$, σ_l^q, $q = 1, \ldots, N$, are the heights of the piecewise-constant control component u_l^N.

Let

$$\boldsymbol{\sigma}^{N,q} = [\sigma_1^{N,q}, \ldots, \sigma_r^{N,q}]^\top, \quad q = 1, \ldots, N,$$

and let

$$\boldsymbol{\sigma}^N = [(\boldsymbol{\sigma}^{N,1})^\top, \ldots, (\boldsymbol{\sigma}^{N,N})^\top]^\top.$$

A function $\boldsymbol{u}^N = [u_1^N, \ldots, u_r^N)]^\top$ with u_l^N, $l = 1, \ldots, r$, given by (4.8) and $\boldsymbol{\sigma}^{N,q} \in \boldsymbol{U}$, is called an admissible piecewise-constant control. Let $\mathcal{U}_N$ be the set of all such admissible piecewise-constant controls, and let $\boldsymbol{\Xi}_N$ be the set containing all the corresponding $\boldsymbol{\sigma}^N$, i.e.,

$$\boldsymbol{\Xi}_N = \{\boldsymbol{\sigma}^N = [(\boldsymbol{\sigma}^{N,1})^\top, \ldots, (\boldsymbol{\sigma}^{N,N})^\top]^\top \in \mathbb{R}^{N\times r} : \boldsymbol{\sigma}^{N,q} \in \boldsymbol{U}, \quad q = 1, \ldots, N\}. \tag{4.11}$$

Clearly, each $\boldsymbol{u} \in \mathcal{U}_N$ corresponds to a unique $\boldsymbol{\sigma}^N \in \boldsymbol{\Xi}_N$ and vice versa. For a $\boldsymbol{\sigma}^N \in \boldsymbol{\Xi}_N$, let $\boldsymbol{u}^N(\cdot|\boldsymbol{\sigma}^N)$ denote the corresponding piecewise-constant control in $\mathcal{U}_N$. For each $\boldsymbol{u} = \boldsymbol{u}^N \in \mathcal{U}_N$, let $\boldsymbol{x}(\cdot|\boldsymbol{u}^N)$ be the corresponding solution of system (4.1) with initial conditions (4.3) and (4.4). Furthermore, let $\tilde{\boldsymbol{x}}(\cdot|\boldsymbol{u}^N)$ be the corresponding delayed state. For convenience, $\boldsymbol{x}(\cdot|\boldsymbol{u}^N)$ and $\tilde{\boldsymbol{x}}(\cdot|\boldsymbol{u}^N)$ are written as $\boldsymbol{x}(\cdot|\boldsymbol{\sigma}^N)$ and $\tilde{\boldsymbol{x}}(\cdot|\boldsymbol{\sigma}^N)$, respectively.

From (4.8), we have

$$u_l^N(t - \beta_j) = \sum_{q=1}^{N} \sigma_l^{N,q} \chi_{I_q}(t - \beta_j), \quad t \in I_k, \quad j = 1, \ldots, p, \quad l = 1, \ldots, r. \tag{4.12}$$

For brevity, define

$$\tilde{\boldsymbol{u}}^N(t|\boldsymbol{\sigma}^N) = [\boldsymbol{u}^N(t - \beta_1|\boldsymbol{\sigma}^N)^\top, \ldots, \boldsymbol{u}^N(t - \beta_j|\boldsymbol{\sigma}^N)^\top]^\top,$$

where $\boldsymbol{u}^N(t - \beta_j|\boldsymbol{\sigma}^N) = [u_1^N(t - \beta_j|\boldsymbol{\sigma}^N), \ldots, u_r^N(t - \beta_j|\boldsymbol{\sigma}^N)]^\top$, $j = 1, \ldots, p$. Thus, by applying the control parametrization technique, system (4.1)-(4.3) becomes

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}\big(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N), \tilde{\boldsymbol{x}}(t|\boldsymbol{\sigma}^N), \boldsymbol{u}^N(t|\boldsymbol{\sigma}^N), \tilde{\boldsymbol{u}}^N(t|\boldsymbol{\sigma}^N)\big), \quad t \in [0, t_f], \tag{4.13}$$

with initial conditions

$$\boldsymbol{x}(t) = \boldsymbol{\phi}(t), \quad t \in [-\tau_M, 0), \tag{4.14}$$

$$\boldsymbol{x}(0) = \boldsymbol{x}^0. \tag{4.15}$$

With $\boldsymbol{u} \in \mathcal{U}_N$, the continuous state inequality constraints (4.6) become:

$$h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) \leq 0, \quad i = 1, \ldots, N_c. \tag{4.16}$$

We may now state an approximate problem of Problem (Q) as follows:

Problem (Q_N). *Given system* (4.13) *with initial conditions* (4.14) *and* (4.15)*, find a control vector* $\boldsymbol{\sigma}^N \in \Xi_N$ *such that the following cost function*

$$\tilde{J}_0(\boldsymbol{\sigma}^N) = \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{\sigma}^N)\big) + \int_0^{t_f} \mathcal{L}_0\big(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N), \tilde{\boldsymbol{x}}(t|\boldsymbol{\sigma}^N), \boldsymbol{u}^N(t|\boldsymbol{\sigma}^N), \tilde{\boldsymbol{u}}^N(t|\boldsymbol{\sigma}^N)\big)dt \tag{4.17}$$

is minimized over Ξ_N *subject to constraints* (4.16)*.*

4.3.2 Constraints transformation

For each of the continuous inequality constraints, it contains infinite number of constraints and hence these continuous inequality constraints are difficult to handle directly. Note that each of these continuous inequality constraints (4.16) is equivalent to the following equality constraint in integral form:

$$g_i(\boldsymbol{\sigma}^N) = \int_0^{t_f} \max\big\{h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)), 0\big\} = 0, \quad i = 1, \ldots, N_c. \tag{4.18}$$

Let $\mathcal{B}_N$ be the feasible region defined by

$$\begin{aligned}\mathcal{B}_N &= \{\boldsymbol{\sigma}^N \in \Xi_N : h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) \leq 0, \quad \forall t \in [0, t_f], \quad i = 1, \ldots, N_c\} \\ &= \{\boldsymbol{\sigma}^N \in \Xi_N : g_i(\boldsymbol{\sigma}^N) = 0, \quad i = 1, \ldots, N_c\}.\end{aligned}$$

Furthermore, let $\mathcal{B}_N^0$ denote the interior of $\mathcal{B}_N$, i.e.,

$$\mathcal{B}_N^0 = \{\boldsymbol{\sigma}^N \in \Xi_N : h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) < 0, \quad \forall t \in [0, t_f], \quad i = 1, \ldots, N_c\}.$$

Note that the sets $\mathcal{B}_N$ and $\mathcal{B}_N^0$ are defined for the control parameter vectors in Ξ_N. The corresponding subsets in $\mathcal{U}_N$ are denoted by $\mathcal{F}_N$ and $\mathcal{F}_N^0$, respectively.

We assume that the following assumptions are satisfied.

(4.A.6). $\mathcal{B}_N^0 \neq \emptyset$.

(4.A.7). Suppose that $\boldsymbol{\sigma}^N \in \mathcal{B}_N$. Then, there exists a control vector $\bar{\boldsymbol{\sigma}} \in \mathcal{B}_N^0$ such that $\alpha\bar{\boldsymbol{\sigma}} + (1-\alpha)\boldsymbol{\sigma}^N \in \mathcal{B}_N^0$ for all $\alpha \in (0, 1]$.

Since $\max\big\{h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)), 0\big\}$, $i = 1, \ldots, N_c$, are non-smooth, they are approximated

by the following smooth functions (see Chapter 8 of [80])

$$\mathcal{L}_{i,\varepsilon}(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N)) = \begin{cases} h_i(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N)), & \text{if } h_i(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N)) > \varepsilon, \\ \frac{\left\{h_i(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N))-\varepsilon\right\}^2}{4\varepsilon}, & \text{if } -\varepsilon \le h_i(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N)) \le \varepsilon, \\ 0, & \text{if } h_i(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N)) < -\varepsilon, \end{cases} \tag{4.19}$$

where $\varepsilon > 0$ is a smoothing parameter controlling the accuracy of the approximation. Thus, constraints (4.18) are approximated by

$$g_{i,\varepsilon}(\boldsymbol{\sigma}^N) = \int_0^{t_f} \mathcal{L}_{i,\varepsilon}(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N)) = 0, \quad i = 1,\ldots,N_c, \tag{4.20}$$

Let $\mathcal{B}_{N,\varepsilon}$ be the feasible region defined by

$$\begin{aligned} \mathcal{B}_{N,\varepsilon} &= \{\boldsymbol{\sigma}^N \in \Xi_N : g_{i,\varepsilon}(\boldsymbol{\sigma}^N) = 0, \quad i = 1,\ldots,N_c\} \\ &= \{\boldsymbol{\sigma}^N \in \Xi_N : h_i(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N)) \le -\varepsilon, \quad t \in [0,t_f], \quad i = 1,\ldots,N_c\}. \end{aligned}$$

Clearly, $\mathcal{B}_{N,\varepsilon} \subset \mathcal{B}_N$ for each $\varepsilon > 0$.

However, it can be shown that constraints (4.20) fail to satisfy the usual constraint qualification (see [82, 89]) because when any of the constraints is satisfied as an equality, its gradient is a zero vector (see Chapter 6 of [80]). It is well-known in the optimization literature that any gradient-based optimization technique will not work well for optimization problems with constraints which fail to satisfy the usual constraint qualification. Here, the concept of the penalty function approach is used to append the approximate constraints to the cost function, forming an augmented cost function given below:

$$\begin{aligned} \tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N) = \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{\sigma}^N)\big) &+ \int_0^{t_f} \mathcal{L}_0\big(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N),\tilde{\boldsymbol{x}}(t|\boldsymbol{\sigma}^N),\boldsymbol{u}^N(t|\boldsymbol{\sigma}^N),\tilde{\boldsymbol{u}}^N(t|\boldsymbol{\sigma}^N)\big)dt \\ &+ \gamma \sum_{i=1}^{N_c} \int_0^{t_f} \mathcal{L}_{i,\varepsilon}\big(t,\boldsymbol{x}(t|\boldsymbol{\sigma}^N)\big)dt, \end{aligned} \tag{4.21}$$

where $\gamma > 0$ is the penalty factor of the continuous state inequality constraints.

In this way, Problem (Q_N) is approximated by a sequence of approximate problems given below:

Problem ($\mathbf{Q}_{N,\varepsilon,\gamma}$). *Given system* (4.13) *with initial conditions* (4.14) *and* (4.15), *find a control vector* $\boldsymbol{\sigma}^N \in \boldsymbol{\Xi}_N$ *such that the augmented cost function* (4.21) *is minimized.*

Problem $(Q_{N,\varepsilon,\gamma})$ can be regarded as a nonlinear optimization problem subject to simple bounds on the decision variables specified by (4.5). It can be solved by a gradient-based optimization technique, such as the sequential quadratic approximate scheme with an active set strategy.

4.3.3 Convergence analysis

In this subsection, we shall show that the optimal cost of Problem $(Q_{N,\varepsilon,\gamma})$ converges to the optimal cost of the original problem (Q).

For each $N \in \{2, 3, \ldots\}$, let S^N denote the set containing the partition points t_q, $q = 1, \ldots, N$. It is chosen such that $S^{N+1} \supset S^N$ and $\lim_{N\to\infty} S^N$ is dense in $[0, t_f]$. Let $\boldsymbol{\sigma}^{N,*}$ be an optimal control vector to Problem (Q_N) and let $\boldsymbol{u}^{N,*}$ be the corresponding piecewise-constant control in $\mathcal{U}$, (in fact, it is in $\mathcal{U}_N$). Thus, for each integer $N \geq 2$, $\boldsymbol{u}^{N,*}$ is a suboptimal control to Problem (Q) such that $J_0(\boldsymbol{u}^{N+1,*}) \leq J_0(\boldsymbol{u}^{N,*})$ for all $N \geq 2$.

We have the following result. Its proof is similar to that given for Theorem 6.4.2 in Chapter 6 of [80].

Lemma 4.1. *Let $\{\boldsymbol{u}^N\}_{N=1}^{\infty}$ be a bounded sequence of functions in L_∞^r. Then, the sequence $\{\boldsymbol{x}(\cdot|\boldsymbol{u}^N)\}_{N=1}^{\infty}$ of the corresponding solutions of system* (4.13) *with initial conditions* (4.14)-(4.15) *is also bounded in L_∞^r.*

We now relate the solutions of Problem (Q) and Problem $(Q_{N,\varepsilon,\gamma})$ in the following theorems.

Theorem 4.1. *For any $\varepsilon > 0$, there exists a $\rho(\varepsilon) > 0$ such that for any ρ, $0 < \rho < \rho(\varepsilon)$, if $g_{i,\varepsilon}(\boldsymbol{\sigma}^N) \leq \rho$, $i = 1, \ldots, N_c$, then $\boldsymbol{\sigma}^N \in \mathcal{B}_N$.*

Proof. Since h_i, $i = 1, \ldots, N_c$, are continuously differentiable in $[0, t_f] \times \mathbb{R}^n$ (recall assumption (4.A.4)), it follows that for each i, $i = 1, \ldots, N_c$, and any $\boldsymbol{\sigma}^N \in \Xi_N$, we have

$$\frac{dh_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))}{dt} = \frac{\partial h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}}(t) + \frac{\partial h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))}{\partial t}.$$

By (4.A.3) and the fact that $\boldsymbol{x}(t|\boldsymbol{\sigma}^N) \in \boldsymbol{X}$ all $t \in [0, t_f]$, where $\boldsymbol{X}$ is a bounded set (see Lemma 4.1), there exists a positive constant m_i such that

$$\Big|\frac{dh_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))}{dt}\Big| \leq m_i, \quad \forall t \in [0, t_f]. \tag{4.22}$$

For any $\varepsilon > 0$, define

$$\kappa_{i,\varepsilon} = \frac{\varepsilon}{16} \min\Big\{t_f, \frac{\varepsilon}{2m_i}\Big\}.$$

Let

$$\mathcal{B}_{N,i} = \{\boldsymbol{\sigma}^N \in \Xi_N : g_i(\boldsymbol{\sigma}^N) = 0\},$$

and

$$\mathcal{B}_{N,i,\varepsilon,\rho} = \{\boldsymbol{\sigma}^N \in \Xi_N : g_{i,\varepsilon}(\boldsymbol{\sigma}^N) \leq \rho\},$$

where ρ is a positive real number. It suffices to show that $\mathcal{B}_{N,i,\varepsilon,\rho} \subset \mathcal{B}_{N,i}$ for any ρ such that $0 < \rho < \kappa_{i,\varepsilon}$. We assume the contrary. Then there exists a $\boldsymbol{\sigma}^N \in \Xi_N$ such that

$$g_{i,\varepsilon}(\boldsymbol{\sigma}^N) \leq \rho < \kappa_{i,\varepsilon}, \tag{4.23}$$

but

$$g_i(\boldsymbol{\sigma}^N) > 0. \tag{4.24}$$

Again by the continuity of h_i, it follows from (4.24) that there exists a $\iota \in [0, t_f]$ such that

$$h_i(\iota, \boldsymbol{x}(\iota|\boldsymbol{\sigma}^N)) > 0,$$

and there exists an interval $\mathcal{I}_i \subset [0, t_f]$ such that

$$h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) > -\frac{\varepsilon}{2}, \quad \forall t \in \mathcal{I}_i. \tag{4.25}$$

Using (4.22), it is clear from (4.25) that

$$|\mathcal{I}_i| \geq \min\Big\{t_f, \frac{\varepsilon}{2m_i}\Big\},$$

where $|\mathcal{I}_i|$ denotes the length of the interval $\mathcal{I}_i$. From the definition of $g_{i,\varepsilon}$, we have

$$\begin{aligned} g_{i,\varepsilon}(\boldsymbol{\sigma}^N) &= \int_0^{t_f} h_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))dt \geq \int_{\mathcal{I}_i} h_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))dt \geq \int_{\mathcal{I}_i} \min_{t\in\mathcal{I}_i} h_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))dt \\ &\geq \min_{t\in\mathcal{I}_i}\{(h_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) + \varepsilon)^2/4\varepsilon\}|\mathcal{I}_i| \geq \frac{\varepsilon}{16}\min\Big\{t_f, \frac{\varepsilon}{2m_i}\Big\} = \kappa_{i,\varepsilon}. \end{aligned}$$

This is a contradiction to (4.23). Thus, the proof is completed. □

Theorem 4.2. *For any $\varepsilon > 0$, there exists a $\gamma(\varepsilon) > 0$ such that for any $\gamma > \gamma(\varepsilon)$, if $\boldsymbol{\sigma}^{N,*}_{\varepsilon,\gamma}$ is an optimal control vector of Problem ($Q_{N,\varepsilon,\gamma}$), then it satisfies the continuous inequalities constraints* (4.16) *of Problem (Q_N).*

Proof. Let $\boldsymbol{\sigma}^{N,*}_{\varepsilon,\gamma}$ bet an optimal control vector of Problem ($Q_{N,\varepsilon,\gamma}$). Then, for any $\boldsymbol{\sigma}^N \in \Xi_N$,

$$\tilde{J}_{\varepsilon,\eta}(\boldsymbol{\sigma}^{N,*}_{\varepsilon,\gamma}) = \tilde{J}_0(\boldsymbol{\sigma}^{N,*}_{\varepsilon,\gamma}) + \gamma\sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}^{N,*}_{\varepsilon,\gamma}) \leq \tilde{J}_0(\boldsymbol{\sigma}^N) + \gamma\sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}^N).$$

Let $\boldsymbol{\sigma}^N_\varepsilon \in \mathcal{B}_{N,\varepsilon}$ be fixed. Then, by the definition of $g_{i,\varepsilon}$ given in (4.20), we have

$$\gamma\sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}^N_\varepsilon) = 0.$$

Thus,

$$\tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) + \gamma \sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon}^{N}) + \gamma \sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon}^{N}) = \tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon}^{N}). \tag{4.26}$$

Since Ξ_N is a compact set, there exists, by (4.A.1)-(4.A.3) and (4.A.4)-(4.A.5), a $\bar{\boldsymbol{\sigma}}^N \in \Xi_N$ such that $\tilde{J}_0(\bar{\boldsymbol{\sigma}}^N) \leq \tilde{J}_0(\boldsymbol{\sigma}^N)$ for all $\boldsymbol{\sigma}^N \in \Xi_N$. Clearly,

$$\tilde{J}_0(\bar{\boldsymbol{\sigma}}^N) \leq \tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}). \tag{4.27}$$

Then, adding the penalty term $\gamma \sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*})$ to each side of (4.27), we have

$$\tilde{J}_0(\bar{\boldsymbol{\sigma}}^N) + \gamma \sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) + \gamma \sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}).$$

Using (4.26) gives

$$\tilde{J}_0(\bar{\boldsymbol{\sigma}}^N) + \gamma \sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon}^{N}). \tag{4.28}$$

Rearranging (4.28),

$$\gamma \sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon}^{N}) - \tilde{J}_0(\bar{\boldsymbol{\sigma}}^N). \tag{4.29}$$

Letting $z = \tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon}^{N}) - \tilde{J}_0(\bar{\boldsymbol{\sigma}}^N)$, we obtain

$$\sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \frac{z}{\gamma}.$$

By choosing $\rho(\varepsilon) = \frac{z}{\gamma(\varepsilon)}$, it follows that for any $\gamma > \gamma(\varepsilon)$,

$$0 < \rho < \rho(\varepsilon)$$

and

$$\sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \rho.$$

Consequently, $g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \rho$, $i = 1, \ldots, N_c$. Hence, it follows from Theorem 4.1 that $\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*} \in \mathcal{B}_N$. This completes the proof. □

Theorem 4.3. *Let $\boldsymbol{\sigma}^{N,*}$ be an optimal control vector of Problem (Q_N) and let $\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}$ be an optimal control vector of Problem ($Q_{N,\varepsilon,\gamma}$), where $\gamma(\varepsilon)$ is chosen such that $\boldsymbol{\sigma}_{\varepsilon,\gamma(\varepsilon)}^{N,*} \in \mathcal{B}_N$. Then,*

$$\lim_{\varepsilon \to 0} \tilde{J}_{\varepsilon,\gamma(\varepsilon)}(\boldsymbol{\sigma}_{\varepsilon,\gamma(\varepsilon)}^{N,*}) = \tilde{J}_0(\boldsymbol{\sigma}^{N,*}).$$

Proof. By (4.A.7), there exists a $\bar{\boldsymbol{\sigma}}^N \in \mathcal{B}_N^0$ such that

$$\boldsymbol{\sigma}_\alpha^N = (1-\alpha)\boldsymbol{\sigma}^{N,*} + \alpha\bar{\boldsymbol{\sigma}}^N \in \mathcal{B}_N^0, \quad \alpha \in (0,1].$$

Now, for any $\delta_1 > 0$, there exists an $\alpha_1 \in (0,1]$ such that

$$\tilde{J}_0(\boldsymbol{\sigma}_\alpha^N) \leq \tilde{J}_0(\boldsymbol{\sigma}^{N,*}) + \delta_1, \quad \forall \alpha \in (0,\alpha_1). \tag{4.30}$$

Choose $\alpha_2 = \alpha_1/2$. Then it is clear that $\boldsymbol{\sigma}_{\alpha_2}^N \in \mathcal{B}_N^0$. Thus, there exists a $\delta_2 > 0$ such that

$$\max\left(h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}_{\alpha_2}^N))\right) < -\delta_2, \quad i = 1,\ldots,N_c.$$

Choosing $\varepsilon = \delta_2$, thus $\boldsymbol{\sigma}_{\alpha_2}^N$ satisfies (4.20). Since, $\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}$ is an optimal control vector of Problem $(Q_{N,\varepsilon,\gamma})$, it follows that

$$\tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) = \tilde{J}_0(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) + \gamma\sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \tilde{J}_0(\boldsymbol{\sigma}_{\alpha_2}^N) + \gamma\sum_{i=1}^{N_c} g_{i,\varepsilon}(\boldsymbol{\sigma}_{\alpha_2}^N) = \tilde{J}_0(\boldsymbol{\sigma}_{\alpha_2}^N).$$

Noting that the penalty term is non-negative, we have

$$\tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \tilde{J}_0(\boldsymbol{\sigma}_{\alpha_2}^N). \tag{4.31}$$

Since, by Theorem 4.2, $\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}$ is a feasible point of Problem (Q_N), it follows from (4.30) and (4.31) that

$$\tilde{J}_0(\boldsymbol{\sigma}^{N,*}) \leq \tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}_{\varepsilon,\gamma}^{N,*}) \leq \tilde{J}_0(\boldsymbol{\sigma}^{N,*}) + \delta_1.$$

Since $\delta_1 > 0$ is arbitrary, letting $\varepsilon \to 0$, the results follows. □

To proceed, we need the following lemmas. The first is quoted from Lemma 6.4.1 in [80].

Lemma 4.2. *For each $\boldsymbol{u} \in \mathcal{U}$, let*

$$\boldsymbol{u}^N(t) = \sum_{q=1}^{N} \boldsymbol{\sigma}^{N,q}\chi_{I_q}(t), \quad t \in [0,t_f], \tag{4.32}$$

where $I_q = [t_{q-1}, t_q)$, $q = 1,\ldots,N$,

$$\boldsymbol{\sigma}^q = \frac{1}{|I_q|}\int_{I_q} \boldsymbol{u}(s)ds, \tag{4.33}$$

and $|I_q| = t_q - t_{q-1}$. Then, $\boldsymbol{u}^N \to \boldsymbol{u}$ almost everywhere in $[0,t_f]$ as $N \to \infty$; and furthermore,

$$\lim_{N\to\infty}\int_0^{t_f} |\boldsymbol{u}^N(t) - \boldsymbol{u}(t)|dt = 0,$$

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

Lemma 4.3. *Let $\{\boldsymbol{u}^N\}_{N=2}^{\infty}$ be a bounded sequence of functions in $\mathcal{L}_{\infty}^{r}$. Suppose that $\boldsymbol{u}^N \to \boldsymbol{u}$ almost everywhere in $[0, t_f]$ as $N \to \infty$. Then,*

$$\lim_{N\to\infty} |\boldsymbol{x}(t|\boldsymbol{u}^N) - \boldsymbol{x}(t|\boldsymbol{u})| = 0,$$

$$\lim_{N\to\infty} J_0(\boldsymbol{u}^N) = J_0(\boldsymbol{u}).$$

Proof. The proof is similar to that given for Lemma 6.4.3 and Lemma 6.4.4 in [80]. □

Define

$$\boldsymbol{\Omega} = \{\boldsymbol{u} \in \mathcal{U} : h_i(t, \boldsymbol{x}(t|\boldsymbol{u})) \leq 0, \quad t \in [0, t_f], \quad i = 1, \ldots, N_c\}.$$

Furthermore, let $\boldsymbol{\Omega}^0$ denote the interior of $\boldsymbol{\Omega}$, i.e.,

$$\boldsymbol{\Omega}^0 = \{\boldsymbol{u} \in \mathcal{U} : h_i(t, \boldsymbol{x}(t|\boldsymbol{u})) < 0, \quad t \in [0, t_f], \quad i = 1, \ldots, N_c\}.$$

We assume that the following assumptions are satisfied.

(4.A.8). $\boldsymbol{\Omega}^0 \neq \emptyset$.

(4.A.9). Suppose that $\boldsymbol{u} \in \boldsymbol{\Omega}$. Then, there exists a control $\bar{\boldsymbol{u}} \in \boldsymbol{\Omega}^0$ such that

$$\alpha\bar{\boldsymbol{u}} + (1-\alpha)\boldsymbol{u} \in \boldsymbol{\Omega}^0, \quad \forall \alpha \in (0, 1].$$

Theorem 4.4. *Suppose $\boldsymbol{u}^*$ be an optimal control of the original problem (Q). Furthermore, for each $N \geq 2$, let $\boldsymbol{u}^{N,*}$ be an optimal piecewise-constant control of Problem (Q_N). Then,*

$$\lim_{N\to\infty} J_0(\boldsymbol{u}^{N,*}) = J_0(\boldsymbol{u}^*).$$

Proof. By (4.A.9), there exists a $\bar{\boldsymbol{u}} \in \boldsymbol{\Omega}^0$ such that

$$\bar{\boldsymbol{u}}_\alpha = \alpha\bar{\boldsymbol{u}} + (1-\alpha)\boldsymbol{u}^* \in \boldsymbol{\Omega}^0, \quad \alpha \in (0, 1]. \tag{4.34}$$

Equation (4.34) implies that $\bar{\boldsymbol{u}}_\alpha \to \boldsymbol{u}^*$ as $\alpha \to 0$ for almost all $t \in [0, t_f]$. Hence, for any real number $\delta > 0$, there exists, by Lemma 4.3, an $\alpha_1 \in (0, 1)$ such that

$$|J_0(\bar{\boldsymbol{u}}_\alpha) - J_0(\boldsymbol{u}^*)| < \frac{\delta}{2}, \quad \forall \quad 0 < \alpha \leq \alpha_1. \tag{4.35}$$

Since $\bar{\boldsymbol{u}}_\alpha \in \boldsymbol{\Omega}^0$, it is clear that for any α, there is a corresponding real number $\rho > 0$ such that

$$h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}_\alpha)) < -\rho, \quad i = 1, \ldots, N_c, \quad t \in [0, t_f). \tag{4.36}$$

We now fix α. Let $\{\bar{\boldsymbol{u}}_\alpha^N\}_{N=2}^\infty$ denote the sequence of piecewise-constant controls constructed form $\bar{\boldsymbol{u}}_\alpha$ according to (4.32)-(4.33). Thus, by Lemma 4.2, $\{\bar{\boldsymbol{u}}_\alpha^N\}_{N=2}^\infty \to \bar{\boldsymbol{u}}_\alpha$ almost everywhere in $[0, t_f]$ as $N \to \infty$. By using (4.A.1) and (4.A.2) and Lemma 4.3, there exists an integer $N_0 \geq 2$, such that for all $N > N_0$

$$|h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}_\alpha^N)) - h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}_\alpha))| < \frac{\rho}{2}, \quad i = 1, \ldots, N_c, \quad t \in [0, t_f). \tag{4.37}$$

From (4.36)-(4.37), it follows that

$$h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}_\alpha^N)) < -\frac{\rho}{2}, \quad i = 1, \ldots, N_c, \quad t \in [0, t_f),$$

for all $N \geq N_0$. This implies that $\bar{\boldsymbol{u}}_\alpha^N \in \mathcal{B}_N^0$. Furthermore, by Lemma 4.3, there exists an $N_1 \geq 2$ such that for all $N > N_1$,

$$|J_0(\bar{\boldsymbol{u}}_\alpha^N) - J_0(\bar{\boldsymbol{u}}_\alpha)| < \frac{\delta}{2}. \tag{4.38}$$

Set $N_2 = \max\{N_0, N_1\}$. Then, it follows from (4.35) and (4.38) that

$$|J_0(\bar{\boldsymbol{u}}_\alpha^N) - J_0(\boldsymbol{u}^*)| \leq |J_0(\bar{\boldsymbol{u}}_\alpha^N) - J_0(\bar{\boldsymbol{u}}_\alpha)| + |J_0(\bar{\boldsymbol{u}}_\alpha) - J_0(\boldsymbol{u}^*)| < \delta, \tag{4.39}$$

for all $N > N_2$. Since $\boldsymbol{u}^*$ is the optimal control of Problem (Q), it follows that

$$J_0(\boldsymbol{u}^*) \leq J_0(\boldsymbol{u}^{N,*}).$$

Furthermore, $\boldsymbol{u}^{N,*}$ is an optimal control of Problem (Q_N). Thus,

$$J_0(\boldsymbol{u}^{N,*}) \leq J_0(\bar{\boldsymbol{u}}_\alpha^N).$$

On the other hand, $\boldsymbol{u}^{N,*}$ is a suboptimal control of Problem (Q). It follows that, for all $N > N_2$,

$$J_0(\boldsymbol{u}^*) \leq J_0(\boldsymbol{u}^{N,*}) \leq J_0(\bar{\boldsymbol{u}}_\alpha^N). \tag{4.40}$$

Combining (4.39) and (4.40) gives

$$J_0(\boldsymbol{u}^*) \leq J_0(\boldsymbol{u}^{N,*}) \leq J_0(\bar{\boldsymbol{u}}_\alpha^N) \leq J_0(\boldsymbol{u}^*) + \delta.$$

Since $\delta > 0$ can be chosen arbitrarily, and $\bar{\boldsymbol{u}}_\alpha^N \in \mathcal{B}_N^0$, it is clear that $J_0(\boldsymbol{u}^{N,*}) \to J_0(\boldsymbol{u}^*)$ as $N \to \infty$. □

Theorem 4.4 indicates that the optimal cost of Problem (Q_N) will converge to the optimal cost of Problem (Q) as $N \to \infty$. However, there is no guarantee that the optimal control of Problem (Q_N) itself will converge to the optimal control of Problem (Q). However, we have the following result.

Theorem 4.5. *let $\boldsymbol{u}^{N,*}$ be a piecewise-constant control constructed from the optimal control vector $\boldsymbol{\sigma}^{N,*}$ of Problem (Q_N). Let $\boldsymbol{u}^*$ be an optimal control of the original problem (Q). If $\{\boldsymbol{u}^{N,*}\}_{N=2}^{\infty}$ converges to $\bar{\boldsymbol{u}} \in \mathcal{U}$ almost everywhere on $[0, t_f]$ as $N \to \infty$, then $\bar{\boldsymbol{u}}$ is also an optimal control of Problem (Q).*

Proof. First, by Lemma 4.3, we have

$$\lim_{N\to\infty} J_0(\boldsymbol{u}^{N,*}) \to J_0(\bar{\boldsymbol{u}}).$$

From Theorem 4.4, we recall that

$$\lim_{N\to\infty} J_0(\boldsymbol{u}^{N,*}) \to J_0(\boldsymbol{u}^*).$$

Since the limit of a convergence sequence is unique, we have

$$J_0(\bar{\boldsymbol{u}}) = J_0(\boldsymbol{u}^*).$$

It remains to show that $\bar{\boldsymbol{u}}$ is a feasible control of Problem (Q). On the contrary, suppose that it is not true. Then, there exists an integer $i \in \{1, \ldots, N_c\}$ and a non-zero interval $\mathcal{I} \subset [0, t_f]$ such that

$$h_i(\bar{t}, \boldsymbol{x}(\bar{t}|\bar{\boldsymbol{u}})) > 0, \quad \forall t \in \mathcal{I}. \tag{4.41}$$

Since, by (4.A.4), h_i is continuous, it follows from Lemma 4.3 that

$$\lim_{N\to\infty} |\boldsymbol{x}(t|\boldsymbol{u}^{N,*}) - \boldsymbol{x}(t|\bar{\boldsymbol{u}})| = 0,$$

for each $t \in [0, t_f]$. Furthermore, $\boldsymbol{x}(t|\boldsymbol{u}^{N,*}) \in \boldsymbol{X}$ for all $t \in [0, t_f]$, where $\boldsymbol{X}$ is a bounded set (see Lemma 4.1). Thus,

$$\lim_{N\to\infty} \int_{\mathcal{I}} |h_i(t, \boldsymbol{x}(t|\boldsymbol{u}^{N,*})) - h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}))| dt = 0.$$

Therefore, if (4.41) is valid, then

$$\int_{\mathcal{I}} h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}))dt = \int_{\mathcal{I}} \{h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}) - h_i(t, \boldsymbol{x}(t|\boldsymbol{u}^{N,*}))\}dt + \int_{\mathcal{I}} h_i(t, \boldsymbol{x}(t|\boldsymbol{u}^{N,*}))dt > 0.$$

Since $\boldsymbol{u}^{N*}$ is an optimal control of Problem (Q_N), we have

$$h_i(t, \boldsymbol{x}(t|\boldsymbol{u}^{N,*})) \leq 0, \quad \forall t \in [0, t_f].$$

Thus,

$$0 < \int_{\mathcal{I}} h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}))dt = \lim_{N\to\infty} \int_{\mathcal{I}} \{h_i(t, \boldsymbol{x}(t|\bar{\boldsymbol{u}}) - h_i(t, \boldsymbol{x}(t|\boldsymbol{u}^{N,*}))\}dt = 0. \tag{4.42}$$

This is a contradiction. Thus, $\bar{\boldsymbol{u}}$ is feasible as required.

□

4.3.4 Computational method

To solve Problem $(Q_{N,\varepsilon,\gamma})$, the gradient formula of the augmented cost function (4.21) with respect to the control vector $\boldsymbol{\sigma}^N$ is needed. It is derived, although rather involved, via variational formulae given below. Let $H_{\varepsilon,\gamma}$ be the Hamiltonian function defined by

$$\begin{aligned} H_{\varepsilon,\gamma} = \mathcal{L}_0(t) + \boldsymbol{\lambda}(t)^\top \boldsymbol{f}(t) + \gamma \sum_{i=1}^{N_c} \mathcal{L}_{i,\varepsilon}\big(t, \boldsymbol{x}(t|\boldsymbol{u})\big) + \sum_{k=1}^{M} \mathcal{L}_0(t+\tau_k) \\ + \sum_{k=1}^{M} (\bar{\boldsymbol{\lambda}}^k)^\top(t) \bar{\boldsymbol{f}}_k(t) e(t_f - t - \tau_k), \end{aligned} \tag{4.43}$$

where the following abbreviations are used

$$\begin{aligned} \boldsymbol{f}(t) &= \boldsymbol{f}\big(t, \boldsymbol{x}(t|\boldsymbol{u}), \tilde{\boldsymbol{x}}(t|\boldsymbol{u}), \boldsymbol{u}(t), \tilde{\boldsymbol{u}}(t)\big) \\ \mathcal{L}_0\big(t\big) &= \mathcal{L}_0\big(t, \boldsymbol{x}(t|\boldsymbol{u}), \tilde{\boldsymbol{x}}(t|\boldsymbol{u}), \boldsymbol{u}(t), \tilde{\boldsymbol{u}}(t)\big), \end{aligned}$$

and for each k, $k = 1, \ldots, M$,

$$\bar{\boldsymbol{\lambda}}^k(t) = \boldsymbol{\lambda}(t+\tau_k), \tag{4.44}$$

$$\bar{\boldsymbol{f}}_k(t) = \boldsymbol{f}\big(t+\tau_k, \boldsymbol{x}(t+\tau_k|\boldsymbol{u}), \tilde{\boldsymbol{x}}(t+\tau_k|\boldsymbol{u}), \boldsymbol{u}(t+\tau_k), \tilde{\boldsymbol{u}}(t+\tau_k)\big), \tag{4.45}$$

$$\mathcal{L}_0\big(t+\tau_k\big) = \mathcal{L}_0\big(t+\tau_k, \boldsymbol{x}(t+\tau_k|\boldsymbol{u}), \tilde{\boldsymbol{x}}(t+\tau_k|\boldsymbol{u}), \boldsymbol{u}(t+\tau_k), \tilde{\boldsymbol{u}}(t+\tau_k)\big),$$

while $e(\cdot)$ is the unit step function defined by

$$e(\cdot) = \begin{cases} 1, & \text{if} \quad t \geq 0, \\ 0, & \text{if} \quad t < 0 \end{cases}$$

and

$$\mathcal{L}_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{u})) = \begin{cases} h_i(t, \boldsymbol{x}(t|\boldsymbol{u})), & \text{if} \quad h_i(t, \boldsymbol{x}(t|\boldsymbol{u})) > \varepsilon, \\ \frac{\left\{h_i(t, \boldsymbol{x}(t|\boldsymbol{u})) - \varepsilon\right\}^2}{4\varepsilon}, & \text{if} \quad -\varepsilon \leq h_i(t, \boldsymbol{x}(t|\boldsymbol{u})) \leq \varepsilon, \\ 0, & \text{if} \quad h_i(t, \boldsymbol{x}(t|\boldsymbol{u})) < -\varepsilon. \end{cases} \tag{4.46}$$

Let $\boldsymbol{\lambda}(t)$ be the corresponding solution of the co-state system defined by

$$\dot{\boldsymbol{\lambda}}(t) = -\frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{x}}, \tag{4.47}$$

with boundary conditions

$$\boldsymbol{\lambda}(t_f) = \frac{\partial \Phi_0(\boldsymbol{x}(t_f|\boldsymbol{u}))}{\partial \boldsymbol{x}}, \tag{4.48}$$

$$\boldsymbol{\lambda}(t) = \mathbf{0}, \quad t > t_f. \tag{4.49}$$

Then, for each pair of ε and γ, we have the following result.

Theorem 4.6. *Let $\boldsymbol{u}$ be any control in $\mathcal{U}$ and let $\Delta\boldsymbol{u}(t) \in \mathbb{R}^r$ be any bounded measurable function defined in $[-\tau_M, t_f]$ with $\Delta\boldsymbol{u}(t) = 0$ for all $t \in [-\tau_M, 0]$. Then, the directional derivative of the function $\tilde{J}_{\varepsilon,\gamma}$ given by (4.21) is:*

$$\Delta J_{\varepsilon,\gamma}(\boldsymbol{u}) = \int_0^{t_f} \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{u}} \Delta\boldsymbol{u}(t)dt,$$

where

$$\begin{aligned} J_{\varepsilon,\gamma}(\boldsymbol{u}) = \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{u})\big) + \int_0^{t_f} \mathcal{L}_0\big(t, \boldsymbol{x}(t|\boldsymbol{u}), \tilde{\boldsymbol{x}}(t|\boldsymbol{u}), \boldsymbol{u}(t), \tilde{\boldsymbol{u}}(t)\big)dt \\ + \gamma \sum_{i=1}^{N_c} \int_0^{t_f} \mathcal{L}_{i,\varepsilon}\big(t, \boldsymbol{x}(t|\boldsymbol{u})\big)dt. \end{aligned} \tag{4.50}$$

Proof. Let $\boldsymbol{u}(t) \in \mathcal{U}$ be arbitrary but fixed. Let the control vector $\boldsymbol{u}(t)$ be perturbed by $\epsilon\Delta\boldsymbol{u}(t)$, where $\epsilon > 0$ is a small real number and $\Delta\boldsymbol{u}(t)$ is an arbitrary but fixed perturbation of $\boldsymbol{u}(t)$ given by

$$\begin{aligned} &\Delta\boldsymbol{u}(t) = [\Delta u_1(t), \Delta u_2(t), \ldots, \Delta u_r(t)]^\top, \quad t \in [0, t_f] \\ &\Delta\boldsymbol{u}(t) = \mathbf{0}, \quad t < 0. \end{aligned}$$

where $\Delta u_j(t)$, $j = 1, \ldots, r$, are arbitrary but given functions. Let,

$$\boldsymbol{u}_\epsilon(t) = \boldsymbol{u}(t) + \epsilon\Delta\boldsymbol{u}(t). \tag{4.51}$$

Furthermore, let

$$\tilde{\boldsymbol{u}}_\epsilon^k(t) = \boldsymbol{u}_\epsilon(t - \tau_k) = \boldsymbol{u}(t - \tau_k) + \epsilon\Delta\boldsymbol{u}(t - \tau_k), \quad k = 1, \ldots, M,$$

and

$$\tilde{\boldsymbol{u}}_\epsilon(t) = [\tilde{\boldsymbol{u}}_\epsilon^1(t)^\top, \ldots, \tilde{\boldsymbol{u}}_\epsilon^M(t)^\top]^\top.$$

For brevity, let $\boldsymbol{x}(\cdot)$ denote the solution of system (4.1)-(4.3) with the control $\boldsymbol{u}$, and $\boldsymbol{x}_\epsilon(\cdot)$ denote the solution of system (4.1)-(4.3) with the control $\boldsymbol{u}_\epsilon$. Clearly,

$$\begin{aligned}\boldsymbol{x}(t) &= \boldsymbol{x}(0) + \int_0^t \boldsymbol{f}\big(s, \boldsymbol{x}(s|\boldsymbol{u}), \tilde{\boldsymbol{x}}(s|\boldsymbol{u}), \boldsymbol{u}(s), \tilde{\boldsymbol{u}}(t)\big)ds,\\ \boldsymbol{x}_\epsilon(t) &= \boldsymbol{x}(0) + \int_0^t \boldsymbol{f}\big(s, \boldsymbol{x}(s|\boldsymbol{u}_\epsilon), \tilde{\boldsymbol{x}}(s|\boldsymbol{u}_\epsilon), \boldsymbol{u}_\epsilon, \tilde{\boldsymbol{u}}_\epsilon(s)\big)ds.\end{aligned}$$

We will use the notation $\frac{\partial}{\partial \tilde{\boldsymbol{x}}^k}$ to denote the partial differentiation with respect to the kth delayed state in $\tilde{\boldsymbol{x}}(t)$ (i.e. the partial differentiation with respect to $\boldsymbol{x}(t-\tau_k)$), and the notation $\frac{\partial}{\partial \tilde{\boldsymbol{u}}^k}$ to denote the partial differentiation with respect to the kth delayed control in $\tilde{\boldsymbol{u}}(t)$ (i.e. the partial differentiation with respect to $\boldsymbol{u}(t-\tau_k)$). Then, by the chain rule, we have

$$\begin{aligned}\Delta\boldsymbol{x}(t) &= \frac{d\boldsymbol{x}_\epsilon(t)}{d\epsilon}\Big|_{\epsilon=0}\\ &= \int_0^t \Big\{\frac{\partial \boldsymbol{f}(s)}{\partial \boldsymbol{x}}\Delta\boldsymbol{x}(s) + \frac{\partial \boldsymbol{f}(s)}{\partial \boldsymbol{u}}\Delta\boldsymbol{u}(s) + \sum_{k=1}^M \frac{\partial \boldsymbol{f}(s)}{\partial \tilde{\boldsymbol{x}}^k}\Delta\boldsymbol{x}(s-\tau_k)\Big\}ds\\ &\quad + \sum_{k=1}^M \int_0^t \frac{\partial \boldsymbol{f}(s)}{\partial \tilde{\boldsymbol{u}}^k}\Delta\boldsymbol{u}(s-\tau_k)ds.\end{aligned}$$

Clearly,

$$\begin{aligned}\frac{d(\Delta\boldsymbol{x}(t))}{dt} &= \frac{\partial \boldsymbol{f}(t)}{\partial \boldsymbol{x}}\Delta\boldsymbol{x}(t) + \frac{\partial \boldsymbol{f}(t)}{\partial \boldsymbol{u}}\Delta\boldsymbol{u}(t) + \sum_{k=1}^M \frac{\partial \boldsymbol{f}(t)}{\partial \tilde{\boldsymbol{x}}^k}\Delta\boldsymbol{x}(t-\tau_k)\\ &\quad + \sum_{k=1}^M \frac{\partial \boldsymbol{f}(t)}{\partial \tilde{\boldsymbol{u}}^k}\Delta\boldsymbol{u}(t-\tau_k).\end{aligned} \tag{4.52}$$

Then, by the chain rule,

$$\begin{aligned}\Delta J_{\varepsilon,\gamma}(\boldsymbol{u}) &= \frac{dJ_{\varepsilon,\gamma}(\boldsymbol{u}_\epsilon)}{d\epsilon}\Big|_{\epsilon=0}\\ &= \frac{\partial \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{u})\big)}{\partial \boldsymbol{x}(t_f)}\Delta\boldsymbol{x}(t_f) + \int_0^{t_f}\Big\{\frac{\partial \mathcal{L}_0(t)}{\partial \boldsymbol{x}}\Delta\boldsymbol{x}(t) + \frac{\partial \mathcal{L}_0(t)}{\partial \boldsymbol{u}}\Delta\boldsymbol{u}(t)\Big\}dt\\ &\quad + \int_0^{t_f}\Big\{\sum_{k=1}^M \frac{\partial \mathcal{L}_0(t)}{\partial \tilde{\boldsymbol{x}}^k}\Delta\boldsymbol{x}(t-\tau_k) + \sum_{k=1}^M \frac{\partial \mathcal{L}_0(t)}{\partial \tilde{\boldsymbol{u}}^k}\Delta\boldsymbol{u}(t-\tau_k)\Big\}dt\\ &\quad + \gamma\sum_{i=1}^{N_c}\int_0^{t_f}\frac{\partial \mathcal{L}_{i,\varepsilon}\big(t, \boldsymbol{x}(t|\boldsymbol{u})\big)}{\partial \boldsymbol{x}}\Delta\boldsymbol{x}(t)dt.\end{aligned} \tag{4.53}$$

Consider the definition of $\bar{\boldsymbol{f}}_k(t)$ in (4.45), we have

$$\sum_{k=1}^{M} \frac{\partial \mathcal{L}_0(t)}{\partial \tilde{\boldsymbol{x}}^k} \Delta \boldsymbol{x}(t-\tau_k) = \sum_{k=1}^{M} e(t_f - t - \tau_k) \frac{\partial \mathcal{L}_0(t+\tau_k)}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t), \tag{4.54}$$

$$\sum_{k=1}^{M} \frac{\partial \mathcal{L}_0(t)}{\partial \tilde{\boldsymbol{u}}^k} \Delta \boldsymbol{u}(t-\tau_k) = \sum_{k=1}^{M} e(t_f - t - \tau_k) \frac{\partial \mathcal{L}_0(t+\tau_k)}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t). \tag{4.55}$$

Substituting (4.54)-(4.55) into (4.56) yields,

$$\begin{aligned}
\Delta J_{\varepsilon,\gamma}(\boldsymbol{u}) = & \frac{\partial \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{u})\big)}{\partial \boldsymbol{x}(t_f)} \Delta \boldsymbol{x}(t_f) + \int_0^{t_f} \frac{\partial \mathcal{L}_0(t)}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) dt \\
& + \sum_{k=1}^{M} \int_0^{t_f} \frac{\partial \mathcal{L}_0(t+\tau_k)}{\partial \boldsymbol{u}} e(t_f - t - \tau_k) \Delta \boldsymbol{u}(t) dt + \int_0^{t_f} \frac{\partial \mathcal{L}_0(t)}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t) dt \\
& + \int_0^{t_f} \Big\{ \sum_{k=1}^{M} \frac{\partial \mathcal{L}_0(t+\tau_k)}{\partial \boldsymbol{x}} e(t_f - t - \tau_k) + \gamma \sum_{i=1}^{N_c} \frac{\partial \mathcal{L}_{i,\varepsilon}\big(t, \boldsymbol{x}(t|\boldsymbol{u})\big)}{\partial \boldsymbol{x}} \Big\} \Delta \boldsymbol{x}(t) dt.
\end{aligned} \tag{4.56}$$

By using the definition of the Hamiltonian function $H_{\varepsilon,\gamma}$, we have

$$\begin{aligned}
& \mathcal{L}_0(t) + \sum_{k=1}^{M} \mathcal{L}_0(t+\tau_k) e(t_f - t - \tau_k) + \gamma \sum_{i=1}^{N_c} \mathcal{L}_{i,\varepsilon}\big(t, \boldsymbol{x}(t|\boldsymbol{u})\big) \\
& \qquad = H_{\varepsilon,\gamma} - \boldsymbol{\lambda}(t)^\top \boldsymbol{f}(t) + \sum_{k=1}^{M} (\bar{\boldsymbol{\lambda}}^k)^\top(t) \bar{\boldsymbol{f}}_k(t) e(t_f - t - \tau_k).
\end{aligned}$$

Thus,

$$\begin{aligned}
\Delta J_{\varepsilon,\gamma}(\boldsymbol{u}) = & \frac{\partial \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{u})\big)}{\partial \boldsymbol{x}(t_f)} \Delta \boldsymbol{x}(t_f) + \int_0^{t_f} \Big\{ \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t) + \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) \Big\} dt \\
& - \int_0^{t_f} \boldsymbol{\lambda}(t)^\top \big\{ \frac{\partial \boldsymbol{f}(t)}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t) + \frac{\partial \boldsymbol{f}(t)}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) \big\} dt \\
& - \sum_{k=1}^{M} \int_0^{t_f} (\bar{\boldsymbol{\lambda}}^k)^\top(t) e(t_f - t - \tau_k) \Big\{ \frac{\partial \bar{\boldsymbol{f}}_k(t)}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t) + \frac{\partial \bar{\boldsymbol{f}}_k(t)}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) \Big\} dt.
\end{aligned} \tag{4.57}$$

By (4.43)-(4.45), and the definition of $e(\cdot)$, it follows that

$$\begin{aligned}
& \sum_{k=1}^{M} \int_0^{t_f} (\bar{\boldsymbol{\lambda}}^k)^\top(t) e(t_f - t - \tau_k) \Big\{ \frac{\partial \bar{\boldsymbol{f}}_k(t)}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t) + \frac{\partial \bar{\boldsymbol{f}}_k(t)}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) \Big\} \\
& \qquad = \sum_{k=1}^{M} \int_0^{t_f - \tau_k} (\bar{\boldsymbol{\lambda}}^k)^\top(t) e(t_f - t - \tau_k) \Big\{ \frac{\partial \bar{\boldsymbol{f}}_k(t)}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t) + \frac{\partial \bar{\boldsymbol{f}}_k(t)}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) \Big\}
\end{aligned}$$

$$= \sum_{k=1}^{M} \int_{\tau_k}^{t_f} (\boldsymbol{\lambda}^k)^\top(t) \Big\{ \frac{\partial \boldsymbol{f}(t)}{\partial \tilde{\boldsymbol{x}}} \Delta \boldsymbol{x}(t-\tau_k) + \frac{\partial \hat{\boldsymbol{f}}_k(t)}{\partial \tilde{\boldsymbol{u}}} \Delta \boldsymbol{u}(t-\tau_k) \Big\}. \tag{4.58}$$

Since $\Delta \boldsymbol{x}(t-\tau_k) = \boldsymbol{0}$, for $0 \le t \le \tau_k$, and $\Delta \boldsymbol{u}(t-\tau_k) = \boldsymbol{0}$, for $0 \le t < \tau_k$, we have

$$\begin{aligned} &\sum_{k=1}^{M} \int_{\tau_k}^{t_f} (\boldsymbol{\lambda}^k)^\top(t) \Big\{ \frac{\partial \boldsymbol{f}(t)}{\partial \tilde{\boldsymbol{x}}} \Delta \boldsymbol{x}(t-\tau_k) + \frac{\partial \boldsymbol{f}(t)}{\partial \tilde{\boldsymbol{u}}} \Delta \boldsymbol{u}(t-\tau_k) \Big\} \\ &\quad = \sum_{k=1}^{M} \int_{0}^{t_f} (\boldsymbol{\lambda}^k)^\top(t) \Big\{ \frac{\partial \boldsymbol{f}(t)}{\partial \tilde{\boldsymbol{x}}} \Delta \boldsymbol{x}(t-\tau_k) + \frac{\partial \boldsymbol{f}(t)}{\partial \tilde{\boldsymbol{u}}} \Delta \boldsymbol{u}(t-\tau_k) \Big\}. \end{aligned} \tag{4.59}$$

Substituting (4.59) and (4.58) into (4.57), and then combining with (4.58), it gives

$$\begin{aligned} \Delta J_{\varepsilon,\gamma}(\boldsymbol{u}) = &\frac{\partial \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{u})\big)}{\partial \boldsymbol{x}(t_f)} \Delta \boldsymbol{x}(t_f) + \int_0^{t_f} \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) dt \\ &\int_0^{t_f} \Big\{ \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t) - \boldsymbol{\lambda}(t)^\top \frac{d(\Delta \boldsymbol{x}(t))}{dt} \Big\} dt. \end{aligned} \tag{4.60}$$

Integrating the last term of (4.60) by parts gives

$$\begin{aligned} \Delta J_{\varepsilon,\gamma}(\boldsymbol{u}) = &\frac{\partial \Phi_0\big(\boldsymbol{x}(t_f|\boldsymbol{u})\big)}{\partial \boldsymbol{x}(t_f)} \Delta \boldsymbol{x}(t_f) - \boldsymbol{\lambda}(t)^\top \Delta \boldsymbol{x}(t) \Big|_0^{t_f} \\ &+ \int_0^{t_f} \Big\{ \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{x}} \Delta \boldsymbol{x}(t) + \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) + \frac{d(\boldsymbol{\lambda}(t))^\top}{dt} \Delta \boldsymbol{x}(t) \Big\} dt. \end{aligned} \tag{4.61}$$

Since $\boldsymbol{x}(0)$ is a given constant, it is clear that $\Delta \boldsymbol{x}(0) = 0$. Hence, $\boldsymbol{\lambda}(0)^\top \Delta \boldsymbol{x}(0) = 0$. Since $\Delta \boldsymbol{u}(t)$ is arbitrary on $[0, t_f]$, substituting (4.47), (4.48), and (4.49) into (4.61) yields

$$\Delta J_{\varepsilon,\gamma}(\boldsymbol{u}) = \int_0^{t_f} \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{u}} \Delta \boldsymbol{u}(t) dt. \tag{4.62}$$

This completes the proof. □

Theorem 4.7. *For each $q = 1, \ldots, N$, the gradient of the augmented cost function $\tilde{J}_{\varepsilon,\gamma}$ with respect to $\boldsymbol{\sigma}^{N,q}$ is*

$$\frac{\partial \tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N)}{\partial \boldsymbol{\sigma}^{N,q}} = \int_{t_{q-1}}^{t_q} \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{u}} dt. \tag{4.63}$$

Proof. Let $\boldsymbol{u}_\epsilon(t)$ be defined as (4.51), where

$$\Delta \boldsymbol{u}(t) = \Delta \boldsymbol{\sigma}^N \chi_{[t_{q-1}, t_q)}(t), \tag{4.64}$$

$$\Delta \boldsymbol{u}(t) = \boldsymbol{0}, \quad t < 0, \tag{4.65}$$

and

$$\Delta\boldsymbol{\sigma}^N = [\mathbf{0}^\top, \ldots, \mathbf{0}^\top, (\Delta\boldsymbol{\sigma}^{N,q})^\top, \mathbf{0}^\top, \ldots, \mathbf{0}^\top]^\top.$$

Substitute (4.64) into (4.62), it is clear that

$$\Delta J_{\varepsilon,\gamma}(\boldsymbol{u}) = \int_{t_{q-1}}^{t_q} \frac{\partial H_{\varepsilon,\gamma}}{\partial \boldsymbol{u}} \Delta\boldsymbol{\sigma}^{N,q} dt. \tag{4.66}$$

Restricting the controls to $\mathcal{U}^N$ yields

$$J_{\varepsilon,\gamma}(\boldsymbol{u}^N) = \tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N).$$

Using Theorem 4.6, it is clear that

$$\begin{aligned}
\Delta J_{\varepsilon,\gamma}(\boldsymbol{u}^N) &= \lim_{\epsilon\to 0}\Big\{\frac{\partial J_{\varepsilon,\gamma}(\boldsymbol{u}^N + \epsilon\Delta\boldsymbol{u}) - J_{\varepsilon,\gamma}(\boldsymbol{u}^N)}{\partial\epsilon}\Big\} \\
&= \lim_{\epsilon\to 0}\Big\{\frac{\partial \tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N + \epsilon\Delta\boldsymbol{\sigma}^N) - \tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N)}{\partial\epsilon}\Big\} \\
&= \big\langle \frac{\partial \tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N)}{\partial\boldsymbol{\sigma}^N}, \Delta\boldsymbol{\sigma}^N\big\rangle \\
&= \big\langle \frac{\partial \tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N)}{\partial\boldsymbol{\sigma}^{N,q}}, \Delta\boldsymbol{\sigma}^{N,q}\big\rangle.
\end{aligned} \tag{4.67}$$

Combining (4.66) and (4.67), and noting that $\Delta\boldsymbol{\sigma}^{N,q}$ is arbitrary, the theorem follows readily. □

With the gradient formula given in Theorem 4.7, Problem $(Q_{N,\varepsilon,\gamma})$, for each $\varepsilon > 0$ and $\gamma > 0$, can be solved by using a gradient-based optimization technique, such as the sequential quadratic programming approximation method. We propose the following algorithm.

Algorithm 4.1.

Step 1. Set $\varepsilon = 0.01$ and $\gamma = 10$.

Step 2. Solve the state differential equation (4.1) with the initial conditions (4.3) forward in time from $t = 0$ to $t = t_f$. Let the solution obtained be denoted by $\boldsymbol{x}(\cdot|\boldsymbol{\sigma}^N)$.

Step 3. Compute the value of the augmented cost function given by (4.21).

Step 4. Solve the co-state system (4.47) backward in time from $t = t_f$ to $t = 0$ with the boundary condition (4.48) and (4.49), where $\boldsymbol{x}(\cdot|\boldsymbol{\sigma}^N)$ is from Step 2. Let the solution obtained be referred to as $\boldsymbol{\lambda}(\cdot|\boldsymbol{\sigma}^N)$.

Step 5. Compute the gradient of the augmented cost function (4.21) with respect to $\boldsymbol{\sigma}^N$ according to (4.63).

Step 6. Solve the approximate Problem $(Q_{N,\varepsilon,\gamma})$ by using the sequential quadratic programming approximation scheme with active set strategy. Let the optimal control vector obtained be denoted as $\boldsymbol{\sigma}^{N,*}_{\varepsilon,\gamma}$.

Step 7. Check the feasibility of the continuous state inequality constraints for all $t \in [0, t_f]$. If $\boldsymbol{\sigma}^{N,*}_{\varepsilon,\gamma}$ is feasible, go to Step 8; otherwise, set $\gamma := 10 \times \gamma$ and go to Step 2.

Step 8. Set $\varepsilon := \varepsilon/10$. If $\varepsilon > \varepsilon_{min}$, go to Step 2, else successfully exist.

Remark 4.1. Problem $(Q_{N,\varepsilon,\gamma})$ with $\varepsilon > 0$ and $\eta_z > 0$, $z = 1, \ldots, N_g$, chosen as detailed in Step 7 of Algorithm 4.1 is an approximate problem of Problem (Q). By using arguments similar to that given for Theorem 4.4, it can be shown that the approximate optimal cost will converge to the true optimal cost as $N \to \infty$. In practice, we could start with a small integer N, and obtain the optimal control vector of the corresponding Problem $(Q_{N,\varepsilon,\gamma})$. We then double the value of N and re-calculate the optimal control vector of the corresponding Problem $(Q_{N,\varepsilon,\gamma})$ with the previous optimal control vector taken as the initial guess in the optimization process. We repeat this process until the reduction in the cost value is negligible. From extensive simulation studies, it is observed that N does not need to be very large, certainly the one used in our simulation study is more than sufficient. In fact, the approximation of the control by a piecewise-constant function should also be followed in some real applications. The extension to the case where the control is approximated by piecewise linear or piecewise smooth function is straightforward. The switching times will affect the cost value but is insignificant. The main advantage of taking the switching times as decision variables is that the number of switching times could be reduced for achieving the same cost value. However, the price to pay is a significant increase in the computational burden.

Remark 4.2. Theorem 4.2 ensures that for each $\varepsilon > 0$, a suitable $\gamma(\varepsilon)$ can be obtained in finite number of iterations. This is due to the special structure of the penalty function used.

4.4 Application: optimal control of an evaporation process

We now demonstrate the applicability of our approach to a realistic optimal control problem arising in evaporation process. Specifically, we consider the industrial evaporation process described in [119]. The alumina production process mainly includes aluminium

hydroxide solution preparing process, clarifying process, dissolving process, decomposing process, evaporation process, and roasting process. The main contents of the mother liquor discharged from the decomposing process are sodium hydroxide and aluminum oxide which are valuable materials needed in the recycling. However, the concentration of the mother liquor is lower than the required concentration needed for the leaching process or the grinding process. Thus, it cannot be used directly, and hence the evaporation process is needed to improve the concentration of the mother liquor, such that the acid and caustic materials can be re-used.

The returned lye discharged from the evaporation process is one of the main raw materials needed for converting the bauxite to aluminium hydroxide solution (which is also called the raw slurry). It is known [157] that the quality of raw slurry has a direct influence on the quality of the final product. Unacceptable fluctuation in the composition of the returned lye can lead to instability of the blending process during the preparation of the raw slurry. Consequently, the quality of the product obtained cannot be guaranteed. Usually, only the solution concentration at the outlet of the evaporation process is measured at every two-hour interval, but it takes about one hour for the feed flowing though the evaporation process. Clearly, inappropriate control of the evaporation process will lead to unacceptable output solution, yet with high steam consumption. Hence, optimal control is needed.

In this section, the control method proposed in Section 4.3 is applied to study the optimal control of a practical alumina evaporation process, in which the objective is to find a control such that the specific quality of the sodium aluminate solution control is met with the least energy usage, while the constraints on the state and the control are satisfied.

4.4.1 The evaporation system

In a typical alumina production factory in China, the objective of the evaporation process is to increase the concentration of the industrial sodium aluminate solution (the sodium hydroxide content of the solution is to be increased to about 160∼170g/L for Bayer process). The industrial sodium aluminate solution is highly viscous. It contains many impurities, such as sodium carbonate and sodium sulfate, which can easily emit from the solution due to the increase of caustic alkali concentration or the decrease of temperature. Crystallization of the impurities will cause serious pipe plug problem. To avoid the forming of high viscosity fluid at low temperature, the alumina production factory employs multiple falling film evaporators for the evaporation process as shown in Figure 4.1. It consists of four falling film tube evaporators, three direct pre-heaters, three flash evaporators, four flash tanks and a condenser.

The four falling film tube evaporators are connected in series with reference to their

vapor and liquor lines. Heat is supplied to the first evaporator by live steam generated in the power plant after the pressure is reduced to about 0.5Mpa. The vapor, produced by each of the first three evaporators, is used as the heating source for the next evaporators in series. The vapor produced by the fourth evaporator is condensed and then discharged from the process through the condenser installed in the water circuit.

Between each of the two adjacent evaporators there is a preheater, which is used to preheat the solution fed into the previous (with respect to the vapor lines) evaporator. The heating source of each preheater comes from the vapor produced by the previous (with respect to the vapor lines) evaporator and flash evaporator.

The feed enters the system at the third and the fourth evaporators, and flows backward. Finally the solution leaves from the first evaporator and is fed into the flash evaporators where the final product is drained. The structure of a falling tube evaporator is as shown

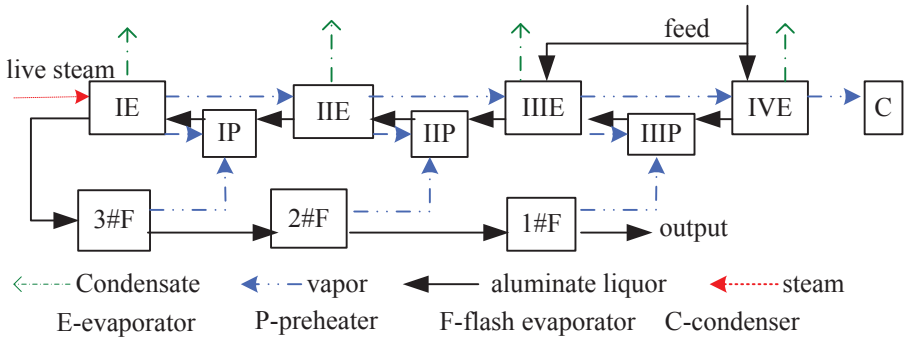

Figure 4.1: Flow sheet of alumina evaporation process

in Figure 4.2. It consists of an evaporation vessel (A), a heat exchanger (B), a number of pumps that realizes the transfer of solution in and out the evaporator, and valves that control the steam in and out of the evaporator.

The liquor, i.e. the industrial sodium aluminate solution, is injected at the bottom of the evaporation vessel through Pump 1. It is then pumped to the top of the evaporator, where a distributor is used to provide a uniformly distributed falling liquid film inside each heating tube. The effect of the distribution depends on the viscosity of the solution and the cycling rate. Because the broken up of the falling film will cause serious scaring problem [109], the change of the cycling rate should only be slight such that the change in the surface tension force of the film is mild.

The liquor flows through the inside of the heating tubes, and the heating steam fills the outside of the heating tubes. It leads to a heat transfer from the steam to the liquor. The heating steam is governed by Valve 1. As the liquor goes through the heating tube, it reaches its boiling temperature, causing water within the liquor to evaporate. A separator is used to split the vapor from the solution at the top of the evaporator vessel. The vapor is drained through Valve 2 which is used to connect the two adjacent evaporators, and

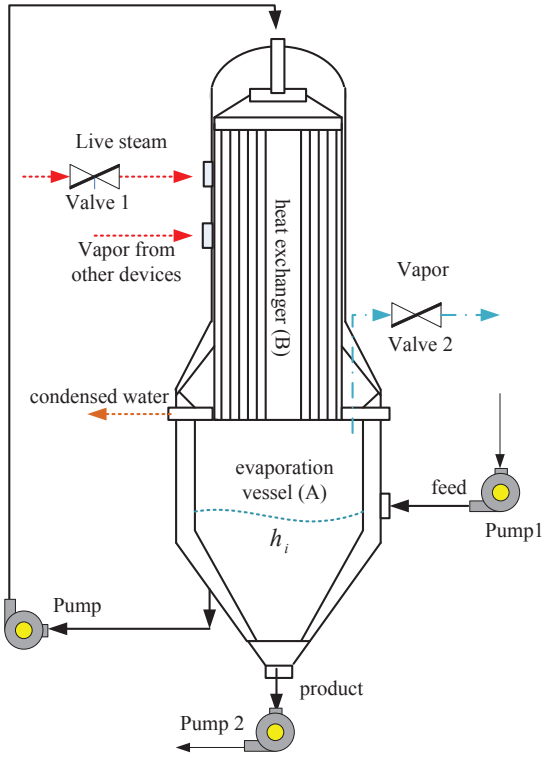

Figure 4.2: Structure of a vertical tube falling film evaporator

hence cannot be adjusted. Finally, the liquor is drained through Pump 2.

In practice, once the control variables are set, they must be used for at least 5 minutes. In other words, the control variables (i.e. flow rate of live steam) are to be adjusted in a piecewise-constant manner. Thus, the control variables can be approximated by piecewise-constant functions with possible discontinuities at the preset switching points. Hence, the optimal control approach introduced in Section 4.3 can be used to deal with the optimal control for the evaporation process.

4.4.2 Mathematical model of the evaporation system

A Dynamic system for the evaporation process

Three types of dynamical process models have been developed for a falling-film evaporator [46,145]. In this chapter, a pilot-scale evaporation process is considered. As the direct preheaters are used to mix the solution and vapor, there is litter change in its level. Thus, the dynamics of the preheaters are omitted; and the evaporator and the corresponding preheater are taken as a whole. Also, we neglect the dynamics of the distributor and the separator. Only the dynamics involved in the evaporation vessel is considered.

A fundamental model that describes the multi-effect falling film evaporation process can be derived under the following assumptions:

- Perfect mixing in each phase in each vessel;
- The absence of any non-condensable gases and the process is adiabatic;
- No transportation lags associated with the movement of steam;
- The cross-sectional area of the evaporator and the specific heat capacity of water are constant;
- All pipes are full.

According to experienced engineers in the factory, the changes of the input solution, for the evaporator, take about 15 minutes to cause an effect on the changes in the evaporation vessel. This is due to the hysteresis during the solution flowing to the evaporation vessel, such as in the distributor and the place where the input solution is injected. Similarly, the time-delays for the flash evaporators are about 5 minutes. From simulation studies, it appears that the values for these delays are acceptable in practice. The variables that are of interest are product temperature in the effect, solution level in the evaporation vessel and product concentration of each effect. In particular, sodium carbonate, sodium hydroxide, and alumina are the three components of the industrial sodium aluminate solution which are measured at every two-hour interval. Thus, under the aforementioned assumptions, the alumina evaporation process model is built based on the principles of heat balance and material balance in unit operations (evaporator and flash evaporator). They are described by the following differential equations with multiple time-delays [119]:

(a) The time variations of the solution levels are:

$$\frac{dh_i(t)}{dt} = \frac{1}{A_i \rho_i(t)} \Delta M_i(t), \quad i = 1, \ldots, 7. \tag{4.68}$$

(b) The concentration of the solution in the evaporation vessel is assumed to be equal to that of the output solution. For each i, $i = 1, \ldots, 7$, the three ingredients of the solution, i.e. sodium carbonate, sodium hydroxide, and alumina, are denoted by C_i^j, $j = 1, 2, 3$. For each j, $j = 1, 2, 3$, the time variations of the concentrations can be expressed as:

$$\frac{dC_i^j(t)}{dt} = \frac{1}{A_i h_i(t)} \big[F_{i+1}(t-\tau_1) C_{i+1}^j(t-\tau_1) - F_i(t) C_i^j(t) - \frac{dh_i(t)}{dt} A_i C_i^j(t)\big], \quad i = 1, 2, 3, \tag{4.69a}$$

$$\frac{dC_i^j(t)}{dt} = \frac{1}{A_i h_i(t)} \big[F_{i+1}(t-\tau_2) C_{i+1}^j(t-\tau_2) - F_i(t) C_i^j(t) - \frac{dh_i(t)}{dt} A_i C_i^j(t)\big], \quad i = 4, 5, \tag{4.69b}$$

$$\frac{dC_i^j(t)}{dt} = \frac{1}{A_i h_i(t)} \big[F_{i+1}(t-\tau_2) C_{i+1}^j(t-\tau_2) + F_{01}(t-\tau_2) C_0^j(t-\tau_2) - F_i(t) C_i^j(t) - \frac{dh_i(t)}{dt} A_i C_i^j(t)\big], \quad i = 6, \tag{4.69c}$$

$$\frac{dC_i^j(t)}{dt} = \frac{1}{A_i h_i(t)}\big[F_0(t-\tau_2)C_0^j(t-\tau_2) - F_i(t)C_i^j(t) - \frac{dh_i(t)}{dt}A_i C_i^j(t)\big], \quad i = 7. \quad (4.69d)$$

(c) The time variations of the solution temperatures are:

$$\frac{dT_i(t)}{dt} = \frac{\Delta Q_i(t)}{dt}\frac{1}{A_i h_i(t) cp_i(t)} - \frac{T_i(t)}{cp_i(t)}\frac{dcp_i}{dt} - \frac{T_i(t)\rho_i(t)}{h_i(t)}\frac{dh_i(t)}{dt}, \quad i = 1, \ldots, 7. \quad (4.70)$$

Here, C_i^j, for each $i = 1, 2, 3$; $j = 1, 2, 3$, denotes the concentration of the jth ingredient of the ith flash evaporator; C_i^j, for each $i = 4, \ldots, 7$; $j = 1, 2, 3$, denotes the concentration of the jth ingredient of the $(i-3)$th evaporator; h is the solution level of the evaporation vessel; A is the cross-sectional area of the evaporation vessel; cp and ρ are, respectively, the specific heat capacity and the density of the output solution; ΔQ_i and ΔM_i, $i = 1, \ldots, 7$, are the heat changes and the mass changes in the evaporation vessel, respectively; T is the product temperature; F is the flow rate of the output solution; F_0 and F_{01} are, respectively, the feed inputs into the third and fourth evaporators; C_0 is the condensation of the feed injected to the process; τ_1 and τ_2 are the time-delays, where $\tau_1 = 5$ minutes and $\tau_1 = 15$ minutes.

The mass changes in the evaporation vessels can be calculated by using the following formulas:

$$\Delta M_i(t) = F_{i+1}(t-\tau_1)\rho_{i+1}(t-\tau_1) - V_i(t) - F_i(t)\rho_i(t), \quad i = 1, 2, 3, \quad (4.71)$$

$$\Delta M_4(t) = F_5(t-\tau_2)\rho_5(t-\tau_2) - V_4(t) - F_4(t)\rho_4(t) + V_3(t), \quad (4.72)$$

$$\Delta M_5(t) = F_6(t-\tau_2)\rho_6(t-\tau_2) - V_5(t) - F_5(t)\rho_5(t) + V_2(t), \quad (4.73)$$

$$\Delta M_6(t) = F_7(t-\tau_2)\rho_7(t-\tau_2) - V_6(t) - F_6(t)\rho_6(t) + V_1(t) + F_{01}(t-\tau_2)\rho_0(t-\tau_2), \quad (4.74)$$

$$\Delta M_7(t) = F_0(t-\tau_2)\rho_0(t-\tau_2) - V_7(t) - F_7(t)\rho_7(t), \quad (4.75)$$

where V_i, for each $i = 1, 2, 3$, is the vapor produced by the ith flash evaporator; V_i, for each $i = 4, \ldots, 7$, is the vapor produced by the $(i-3)$th evaporator. They are calculated as follows:

$$V_i(t) = \frac{F_{i+1}(t-\tau_1)\rho_{i+1}(t-\tau_1)[cp_{i+1}(t-\tau_1)T_{i+1}(t-\tau_1) - cp_i(t)T_i(t)]}{H_i(t) - cp_i(t)T_i(t)}, \quad (4.76a)$$

$$V_4(t) = \frac{V_0(t)r_0(t) + V_3(t)(H_3(t) - cp_4(t)T_4(t)) - cp_4(t)T_4(t)]}{H_4(t) - cp_4(t)T_4(t)} \quad (4.76b)$$

$$+ \frac{F_5(t-\tau_2)\rho_5(t-\tau_2)[cp_5(t-\tau_2)T_5(t-\tau_2)}{H_4(t) - cp_4(t)T_4(t)}, \quad (4.76c)$$

$$V_5(t) = \frac{V_4(t)r_4(t) + V_2(t)(H_2(t) - cp_5(t)T_5(t))}{H_5(t) - cp_5(t)T_5(t)} \quad (4.76d)$$

$$+ \frac{F_6(t-\tau_2)\rho_6(t-\tau_2)[cp_6(t-\tau_2)T_6(t-\tau_2) - cp_5(t)T_5(t)]}{H_5(t) - cp_5(t)T_5(t)}, \quad (4.76e)$$

$$V_6(t) = \frac{V_1(t)(H_1(t) - cp_6(t)T_6(t)) + V_5(t)r_5(t)}{H_6(t) - cp_6(t)T_6(t)} + \frac{F_7(t-\tau_2)\rho_7(t-\tau_2)[cp_7(t-\tau_2)T_7(t-\tau_2) - cp_6(t)T_6(t)]}{H_6(t) - cp_6(t)T_6(t)} + \frac{V_5(t)r_5(t) + F_{01}(t-\tau_2)\rho_0(t-\tau_2)[cp_0(t-\tau_2)T_0(t-\tau_2) - cp_6(t)T_6(t)]}{H_6(t) - cp_6(t)T_6(t)}, \quad (4.76f)$$

$$V_7(t) = \frac{V_6(t)r_6(t) + F_0(t-\tau_2)\rho_0(t-\tau_2)[cp_0(t-\tau_2)T_0(t-\tau_2) - cp_7(t)T_7(t)]}{H_7(t) - cp_7(t)T_7(t)}, \quad (4.76g)$$

where H and r are, respectively, the enthalpy and the latent heat of the output vapor; V_0 is the flow rate of the live steam; r_0 is the latent heat of the live steam; and ρ_0, cp_0, T_0 are, respectively, the density, specific heat capacity, and temperature of the feed injected to the process. It is assumed that the vapor is saturated. The relationships between the latent heat and the enthalpy are obtained in [21] as given below:

$$r_i = 2495.0 - 2.219T_{vi} - 0.002128(T_{vi})^2, \quad i = 1, \ldots, 7, \quad (4.77)$$

$$H_i = r_i + 4.18T_{vi}, \quad i = 1, \ldots, 7, \quad (4.78)$$

where

$$T_{vi} = T_i - \Delta T_i, \quad i = 1, \ldots, 7. \quad (4.79)$$

The density and the boiling point rise are important properties that must be specified in a multiple-effect evaporator [46], especially, when the soluble solid concentration is high. In order to obtain their relationships, several thermal balance tests were carried out during the whole acid cycle of the evaporation process. Correlations describing the relations between the boiling point rise and the density are determined by the regression method using the experimental data. For each i, $i = 1, \ldots, 7$, they are obtained as follows:

$$\rho_i = 1045 + 1.2C_i^1 + C_i^2 + 0.8C_i^3, \quad (4.80)$$

$$\Delta T_i = \frac{0.0162(T_i + 273)^2(75.77C_i^2\rho^{-1} - 3.608)}{r_i} - 0.23C_i^1 - 0.073C_i^2 - 0.1094C_i^3 + 0.3206T_i, \quad (4.81)$$

where ΔT_i, for each $i = 1, 2, 3$, is the boiling point rise of the solution output from the ith flash evaporator; and ΔT_i, for each $i = 4, \ldots, 7$, is the boiling point rise of the solution output from the $(i-3)$th evaporator.

The specific heat of the solution at each of these evaporators can be calculated from its component concentration as follows:

$$cp_i = 4.18 - \rho^{-1}(2.994C_i^1 + 2.923C_i^2 + 3.266C_i^3), \quad i = 1, \ldots, 7 \quad (4.82)$$

Furthermore, the heat changes in the evaporation vessels are:

$$\begin{aligned}\Delta Q_i(t) &= F_{i+1}(t-\tau_1)\rho_{i+1}(t-\tau_1)cp_{i+1}(t-\tau_1)T_{i+1}(t-\tau_1)\\ &\quad - V_i(t)H_i(t) - F_i(t)\rho_i(t)cp_i(t)T_i(t), \quad i=1,2,3, \qquad (4.83a)\\ \Delta Q_4(t) &= V_0(t)r_0(t) + V_3(t)H_3(t) + F_5(t-\tau_2)\rho_5(t-\tau_2)cp_5(t-\tau_2)T_5(t-\tau_2)\\ &\quad - V_4(t)H_4(t) - F_4(t)\rho_4(t)cp_4(t)T_4(t), \qquad (4.83b)\\ \Delta Q_5(t) &= V_4(t)r_4(t) + V_2(t)H_2(t) + F_6(t-\tau_2)\rho_6(t-\tau_2)cp_6(t-\tau_2)T_6(t-\tau_2)\\ &\quad - V_5(t)H_5(t) - F_5(t)\rho_5(t)cp_5(t)T_5(t), \qquad (4.83c)\\ \Delta Q_6(t) &= V_5(t)r_5(t) + V_1(t)H_1(t) + F_7(t-\tau_2)\rho_7(t-\tau_2)cp_7(t-\tau_2)T_7(t-\tau_2)\\ &\quad - V_6(t)H_6(t) - F_6(t)\rho_6(t)cp_6(t)T_6(t)\\ &\quad + F_{01}(t-\tau_2)\rho_0(t-\tau_2)cp_0(t-\tau_2)T_0(t-\tau_2), \qquad (4.83d)\\ \Delta Q_7(t) &= V_6(t)r_6(t) + F_0(t-\tau_2)\rho_0(t-\tau_2)cp_0(t-\tau_2)T_0(t-\tau_2)\\ &\quad - V_7(t)H_7(t) - F_7(t)\rho_7(t)cp_7(t)T_7(t). \qquad (4.83e)\end{aligned}$$

Let

$$\boldsymbol{x} = [x_1,\ldots,x_{35}]^\top = [h_1,\ldots,h_7,C_1^1,\ldots,C_7^1,C_1^2,\ldots,C_7^2,C_1^3,\ldots,C_7^3,T_1,\ldots,T_7]^\top \in \mathbb{R}^{35},$$

denote the state with 35 variables. Taking into account the thermophysical properties of the solution and vapor, and substituting (4.71)-(4.83e) into (4.68) to (4.70), the evaporation system model can be expressed as a system of 35 differential equations. The split flow rate of feed, product flow rates of each of the evaporators and flash evaporators, and the live steam flow rate are the control variables denoted as $\boldsymbol{u} = [u_1,\ldots,u_9]^\top = [F_1,\ldots,F_7,F_{01},V_0]^\top \in \mathbb{R}^9$. Let $\boldsymbol{f} = [f_1,\ldots,f_{35}]^\top$, where f_i, $i=1,\ldots,35$, denote the functions appeared on the right-hand sides of (4.68)-(4.70).

Let this system model be referred to as System (S1).

B Initial conditions for the evaporation system

For System (S1), the initial values for the state and the control variables at and prior to $t=0$ are obtained from a real-life evaporation process of an alumina production factory in China. Specifically, for each of the temperature T_i, $i=1,\ldots,7$, it has relatively little fluctuation. However, the values of the seven temperatures T_i, $i=1,\ldots,7$, are quit different. Thus, we shall minus each of the temperature T_i, $i=1,\ldots,7$, by 97.7, 106.15, 116.1, 131.2, 105.6, 75, 54.5, respectively. Then, mark the values after treatments in Figure 4.3(a). The values of the levels are marked in Figure 4.3(b). Let $\{\zeta_k\}_{k=1}^K$ with $\zeta_k < \zeta_{k+1}$, $k=1,\ldots,K-1$, be the set of the observed time points on the interval $[-15,0]$. The value of the ith state variable x_i at the observed time point ζ_k is denoted as ϕ_i^k. The

temperatures and the liquor levels at and prior to $t = 0$ are calculated according to

$$x_i(t) = \phi_i^k + \frac{\phi_i^k - \phi_i^{k+1}}{\zeta_k - \zeta_{k+1}}(t - \zeta_k), \quad t \in [\zeta_k, \zeta_{k+1}], \quad k = 1, \ldots, K-1; \quad i = 1, \ldots, 7, 29, \ldots, 35. \tag{4.84}$$

In addition, the concentrations of the solution x_i, $i = 8, \ldots, 28$, can only be accessed every two hours through analyzing sample solution collected from the practical evaporation process in the factory. It is assumed that the concentrations do not change during the time interval $[-15, 0]$. They are list below.

$$\begin{aligned}
[x_8(t), \ldots, x_{14}(t)]^\top &= [74.85, 73.27, 71.75, 66.39, 54.86, 47.39, 44.83]^\top, \quad t \in [-15, 0], \\
[x_{15}(t), \ldots, x_{21}(t)]^\top &= [163.94, 161.07, 157.70, 145.94, 120.57, 104.17, 98.54]^\top, \quad t \in [-15, 0], \\
[x_{22}(t), \ldots, x_{28}(t)]^\top &= [75.61, 74.29, 72.73, 67.31, 55.61, 48.04, 45.45]^\top, \quad t \in [-15, 0].
\end{aligned}$$

The control in time horizon $t \in [-15, 0)$ for System (S1) are given by

$$\boldsymbol{\psi}(t) = [\psi_1(t), \ldots, \psi_9(t)]^\top = [2.54, 2.59, 2.64, 2.76, 3.44, 3.99, 4.22, 0.165, 986.5]^\top. \tag{4.85}$$

Let these initial state and control conditions obtained over the time interval $[-15, 0)$ be referred to as Initial Condition (IC).

4.4.3 Optimal control problem formulation

The optimal control of the evaporation process is to find a control such that the tasks listed below are accomplished.

(i) The energy usage is minimized.

(ii) The specific requirements of the industrial sodium aluminate solution are met.

(iii) The solution level for each effect is maintained to within its operation limits.

The energy usage is measured in terms of the mass units of live steam used for evaporating one mass unit of water. It is defined by the following equation:

$$J_0 = V_0(t)/W(t) = u_9(t)/W(t),$$

where $W(t)$ is the total water evaporated for the evaporation process at time t. It is the difference between the mass flow rate of the feed and the final product, and is calculated

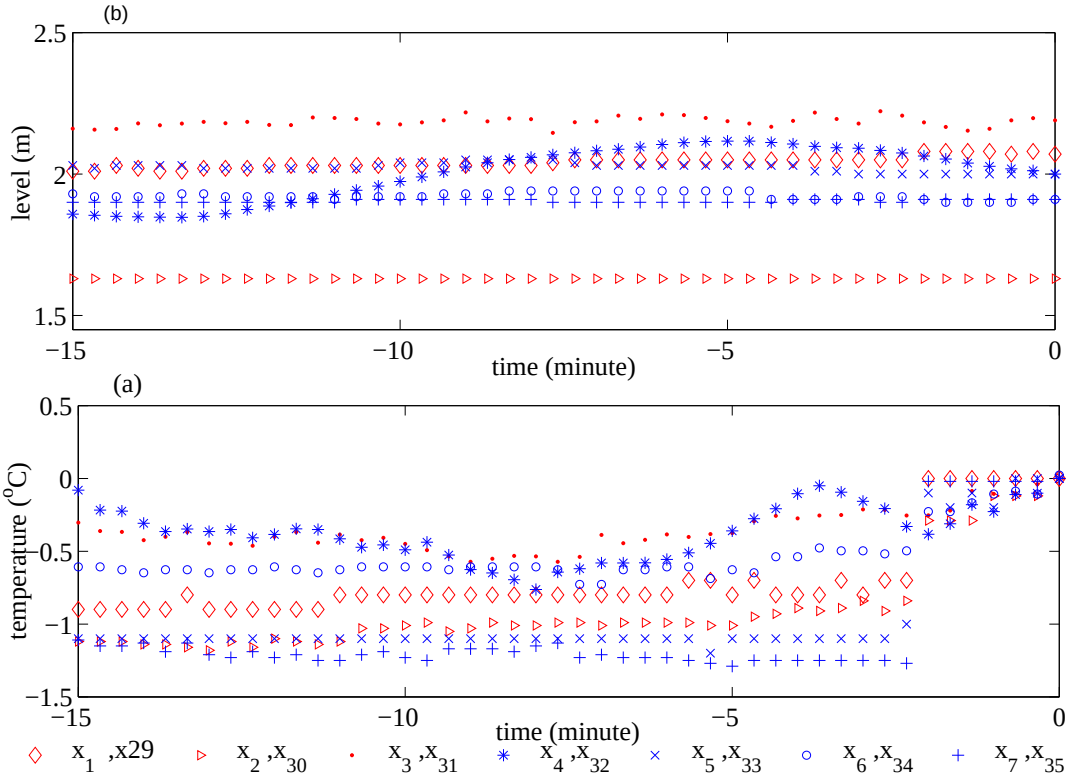

Figure 4.3: The variations of temperatures and liquor levels at observed time points

as given below:

$$\begin{aligned} W(t) &= (F_0(t) + F_{01}(t))\rho_0(t) - F_1(t)\rho_1(t) \\ &= (F_0(t) + F_{01}(t))\rho_0(t) - u_1(t)(1045 + 1.2x_8(t) + x_{15}(t) + 0.8x_{22}(t)). \end{aligned}$$

Thus, the cost function to be minimized is

$$\begin{aligned} J &= \Phi_0(\boldsymbol{x}(t_f|\boldsymbol{u})) \\ &\quad + \int_0^{t_f} \left\{ \frac{u_9(t)}{(F_0(t) + F_{01}(t))\rho_0(t) - u_1(t)(1045 + 1.2x_8(t) + x_{15}(t) + 0.8x_{22}(t))} \right\}^2 dt, \end{aligned} \tag{4.86}$$

where t_f is the final time of the time horizon $[0, t_f]$. Note that the change of the cost function value caused by the change of the product concentration takes about 75 minutes to be evaluated. Thus, the final time t_f should be much larger than 75. Φ_0 is the terminal cost given by

$$\Phi_0(\boldsymbol{x}(t_f|\boldsymbol{u})) = \sum_{i=1}^{7} (\hat{x}_i - x_i(t_f|\boldsymbol{u}))^2, \tag{4.87}$$

where $\hat{x}_i$, $i = 1, \ldots, 7$, are specified desired solution levels, which can be determined according to experience. One way to specify these desired solution levels is to solve the optimal control problem with Φ_0 deleted initially. Then, the time when the concentration of the product solution has become stable is identified. The values of the solution levels at this particular time instant are chosen as these specified desired solution levels. We then re-solve the optimal problem with Φ_0 being included. The inclusion of Φ_0 is for regulating the solution levels toward the end of the time horizon.

To proceed further, the bounds on the state and control variables are specified through analyzing the production data as well as utilizing the experience of the operators from the evaporation process of an alumina production factory in China. It is found that the solution levels of the flash evaporators are less important than those of the evaporators. Moreover, according to the production data, the solution levels of the flash evaporators must be in the range of 1.5∼2.5 m; the solution level of the first evaporator is limited to 1.8∼2.3 m; and the solution levels of the last three evaporators are constrained to lie between 1.9 m and 2.1 m; and the sodium hydroxide concentration of the final product must reach 160∼170 g/L. For the control variables, the flow rate of the solution should be operated within 1.6∼5.3 m^3/min; the split flow rate of the feed input into the third evaporator is allowed to vary between 0 m^3/min to 0.6 m^3/min. The live steam supplied to the process can be adjusted within 566 kg/min to 1230 kg/min. Let us write down explicitly the bounds for these variables as follows:

$$\boldsymbol{a} = [a_1, \ldots, a_8]^\top = [2.5, 2.5, 2.5, 2.3, 2.1, 2.1, 2.1, 170]^\top, \tag{4.88a}$$

$$\boldsymbol{b} = [b_1, \ldots, b_8]^\top = [1.5, 1.5, 1.5, 1.8, 1.9, 1.9, 1.9, 160]^\top, \tag{4.88b}$$

$$\boldsymbol{c} = [c_1, \ldots, c_9]^\top = [5.3, 5.3, 5.3, 5.3, 5.3, 5.3, 5.3, 0.6, 1230]^\top, \tag{4.88c}$$

$$\boldsymbol{d} = [d_1, \ldots, d_8]^\top = [1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 0, 566]^\top, \tag{4.88d}$$

where a_i and b_i, for each $i = 1, \ldots, 7$, are the upper and lower bounds for the level of the ith equipment, respectively; a_8 and b_8 are the upper and lower bounds for the product concentration, respectively; d_l and c_l, for each $l = 1, \ldots, 9$, are the lower and upper bounds for the level of the lth control, respectively. The continuous inequality constraints on the states and controls may now be stated explicitly as follows:

$$b_i \le x_i(t) \le a_i, \quad i = 1, \ldots, 7, \quad t \in [0, t_f], \tag{4.89a}$$

$$b_8 \le x_{15}(t) \le a_8, \quad t \in [0, t_f], \tag{4.89b}$$

$$d_i \le u_l(t) \le c_i, \quad l = 1, \ldots, 9, \quad t \in [0, t_f]. \tag{4.89c}$$

Any measurable function $\boldsymbol{u} = [u_1, \ldots, u_9]^\top : [0, t_f] \to \mathbb{R}^9$ such that the constraints (4.89c) are satisfied is called an admissible control. Let $\boldsymbol{\mathcal{U}}$ be the set which consists of all such admissible controls.

The optimal control problem may now be stated formally below.

Problem (Q1). *Given System (S1) with Initial Conditions (IC), find a control* $\boldsymbol{u} \in \boldsymbol{\mathcal{U}}$ *such that the cost functional* (4.86) *is minimized subject to the continuous inequality constraints on the states given by* (4.89a)-(4.89b).

For Problem (Q1), the dimension of the state variables is 35 and the differential equations of the dynamics are nonlinear with multiple delays. Furthermore, there are eight continuous inequality constraints on the state variables which are not allowed to be violated at any time point during the time horizon. The final time t_f is taken as 8 hours, which is rather long. It does not appear that the nonlinear model predictive control (NMPC) technique could be applied directly due to the complexity of Problem (Q1). Nonetheless, Problem (Q1) will be solved by using the NMPC. As expected, the computational time is much too long for it to used in real operation. Thus, we shall make use of the control parameterization method and apply the proposed optimal control method to solve Problem (Q1). We shall also check the robustness of the optimal control obtained. Furthermore, the results will be compared with the data collected from the real plant to ensure proper operation of the process and those obtained by NMPC. Under normal operation, the problem will be re-solved 3 hours before the end of the 8 hours period so that a new optimal control can be obtained and used for the next 8 hours period.

4.4.4 Numerical results

A Numerical calculation

Consider Problem (Q1), i.e., the optimal control problem with its dynamical system, initial condition, and the cost function described by System (S1), Initial Condition (IC) and (4.86), respectively. Clearly,

$$\mathcal{L}_0(t, \boldsymbol{x}(t), \boldsymbol{u}) = \left[\frac{u_9(t)}{(F_0(t) + F_{01}(t))\rho_0(t) - u_1(t)(1045 + 0.8x_{22}(t) + 1.2x_8(t) + x_{15}(t))}\right]^2,$$

$$\Phi_0(\boldsymbol{x}(t_f|\boldsymbol{u})) = \sum_{i=1}^{7}(\hat{x}_i - x_i(t_f|\boldsymbol{u}))^2.$$

By using the control parameterization technique and the constraints transformation method described in Sections 4.3.1 and 4.3.2, the augmented cost function is:

$$\tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N) = \Phi_0(\boldsymbol{x}(t_f|\boldsymbol{\sigma}^N)) + \int_0^{t_f} \left\{\mathcal{L}_0(t) + \gamma \sum_{i=1}^{16} \left[\mathcal{L}_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))\right]\right\} dt \tag{4.90}$$

Here, $\mathcal{L}_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))$, $i = 1, \ldots, 16$, are specified in (4.19), where $h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N))$ are:

$$\begin{aligned}
h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) &= x_i(t|\boldsymbol{\sigma}^N) - a_i, \quad i = 1, \ldots, 7, \\
h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) &= b_{i-7} - x_{i-7}(t|\boldsymbol{\sigma}^N), \quad i = 8, \ldots, 14, \\
h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) &= x_{15}(t|\boldsymbol{\sigma}^N) - a_{i-7}, \quad i = 15, \\
h_i(t, \boldsymbol{x}(t|\boldsymbol{\sigma}^N)) &= b_{i-8} - x_{15}(t|\boldsymbol{\sigma}^N), \quad i = 16,
\end{aligned}$$

where a_i and b_i, $i = 1, \ldots, 8$, are given in (4.88a)-(4.88b).

Define the Hamiltonian function as:

$$\begin{aligned}
H_{\varepsilon,\gamma} = &\mathcal{L}_0(t, \boldsymbol{x}(t), \boldsymbol{u}^N(t)) \\
&+ \sum_{k=1}^{2} \left[(\bar{\boldsymbol{\lambda}}^k(t))^\top \bar{\boldsymbol{f}}_k e(t_f - t + \tau_k) \right] + \gamma \sum_{i=1}^{16} \left[\mathcal{L}_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{u}^N)) \right] + (\boldsymbol{\lambda}(t))^\top \boldsymbol{f}(t),
\end{aligned}$$

where $\bar{\boldsymbol{\lambda}}^k$ and $\bar{\boldsymbol{f}}_k$, $k = 1, 2$, are defined by (4.44) and (4.45), and $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_{35}]^\top \in \mathbb{R}^{35}$ is the solution of the co-state system defined by

$$\frac{d\boldsymbol{\lambda}(t)^\top}{dt} = -\frac{\partial \{\mathcal{L}_0(t) + \gamma \sum_{i=1}^{16} \mathcal{L}_{i,\varepsilon}(t, \boldsymbol{x}(t|\boldsymbol{u}^N))\}}{\partial \boldsymbol{x}} + (\boldsymbol{\lambda}(t))^\top \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} + \sum_{k=1}^{2} (\bar{\boldsymbol{\lambda}}^k(t))^\top \frac{\partial \bar{\boldsymbol{f}}_k}{\partial \boldsymbol{x}} e(t_f - t + \tau_k),$$

with terminal conditions

$$\begin{aligned}
\boldsymbol{\lambda}(t_f) &= \frac{\partial \boldsymbol{\Phi}_0(\boldsymbol{x}(t_f|\boldsymbol{\sigma}^N))^\top}{\partial \boldsymbol{x}}, \\
\boldsymbol{\lambda}(t) &= \mathbf{0}, \quad t > t_f.
\end{aligned}$$

The gradient formula of the augmented cost function with respect to each component of the control parameter vector can be calculated by using Theorem 4.7, where

$$\begin{aligned}
\frac{\partial H_{\varepsilon,\gamma}}{\partial u_1} = &\frac{2u_9(t)^2(1045 + 0.8x_{22}(t) + 1.2x_8(t) + x_{15}(t))}{\left[(F_0(t) + F_{01}(t))\rho_0(t) - u_1(t)(1045 + 0.8x_{22}(t) + 1.2x_8(t) + x_{15}(t))\right]^3} \\
&+ (\boldsymbol{\lambda}(t))^\top \frac{\partial \boldsymbol{f}}{\partial u_1} + \sum_{k=1}^{2} (\bar{\boldsymbol{\lambda}}^k(t))^\top \frac{\partial \bar{\boldsymbol{f}}_k}{\partial u_1} e(t_f - t + \tau_k), \\
\frac{\partial H_{\varepsilon,\gamma}}{\partial u_l} = &(\boldsymbol{\lambda}(t))^\top \frac{\partial \boldsymbol{f}}{\partial u_l} + \sum_{k=1}^{2} (\bar{\boldsymbol{\lambda}}^k(t))^\top \frac{\partial \bar{\boldsymbol{f}}_k}{\partial u_l} e(t_f - t + \tau_k), \quad l = 2, \ldots, 8, \\
\frac{\partial H_{\varepsilon,\gamma}}{\partial u_9} = &\frac{2u_9(t)}{\left[(F_0(t) + F_{01}(t))\rho_0(t) - u_1(t)(1045 + 0.8x_{22}(t) + 1.2x_8(t) + x_{15}(t))\right]^2} \\
&+ (\boldsymbol{\lambda}(t))^\top \frac{\partial \boldsymbol{f}}{\partial u_9} + \sum_{k=1}^{2} (\bar{\boldsymbol{\lambda}}^k(t))^\top \frac{\partial \bar{\boldsymbol{f}}_k}{\partial u_9} e(t_f - t + \tau_k),
\end{aligned}$$

and the partial derivatives of the functions $H_{\varepsilon,\gamma}$, $\boldsymbol{f}$, $\bar{\boldsymbol{f}}_1$, and $\bar{\boldsymbol{f}}_2$, with respect to x_i, $i = 1, \ldots, 35$, are calculated by using Maple software. Now, by using Algorithm 4.1, the optimal control problem can be solved using any gradient-based optimization algorithm, where the gradient of $\tilde{J}_{\varepsilon,\gamma}(\boldsymbol{\sigma}^N)$ with respect to $\boldsymbol{\sigma}^N$ are normalized at each iteration of the optimization process.

B Result and discussion

Simulation studies are performed using MATLAB on a computer with Intel Core 2 Quad Q9400 processor, where the final time t_f is taken as 480 minutes.

The optimal control problem is solved with the penalty factor taken as $\gamma = 10^5$. According to Remark 4.1, we choose $N = 80$, which means that the control is allowed to switch its value at every 6 minutes. Furthermore, the desired solution levels $\hat{x}_i$, $i = 1, \ldots, 7$, are chosen as 1.73, 2.25, 2.22, 2.14, 1.98, 2.03, and 1.98, respectively.

The disturbances that commonly affect the evaporation process are: disturbances due to the changes of the concentrations of feed; and the disturbances on the flow rate of the live steam. We now consider the same optimal control problem under the optimal control obtained. However, we assume that the feed concentration and the flow rate of the live steam are perturbed by a Gaussian noise with standard deviation of $\pm 5\%$.

For comparison, we shall use the model predictive control (MPC) [61,136] to construct the controller for the problem considered, where the objective function for MPC is given below:

$$J_M = 100 \sum_{k=1}^{M} \left\{ \frac{u_9(t_k)}{(F_0(t_\iota) + F_{01}(t_\iota))\rho_0(t_k) - u_1(t_k)(1045 + 0.8x_{22}(t_k) + 1.2x_8(t_k) + x_{15}(t_k))} \right\}^2 + 10 \sum_{k=1}^{P} (x_{15}(t_k) - x_{ref})^2 + \sum_{k=1}^{M} \Delta \boldsymbol{u}(t_k)^\top \boldsymbol{R} \Delta \boldsymbol{u}(t_k)$$

where $\boldsymbol{R} = diag[1, 1, 1, 1, 1, 1, 1, 0.1, 1]$. t_k, $k = 1, \ldots, P$, are kth sampling time. It is 0.1 hours. $\Delta \boldsymbol{u}(t_k)$ is the change rate of the control at time t_k. $M = 10$ is the control horizon, $P = 13$ is the predict horizon. x_{ref} is the reference trajectory, which is taken as 162.5—the concentration of the sodium hydroxide achieved by the optimal control at the end of the simulation time. Furthermore, the states and the controls are required to satisfy constraints (4.89a)-(4.89b).

Consider the problem with disturbance as described above. Figure 4.4 shows the results for this problem under the optimal control, the level controller used in the current practice, and the MPC. These controls are depicted in Figure 4.5. It takes about 3 hours to accomplish the optimal control calculation. The computational time of MPC for each predicted horizon is about 20 minute, which is significantly larger than the sampling time 6 minutes. The total computational time of MPC is over 5 times longer when compared with the optimal control method proposed in this chapter. Detailed comparisons between

the results obtained by MPC, the level controller and those obtained by the optimal control method are as follows.

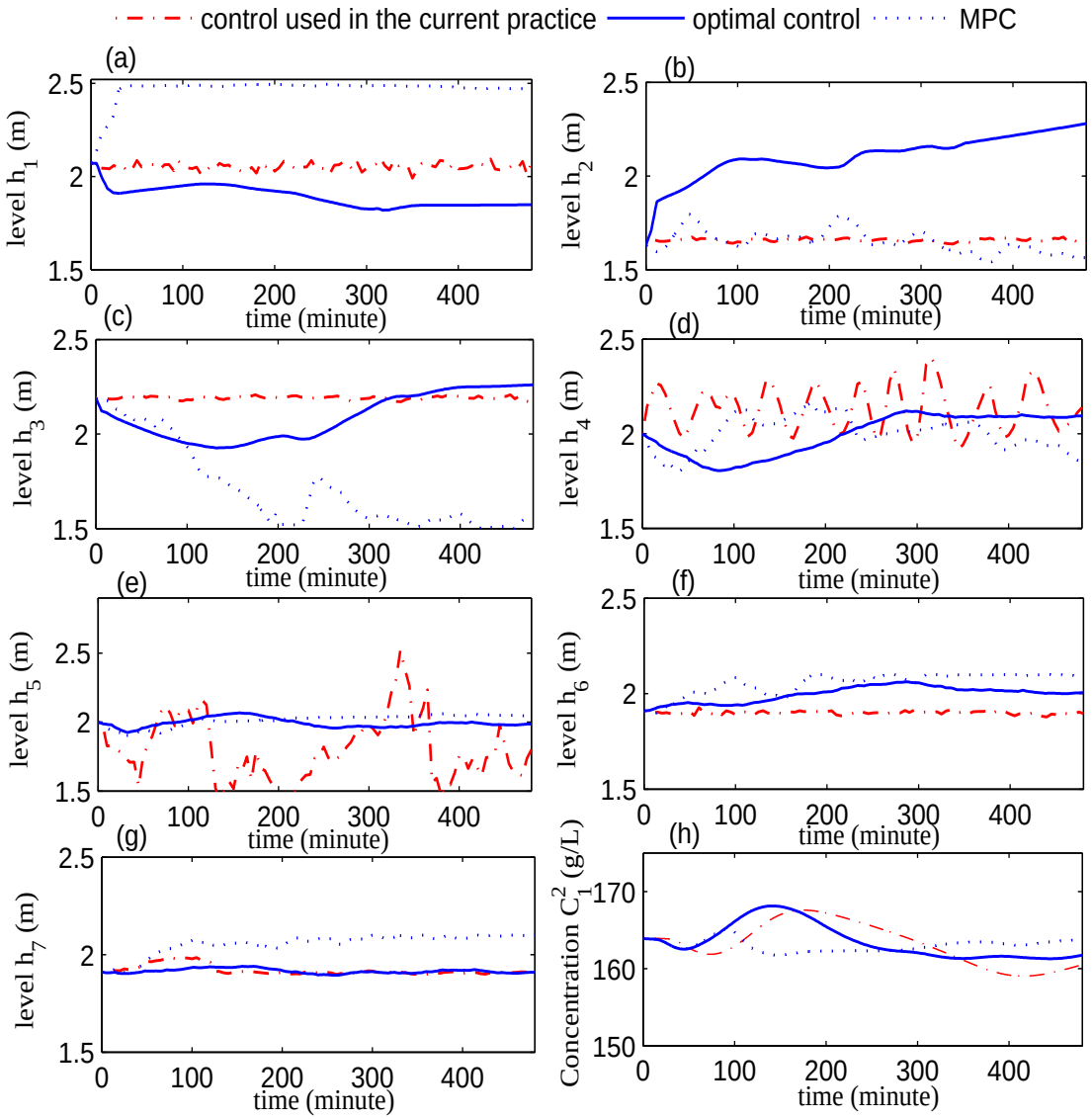

Figure 4.4: State of the evaporation process under the disturbances of the feed concentration and the live steam flow rate

The energy usage value (the mass units of live steam used for evaporating one mass unit of water) obtained by the proposed optimal control is 0.368. With disturbances, the energy usage is 0.369. The energy usage obtained by the MPC controller is 0.371. It is 0.38 under the level controller used in current actual operation.

Figures 4.4(a)-4.4(g) show the changes of the solution levels. Although the solution levels touch the permitted bounds at certain time points, both the optimal control and the MPC drive the solution levels towards inside of the permitted ranges as the simulation time increases. The results obtained by the optimal control have much less oscillations in the solution levels when compared with those obtained using the level controller in the

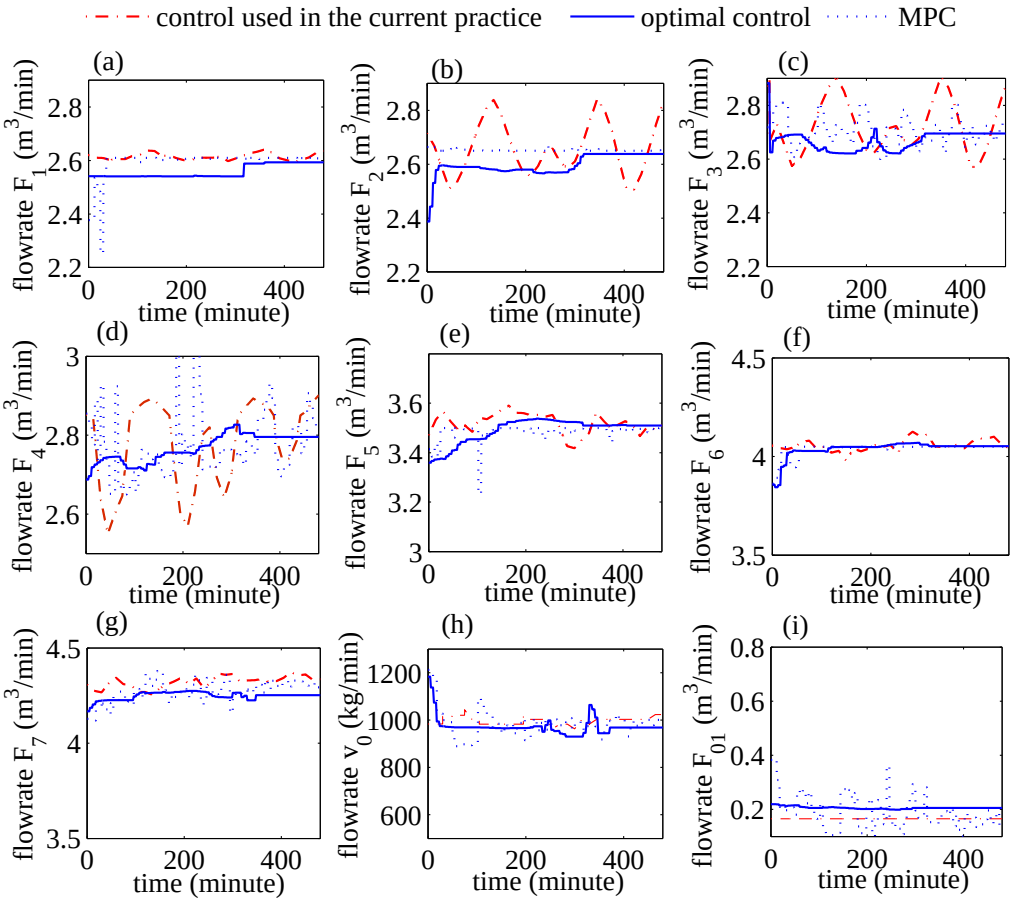

Figure 4.5: Control of the evaporation process

current practice.

In Figure 4.4(h), the plot of the product concentrations shows that the solution concentration is above 160g/L and below 170g/L during the whole time horizon. It means that the optimal control achieves disturbance rejection while maintaining the quality of product concentrations when disturbances occur in the feed concentration and live steam. Furthermore, we can see that only slight fluctuation in the concentration of the final product is observed after 250 minutes. The concentration obtained by using the MPC takes more than 8 hours to approach the desired value.

The live steam flow rate is as shown in Figure 4.5(h). By using the optimal control, the total live steam consumption is 466.96T. Under disturbances, it is 468.50T. On the other hand, by using the level controller in the current practice, the total live steam consumption is 481.76T. By using MPC, the live steam consumption is 471.5T. This represents a significant reduction of live steam consumption being achieved by using the optimal control, even in the presence of disturbances. The two main reasons are: (i) The optimal control improves the performance of the flash evaporator, and (ii) the live steam consumption and the final concentration are involved in the objective function which

is being minimized. Thus, unnecessary live steam usage is reduced while achieving the quality of the final product concentration.

As shown in Figure 4.5, the changes of all the variables of the optimal control and MPC vary strictly inside their bounds. This is due to the imposed continuous inequality constraints on the levers. It is a useful feature in practice, as there are rooms for adjusting the pumps or valves with dead-zone of the alumina evaporation process. The variations of the optimal control variables are much less when compared with those of the MPC.

4.5 Conclusion

In this chapter, we consider a time-delayed optimal control problem subject to continuous state inequality constraints and control constraints. This optimal control problem arises from practical production processes, where the systems are time-delayed dynamic systems. The objective is to find an admissible control such that the energy consumption or the material consumption are minimized, while practical limitations and engineering specifications, which are expressed as continuous inequality constraints on the state variables and the control constants, are satisfied. An efficient numerical algorithm is developed based on the control parameterization technique for solving this constrained time-delayed optimal problem. From solving an optimal control problem in a practical evaporation process, it is observed that the results obtained by the optimal control are superior to those obtained by MPC controller and the controller used in the current practice.

CHAPTER 5

A max-min control problem arising in gradient elution chromatography

5.1 Introduction

Chromatography plays an important role as a separation and purification process in many industrial settings, especially in the preparation of biochemical and pharmaceutical products. A typical chromatographic system shown in Figure 5.1 consists of a column containing an absorbent called the stationary phase, and a liquid that flows through the column called the mobile phase. The absorbent is fixed in the column. The mixture to be separated is injected into the mobile phase and flows through the column. Because different components in the mixture are attracted to the stationary phase at varying degrees, they travel through the column at different speeds, and thus they exit the column at different times (called peak times or retention times). Therefore, the mixture is gradually separated while moving through the column. The separated components are analyzed by a detector at the outlet of the column. The chromatography signal is shown in Figure 5.2.

In practice, chromatography is time-consuming, and the purity of the final product must satisfy strict conditions. Thus, it is essential that the chromatographic process be controlled judiciously by varying mobile phase conditions such as pH value, ionic strength, and flow rate. To achieve a high-quality separation and improve productivity, the minimum duration between successive retention times should be maximized.

In this chapter, we consider an optimal control problem in which manipulative variables in the chromatographic process need to be determined to maximize separation efficiency and achieve optimum separation. This problem has been formulated as a max-min optimal control problem in [90]. It has two non-standard characteristics: (i) The objective function is non-smooth; and (ii) each state variable is defined over a different time horizon. The final time for each state variable, the so-called retention time, is not fixed and actually depends on the control variables. A computational method for solving this problem, based on the control parameterization technique is proposed in [80]. This method involves reformulating the max-min objective function into a more convenient form, then

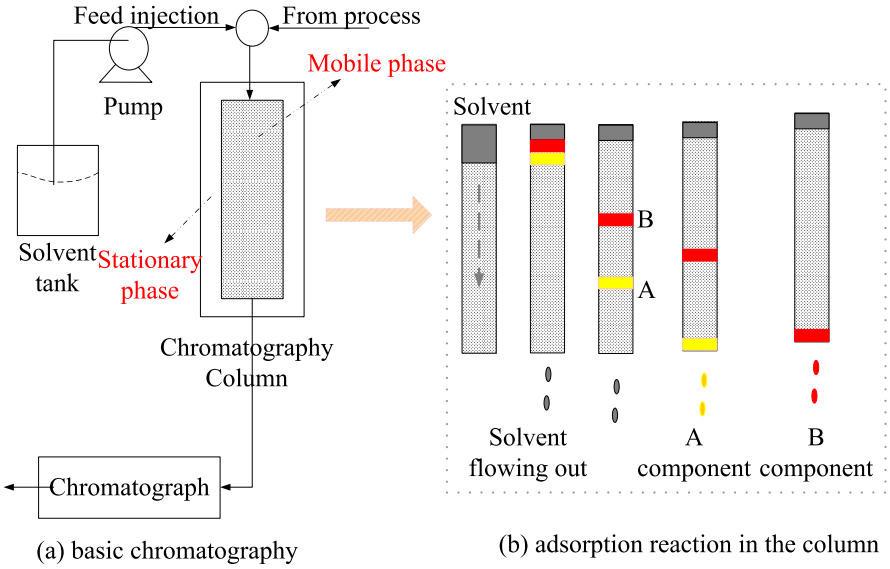

Figure 5.1: Basic chromatography and the reaction principle

approximating the control by a piecewise-constant function, before finally transforming the time horizon into the fixed interval $[0, 1]$. This yields an approximate mathematical programming problem that can be solved using existing optimization algorithms.

The time transformation method used in [90] is based on the substitution $t = st_f$, where $t \in [0, t_f]$ is the original time variable, $s \in [0, 1]$ is the new time variable, and t_f is the final time of the process. The same transformation has been successfully applied to solve time-optimal control problems in [13, 67]. When applied to the chromatography optimal control problem, this transformation does not map the retention times to fixed points, only the final time. Furthermore, the retention times do not necessarily coincide with the control switching times in the new time horizon. In fact, the retention times remain variable under this transformation, which makes them very difficult to compute numerically. This is a major disadvantage, as studies show that the productivity of a chromatographic process is highly sensitive to the retention times [11, 98]. However, although difficult, accurate determination of the retention times is crucial for industrial applications [152].

In the chromatography optimal control problem, an equality state constraint is imposed at each retention time. Such constraints are called characteristic - time constraints in the optimal control literature [120, 121]. Computational methods for solving optimal control problems with characteristic-time constraints is developed in [120, 121]. These methods, however, assume that the ordering of the characteristic times is fixed and known. In the chromatography optimal control problem, the characteristic times are the retention times, and their ordering depends on the control variables. One approach that can be

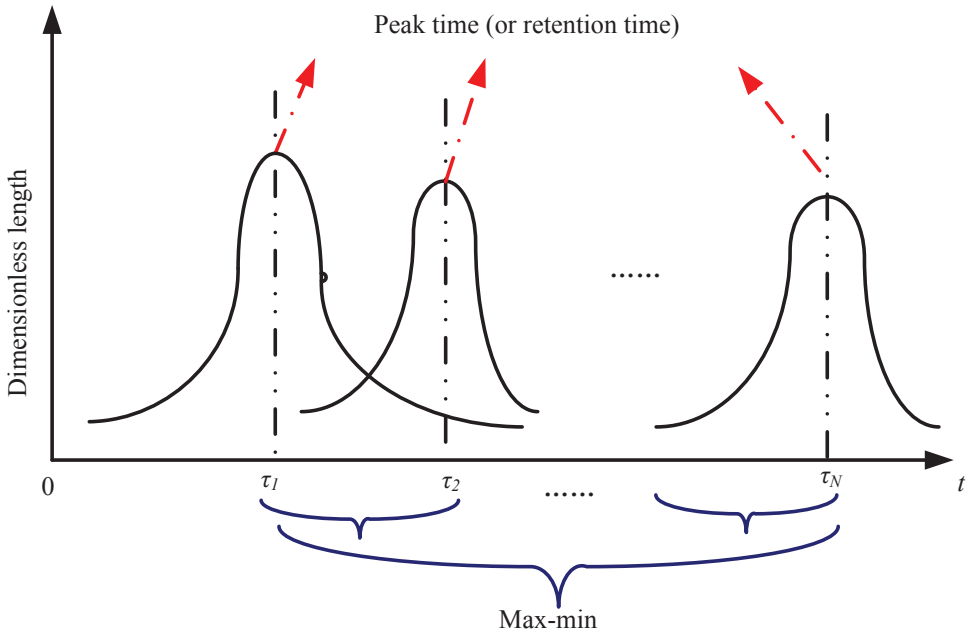

Figure 5.2: The chromatography signal

used to deal with this situation is the bilevel optimization approach [14], whereby the ordering is optimized in the outer level, and the control variables and retention times are optimized in the inner level. Another possible approach is the grid search algorithm [152], whereby the ordering of the retention times is determined after the controls are chosen. However, both the bilevel approach and the grid search algorithm are inefficient because they involve solving a computationally-intensive discrete optimization problem.

In this chapter, we consider the same chromatography optimal control problem formulated in [90]. We propose a new method for reformulating this problem that facilitates accurate determination of the retention times. First, a set of auxiliary decision variables are introduced to govern the ordering of the retention times. Then, after approximating the control variables by piecewise-constant functions, a novel time-scaling transformation is used to map the retention times to fixed points in a new time horizon. We then show that the max-min optimal control problem under consideration is equivalent to a minimization problem subject to additional inequality constraints. This minimization problem can be solved using an exact penalty method [41]. This method is then applied to solve two real world problems. The results show that the approach is both accurate and efficient.

5.2 Problem statement

A gradient elution chromatographic process with N components can be described by the following dynamical system of differential equations [90]:

$$\dot{x}_k(t) = f_k(t, x_k(t), \boldsymbol{u}(t)), \quad t > 0, \quad k = 1, \ldots, N, \tag{5.1}$$

with initial conditions

$$x_k(0) = 0, \quad k = 1, \ldots, N, \tag{5.2}$$

where x_k is the chromatography signal corresponding to the kth component, f_k is the signal velocity corresponding to the kth component, and $\boldsymbol{u}$ is a vector representing mobile phase conditions such as pH value, ionic strength, temperature, and flow rate. In the language of control theory, $\boldsymbol{x} = [x_1, \ldots, x_N]^\top \in \mathbb{R}^N$ is called the state, and $\boldsymbol{u} = [u_1, \ldots, u_p]^\top \in \mathbb{R}^p$ is called the control. Each $f_k : \mathbb{R} \times \mathbb{R} \times \mathbb{R}^p \to \mathbb{R}$ is assumed to be a given continuously differentiable function.

For each $k = 1, \ldots, N$, the retention time τ_k for the kth component is defined by the following equality constraint:

$$x_k(\tau_k) = \int_0^{\tau_k} f_k(t, x_k(t), \boldsymbol{u}(t))\mathrm{d}t = L_k, \quad k = 1, \ldots, N, \tag{5.3}$$

where L_k is the peak height of the chromatography signal x_k. Thus, the state variable x_k is defined on the time horizon $[0, \tau_k]$.

We assume that the p control variables in the chromatographic process are bounded:

$$a_j \leq u_j(t) \leq b_j, \quad t \geq 0, \quad j = 1, \ldots, p, \tag{5.4}$$

where a_j and b_j are the lower and upper bounds of the jth control variable, respectively. Given a piecewise continuous function $\boldsymbol{u} : [0, \infty) \to \mathbb{R}^p$ satisfying (5.4), we can solve the system defined by (5.1) and (5.2) to yield a corresponding state trajectory.

To ensure that the gradient elution process is effective, the control variables should be chosen so that the minimum duration between successive retention times is maximized. Hence, we introduce the following objective function:

$$J(\boldsymbol{u}) = \min_{i \neq j} \left\{ \frac{(\tau_j - \tau_i)^2}{\tau_f} \right\}, \tag{5.5}$$

where τ_f is the terminal time for the chromatographic process defined by

$$\tau_f = \max_{k=1,\ldots,N} \{\tau_k\}.$$

We now introduce the following optimal control problem.

Problem (P). *Given the system defined by* (5.1) *and* (5.2)*, choose a control* $\boldsymbol{u} : [0, \infty) \to \mathbb{R}^p$ *and the corresponding retention times* τ_k*,* $k = 1, \ldots, N$*, such that the objective function* (5.5) *is maximized subject to the terminal constraints* (5.3) *and the control constraints* (5.4)*.*

Compared with standard optimal control problems, Problem (P) has several unusual characteristics:

(i) The max-min objective function is non-smooth.

(ii) Each state variable is defined over a different time horizon.

(iii) The retention times and the ordering of the retention times are not fixed, but instead depend on the control through (5.3).

Thus, Problem (P) is a highly non-standard optimal control problem and cannot be solved directly using standard methods such as the Pontryagin minimum principle [92] or state discretization [38, 148].

5.3 Problem transformation

Let $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_N]^\top$ be a vector containing the retention times. Furthermore, let v_{ij}, $i = 1, \ldots, N$; $j = 1, \ldots, N$, be a set of auxiliary decision variables controlling the order of the retention times, where

$$v_{ij} = \begin{cases} 1, & \text{if component } j \text{ has the } i\text{th earliest retention time,} \\ 0, & \text{otherwise.} \end{cases} \tag{5.6}$$

Clearly,

$$\sum_{j=1}^{N} v_{ij} = 1, \quad i = 1, \ldots, N, \tag{5.7}$$

and

$$\sum_{i=1}^{N} v_{ij} = 1, \quad j = 1, \ldots, N. \tag{5.8}$$

We collect the auxiliary variables into a vector $\boldsymbol{v} = [\boldsymbol{v}_1^\top, \ldots, \boldsymbol{v}_N^\top]^\top \in \mathbb{R}^{NN}$, where $\boldsymbol{v}_i = [v_{i1}, \ldots, v_{iN}]^\top$. As an example, consider a 3-component mixture in which component 2 is the first component to exit the chromatography column, component 1 is the second, and component 3 is the last. Then $\tau_2 < \tau_1 < \tau_3$, and thus $\boldsymbol{v}_1 = [0, 1, 0]^\top$, $\boldsymbol{v}_2 = [1, 0, 0]^\top$, and $\boldsymbol{v}_3 = [0, 0, 1]^\top$.

Standard gradient-based optimization techniques cannot handle binary constraints such as (5.6). Thus, we replace the 0-1 constraints on v_{ij} by the following constraints:

$$\sum_{j=1}^{N} v_{ij}(j^2 - j + \tfrac{1}{3}) - \left\{ \sum_{j=1}^{N} v_{ij}(j - \tfrac{1}{2}) \right\}^2 = \frac{1}{12}, \quad i = 1, \ldots, N, \tag{5.9}$$

and

$$0 \leq v_{ij} \leq 1, \quad i = 1, \ldots, N, \quad j = 1, \ldots, N. \tag{5.10}$$

The following result shows that (5.9) and (5.10) imply $v_{ij} \in \{0, 1\}$.

Theorem 5.1. *Suppose that v_{ij}, $i = 1, \ldots, N$; $j = 1, \ldots, N$, satisfy* (5.7) *and* (5.10). *Then, for each $i = 1, \ldots, N$,* (5.9) *holds if and only if there exists a $k \in \{1, \ldots, N\}$ such that $v_{ik} = 1$ and $v_{ij} = 0$ for all $j \neq k$.*

Proof. Let $i \in \{1, \ldots, N\}$ be fixed but arbitrary, and assume that $v_{ik} = 1$ and $v_{ij} = 0$ for all $j \neq k$. Then

$$\sum_{j=1}^{N} v_{ij}(j^2 - j + \tfrac{1}{3}) - \left\{ \sum_{j=1}^{N} v_{ij}(j - \tfrac{1}{2}) \right\}^2 = k^2 - k + \tfrac{1}{3} - (k - \tfrac{1}{2})^2 = \frac{1}{12}. \tag{5.11}$$

To prove the opposite implication, we use similar arguments to those used in the proof of Lemma 3.1 in [65]. First, consider the following optimization problem, which we call Problem (Q):

$$\min_{v_{i1}, \ldots, v_{iN}} \quad \sum_{j=1}^{N} v_{ij}(j^2 - j + \tfrac{1}{3}) - \left\{ \sum_{j=1}^{N} v_{ij}(j - \tfrac{1}{2}) \right\}^2 \tag{5.12}$$

$$\text{s.t.} \quad \sum_{j=1}^{N} v_{ij} = 1, \tag{5.13}$$

$$v_{ij} \geq 0, \quad j = 1, \ldots, N. \tag{5.14}$$

Note that Problem (Q) has a continuous objective function and a compact feasible region. Hence, it admits at least one optimal solution. The Lagrangian for Problem (Q) is defined by

$$\mathcal{L} = \sum_{j=1}^{N} v_{ij}(j^2 - j + \tfrac{1}{3}) - \left\{ \sum_{j=1}^{N} v_{ij}(j - \tfrac{1}{2}) \right\}^2 - \rho \left\{ \sum_{j=1}^{N} v_{ij} - 1 \right\} - \sum_{j=1}^{N} \eta_j v_{ij},$$

where ρ is the Lagrange multiplier for the equality constraint, and $\eta_j \geq 0$, $j = 1, \ldots, N$, are the Lagrange multipliers for the inequality constraints.

Since the linear independence constraint qualification is clearly satisfied in Prob-

lem (Q), any optimal solution must satisfy the following Kuhn-Tucker conditions [73]:

$$\frac{\partial \mathcal{L}}{\partial v_{ik}} = k^2 - k + \tfrac{1}{3} - 2\Big\{\sum_{j=1}^{N} v_{ij}(j-\tfrac{1}{2})\Big\}(k-\tfrac{1}{2}) - \rho - \eta_k = 0, \qquad k = 1, \ldots, N, \tag{5.15}$$

and

$$\eta_k v_{ik} = 0, \quad k = 1, \ldots, N. \tag{5.16}$$

From (5.15), we obtain

$$g(k) - \eta_k = 0, \quad k = 1, \ldots, N,$$

where $g : \mathbb{R} \to \mathbb{R}$ is a quadratic function defined by

$$\begin{aligned} g(y) &= y^2 - y + \tfrac{1}{3} - 2\Big\{\sum_{j=1}^{N} v_{ij}(j-\tfrac{1}{2})\Big\}(y-\tfrac{1}{2}) - \rho \\ &= y^2 - \Big[1 + 2\Big\{\sum_{j=1}^{N} v_{ij}(j-\tfrac{1}{2})\Big\}\Big]y + \Big[\tfrac{1}{3} + \sum_{j=1}^{N} v_{ij}(j-\tfrac{1}{2}) - \rho\Big]. \end{aligned}$$

Clearly $g(k) = 0$ if and only if $\eta_k = 0$. Since $g(y)$ is a quadratic function with at most two real roots, no more than two of the multipliers for the inequality constraints in Problem (Q) are zero.

Suppose that $\eta_k > 0$ for all $k = 1, \ldots, N$. Then, by (5.16), $v_{ik} = 0$ for all $k = 1, \ldots, N$, contradicting (5.13) in Problem (Q). Thus, it suffices to consider the following two cases:

(a) There exist an integer k_1 such that $\eta_{k_1} = 0$ and $\eta_k > 0$ for all $k \neq k_1$ (exactly one of the multipliers is zero).

(b) There exists integers k_1 and k_2 such that $\eta_{k_1} = \eta_{k_2} = 0$ and $\eta_k > 0$ for all $k \neq k_1, k_2$ (exactly two of the multipliers are zero).

Consider Case (a). In this case, (5.16) and the equality constraint (5.13) in Problem (Q) imply that $v_{ik_1} = 1$ and $v_{ik} = 0$ for all $k \neq k_1$. Thus, as shown in (5.11), the optimal cost of Problem (Q) is equal to $\frac{1}{12}$.

We now consider Case (b). In this case, (5.16) implies that $v_{ik} = 0$ for all $k \neq k_1, k_2$. Hence, from the equality constraint (5.13) in Problem (Q), we have $v_{ik_2} = 1 - v_{ik_1}$. Thus,

the optimal cost of Problem (Q) is

$$\begin{aligned}
&\sum_{j=1}^{N} v_{ij}(j^2 - j + \tfrac{1}{3}) - \Big\{\sum_{j=1}^{N} v_{ij}(j - \tfrac{1}{2})\Big\}^2 \\
&\qquad = v_{ik_1}(k_1^2 - k_1 + \tfrac{1}{3}) + v_{ik_2}(k_2^2 - k_2 + \tfrac{1}{3}) - \Big\{v_{ik_1}(k_1 - \tfrac{1}{2}) + v_{ik_2}(k_2 - \tfrac{1}{2})\Big\}^2 \\
&\qquad = v_{ik_1}(k_1^2 - k_1 + \tfrac{1}{3}) + (1 - v_{ik_1})(k_2^2 - k_2 + \tfrac{1}{3}) \\
&\qquad\quad - \Big\{v_{ik_1}(k_1 - \tfrac{1}{2}) + (1 - v_{ik_1})(k_2 - \tfrac{1}{2})\Big\}^2 \\
&\qquad = \tfrac{1}{12} + (v_{ik_1} - v_{ik_1}^2)(k_1 - k_2)^2 \geq \tfrac{1}{12},
\end{aligned}$$

where the last inequality follows from $0 \leq v_{ij} \leq 1$. Clearly, from Cases (a) and (b), the minimum value of the cost function in Problem (Q) is $\frac{1}{12}$. Furthermore, this minimum is achieved only when there exists a $k \in \{1, \ldots, N\}$ such that $v_{ik} = 1$ and $v_{ij} = 0$, $j \neq k$. This completes the proof. □

On the basis of Theorem 5.1, we can replace the binary constraints (5.6) by the non-discrete constraints (5.7), (5.9), and (5.10). As we will see later, this reformulation enables us to determine the optimal retention time ordering by using an exact penalty function method.

To proceed, we now use the control parameterization technique [80] to approximate Problem (P) by a finite-dimensional optimization problem. This is done by approximating the control $\boldsymbol{u}$ by a piecewise-constant function that switches value at each retention time and at $q-1$ times between each pair of successive retention times. Thus, the time horizon is divided into qN subintervals, with q subintervals between each pair of successive retention times, and the control is approximated by a constant value on each subinterval.

For each $j = 1, \ldots, p$, the control u_j is approximated as follows:

$$u_j(t) = \sum_{i=1}^{qN} \sigma_j^i \chi_{[t_{i-1}, t_i)}(t), \quad t \in [0, \tau_f], \tag{5.17}$$

where t_i, $i = 0, \ldots, qN$, are the control switching times; σ_j^i, $i = 1, \ldots, qN$; $j = 1, \ldots, p$, are the control heights; and $\chi_{[t_{i-1}, t_i)}$ is the indicator function for the subinterval $[t_{i-1}, t_i)$ defined by

$$\chi_{[t_{i-1}, t_i)}(t) = \begin{cases} 1, & \text{if } t \in [t_{i-1}, t_i), \\ 0, & \text{otherwise.} \end{cases} \tag{5.18}$$

Here, $0 \leq t_0 \leq t_1 \leq \cdots \leq t_{qN} = \tau_f$. Furthermore, for each $k = 1, \ldots, N$, the switching time t_{kq} (the right end-point of subinterval $[t_{kq-1}, t_{kq}]$) coincides with one of the retention times. Our objective is to choose the control heights and control switching times in (5.17) appropriately so that the objective function (5.5) is maximized. Note that the control

approximation scheme used in (5.17) is more flexible than the one used in [90], which does not allow the switching times to be determined optimally (instead they are pre-fixed).

It is well-known that treating the control switching times as decision variables causes major problems in numerical computation, see [67, 121]. Hence, we will use the so-called time-scaling transformation [81] to map the switching times to fixed points in a new time horizon. This involves introducing a new time variable $s \in [0, qN]$, and then relating s to t through the following differential equation:

$$\begin{aligned} \frac{dt(s)}{ds} &= \omega(s), \\ t(0) &= 0, \end{aligned} \tag{5.19}$$

where $\omega : [0, qN] \to [0, \infty)$ is a piecewise-constant function with switching points at the fixed locations $s = i$, $i = 1, \ldots, qN - 1$. We express ω mathematically as follows:

$$\omega(s) = \sum_{i=1}^{qN} \theta_i \chi_{[i-1,i)}(s), \tag{5.20}$$

where

$$\theta_i = t_i - t_{i-1} \geq 0, \quad i = 1, \ldots, qN. \tag{5.21}$$

It follows from (5.19) that for $s \in [l-1, l]$, we have

$$t(s) = \int_0^s \omega(\eta) d\eta = \sum_{j=1}^{l-1} \theta_j + \theta_l (s - l + 1).$$

Furthermore, for each $l = 1, \ldots, qN$,

$$t(l) = \sum_{j=1}^{l} \theta_j = \sum_{j=1}^{l} (t_j - t_{j-1}) = t_l.$$

This shows that the time-scaling transformation defined by (5.19) and (5.20) maps the control switching times to fixed integers.

After applying the time-scaling transformation, the control variables defined in (5.17) are written as:

$$\tilde{u}_j(s) = u_j(t(s)) = \sum_{i=1}^{qN} \sigma_j^i \chi_{[i-1,i)}(s), \quad j = 1, \ldots, p, \tag{5.22}$$

where the control heights satisfy the following constraints:

$$a_j \leq \sigma_j^i \leq b_j, \quad j = 1, \ldots, p, \quad i = 1, \ldots, qN. \tag{5.23}$$

Define $\boldsymbol{\sigma} = [(\boldsymbol{\sigma}^1)^\top, \ldots, (\boldsymbol{\sigma}^{qN})^\top]^\top \in \mathbb{R}^{qNp}$, where $\boldsymbol{\sigma}^i = [\sigma_1^i, \ldots, \sigma_p^i]^\top \in \mathbb{R}^p$, $i = 1, \ldots, qN$.

Furthermore, let $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_{qN}]^\top$.

Under the time-scaling transformation, the system defined by (5.1) and (5.2) becomes

$$\begin{aligned} \frac{d\tilde{x}_k(s)}{ds} &= \sum_{i=1}^{qN} \theta_i f_k(t(s), \tilde{x}_k(s), \boldsymbol{\sigma}^i)\chi_{[i-1,i)}(s), \quad k = 1, \ldots, N, \\ \frac{dt(s)}{ds} &= \omega(s), \end{aligned} \tag{5.24}$$

with initial conditions

$$\begin{aligned} \tilde{x}_k(0) &= 0, \quad k = 1, \ldots, N, \\ t(0) &= 0. \end{aligned} \tag{5.25}$$

The objective function (5.5) becomes

$$\begin{aligned} \tilde{J}(\boldsymbol{\sigma}, \boldsymbol{\theta}, \boldsymbol{v}) &= \min_{i=1,\ldots,N-1} \left\{ \frac{(t_{(i+1)q} - t_{iq})^2}{t_{qN}} \right\} \\ &= \min_{i=1,\ldots,N-1} \frac{(\theta_{iq+1} + \theta_{iq+2} + \cdots + \theta_{iq+q})^2}{\theta_1 + \theta_2 + \cdots + \theta_{qN}}. \end{aligned} \tag{5.26}$$

For each $i = 1, \ldots, N$, exactly one component in the chromatography system will have its retention time at $t = t_{iq}$. Hence, we have the following interior point constraints:

$$\sum_{k=1}^{N} v_{ik}(\tilde{x}_k(iq) - L_k) = 0, \quad i = 1, \ldots, N. \tag{5.27}$$

In view of our discussion above, Problem (P) can be approximated by the following optimization problem.

Problem (P1). *Given the system defined by* (5.24) *and* (5.25)*, find vectors* $\boldsymbol{\sigma} \in \mathbb{R}^{qNp}$*,* $\boldsymbol{\theta} \in \mathbb{R}^{qN}$*, and* $\boldsymbol{v} \in \mathbb{R}^{NN}$*, such that the objective function* (5.26) *is maximized subject to* (5.7)-(5.10)*,* (5.21)*,* (5.23) *and interior point constraints* (5.27)*.*

Although Problem (P1) is a finite-dimensional optimization problem, it is still difficult to solve because the objective function (5.26) is non-smooth. Thus, we introduce a new decision parameter ξ, where

$$\xi = \min_{i=1,\ldots,N-1} \frac{(\theta_{iq+1} + \theta_{iq+2} + \cdots + \theta_{iq+q})^2}{\theta_1 + \theta_2 + \cdots + \theta_{qN}}.$$

Clearly, for each $i = 1, \ldots, N-1$, the following inequality is satisfied:

$$\frac{(\theta_{iq+1} + \theta_{iq+2} + \cdots + \theta_{iq+q})^2}{\theta_1 + \theta_2 + \cdots + \theta_{qN}} \geq \xi.$$

Since $\theta_1 + \cdots + \theta_{qN} > 0$, this inequality can be rewritten as:

$$\xi \sum_{k=1}^{qN} \theta_k - \left\{ \sum_{l=iq+1}^{iq+q} \theta_l \right\}^2 \leq 0, \quad i = 1, \ldots, N-1. \tag{5.28}$$

Thus, Problem (P1) is equivalent to the following problem.

Problem (P2). *Given the system defined by* (5.24) *and* (5.25)*, find vectors* $\boldsymbol{\sigma} \in \mathbb{R}^{qNp}$*,* $\boldsymbol{\theta} \in \mathbb{R}^{qN}$*, and* $\boldsymbol{v} \in \mathbb{R}^{NN}$*, and the parameter* ξ*, to minimize the cost function*

$$\hat{J}(\boldsymbol{\sigma}, \boldsymbol{\theta}, \boldsymbol{v}, \xi) = -\xi \tag{5.29}$$

subject to (5.7)-(5.10)*,* (5.21)*,* (5.23)*,* (5.27) *and* (5.28)*.*

5.4 A computational method

Problem (P2) is a finite-dimensional optimization problem with nonlinear equality and nonlinear inequality constraints. The major difficulty with solving this problem is that the constraints (5.7) - (5.10) force v_{ij} to be binary decision variables, and thus the feasible region for Problem (P2) is disjoint. Standard gradient-based optimization methods usually fail miserably when applied to problems with disjoint feasible regions. In this section, we will show how to solve Problem (P2) using an exact penalty function method.

First, let

$$g_m(\boldsymbol{\theta}, \xi) = \xi \sum_{k=1}^{qN} \theta_k - \left\{ \sum_{l=mq+1}^{mq+q} \theta_l \right\}^2, \quad m = 1, \ldots, N-1.$$

Furthermore, for each $m = 1, \ldots, N$, let

$$\begin{aligned}
h_m(\boldsymbol{v}) &= \sum_{j=1}^{N} v_{mj} - 1, \\
\hat{h}_m(\boldsymbol{v}) &= \sum_{i=1}^{N} v_{im} - 1, \\
\tilde{h}_m(\boldsymbol{v}) &= \sum_{j=1}^{N} v_{mj}(j^2 - j + \tfrac{1}{3}) - \left\{ \sum_{j=1}^{N} v_{mj}(j - \tfrac{1}{2}) \right\}^2 - \frac{1}{12}.
\end{aligned}$$

We will use an exact penalty function approach, recently developed in [41], to handle the equality constraints (5.7)- (5.9), and the inequality constraints (5.28). This approach has been successively used to solve semi-infinite and discrete optimization problems [40–42].

Consider the following penalty function:

$$\hat{J}_\eta(\boldsymbol{\sigma},\boldsymbol{\theta},\boldsymbol{v},\xi,\varepsilon) = -\xi + \varepsilon^{-\alpha}(\Delta_1(\boldsymbol{\theta},\xi,\varepsilon) + \Delta_2(\boldsymbol{v})) + \eta\varepsilon^{\beta}, \tag{5.30}$$

where $\varepsilon > 0$ is a new decision variable and

$$\Delta_1(\boldsymbol{\theta},\xi,\varepsilon) = \sum_{m=1}^{N-1}\left\{\max\{0, g_m(\boldsymbol{\theta},\xi) - \varepsilon^{\gamma}\kappa_m\}\right\}^2,$$
$$\Delta_2(\boldsymbol{v}) = \sum_{m=1}^{N}\left\{(h_m(\boldsymbol{v}))^2 + (\hat{h}_m(\boldsymbol{v}))^2 + (\tilde{h}_m(\boldsymbol{v}))^2\right\}.$$

Here, $\alpha > 0$, $\beta > 2$, $\gamma > 0$, and $\kappa_m \in (0,1)$, $m = 1,\ldots,N-1$, are fixed constants, and $\eta > 0$ is the penalty parameter. Note that $\Delta_1(\boldsymbol{\theta},\xi,\varepsilon)$ measures violations in (5.28), while $\Delta_2(\boldsymbol{v})$ measures violations in (5.7) - (5.9). The idea is that when the penalty parameter η is large, the final term in (5.30) forces ε to be small, which in turn causes the middle term in (5.30) to penalize constraint violations very severely. Hence, minimizing the penalty function will lead to a feasible point of Problem (P2). With this in mind, we introduce the following exact penalty function problem for Problem (P2).

Problem (P3). *Given the system defined by* (5.24) *and* (5.25)*, find vectors* $\boldsymbol{\sigma} \in \mathbb{R}^{qNp}$*,* $\boldsymbol{\theta} \in \mathbb{R}^{qN}$*,* $\boldsymbol{v} \in \mathbb{R}^{NN}$*, and the parameters* ξ *and* ε*, such that the penalty function* (5.30) *is minimized subject to the bound constraints* (5.10)*,* (5.21)*, and* (5.23)*, and the interior point constraints* (5.27)*.*

Problem (P3) is an optimal parameter selection problem with bound constraints on decision parameters $\boldsymbol{\sigma}$, $\boldsymbol{\theta}$, $\boldsymbol{v}$, ξ, ε and interior point constraints (5.27). Unlike the non-smooth approximate problem derived in [90], Problem (P3) can be solved using standard computational methods such as those developed in Chapter 5 of [80]. These computational methods have been implemented in the optimal control software package MISER [91].

To solve Problem (P3), MISER requires gradient formulae for both the objective function and the constraint functions. The gradients of $\hat{J}_\eta$ are given below:

$$\frac{\partial \hat{J}_\eta}{\partial \theta_i} = 2\varepsilon^{-\alpha}\sum_{m=1}^{N-1}\left[\max\{0, g_m(\boldsymbol{\theta},\xi) - \varepsilon^{\gamma}\kappa_m\}\right]\left[\xi - 2\left\{\sum_{k=mq+1}^{mq+q}\theta_k\right\}\chi_{[mq+1,mq+q]}(i)\right],$$
$$\frac{\partial \hat{J}_\eta}{\partial v_{ij}} = 2\varepsilon^{-\alpha}\sum_{m=1}^{N}\left\{h_m(\boldsymbol{v})\frac{\partial h_m(\boldsymbol{v})}{\partial v_{ij}} + \hat{h}_m(\boldsymbol{v})\frac{\partial \hat{h}_m(\boldsymbol{v})}{\partial v_{ij}} + \tilde{h}_m(\boldsymbol{v})\frac{\partial \tilde{h}_m(\boldsymbol{v})}{\partial v_{ij}}\right\},$$
$$\frac{\partial \hat{J}_\eta}{\partial \xi} = -1 + 2\varepsilon^{-\alpha}\sum_{m=1}^{N-1}\left\{\left[\max\{0, g_m(\boldsymbol{\theta},\xi) - \varepsilon^{\gamma}\kappa_m\}\right]\sum_{k=1}^{qN}\theta_k\right\},$$

$$\frac{\partial \hat{J}_\eta}{\partial \varepsilon} = 2\varepsilon^{-\alpha} \sum_{m=1}^{N-1} \left\{ -\gamma \kappa_m \varepsilon^{\gamma-1} \left[\max\{0, g_m(\boldsymbol{\theta}, \xi) - \varepsilon^\gamma \kappa_m\} \right] \right\} - \alpha \varepsilon^{-\alpha-1} (\Delta_1(\boldsymbol{\theta}, \xi, \varepsilon) + \Delta_2(\boldsymbol{v})) + \beta\eta\varepsilon^{\beta-1},$$

$$\frac{\partial \hat{J}_\eta}{\partial \sigma_j^i} = 0,$$

where

$$\frac{\partial \tilde{h}_m(\boldsymbol{v})}{\partial v_{ij}} = \begin{cases} (j^2 - j + \frac{1}{3}) - 2(j - \frac{1}{2}) \left\{ \sum_{j=1}^N v_{ij}(j - \frac{1}{2}) \right\}, & \text{if } i = m, \\ 0, & \text{if } i \neq m, \end{cases}$$

and

$$\frac{\partial h_m(\boldsymbol{v})}{\partial v_{ij}} = \begin{cases} 1, & \text{if } i = m, \\ 0, & \text{if } i \neq m, \end{cases} \qquad \frac{\partial \hat{h}_m(\boldsymbol{v})}{\partial v_{ij}} = \begin{cases} 1, & \text{if } j = m, \\ 0, & \text{if } j \neq m. \end{cases}$$

We now consider the gradient of the interior point constraints (5.27). First, define

$$\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v}) = \sum_{k=1}^N v_{mk}(\tilde{x}_k(mq) - L_k), \quad m = 1, \ldots, N. \tag{5.31}$$

That is, Φ_m is the left-hand side of the interior point constraints (5.27) (with index i replaced by m). The partial derivatives of Φ_m with respect to ξ, ε, and v_{ij} can be computed directly:

$$\frac{\partial \Phi_m}{\partial \varepsilon} = \frac{\partial \Phi_m}{\partial \xi} = 0,$$

and

$$\frac{\partial \Phi_m}{\partial v_{ij}} = \begin{cases} \tilde{x}_j(mq) - L_j, & \text{if } i = m, \\ 0, & \text{if } i \neq m. \end{cases}$$

The partial derivatives with respect to $\boldsymbol{\sigma}$ and $\boldsymbol{\theta}$, however, are more difficult because $\boldsymbol{\sigma}$ and $\boldsymbol{\theta}$ influence Φ_m implicitly through the dynamic system defined by (5.24) and (5.25). They are derived via formulas given below. Define the Hamiltonian function

$$H_m = \sum_{k=1}^N \lambda_k^m(s)\omega(s) f_k(t(s), \tilde{x}_k(s), \tilde{\boldsymbol{u}}(s)) + \lambda_{N+1}^m(s)\omega(s), \tag{5.32}$$

where the costate functions $\lambda_k^m : \mathbb{R} \to \mathbb{R}$ satisfy the dynamic systems

$$\dot{\lambda}_k^m(s) = -\frac{\partial H_m}{\partial \tilde{x}_k}, \quad s \in [0, mq], \quad k = 1, \ldots, N, \tag{5.33}$$

$$\lambda_k^m(mq) = v_{mk}, \tag{5.34}$$

and

$$\dot{\lambda}^m_{N+1}(s) = -\frac{\partial H_m}{\partial t}, \quad s \in [0, mq], \tag{5.35}$$

$$\lambda^m_{N+1}(mq) = 0. \tag{5.36}$$

For each m, $m = 1, \ldots, N$, we have the following results.

Theorem 5.2. *The partial derivatives of Φ_m with respect to $\boldsymbol{\sigma}$ is*

$$\frac{\partial \Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v})}{\partial \sigma^i_j} = \begin{cases} \int_{i-1}^{i} \frac{\partial H_m}{\partial \sigma_j} \mathrm{d}s, & \text{if } i \le mq, \\ 0, & \text{if } i > mq, \end{cases} \tag{5.37}$$

Proof. Let the control parameter vector $\boldsymbol{\sigma}$ be perturbed by $\epsilon\boldsymbol{\rho}$, where $\epsilon > 0$ is a small real number and $\boldsymbol{\rho}$ is an arbitrary fixed perturbation of $\boldsymbol{\sigma}$. Then, we have

$$\boldsymbol{\sigma}(\epsilon) = \boldsymbol{\sigma} + \epsilon\boldsymbol{\rho},$$

where $\boldsymbol{\rho} = [(\boldsymbol{\rho}^1)^\top, \ldots, (\boldsymbol{\rho}^{qN})^\top]^\top$, and $\boldsymbol{\sigma}(\epsilon) = [(\boldsymbol{\sigma}^1(\epsilon))^\top, \ldots, (\boldsymbol{\sigma}^{qN}(\epsilon))^\top]^\top$. Consequently, the system state $\tilde{\boldsymbol{x}}$ as well as the function Φ_m will be perturbed. Let

$$\tilde{x}_k(s, \epsilon) = \tilde{x}_k(s|\boldsymbol{\sigma}(\epsilon)), \quad k = 1, \ldots, N.$$

Clearly,

$$\tilde{x}_k(s, \epsilon) = \tilde{x}_k(0) + \sum_{i=1}^{qN} \int_0^s \theta_i f_k(t(l), \tilde{x}_k(l), \boldsymbol{\sigma}^i(\epsilon))(l) dl.$$

Let $f_k(t(\cdot), \tilde{x}_k(\cdot), \boldsymbol{\sigma}^i(\epsilon))(\cdot)$ be written as $f_k(\cdot)$ for brevity. Then, by the chain rule, we have

$$\Delta\tilde{x}_k(s) = \frac{d\tilde{x}_k(s, \epsilon)}{d\epsilon}\Big|_{\epsilon=0} = \sum_{i=1}^{qN} \int_0^s \theta_i \Big\{ \frac{\partial f_k(l)}{\partial \tilde{x}_k} \Delta\tilde{x}_k(l) + \frac{\partial f_k(l)}{\partial \boldsymbol{\sigma}^i} \boldsymbol{\rho}^i \Big\} dl.$$

Clearly,

$$\frac{d\Delta\tilde{x}_k(s)}{ds} = \sum_{i=1}^{qN} \theta_i \Big\{ \frac{\partial f_k(s)}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) + \frac{\partial f_k(s)}{\partial \boldsymbol{\sigma}^i} \boldsymbol{\rho}^i \Big\}. \tag{5.38}$$

By using (5.32), we have

$$\Delta H_m = \frac{\partial H_m}{\partial \boldsymbol{\sigma}} \boldsymbol{\rho} + \sum_{k=1}^{N} \frac{\partial H_m}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) = \sum_{k=1}^{N} \lambda^m_k(s)\omega(s) \Big\{ \frac{\partial f_k(s)}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) + \frac{\partial f_k(s)}{\partial \boldsymbol{\sigma}} \boldsymbol{\rho} \Big\}.$$

Using the definition of $\omega(s)$ in (5.20), it is clear that

$$\frac{\partial H_m}{\partial \boldsymbol{\sigma}}\boldsymbol{\rho} + \sum_{k=1}^{N} \frac{\partial H_m}{\partial \tilde{x}_k}\Delta\tilde{x}_k(s) = \sum_{i=1}^{qN}\sum_{k=1}^{N} \lambda_k^m(s)\theta_i\Big\{\frac{\partial f_k(s)}{\partial \tilde{x}_k}\Delta\tilde{x}_k(s) + \frac{\partial f_k(s)}{\partial \boldsymbol{\sigma}^i}\boldsymbol{\rho}^i\Big\}. \tag{5.39}$$

Substituting (5.38) into (5.39) yields

$$\frac{\partial H_m}{\partial \boldsymbol{\sigma}}\boldsymbol{\rho} + \sum_{k=1}^{N} \frac{\partial H_m}{\partial \tilde{x}_k}\Delta\tilde{x}_k(s) - \sum_{k=1}^{N} \lambda_k^m(s)\frac{d\Delta\tilde{x}_k(s)}{ds} = 0. \tag{5.40}$$

Then, by the chain rule, we obtain

$$\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v}) = \frac{d\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v})}{d\epsilon}\Big|_{\epsilon=0} = \frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v})}{\partial \tilde{x}_k}\Delta\tilde{x}_k(mq). \tag{5.41}$$

Adding (5.40) to the right hand side of (5.41), gives

$$\begin{aligned}\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v}) = &\sum_{k=1}^{N} \frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v})}{\partial \tilde{x}_k}\Delta\tilde{x}_k(mq) + \int_0^{mq} \frac{\partial H_m}{\partial \boldsymbol{\sigma}}\boldsymbol{\rho} dl \\ &+ \sum_{k=1}^{N}\int_0^{mq} \Big\{\frac{\partial H_m}{\partial \tilde{x}_k}\Delta\tilde{x}_k(l) - \lambda_k^m(l)\frac{d\Delta\tilde{x}_k(l)}{dl}\Big\} dl\end{aligned} \tag{5.42}$$

Integrating the last term of (5.42) by parts gives

$$\begin{aligned}\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v}) = &\sum_{k=1}^{N} \frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v})}{\partial \tilde{x}_k}\Delta\tilde{x}_k(mq) - \sum_{k=1}^{N} \lambda_k^m(l)\Delta\tilde{x}_k(l)\Big|_0^{mq} \\ &+ \int_0^{mq} \frac{\partial H_m}{\partial \boldsymbol{\sigma}}\boldsymbol{\rho} dl + \sum_{k=1}^{N}\int_0^{mq} \Big\{\frac{\partial H_m}{\partial \tilde{x}_k}\Delta\tilde{x}_k(l) + \frac{d\lambda_k^m(l)}{dl}\Delta\tilde{x}_k(l)\Big\} dl.\end{aligned} \tag{5.43}$$

Since $\tilde{x}_k(0) = 0$, $k = 1, \ldots, N$, it follows that $\lambda_k^m(0)\Delta\tilde{x}_k(0) = 0$, $k = 1, \ldots, N$. Substituting (5.33)-(5.34) into (5.43) yields

$$\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v}) = \int_0^{mq} \frac{\partial H_m}{\partial \boldsymbol{\sigma}}\boldsymbol{\rho} dl.$$

Since $\boldsymbol{\rho}$ is chosen arbitrary, using the definition of $\boldsymbol{\sigma}$, the result follows directly. □

Theorem 5.3. *For each m, $m = 1, \ldots, N$, the partial derivatives of Φ_m with respect to $\boldsymbol{\theta}$ is*

$$\frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v})}{\partial\theta_i} = \begin{cases} \int_{i-1}^{i} \frac{\partial H_m}{\partial\theta_i} \mathrm{d}s, & \text{if } i \le mq, \\ 0, & \text{if } i > mq. \end{cases} \tag{5.44}$$

Proof. Let the time scaling parameter vector $\boldsymbol{\theta}$ be perturbed by $\epsilon\boldsymbol{\rho}$, where $\epsilon > 0$ is a small

real number and $\boldsymbol{\rho}$ is an arbitrary fixed perturbation of $\boldsymbol{\theta}$. Then, we have

$$\boldsymbol{\theta}(\epsilon) = \boldsymbol{\theta} + \epsilon\boldsymbol{\rho}.$$

where $\boldsymbol{\rho} = [\rho_1, \ldots, \rho_{qN}]^\top$, and $\boldsymbol{\theta}(\epsilon) = [\theta_1(\epsilon), \ldots, \theta_{qN}(\epsilon)]^\top$. Consequently, the system state $\tilde{\boldsymbol{x}}$ as well as the function Φ_m will be perturbed. Let

$$\tilde{x}_k(s, \epsilon) = \tilde{x}_k(s|\boldsymbol{\theta}(\epsilon)), \quad k = 1, \ldots, N.$$

Clearly,

$$\tilde{x}_k(s, \epsilon) = \tilde{x}_k(0) + \sum_{i=1}^{qN} \int_0^s \theta_i(\epsilon) f_k(t(l), \tilde{x}_k(l), \boldsymbol{\sigma}^i)(l) dl.$$

Let $f_k(t(\cdot), \tilde{x}_k(\cdot), \boldsymbol{\sigma}^i)(\cdot)$ be written as $f_k(\cdot)$ for brevity. Then, by the chain rule, we have

$$\Delta\tilde{x}_k(s) = \frac{d\tilde{x}_k(s, \epsilon)}{d\epsilon}\Big|_{\epsilon=0} = \sum_{i=1}^{qN} \int_0^s \Big\{\rho_i f_k(l) + \theta_i \frac{\partial f_k(l)}{\partial \tilde{x}_k} \Delta\tilde{x}_k(l) + \theta_i \frac{\partial f_k(l)}{\partial t} \Delta t(l)\Big\} dl.$$

Clearly,

$$\frac{d\Delta\tilde{x}_k(s)}{ds} = \sum_{i=1}^{qN} \Big\{\rho_i f_k(s) + \theta_i \frac{\partial f_k(s)}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) + \theta_i \frac{\partial f_k(s)}{\partial t} \Delta t(s)\Big\}. \tag{5.45}$$

By (5.32), we have

$$\begin{aligned}
\Delta H_m &= \frac{\partial H_m}{\partial \boldsymbol{\theta}} \boldsymbol{\rho} + \sum_{k=1}^{N} \frac{\partial H_m}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) + \frac{\partial H_m}{\partial t} \Delta t(s) \\
&= \sum_{k=1}^{N} \lambda_k^m(s) \Big\{\boldsymbol{\rho} f_k(s) + \omega(s) \frac{\partial f_k(s)}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) + \omega(s) \frac{\partial f_k(s)}{\partial t} \Delta t(s)\Big\} + \lambda_{N+1}^m(s) \frac{d\omega(s)}{ds}.
\end{aligned}$$

Using the definition of $\omega(s)$ in (5.20), it is clear that

$$\begin{aligned}
&\frac{\partial H_m}{\partial \boldsymbol{\theta}} \boldsymbol{\rho} + \sum_{k=1}^{N} \frac{\partial H_m}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) + \frac{\partial H_m}{\partial t} \Delta t(s) \\
&\quad = \sum_{i=1}^{qN} \sum_{k=1}^{N} \lambda_k^m(s) \Big\{\boldsymbol{\rho}_i f_k(s) + \theta_i \frac{\partial f_k(s)}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) + \theta_i \frac{\partial f_k(s)}{\partial t} \Delta t(s)\Big\} + \lambda_{N+1}^m(s) \frac{d\omega(s)}{ds}.
\end{aligned} \tag{5.46}$$

Substituting (5.45) into (5.46) yields

$$\frac{\partial H_m}{\partial \boldsymbol{\theta}} \boldsymbol{\rho} + \frac{\partial H_m}{\partial t} \Delta t(s) - \lambda_{N+1}^m(s) \frac{d\omega(s)}{ds} + \sum_{k=1}^{N} \frac{\partial H_m}{\partial \tilde{x}_k} \Delta\tilde{x}_k(s) - \sum_{k=1}^{N} \lambda_k^m(s) \frac{d\Delta\tilde{x}_k(s)}{ds} = 0. \tag{5.47}$$

Then, by the chain rule, we obtain

$$\begin{aligned}\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v}) &= \frac{d\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{d\epsilon}\Big|_{\epsilon=0}\\ &= \frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial t}\Delta t(mq) + \frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial\tilde{x}_k}\Delta\tilde{x}_k(mq).\end{aligned} \tag{5.48}$$

Adding (5.47) to the right hand side of (5.48), we obtain

$$\begin{aligned}\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v}) = &\frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial t}\Delta t(mq) + \sum_{k=1}^{N}\frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial\tilde{x}_k}\Delta\tilde{x}_k(mq)\\ &+\int_0^{mq}\frac{\partial H_m}{\partial\boldsymbol{\theta}}\boldsymbol{\rho}dl + \int_0^{mq}\frac{\partial H_m}{\partial t}\Delta t(l)dl - \int_0^{mq}\lambda_{N+1}^m(l)\frac{d\omega(l)}{dl}dl\\ &+\sum_{k=1}^{N}\int_0^{mq}\Big\{\frac{\partial H_m}{\partial\tilde{x}_k}\Delta\tilde{x}_k(l) - \lambda_k^m(l)\frac{d\Delta\tilde{x}_k(l)}{dl}\Big\}dl\end{aligned} \tag{5.49}$$

Integrating the last term of (5.49) by parts gives

$$\begin{aligned}\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v}) = &\frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial t}\Delta t(mq) + \sum_{k=1}^{N}\frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial\tilde{x}_k}\Delta\tilde{x}_k(mq)\\ &+\int_0^{mq}\frac{\partial H_m}{\partial\boldsymbol{\theta}}\boldsymbol{\rho}dl + \int_0^{mq}\frac{\partial H_m}{\partial t}\Delta t(l)dl - \int_0^{mq}\lambda_{N+1}^m(l)\frac{d\omega(l)}{dl}dl\\ &-\sum_{k=1}^{N}\lambda_k^m(l)\Delta\tilde{x}_k(l)\Big|_0^{mq} + \sum_{k=1}^{N}\int_0^{mq}\Big\{\frac{\partial H_m}{\partial\tilde{x}_k} - \frac{d\lambda_k^m(l)}{dl}\Big\}\Delta\tilde{x}_k(l)dl\end{aligned} \tag{5.50}$$

Since $\tilde{x}_k(0) = 0$, $k = 1,\ldots,N$, it follows that $\lambda_k^m(0)\Delta\tilde{x}_k(0) = 0$, $k = 1,\ldots,N$. Substituting (5.33)-(5.34) into (5.50) yields

$$\begin{aligned}\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v}) = &\frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial t}\Delta t(mq) + \int_0^{mq}\frac{\partial H_m}{\partial\boldsymbol{\theta}}\boldsymbol{\rho}dl\\ &+\int_0^{mq}\Big\{\frac{\partial H_m}{\partial t}\Delta t(l) - \lambda_{N+1}^m(s)\frac{d\omega(l)}{dl}\Big\}dl\end{aligned} \tag{5.51}$$

Integrating the last term of (5.51) by parts gives

$$\begin{aligned}\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v}) = &\frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial t}\Delta t(mq) - \lambda_{N+1}^m(s)\omega(l)\Big|_0^{mq} + \int_0^{mq}\frac{\partial H_m}{\partial\boldsymbol{\theta}}\boldsymbol{\rho}dl\\ &+\int_0^{mq}\Big\{\frac{\partial H_m}{\partial t}\Delta t(l) + \frac{d\lambda_{N+1}^m(s)}{ds}\omega(l)\Big\}dl\end{aligned} \tag{5.52}$$

By (5.19), clearly, $\omega(l) = \Delta t(l)$. By (5.31), we have

$$\frac{\partial\Phi_m(\tilde{\boldsymbol{x}}(mq),\boldsymbol{v})}{\partial t} = 0. \tag{5.53}$$

Substituting (5.35)-(5.36) and (5.53) into (5.52) yields

$$\Delta\Phi_m(\tilde{\boldsymbol{x}}(mq), \boldsymbol{v}) = \int_0^{mq} \frac{\partial H_m}{\partial \boldsymbol{\theta}} \boldsymbol{\rho} dl.$$

Since $\boldsymbol{\rho}$ is chosen arbitrary, using the definition of $\boldsymbol{\theta}$, the result follows directly. □

The gradient formula given above for the penalty and constraint functions can be combined with a standard optimization algorithm—for example, a conjugate gradient method or sequential quadratic programming [48]—to solve Problem (P3) as a nonlinear programming problem. The optimal control software MISER does this automatically. It can be shown that a local solution of Problem (P3) converges to a local solution of Problem (P2) as the penalty parameter becomes sufficiently large [41, 42]. Hence, we can obtain an approximate solution of Problem (P2) by solving Problem (P3) for large η. A corresponding suboptimal control for Problem (P) can then be constructed according to (5.17). In the next section, we use this approach to solve two examples.

5.5 Numerical examples

5.5.1 Example 5.1

In [90], a linear gradient elution chromatographic process for separating four protein solutes is described. The retention time for each protein solute is controlled by adjusting the ionic strength of the mobile phase composition. The rate of change of each chromatography signal is given by the following system functions:

$$\begin{aligned}
f_1(u(t)) &= \frac{(u(t))^2}{1.4(u(t))^2 + 3.6}, \\
f_2(u(t)) &= \frac{u(t)}{1.4u(t) + 5.4}, \\
f_3(u(t)) &= \frac{u(t)}{1.4u(t) + 7.2}, \\
f_4(u(t)) &= \frac{(u(t))^2}{1.4(u(t))^2 + 9},
\end{aligned}$$

where the control is subject to the bound constraints $0.5 \leq u(t) \leq 2$. A chromatography signal of 1 indicates that the protein solute has left the chromatography column. Thus, the state variables must satisfy the following interior point constraints at the retention times:

$$x_k(\tau_k) = 1, \quad k = 1, 2, 3, 4. \tag{5.54}$$

We consider the problem of maximizing the objective function (5.5) with $N = 4$ subject

to the interior point constraints (5.54). This problem is in the form of Problem (P). We use the method outlined in Sections 5.3 and 5.4 with $q = 1$ to approximate Problem (P) by Problem (P3). Problem (P3), the exact penalty function problem, can be solved by using the software package MISER 3. The parameters in the penalty function for Problem (P3) are set as follows:

$$\alpha = 1.5; \quad \beta = 2.2; \quad \gamma = 3; \quad \kappa_1 = \kappa_2 = \kappa_3 = 0.3.$$

We start by solving Problem (P3) using MISER 3 with $\eta = 10$. Then we increase η and re-solve Problem (P3), using the previous solution as the initial guess. We continue to increase η until $\eta = 30$, at which point the objective function value is 0.8205 with $\varepsilon = 0.0024$. The objective function value is slightly better than the result reported in [90]. The largest violation of the interior point constraints (5.27) is 6×10^{-4}, which is smaller than the violation of 1.07×10^{-3} reported in [90].

We next consider Problem (P3) with $q = 3$. Then, the time horizon is divided into 12 sub-intervals. We again use MISER 3 to solve Problem (P3). The optimal trajectories and optimal control computed by MISER 3 are shown in Figure 5.3. The optimal control values and the corresponding time-scaling transformation parameters are given in Tables 5.1 and 5.2, respectively. The optimal objective function value is 0.8276 when $\varepsilon = 0.001$, $\eta = 300$, and $\boldsymbol{\tau} = [2.7507, 5.9795, 9.1807, 12.3820]$. As expected, the optimal value for $q = 3$ is higher than the optimal value for $q = 1$.

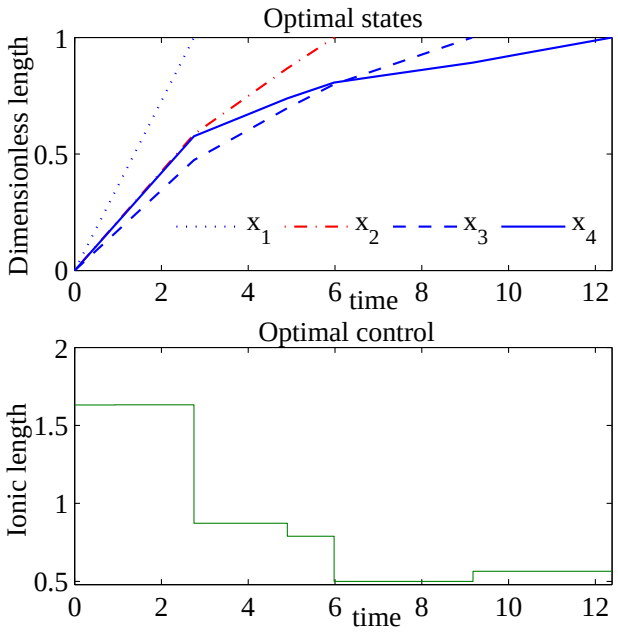

Figure 5.3: Optimal controls and states for Example 5.1.

The equality and inequality constraints (5.7)-(5.10), (5.21), (5.23), and 5.28 in Problem (P2) are all satisfied. Compared with the results obtained in [90], our optimal ob-

Table 5.1: Optimal control values for Example 5.1.

σ_1	σ_2	σ_3	σ_4	σ_5	σ_6
1.6314	1.6324	1.6326	0.8729	0.8730	0.7901
σ_7	σ_8	σ_9	σ_{10}	σ_{11}	σ_{12}
0.5000	0.5000	0.5000	0.5644	0.5645	0.5645

Table 5.2: Optimal interval durations for Example 5.1.

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
0.9347	0.9153	0.9007	1.0740	1.0777	1.0770
σ_7	σ_8	σ_9	σ_{10}	σ_{11}	σ_{12}
1.0801	1.0608	1.0603	1.0541	1.0728	1.0743

jective function value is better. Moreover, the largest violation of the interior point constraints (5.27) is 8×10^{-4} which is slightly smaller than the violation achieved in [90].

5.5.2 Example 5.2

We now apply our optimal control method to a more complicated system with multiple inputs. Consider the ion-exchange chromatographic process for separating three protein species described in [112]. The peak time for each protein solute is controlled by adjusting the ionic strength (I) and the pH value (z) of the mobile phase composition. The rate of change of each chromatography signal is given by the following system functions:

$$f_k(I(t), z(t)) \;= \left[1 + \tfrac{(1-\varepsilon_b)\varepsilon_m}{\varepsilon_b} + \tfrac{(1-\varepsilon_b)(1-\varepsilon_m)}{\varepsilon_b} K_k(I(t), z(t))\right]^{-1}, \quad k = 1, 2, 3$$

where $\varepsilon_b = 0.25$, $\varepsilon_m = 0.8$, and the affinity distribution coefficients are given by

$$\begin{aligned} K_1(I(t), z(t)) &= 0.49 \times 10^6 (0.1867 I^{-1})^{-18.5+9.9z-0.77z^2}, \\ K_2(I(t), z(t)) &= 5.48 \times 10^6 (0.1867 I^{-1})^{-26.9+13.8z-1.15z^2}, \\ K_3(I(t), z(t)) &= 2.21 \times 10^6 (0.1867 I^{-1})^{-24.7+12.9z-1.1z^2}. \end{aligned}$$

The controls are bounded by

$$0.405 \leq I(t) \leq \; 0.495, \quad 4.8 \leq z(t) \leq 5.2.$$

A protein species is considered to be separated when its corresponding chromatography signal reaches 1. That is, the state variables must satisfy the following interior point constraints:

$$x_k(\tau_k) = 1, \quad k = 1, 2, 3.$$

Our aim is to solve Problem (P), which involves maximizing the objective function (5.5) for the system with $N = 3$ subject to the interior point constraints given above. Using the process outlined in Section 5.3 and Section 5.4 with $q = 3$, we approximate Problem (P) by Problem (P3). The parameters for the exact penalty function are the same as in Example 5.1. Using MISER 3 to solve Problem (P3), we obtain the optimal

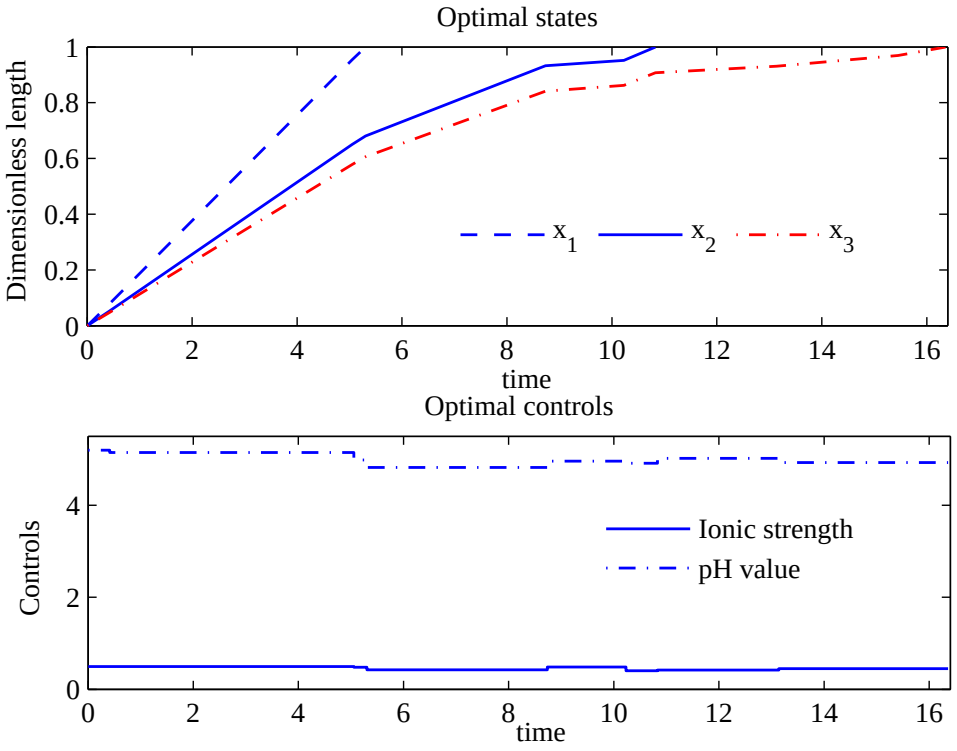

Figure 5.4: Optimal results for Example 5.2

trajectories and optimal controls shown in Figure 5.4. The optimal objective function value is 1.8673, when $\eta = 200$, $\varepsilon = 2.24895 \times 10^{-4}$, and $\boldsymbol{\tau} = [5.3029, 10.8293, 16.3557]$. The largest violation of the interior point constraints (5.27) is 1×10^{-5}. All of the other constraints in Problem (P2) are satisfied.

5.6 Conclusions

In this chapter, we have presented a new computational method for solving a challenging max-min optimal control problem arising in gradient elution chromatography. The state

variables in this problem represent the chromatography signals of the mixture components, and each signal is required to reach its desired target at a different retention time. Our method is based on an approximation scheme whereby the control is approximated by a piecewise-constant function. We use a novel time-scaling transformation to map the retention times to fixed points, and another transformation to convert the max-min objective function into a smooth function. An exact penalty method is then used to solve the resulting approximate optimization problem. The numerical simulations in Section 5.5 show that our new method gives better results than those obtained by using the previous method proposed in [90]. More importantly, our new method is capable of accurately computing the retention times, and it can also be applied to more general problems of larger dimension.

CHAPTER 6

Summary and future research directions

6.1 Main contributions of the thesis

In this thesis, we considered two parameter identification problems and two nonstandard optimal control problems. We developed new methods for solving these problems as nonlinear programming problems, which are based on gradient-based optimization algorithms. This involved combining a variety of novel techniques, including the time-scaling transformation, constraint transcription, control parameterization, as well as complicated derivation of gradient formulas and detailed convergence analysis. We summarize our main contributions below.

In Chapter 2, we considered the problem of identifying unknown delays and unknown system parameters in a general nonlinear delay-differential system with smooth inputs. This problem was formulated as an optimization problem in which the decision variables are the time-delays and the system parameters. We showed that the gradient of the cost function in this optimization problem can be computed by solving a set of auxiliary delay-differential systems. On this basis, the optimization problem can be solved as a nonlinear programming problem using a gradient-based optimization algorithms. Most other methods for delay identification are only applicable to systems containing only a single delay, or to linear systems. The novelties of our approach are: (i) It is applicable to general nonlinear delay-differential systems; and (ii) it makes use of efficient nonlinear programming techniques, and hence it has excellent potential for real-time implementation.

In Chapter 3, we considered a general nonlinear time-delay system in which the delay is involved in a piecewise constant input. We formulated the problem of identifying the unknown state delay and control delay as a nonlinear optimization problem in which the cost function measures the least-squares error between predicted output and observed system output. Since the input function is discontinuous, the dynamics are clearly discontinuous with respect to β. Thus, the gradient of the cost function with respect to input delay does not exist at those time points which lie in the set of the control switching time points. We then showed that, except at those special time points, the gradient of the cost function with respect to the input delay can be computed by integrating

an auxiliary impulsive system with instantaneous jumps forward in time. To solve the identification problem, we propose a heuristic strategy which can be combined with our gradient computation procedure. Then, a computation method is developed based on any standard nonlinear programming algorithm. Two industrial examples are solved by using the proposed method. The results obtained show that this approach is highly effective.

In Chapter 4, we developed a computational method for solving optimal control problems with multiple time-delay systems and subject to continuous state inequality constraints. In this method, the control is approximated by a piecewise constant function whose heights are decision variables to be determined optimally. The approximate control is allowed to change its value at $N-1$ switching times in the time horizon. By novel application of the transformation method used in conjunction with a local smoothing technique, the continuous state inequality constraints are approximated by smooth canonical constraints, which are then appended to the cost function forming an augmented cost function. On this basis, the optimal control problem is approximated by a sequence of optimal parameter selection problems involving time-delay systems and subject to bound constraints on the control vector. Furthermore, we proved under some mild assumptions that the cost of the optimal control vector of the approximate problem converges to the optimal cost of the original problem. Finally, a computational algorithm is developed to solve each of the optimal parameter selection problems. This algorithm involves integrating the time-delay system forward in time and a set of time-delay costate systems backward in time. Then, the gradient of the augmented cost function is calculated. This approach is then applied to solve a control problem arising in a practical evaporation process. The results obtained are highly satisfactory.

In Chapter 5, we considered the problem of determining an optimal operating schedule for a chromatography process. The optimal control problem is formulated as a max-min optimal control problem with multiple characteristic-time equality constraints, which is similar to the one considered in [90]—it involves choosing the retention time for each component so that the minimum time interval between each two successive components is maximized. However, the problem considered in [90] assumes that the order of the retention times is known and fixed, whereas we considered a general problem where the ordering of the retention times is not fixed, but instead needs to be determined optimally. We then developed a new computational method for solving this max-min optimal control problem. The main idea of this method is to approximate the control by a piecewise constant function whose values and switching times are decision variables to be determined optimally. The approximate control is allowed to change its value at each switching time, and also at $N-1$ times between consecutive characteristic times (N is a fixed integer). Then a new time scaling transformation method is used to map these switching times to fixed points in a new time horizon. Note that the method used in [90] only maps the terminal time to a fixed point, but the intermediate retention times remain

as variables which change with the control. Thus, in the numerical computation, it is very difficult to determine accurately the values for these retention times. In our new method, we introduce a superior time-scaling transformation that maps both the control switching times and the retention times to fixed points. The retention times and the control switching times in the original time horizon become optimization variables in the new time horizon. This allows us to simultaneously determine accurate values for the retention times and the optimal control switching times. In Chapter 5, we use a set of auxiliary decision variables to explicitly keep track of the retention time ordering which involves approximating the integer constraints by a set of linear constraints and continuous quadratic constraints. The final approximate problem is solved by a recently developed exact penalty function method.

6.2 Future research directions

The work in this thesis has opened several interesting new areas for future research. We discuss some of them below.

The computational methods developed in Chapter 2 and Chapter 3 are only applicable to the delay system with time-invariant (constant) delays. In some practical applications, the delays are time varying. For example, the chemical reactor recycle system [20], cooling system [2], and the anesthesia control during intensive care [27]. For these systems with time varying delays, the methods developed in Chapters 2 and 3 are not applicable. Thus, considerable effort is needed to extend these methods to delay systems with time varying delays. It is a mathematically challenging and practically significant task to do the parameter identification on-line for systems with uncertainties, where multiple time delays and system parameters are to be identified.

The computational method developed in Chapter 4 involves integrating the time-delay system forward in time and integrating the corresponding time-delay co-state system backward in time. The computational burden is heavy. For example, it takes more than 3 hours to obtain an optimal control for eight hours of simulation time for the optimal control of the evaporation process considered in Section 4.4. In practice, the computation time is required to be light. Since many real practical processes involve complex dynamics, an interesting future research direction is to develop computational methods for which the computational load is light so that they are suitable for online implementation.

In the field of chromatography, the simulated moving bed (SMB) chromatography is becoming more and more popular. The model of the dynamic system for SMB chromatography is usually described by partial differential equations with position and time as the independent variables. Clearly, our method is not directly applicable to SMB chromatography. Thus, it presents an interesting and challenging task to extend our approach to SMB chromatography.

Bibliography

[1] A. Ghasemi, "Application of linear model predictive control and input-output linearization to constrained control of 3D cable robots," *Modern Mechanical Engineering*, vol. 1, no. 2, pp. 69–76, 2011.

[2] A. H. Tan and C. L. Cham, "Continuous-time model identification of a cooling system with variable delay," *IET Control Theory and Applications*, vol. 5, no. 7, pp. 913–922, 2011.

[3] A. Jamali, A. Hajiloo, and N. Nariman-zadeh, "Reliability-based robust pareto design of linear state feedback controllers using a multi-objective uniform-diversity genetic algorithm (muga)," *Expert Systems with Applications*, vol. 37, no. 1, pp. 401–413, 2010.

[4] A. M. Cramer, S. D. Sudhoff, and E. L. Zivi, "Evolutionary algorithms for minimax problems in robust design," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 444–453, 2009.

[5] A. Miele, "Recent advances in gradient algorithms for optimal control problems," *Journal of Optimization Theory and Applications*, vol. 17, no. 5, pp. 361–430, 1975.

[6] A. Miele and R. E. Pritchard, "Gradient methods in control theory, part 2, sequential gradient-restoration algorithm," Rice University, Aero-Astronautics Report 62, 1969.

[7] A. Miele and T. Wang, "Primal-dual properties of sequential gradient-restoration algorithms for optimal control problems 2. general problem," *Journal of Mathematical Analysis and Applications*, vol. 119, no. 1-2, pp. 21–54, 1986.

[8] A. Miele, T. Wang, C. S. Chao, and J. B. Dabney, "Optimal control of a ship for collision avoidance maneuvers," *Journal of optimization theory and applications*, vol. 103, no. 3, pp. 495–519, 1999.

[9] A. Pot, S. Bhulai, and G. Koole, "A simple staffing method for multiskill call centers," *Manufacturing and Service Operations Management*, vol. 10, no. 3, pp. 421–428, 2008.

[10] A. Sbihi, "A cooperative local search-based algorithm for the multiple-scenario max-min knapsack problem," *European Journal of Operational Research*, vol. 202, no. 2, pp. 339–346, 2010.

[11] A. Toumi, F. Hanisch, and S. Engell, "Optimal operation of continuous chromatographic processes: mathematical optimization of the VARICOL process," *Industrial and Engineering Chemistry Research*, vol. 41, pp. 4328–4337, 2002.

[12] A. Woinaroschy, "Time-optimal control of startup distillation columns by iterative dynamic programming," *Industrial and Engineering Chemistry Research*, vol. 47, no. 12, pp. 4158–4169, 2008.

[13] A. Woinaroschy and R. Isopescu, "Time-optimal control of dividing-wall distillation columns," *Industrial and Engineering Chemistry Research*, vol. 49, pp. 9195–9208, 2010.

[14] B. Farhadinia, K. L. Teo, and R. C. Loxton, "A computational method for a class of non-standard time optimal control problems involving multiple time horizons," *Mathematical and Computer Modelling*, vol. 49, pp. 1682–1691, 2009.

[15] B. Gu and P. G. Yash, "Control of nonlinear processes by using linear model predictive control algorithms," *ISA Transactions*, vol. 47, no. 2, pp. 211–216, 2008.

[16] B. Li , K. L. Teo, G. H. Zhao, and G. R. Duan, "An efficient computational approach to a class of minmax optimal control problems with applications," *The ANZIAM Journal*, vol. 51, pp. 162–177, 2009.

[17] B. Pluymers, L. Roobrouck, J. Buijs, J. A. K. Suykens, and M. B. De, "Constrained linear MPC with time-varying terminal cost using convex combinations," *Automatica*, vol. 41, no. 5, pp. 831–837, 2005.

[18] B. Rustem, R. G. Becker, and W. Marty, "Robust min-max portfolio strategies for rival forecast and risk scenarios," *Journal of Economic Dynamics and Control*, vol. 24, no. 112, pp. 1591–1621, 2000.

[19] C. Buskens, "Optimierungs methoden und sensitivitatsanalyse fur optimale steuerprozesse mit steuer-und zustands-beschrankungen," Ph.D. dissertation, Institut fur Numerische und instrumentelle Mathematik, Universitat Munster, Germany, 1998.

[20] C. C. Hua, J. Leng, and X. P. Guan, "Decentralized mrac for large-scale interconnected systems with time-varying delays and applications to chemical reactor systems," *Journal of Process Control*, 2012.

[21] C. Charalambous and A. R. Conn, "An efficient method to solve the minimax problem directly," *SIAM Journal on Numerical Analysis*, vol. 15, no. 1, pp. 162–187, 1978.

[22] C. F. Wen, "Continuous-time generalized fractional programming problems. part i: basic theory," *Journal of Optimization Theory and Applications*, pp. 1–35, 2012.

[23] C. Feller, T. A. Johansen, and S. Olaru, "An improved algorithm for combinatorial multi-parametric quadratic programming," *Automatica*, 2012.

[24] C. Jiang, K. L. Teo, R. Loxton, and G. R. Duan, "A neighboring extremal solution for an optimal switched impulsive control problem," *Journal of Industrial and Management Optimization*, vol. 8, no. 3, pp. 591–609, 2012.

[25] C. Lin, Z. Wang, and F. Yang, "Observer-based networked control for continuous-time systems with random sensor delays," *Automatica*, vol. 45, no. 2, pp. 578–584, 2009.

[26] C. Liu, Z. Gong, E. Feng, and H. Yin, "Modelling and optimal control for nonlinear multistage dynamical system of microbial fed-batch culture," *Journal of Industrial and Management Optimization*, vol. 5, no. 4, pp. 835–850, 2009.

[27] C. M. Ionescu, R. Hodrea, and R. De Keyser, "Variable time-delay estimation for anesthesia control during intensive care," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 2, pp. 363–369, 2011.

[28] C. Pignotti, "A note on stabilization of locally damped wave equations with time delay," *Systems and Control Letters*, vol. 61, no. 1, pp. 92–97, 2012.

[29] C. R. Hargraves and S. W. Paris, "Direct trajectory optimization using nonlinear programming and collocation," in *Astrodynamics Conference*, vol. 3-12, Williamsburg, VA, August 18-20 1986.

[30] ——, "Direct trajectory optimization using nonlinear programming and collocation," *AIAA J. Guidance*, vol. 10, pp. 338–342, 1987.

[31] C. R. Porfrio and D. Odloak, "Optimizing model predictive control of an industrial distillation column," *Control Engineering Practice*, vol. 19, no. 10, pp. 1137–1146, 2011.

[32] C. Sonntag, W. J. Su, O. Stursberg, and S. Engell, "Optimized start-up control of an industrial-scale evaporation system with hybrid dynamics," *Control Engineering Practice*, vol. 16, no. 8, pp. 976–990, 2008.

[33] C. Wu, K. L. Teo, R. Li, and Y. Zhao, "Optimal control of switched systems with time delay," *Applied mathematics letters*, vol. 19, no. 10, pp. 1062–1067, 2006.

[34] C. Y. Kaya, "The leap-frog algorithm and optimal control: background and demonstration," in *Proceedings of ICOTA 98, Perth, Australia*, 1998.

[35] C. Y. Kaya and J. L. Noakes, "Computations and time-optimal controls," *Optimal Control Applications and Methods*, vol. 17, no. 3, pp. 171–185, 1996.

[36] ——, "Geodesics and an optimal control algorithm," in *Proceedings of the 36th IEEE Conference on Decision and Control*, vol. 5, 1997, pp. 4918–4919.

[37] ——, "Leapfrog for optimal control," *SIAM Journal on Numerical Analysis*, vol. 46, no. 6, pp. 2795–2817, 2008.

[38] C. Y. Kaya and J. M. Martínez, "Euler discretization and inexact restoration for optimal control," *Journal of Optimization Theory and Applications*, vol. 134, pp. 191–206, 2007.

[39] C. Y. Kaya, S. K. Lucas, and S. T. Simakov, "Computations for bangbang constrained optimal control using a mathematical programming formulation," *Optimal Control Applications and Methods*, vol. 25, no. 6, pp. 295–308, 2004.

[40] C. Yu, K. L. Teo, and Y. Bai, "An exact penalty function method for nonlinear mixed discrete programming problems," *Optimization Letters*, 2011.

[41] C. Yu, K. L. Teo, L. Zhang, and Y. Bai, "A new exact penalty function method for continuous inequality constrained optimization problems," *Journal of Industrial and Management Optimization*, vol. 6, pp. 895–910, 2010.

[42] C.Yu, B. Li, R. Loxton, and K. L. Teo, "Optimal discrete-valued control computation," *Journal of Global Optimization*, pp. 1–16, 2012.

[43] D. Cao, M. Chen, H. Wang, and J. Ji, "Interval method for global solutions of a class of min-max problems," *Applied Mathematics and Computation*, vol. 196, no. 2, pp. 594–602, 2008.

[44] D. D. Morrison, J. D. Riley, and J. F. Zancanaro, "Multiple shooting method for two-point boundary value problems," *Communications of the ACM*, vol. 5, no. 12, pp. 613–614, 1962.

[45] D. DeHaan and M. Guay, "A real-time framework for model-predictive control of continuous-time nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 11, pp. 2047–2057, 2007.

[46] D. E. Hartley and W. Murgatroyd, "Criteria for the break-up of thin liquid layers flowing isothermally over solid surfaces," *International Journal of Heat and Mass Transfer*, vol. 7, no. 9, pp. 1003–1015, 1964.

[47] D. Eller, J. Aggarwal, and H. Banks, "Optimal control of linear time-delay systems," *IEEE Transactions on Automatic Control,*, vol. 14, no. 6, pp. 678–687, 1969.

[48] D. G. Luenberger and Y. Ye, *Linear and nonlinear programming*, 3rd ed. Springer: New York, 2008.

[49] D. Han, X. Yuan, W. Zhang, and X. Cai, "An adm-based splitting method for separable convex programming," *Computational Optimization and Applications*, vol. 2, no. 54, pp. 343–369, 2013.

[50] E. B. Lee and L. Markus, *Foundation of optimal control theory.* New York, John Wiley and Sons Inc., 1967.

[51] E. Fridman, "Stability of linear descriptor systems with delay: a lyapunov-based approach," *Journal of Mathematical Analysis and Applications*, vol. 273, no. 1, pp. 24–44, 2002.

[52] E. Obasanjo, G. R. Tzallas, and B. Rustem, "An interior-point algorithm for nonlinear minimax problems," *Journal of Optimization Theory and Applications*, vol. 144, no. 2, pp. 291–318, 2010.

[53] E. Polak, J. O. Royset, and R. S. Womersley, "Algorithms with adaptive smoothing for finite minimax problems," *Journal of Optimization Theory and Applications*, vol. 119, no. 3, pp. 459–484, 2003.

[54] F. E. Curtis and M. L. Overton, "A sequential quadratic programming algorithm for nonconvex, nonsmooth constrained optimization," *SIAM Journal on Optimization*, vol. 2, no. 22, pp. 474–500, 2012.

[55] F. E. Haoussi, E. H. Tissir, F. Tadeo, and A. Hmamed, "Delay-dependent stabilisation of systems with time-delayed state and control: application to a quadruple-tank process," *International Journal of Systems Science*, vol. 42, no. 1, pp. 41–49, 2011.

[56] F. Fahroo and I. M. Ross, "Direct trajectory optimization by a chebyshev pseudospectral method," in *Proceedings of the 2000 American Control Conference*, vol. 6, 2000, pp. 3860–3864.

[57] F. Pan, R. C. Han, and D. M. Feng, "An identification method of time-varying delay based on genetic algorithm," *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics*, vol. 2, pp. 781–783, Nov. 2003.

[58] F. S. Mjalli and M. A. Hussain, "Approximate predictive versus self-tuning adaptive control strategies of biodiesel reactors," *Industrial and Engineering Chemistry Research*, vol. 48, no. 24, pp. 11 034–11 047, 2009.

[59] G. A. Fraser, "A multiple-shooting technique for optimal control," *Journal of Optimization Theory and Applications*, vol. 102, no. 2, pp. 299–313, 1999.

[60] G. Fernholz, S. Engell, L. U. Kreul, and A. Gorak, "Optimal operation of a semi-batch reactive distillation column," *Computers and Chemical Engineering*, vol. 24, no. 2-7, pp. 1569–1575, 2000.

[61] G. P. Rangaiah, P. Saha, and M. O. Tad, "Nonlinear model predictive control of an industrial four-stage evaporator system via simulation," *Chemical Engineering Journal*, vol. 87, no. 3, pp. 285–299, 2002.

[62] H. Chung, E. Polak, and S. Sastry, "An external active-set strategy for solving optimal control problems," *IEEE Transactions on Automatic Control*, vol. 54, no. 5, pp. 1129–1133, 2009.

[63] H. Gao, T. Chen, and J. Lam, "A new delay system approach to network-based control," *Automatica*, vol. 44, no. 1, pp. 39–52, 2008.

[64] H. Kubin, M. Benesch, A. Dementjev, K. Kabitzsch, T. Unkelbach, F. H. Rogner, and C. Metzner, "Identification of process models and controller design for vacuum coating processes with a long dead time using an identification tool with advisory support," in *IEEE Conference on Emerging Technologies Factory Automation*, Sept. 2009, pp. 1–8.

[65] H. W. J. Lee, K. L. Teo, and X. Q. Cai, "An optimal control approach to nonlinear mixed integer programming problems," *Computers and Mathematics with Applications*, vol. 36, pp. 87–105, 1998.

[66] H. W. J. Lee, K. L. Teo, L. S. Jennings, and V. Rehbock, "Control parametrization enhancing technique for time optimal control problem," *Dynamical Systems and Applications*, vol. 6, no. , pp. 243–261, 1997.

[67] ——, "Control parametrization enhancing technique for time optimal control problems," *Dynamical Systems and Applications*, vol. 6, pp. 243–261, 1997.

[68] I. Bonis, W. Xie, and C. Theodoropoulos, "A linear model predictive control algorithm for nonlinear large-scale distributed parameter systems," *AIChE Journal*, vol. 58, no. 3, pp. 801–811, 2012.

[69] I. Malloci, J. Daafouz, and C. Iung, "Stability and stabilization of two time scale switched systems in discrete time," *IEEE Transactions on Automatic Control*, vol. 55, no. 6, pp. 1434–1438, 2010.

[70] I. V. Kolmanovsky and T. L. Maizenberg, "Optimal control of continuous-time linear systems with a time-varying, random delay," *Systems and Control Letters*, vol. 44, no. 2, pp. 119–126, 2001.

[71] J. A. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *IEEE Transactions on Information Theory*, vol. 3, no. 52, pp. 1030–1051, 2006.

[72] J. Li and J. Huo, "Inexact smoothing method for large scale minimax optimization," *Applied Mathematics and Computation*, vol. 218, no. 6, pp. 2750–2760, 2011.

[73] J. Nocedal and S.Wright, *Numerical optimization*, 2nd ed. Springer: New York, 2006.

[74] J. P. Richard, "Time-delay systems: an overview of some recent advances and open problems," *Automatica*, vol. 39, no. 10, pp. 1667–1694, 2003.

[75] J. Ruths, A. Zlotnik, and S. Li, "Convergence of a pseudospectral method for optimal control of complex dynamical systems," in *The 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, Dec. 2011, pp. 5553–5558.

[76] J. Zhou and A. L. Tits, "User's guide for FFSQP version 3.7: A Fortran code for solving optimization programs, possibly minimax, with general inequality constraints and linear equality constraints, generating feasible iterates," Institute for Systems Research, University of Maryland, College Park, MD, Tech. Rep. 20742, 1997.

[77] K. Gu, V. L. Kharitonov, and J. Chen, *Stability of Time-Delay Systems*, 3 ed. Berlin, Germany: Birkhauser, 2003.

[78] K. H. Wong, D. J. Clements, and K. L. Teo, "Optimal control computation for nonlinear time-lag systems," *Journal of Optimization Theory and Applications*, vol. 47, pp. 91–107, 1985.

[79] K. H. Wong, L. S. Jennings, and F. Benyah, "The control parametrization enhancing transform for constrained time–delayed optimal control problems," *The ANZIAM Journal*, vol. 43, no. E, pp. 154–185, 2002.

[80] K. L. Teo, C. J. Goh, and K. H. Wong, *A unified computational approach to optimal control problems*. Longman Scientific and Technical: Essex, 1991.

[81] K. L. Teo, L. S. Jennings, H. W. J. Lee, and V. Rehbock, "The control parameterization enhancing transform for constrained optimal control problems," *The Journal of the Australian Mathematical Society. Series B. Applied Mathematics*, vol. 40, no. 3, pp. 314–335, 1999.

[82] K. Teo and C. Goh, "A simple computational procedure for optimization problems with functional inequality constraints," *IEEE Transactions on Automatic Control*, vol. 32, no. 10, pp. 940–941, 1987.

[83] K. Uchida, K. Ikeda, T. Azuma, and A. Kojima, "Finite-dimensional characterizations of h_∞ control for linear systems with delays in input and output," *International Journal of Robust and Nonlinear Control*, vol. 13, no. 9, pp. 833–843, 2003.

[84] K. Watanabe and M. Ito, "A process-model control for linear systems with delay," *IEEE Transactions on Automatic Control*, vol. 26, no. 6, pp. 1261–1269, 1981.

[85] L. Belkoura, J. P. Richard, and M. Fliess, "Parameters estimation of systems with delayed and structured entries," *Automatica*, vol. 45, no. 5, pp. 1117–1125, 2009.

[86] 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*, vol. 30, no. 4, pp. 341–365, 2009.

[87] L. Jin, R. Kumar, and N. Elia, "Model predictive control-based real-time power system protection schemes," *IEEE Transactions on Power Systems*, vol. 25, no. 2, pp. 988–998, 2010.

[88] L. Pronzato and E. Walter, "Robust experiment design via maximin optimization," *Mathematical Biosciences*, vol. 89, no. 2, pp. 161–176, 1988.

[89] L. S. Jennings and K. L. Teo, "A computational algorithm for functional inequality constrained optimization problems," *Automatica*, vol. 26, no. 2, pp. 371–375, 1990.

[90] L. S. Jennings, K. L. Teo, F. Y. Wang, and Q. Yu, "Optimal protein separation," *Computers and Chemical Engineering*, vol. 19, pp. 567–573, 1995.

[91] L. S. Jennings, M. E. Fisher, K. L. Teo, and C. J. Goh, "MISER3: Solving optimal control problems an update," *Advances in Engineering Software and Workstations*, vol. 13, no. 4, pp. 190–196, 1991.

[92] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, *The mathematical theory of optimal processes*. Gordon and Breach Science Publishers: Montreux, 1986.

[93] L. Y. Wang, W. H. Gui, K. L. Teo, R. C. Loxton, and C. H. Yang, "Time delayed optimal control problems with multiple characteristic time points: computation and industrial applications," *Journal of Industrial and Management Optimization*, vol. 5, no. 4, pp. 705–718, 2009.

[94] 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*, vol. 54, no. 2, pp. 307–323, 2012.

[95] 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*, vol. 82, no. 4, p. ID 046212, 2010.

[96] R. Luus, "Application of dynamic programming to differential-algebraic process systems," *Computers and Chemical Engineering*, vol. 17, no. 4, pp. 373–377, 1993.

[97] M. Baric, S. V. Rakovic, T. Besselmann, and M. Morari, "Max-min optimal control of constrained discrete-time systems," in *World Congress*, vol. 17, no. 1, 2008, pp. 8803–8808.

[98] M. G. García, C. E. Balsa, and J. R. Banga, "Dynamic optimization of a simulated moving bed (SMB) chromatographic separation process," *Industrial and Engineering Chemistry Research*, vol. 45, pp. 9033–9041, 2006.

[99] M. K. H. Fan, A. L. Tits, J. L. Zhou, L. S. Wang, and J. Koninckx, *CONSOL-OPTCAD user's manual*. Systems Research Center, University of Maryland, 1992.

[100] M. Kamgarpour and C. Tomlin, "On optimal control of non-autonomous switched systems with a fixed mode sequence," *Automatica*, vol. 48, pp. 1177–1181, 2012.

[101] M. L. A. Marquez and C. H. Moog, "Input-output feedback linearization of time-delay systems," *IEEE Transactions on Automatic Control*, vol. 49, no. 5, pp. 781–785, 2004.

[102] M. L. Kaplan and J. H. Heegaard, "Predictive algorithms for neuromuscular control of human locomotion," *Journal of Biomechanics*, vol. 34, no. 6, pp. 1077–1083, 2001.

[103] M. Liu, Q. G. Wang, B. Huang, and C. C. Hang, "Improved identification of continuous-time delay processes from piecewise step tests," *Journal of Process Control*, vol. 17, no. 1, pp. 51–57, 2007.

[104] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*. New Jersey: John Wiley & Sons, 2006.

[105] M. S. Charalambides, N. Shah, and C. C. Pantelides, "Synthesis of batch reaction/distillation processes using detailed dynamic models," *Computers and Chemical Engineering*, vol. 19, pp. 167–174, 1995.

[106] M. V. Kothare, V. Balakrishnan, and M. Morari, "Robust constrained model predictive control using linear matrix inequalities," *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996.

[107] M. Verstraeten, M. Pursch, P. Eckerle, J. Luong, and G. Desmet, "Modelling the thermal behaviour of the low-thermal mass liquid chromatography system," *Journal of Chromatography A*, vol. 1218, no. 16, pp. 2252–2263, 2011.

[108] N. Lyle, "A global algorithm for geodesics," *Journal of the Australian Mathematical Society*, vol. 65, no. 1, pp. 37–50, 1998.

[109] N. T. Russell, H. H. C. Bakker, and R. L. Chaplin, "A comparison of dynamic models for an evaporation process," *Chemical Engineering Research and Design*, vol. 78, no. 8, pp. 1120–1128, 2000.

[110] N. U. Ahmed, *Elements of finite-dimensional systems and control theory.* Essex: Longman Scientific and Technical, 1988.

[111] ——, *Dynamic Systems and Control with Applications.* Singapore: World Scientific, 2006.

[112] O. Kaltenbrunner, O. Giaverini, D. Woehle, and J. A. Asenjo, "Application of chromatographic theory for process characterization towards validation of an ion-exchange operation," *Biotechnology and Bioengineering*, vol. 98, no. 1, pp. 201–210, 2007.

[113] O. Stryk and R. Bulirsch, "Direct and indirect methods for trajectory optimization," *Annals of Operations Research*, vol. 37, no. 1, pp. 357–373, 1992.

[114] O. V. Stryk, "User's guide for DIRCOL (Version 2.1): A direct collocation method for the numerical solution of optimal control problems," Simulation and Systems Optimization, Technical University of Darmstadt, Tech. Rep., November 1999.

[115] P. Berkmann and H. J. Pesch, "Abort landing in windshear: Optimal control problem with third-order state constraint and varied switching structure," *Journal of Optimization Theory and Applications* , vol. 85, no. 1, pp. 21–57, 1995.

[116] P. J. Gawthrop and M. T. Nihtila, "Identification of time delays using a polynomial identification method," *System and Control Letters*, vol. 5, no. 4, pp. 267–271, 1985.

[117] 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*, vol. 7, pp. 63–81, 2011.

[118] ——, "A new computational method for optimizing nonlinear impulsive systems," *Dynamics of Continuous, Discrete and Impulsive Systems B*, vol. 18, no. 1, pp. 59–76, 2011.

[119] Q. Q. Chai, C. H. Yang, K. L. Teo, and W. H. Gui, "Optimal Control of an Industrial-scale Evaporation Process: Sodium Aluminate Solution," *Control Engineering Practice*, vol. 20, pp. 618–628, 2012.

[120] R. B. Martin, "Optimal control drug scheduling of cancer chemotherapy," *Automatica*, vol. 28, pp. 1113–1123, 1992.

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

[122] R. C. Loxton, K. L. Teo, V. Rehbock, and K. F. C. Yiu, "Optimal control problems with a continuous inequality constraint on the state and the control," *Automatica*, vol. 45, no. 10, pp. 2250–2257, 2009.

[123] R. Datko, "Two examples of ill-posedness with respect to time delays revisited," *IEEE Transactions on Automatic Control*, vol. 42, no. 4, pp. 511–515, 1997.

[124] R. Loxton, K. L. Teo, and V. Rehbock, "Computational method for a class of switched system optimal control problems," *IEEE Transactions on Automatic Control*, vol. 54, no. 10, pp. 2455–2460, 2009.

[125] ——, "An optimization approach to state-delay identification," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2113–2119, 2010.

[126] ——, "Robust suboptimal control of nonlinear systems," *Applied Mathematics and Computation*, vol. 217, no. 14, pp. 6566–6576, 2011.

[127] R. Loxton, Q. Lin, V. Rehbock, and K. L. Teo, "Control parameterization for optimal control problems with continuous inequality constraints: New convergence results," *Numerical Algebra, Control and Optimization*, vol. 2, no. 3, pp. 571–599, 2012.

[128] R. Luus, "Application of dynamic programming to high-dimensional non-linear optimal control problems," *International Journal of Control*, vol. 52, no. 1, pp. 239–250, 1990.

[129] ——, "Piecewise linear continuous optimal control by iterative dynamic programming," *Industrial and Engineering Chemistry Research*, vol. 32, no. 5, pp. 859–865, 1993.

[130] ——, *Iterative dynamic programming.* Chapman and Hall/CRC, London, UK, 2000.

[131] R. Song, H. Zhang, Y. Luo, and Q. Wei, "Optimal control laws for time-delay systems with saturating actuators based on heuristic dynamic programming," *Neurocomputing*, vol. 73, no. 16-18, pp. 3020–3027, 2010.

[132] T. Ruby and V. Rehbock, "Numerical solutions of optimal switching control problems," *Optimization and control with applications*, vol. 96, pp. 447–459, 2005.

[133] S. Dimartino, C. Boi, and G. C. Sarti, "A validated model for the simulation of protein purification through affinity membrane chromatography," *Journal of Chromatography A*, vol. 1218, no. 13, pp. 1677–1690, 2011.

[134] S. Diop, I. Kolmanovsky, P. E. Moraal, and M. V. Nieuwstadt, "Preserving stability/performance when facing an unknown time-delay," *Control Engineering Practice*, vol. 9, no. 12, pp. 1319–1325, 2001.

[135] S. Gonzalez and A. Miele, "Sequential gradient-restoration algorithm for opti- mal control problems with general boundary conditions," *Journal of Optimization Theory and Applications*, vol. 26, pp. 395–425, 1978.

[136] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[137] S. K. Lucas and C. Y. Kaya, "Switching-time computation for bang-bang control laws," in *Proceedings of the 2001 American Control Conference*, vol. 1, 2001, pp. 176–181.

[138] S. Kameswaran and L. Biegler, "Convergence rates for direct transcription of optimal control problems using collocation at radau points," *Computational Optimization and Applications*, vol. 41, pp. 81–126, 2008.

[139] S. Trimboli, S. Di Cairano, A. Bemporad, and I. Kolmanovsky, "Model predictive control with delay compensation for air-to-fuel ratio control," *Time Delay Systems: Methods, Applications and New Trends*, pp. 319–330, 2012.

[140] S. Uma, M. Chidambaram, A. S. Rao, and C. K. Yoo, "Enhanced control of integrating cascade processes with time delays using modified Smith predictor," *Chemical Engineering Science*, vol. 65, no. 3, pp. 1065–1075, 2010.

[141] 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*, vol. 30, no. 2, pp. 143–158, 2006.

[142] R. W. H. Sargent, "Optimal control," *Journal of Computational and Applied Mathematics*, vol. 124, no. 1-2, pp. 361–371, 2000.

[143] V. L. Denis, C. Jauberthie, and B. G. Joly, "Identifiability of a nonlinear delayed-differential aerospace model," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 154–158, 2006.

[144] V. M. Zavala and L. T. Biegler, "The advanced-step NMPC controller: optimality, stability and robustness," *Automatica*, vol. 45, no. 1, pp. 86–93, 2009.

[145] V. Miranda and R. Simpson, "Modelling and simulation of an industrial multiple effect evaporator: tomato concentrate," *Journal of Food Engineering*, vol. 66, no. 2, pp. 203–210, 2005.

[146] V. Rehbock, K. L. Teo, L. S. Jennings, and H. W. J. Lee, "A survey of the control parametrization and control parametrization enhancing methods for constrained optimal control problems," *Progress in Optimization: Contributions from Australasia*, vol. 30, pp. 247–275, 1999.

[147] W. Sun and Y. Yuan, *Optimization theory and methods: nonlinear programming*. Springer Verlag, 2006, vol. 1.

[148] W. W. Hager, "Runge-Kutta methods in optimal control and the transformed adjoint system," *Numerische Mathematik*, vol. 87, pp. 247–282, 2000.

[149] X. L. Liu, G. R. Duan, and K. L. Teo, "Optimal soft landing control for moon lander," *Automatica*, vol. 44, no. 4, pp. 1097–1103, 2008.

[150] X. Liu, "Constrained control of positive systems with delays," *IEEE Transactions on Automatic Control*, vol. 54, no. 7, pp. 1596–1600, 2009.

[151] X. Xu and P. J. Antsaklis, "Optimal control of switched systems based on parameterization of the switching instants," *IEEE Transactions on Automatic Control*, vol. 49, no. 1, pp. 2–16, 2004.

[152] X. Y. Yin, W. F. Yao, and Y. Z. Hu, "Optimization of gradient separation for chromatographic fingerprint of herbal medicine: a predictive model combined with grid search," *Journal of Chromatographic Science*, vol. 46, pp. 722–729, 2008.

[153] Y. B. Kim and S. J. Kang, "Time delay control for fuel cells with bidirectional DC/DC converter and battery," *International Journal of Hydrogen Energy*, vol. 35, no. 16, pp. 8792–8803, 2010.

[154] Y. H. Kim, C. K. Yoo, and I. B. Lee, "Optimization of biological nutrient removal in a sbr using simulation-based iterative dynamic programming," *Chemical Engineering Journal*, vol. 139, no. 1, pp. 11–19, 2008.

[155] Y. Sun, B. van Wyk, and Z. Wang, "A new multi-swarm multi-objective particle swarm optimization based on pareto front set," *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, pp. 203–210, 2012.

[156] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 2, pp. 267–278, 2010.

[157] Z. K. Chen and H. W. Chen, "New research on burden calculation for raw mix slurry in production of alumina with sintering process," *World Nonferrous Met*, pp. 41–45, 2004.

[158] Z. Wan and M. V. Kothare, "Robust output feedback model predictive control using off-line linear matrix inequalities," *Journal of Process Control*, vol. 12, no. 7, pp. 763–774, 2002.