Symbolic-Numeric Approaches Based on Theories of Abstract Algebra to Control, Estimation, and Optimization

Tomoyuki Iori

Department of Systems Science, Graduate School of Informatics

Kyoto University

A thesis submitted for the degree of

PhD in Informatics

2021

Abstract

Various problems in the real world can be formulated as nonlinear optimization problems. In exchange for their full capability of modeling complicated situations and diverse objectives, nonlinearity makes it difficult to provide exact or efficient algorithms for solving them. The exactness of algorithms is useful to guarantee the theoretical properties of their outputs, while the efficiency of solution methods is particularly crucial in applications to optimal control and estimation theory. In this thesis, for optimal control, optimal estimation, and optimization, we aim to propose efficient or exact methods combining numerical and symbolic computation based on theories of abstract algebra. Since symbolic computation can manipulate mathematical expressions including indeterminates, it exactly keeps theoretical properties of mathematical expressions and can deal off-line with variables to be determined on-line, that is, unknown in advance of the implementation of controllers or estimators.

For a class of finite-horizon optimal control and estimation problems, we propose symbolic methods that eliminate auxiliary variables in the problems off-line to make them more tractable. After the variable elimination, the small-sized problems are efficiently solved on-line by numerical computation. Another class of optimal estimation problems called the Bayesian filtering problems is also tackled by using the holonomic gradient method, which is a symbolic-numeric method based on the theory of D-modules, for efficient evaluation of posterior mean and variance of the state.

On the other hand, we also take advantage of the exactness of symbolic computation to obtain necessary optimality conditions for polynomial optimization problems. The obtained necessary optimality conditions are satisfied by all local minimizers, which means that it does not require any constraint qualifications. A limit operation of a parameter, which is introduced to relax the constraints of the original problems, is considered algebraically by utilizing the notions of projective space and tangent cone and exactly performed by symbolic computation. Consequently, we provide an algorithm for constructing a set of equations satisfied by all optimal solutions, namely, a necessary optimality condition without any constraint qualifications.

Acknowledgements

First of all, I would like to express my deepest appreciation to my supervisor, Professor Toshiyuki Ohtsuka, who inspired me to choose a life with research and guided me in my PhD program. Through his words and deeds, he has enlightened me to have a sincere attitude towards research and enthusiasm for science, while he also gave me great freedom to explore the beautiful and exciting world of research. Without his support with great patience, this work would not be completed.

I am grateful to Professor Yoshito Ohta and my co-supervisor, Professor Shin Ishii for being my thesis committee members. I am also grateful to Associate Professor Yu Kawano at Hiroshima University for his helpful discussions, advice, and support about my first journal paper. I would also like to thank Associate Professor Kazunori Sakurama for his valuable comments and discussions in research seminars of Ohtsuka laboratory and thank Assistant Professor Kenta Hoshino for his cordial advice on my career as a researcher. I am also grateful to Asst. Prof. Yuno at Kyushu University, Asst. Prof. Satoh at Tokyo Denki University, and Asst. Prof. Okajima at Joetsu University of Education for their valuable discussions and suggestions.

I would like to thank the members and staff of Ohtsuka laboratory for supporting my daily work and life. My special thanks go to Ms. Tobimatsu and Ms. Suzuki for supporting my research activities as well as other Ohtsuka lab's members who have joined my voluntary book readings. I would also thank Dr. Akutsu, Mr. Hamada, and Mr. Ito for the delightful conversations and discussions during my daily life as a PhD student.

I am grateful for the generous financial support from the Japan Society for the Promotion of Science throughout my PhD study.

Finally, I would like to express my heartfelt gratitude to my family and girlfriend for their strong, continuous, and unconditional support. My odyssey for research would not have started without their sincere understanding which has allowed me to freely pursue my career and would not be completed without her encouragement that saved me many times from dispiritedness.

Contents

List of Figures

Notation

Symbols	Meanings
$\mathbf{Z}_+$	Set of positive integers
$\mathbf{Z}_{\geq 0}$	Set of integers greater than or equal to zero
$\mathbf{R}$	Field of real numbers
$\mathbf{K}[x]^n$	Set of all n-dimensional vectors consisting of polynomials in $\mathbf{K}[x]$
$\mathbf{K}[x]^{n\times m}$	Set of all $n \times m$ matrices consisting of polynomials in $\mathbf{K}[x]$
$\langle F(x) \rangle$	Ideal generated by a set of generators $F(x) \subset \mathbf{R}[x]$
$F(x) = 0$	Set of equations $\{F_1(x) = 0, \ldots, F_m(x) = 0\}$
$\mathcal{V}(I)$	Algebraic set defined by ideal $I \subset \mathbf{R}[x]$
$\mathcal{V}(F(x))$	Algebraic set defined by ideal $\langle F(x) \rangle$
$x^\top$	Transpose of vector or matrix x
$x_{[s:t]}$	Sequence of symbols $\{x_s, x_{s+1}, \ldots, x_t\}$ $(s \in \mathbf{Z}_{\geq 0}, t \in \mathbf{Z}_{\geq 0} \cup \{\infty\}, s < t)$
$\mathrm{diag}[d_1, \ldots, d_n]$	Diagonal matrix whose diagonal components are $d_1, \ldots, d_n$
$\text{block-diag}[D_1, \ldots, D_n]$	Block-diagonal matrix whose diagonal blocks are $D_1, \ldots, D_n$
$\det A$	Determinant of a matrix $A \in \mathbf{R}[x]^{m\times m}$
$\mathrm{PD}(n)$	Set of all positive definite $n \times n$ matrices
$\mathrm{vech}(A)$	Half-vectorization of a positive definite matrix A
$g(x; a)$	Function in variable $x \in \mathbf{R}^n$ having parameter $a \in \mathbf{R}^m$
$g_i(x)$	i-th component of vector-valued function $g(x)\colon \mathbf{R}^n \to \mathbf{R}^m$
$\nabla_x g(x)$	Matrix-valued function whose (i, j) component is $\partial g_j(x)/\partial x_i$
$\partial g(x)/\partial x$	Matrix-valued function whose (i, j) component is $\partial g_i(x)/\partial x_j$
$p(x)$	Probability density function (PDF) of stochastic variable x
$p(x \mid y)$	PDF of stochastic variable x conditionally on stochastic variable y
$E[\cdot]$	Expectation
$f \circ g$	composition of set mappings $g : U \to V$ and $f : V \to W$
$l \bullet f$	Action of differential operator $l \in \mathcal{D}_n$(or $\mathcal{R}_n$) on a sufficiently smooth function f

Chapter 1

Introduction

1.1 Background and Motivation

Optimization problems are the problems of finding a solution under several constraints that maximizes a certain performance index, minimizes some cost and risk, or even achieves both objectives simultaneously. In recent years, more and more problems are formulated as optimization problems because of their capability of describing complicated situations in the real world and considering diverse objectives. Under some convexity assumptions, there are various theoretical results and solution methods for optimization problems, which form a branch of optimization theory called convex optimization [1, 2]. However, for general nonlinear cases, there still remain many open problems.

One of the most important applications of optimization theory is optimal control and estimation theory in the field of systems and control. In the wake of Rudolf E. Kalman's great works on the optimal control and estimation theory [3–5] in the 1960s, linear systems theory has been the main subject of systems and control and intensively studied since the mid-20th century [6, 7]. In linear systems theory, systems are modeled as linear ordinary differential equations (ODEs) in continuous-time cases or as linear difference equations in discrete-time cases. Hence, we can utilize the matured theory and tools provided by linear algebra for system analysis and control. However, if systems have nonlinearity, we can no longer make use of them and have to rely on other fields of mathematics that are suitable for each particular problem setting.

The most common approach to deal with nonlinearity in optimization is the geometric approach [8], where many concepts and arguments such as feasible solutions and optimality are described as abstract sets of points and their relationships such as inclusion and disjointness. Although this approach can take quite general situations

into account, its results are usually non-constructive and cannot be implemented as algorithms on computers directly. Hence, to implement a geometric method, it is necessary to confirm whether or not the problem admits geometric properties required by the method and to construct a problem-specific algorithm that realizes the method by computers, both of which may be difficult depending on each specific problem. Another approach is based on analysis [9], which approximates nonlinearity pointwise by linear functions or other classes of simple functions such as polynomials. This approach tends to focus on the local properties of the optimization problems and usually requires an infinite number of parameters or procedures to achieve optimization. Therefore, we need to cut off it up to a certain finite number to implement algorithms on computers. In systems and control, the differential geometric approach pioneered by Alberto Ishidori is one of the most successful geometric and analytic approaches [10]. However, this approach also tends to yields non-constructive results or an infinite number of procedures similar to the case of optimization theory.

Recently, mathematical tools based on the theories of abstract algebra have been utilized to overcome such difficulties [11–20]. The algorithms based on abstract algebra have been intensively studied from the perspective of symbolic computation. In spite that the notions in abstract algebra are quite abstract, most of them inherit so to speak finiteness of linear algebra. This property is useful for constructing algorithms and particularly for ensuring that they will eventually terminate. Symbolic computation, also called computer algebra, is a field of mathematics that studies algorithms for manipulating mathematical expressions. Since symbolic computation can manipulate the expressions including variables that have no given values or is undetermined off-line, in systems theory, it can be used to transfer a part of on-line computation to off-line, that is, before a controller or estimator is started to operate. Another advantage of symbolic computation is its exactness; contrary to numerical computation, which manipulates floating-point numbers, symbolic computation can manipulate mathematical objects exactly as symbols with no approximation. This feature enables us to derive new theoretical results by ensuring the theoretical properties of mathematical expressions processed by symbolic computation. By taking these advantages, we can push the boundary of solution methods for optimization problems in the sense of the trade-off between exactness and computational cost.

This thesis aims to propose symbolic-numeric methods for optimal control, optimal estimation, and optimization to provide efficient algorithms for solving computationally demanding problems by introducing off-line computation and to overcome

difficulties in numerical computation. We briefly introduce the backgrounds and motivation of using symbolic computation in optimal control, optimal estimation, and optimization.

1.1.1 Optimal Control Theory

The most basic problem in optimal control theory is the linear quadratic regulator (LQR) problem, which had been completely solved in the paper of R. E. Kalman [3]. In general, optimal control problems are reduced to solving the Hamilton-Jacobi-Bellman (HJB) equation, which is derived from the study of dynamic programming pioneered by Richard E. Bellman in the 1950s [21]. The HJB equation is a nonlinear partial differential equation (PDE) including the minimization of the Hamiltonian function with respect to input variables, which is closely related to the well-known minimum principle due to Lev S. Pontryagin [22].

The HJB equation is in general hard to solve. In the LQR, however, by virtue of linearity of the system and quadratic cost, the minimizing input can be explicitly described by the costate under certain mild assumptions. The HJB equation is then reduced to a special type of matrix ODE called the Riccati equation. A number of the solution methods to the Riccati equation have been studied for both continuous-time and discrete-time cases [7, 23]. However, all of these techniques based on linearity can only be used locally when a nonlinear system is well approximated by a linear system. For example, they cannot be used for the systems such as bilinear systems that cannot be accurately approximated by linear systems. Accordingly, the HJB equation for nonlinear systems is generally intractable, and thus some additional assumptions and alternative equations are introduced to make the problem more tractable.

One of such assumptions is that the nonlinear system has the control-affine structure. In this case, like the linear cases, the minimizing input is explicitly obtained as a function of the state and costate under mild assumptions. The HJB equation is then reduced to a set of nonlinear PDEs called the Hamilton-Jacobi (HJ) equation. The HJ equation is still difficult to solve but admits the symplectic structure, which is a special geometric structure of its solution space in symplectic geometry. A lot of methods utilizing the symplectic structure have been proposed such as stable manifold method [24].

Another substitute of the HJB equation is the Euler-Lagrange equations (ELEs), which are obtained by considering first-order necessary optimality conditions in the calculus of variations [23]. The ELEs can be regarded as a two-point boundary value problem, that is, a set of ODEs accompanied by initial and terminal conditions.

This structure is vigorously utilized for many solution methods such as the single shooting method and backward sweep method [23]. The ELEs are more tractable than the HJB equation or HJ equation since they are necessary conditions for the optimality and consist of ODEs and algebraic constraints. Hence, the ELEs tend to be studied in the context of nonlinear model predictive control (NMPC), where the finite-horizon optimal control problem (FHOCP) needs to be solved recursively [25]. However, particularly for nonlinear systems and cost functions, the ELEs are still difficult to solve enough rapidly. If we can simplify the structure of the ELEs by symbolic computation off-line, the on-line computation to solve the ELEs would be reduced.

1.1.2 Optimal Estimation Theory

The basis of optimal estimation theory is the Bayesian estimation, where an optimal estimate is computed from the posterior probability density function (PDF) under a certain optimal criterion [23, 26, 27]. The main advantage of the Bayesian estimation compared to the maximum likelihood estimation is that it makes use of the prior distribution that represents our prior knowledge of the estimated stochastic variables and the model generating them. In the context of the state estimation for dynamical systems, the prior knowledge on the estimated state can be obtained literally from the prior estimate, whence the Bayesian estimation is suitable for estimating the state of dynamical systems.

One of the Bayesian approaches to the state estimation problems is the Bayesian filter. This approach is pioneered, in the field of systems theory, by R. E. Kalman in his paper proposing the most popular filtering algorithm called Kalman filter (KF) [4]. The Kalman filter can be regarded as a realization of the Bayesian filter for linear systems with Gaussian noise. The linearity of systems ensures that all the related distributions are Gaussian, which makes various probability theoretic operations such as marginalizations and expectations drastically tractable.

However, the Bayesian filter for nonlinear systems is hard to realize as an algorithm. One of the most critical obstacles is that the PDF of the state is no longer Gaussian even when the noise is assumed to be Gaussian. Hence, how to approximate the state PDF has been the main interest of nonlinear filtering theory. The most popular extension of the KF to nonlinear cases is the extended KF (EKF) [26]. This method uses the linear approximation of state and observation equations at the previous estimate, and thus the derived Gaussian is just a local approximation of the original non-Gaussian distribution. In another extension called the unscented

KF (UKF) [28], the approximation of non-Gaussian distributions is performed more accurately by the unscented transformation. The unscented transformation can be regarded as a particular Gauss-Hermite quadrature rule [29], which leads to other versions of Gauss-Hermite filter (GHF) [29] or quadrature KF (QKF) [30]. Another approximation approach, which does not rely on Gaussian, is provided by the Monte Carlo method. The particle filter (PF), proposed independently by Kitagawa [31,32] and Gordon [33], approximates non-Gaussian distributions by a finite number of samples. By using the Monte Carlo approximation, we can approximate non-Gaussian distributions arbitrarily accurately and easily perform marginalizations and expectations though the computational cost increases as we use more and more samples.

As mentioned above, it can be said that the most problematic point in the Bayesian filter for nonlinear systems is integrations of nonlinear functions, which are accompanied by several probabilistic operations such as marginalizations and expectations and have to be performed within a short interval of sampling times. To overcome this difficulty, symbolic computation would be used for performing such burdensome computations off-line.

In the Bayesian filter, the current estimate is computed on the basis of the previous estimate, in other words, the estimate of the previous step is fixed when estimating the current state. However, we can estimate the past estimates again by using the observed outputs up to the current time step, which often provides more accurate estimates and is called smoothing in signal processing [27]. This consideration leads to the finite-horizon optimal estimation problem (FHOEP), where a finite number of successive states, including the current and past states, are simultaneously estimated by using the same number of successive observed outputs. The FHOEP can also be understood in the context of the Bayesian filtering problem [26] by considering the PDF of the initial state on the horizon as the prior knowledge. As with the FHOCP, the FHOEP can also be solved recursively, which yields moving horizon estimation (MHE). Moreover, if the maximization of the joint posterior density is selected as the optimality criterion, the FHOEP shares a similar structure with the FHOCP as a nonlinear optimization problem. It is worth exploiting this similar structure to find an efficient algorithm for solving the FHOEP with the analogy to that for solving the FHOCP.

1.1.3 Optimization Theory

For solving the FHOCP and FHOEP, we consider first-order necessary conditions for optimality, called the Karush-Kuhn-Tucker (KKT) conditions, instead of the opti-

mization problems themselves. The KKT conditions, which were obtained independently by William Karush in 1939 [34] and Harold W. Kuhn and Albert W. Tucker in 1951 [35], have been a significant tool in optimization theory. One reason for the KKT conditions to be so popular is that they can be described as relatively simple equalities and inequalities. This feature is particularly suitable for optimization algorithms since we can easily check whether they are satisfied or not at a given point and solve the equalities and inequalities to find optimal solutions.

In the linear quadratic cases, namely, the LQR problems in optimal control and the FHOEPs of linear systems with Gaussian noise, the KKT conditions are also sufficient for optimality. Moreover, the uniqueness of the optimal solution can also be guaranteed in those cases. Hence, solving the KKT conditions is exactly equivalent to solving the original optimization problems [8].

For nonlinear optimization problems, however, any of such preferable properties no longer hold in general. In particular, there are problems where the KKT conditions are not even necessary optimality conditions, that is, there are some optimal solutions violating the KKT conditions. For avoiding such situations, additional conditions to guarantee the KKT conditions to be necessary optimality conditions, namely, constraint qualifications (CQs) have been studied. Various CQs such as the linear independence CQ (LICQ), Mangasarian-Fromovitz CQ (MFCQ), Abadie CQ (ACQ), and Guignard CQ (GCQ) have been proposed [2, 36]. In particular, it is known that the GCQ holds if and only if the KKT conditions are necessary optimality conditions [37]. Although various necessary optimality conditions without CQs have also been obtained [2,8,38], most of them are described in a geometric or analytic way, which are not suitable for solving nonlinear optimization problems numerically. Hence, it is still worth exploring the necessary optimality conditions without CQs in an algebraic way.

1.2 Overview of symbolic methods for control, estimation, and optimization

Symbolic computation, especially as a tool for abstract algebras, has been drastically developed since the invention of an algorithm for manipulating polynomial equations called the Buchberger algorithm, which is proposed by Bruno Buchberger in the 1960s [39, 40]. For a given set of polynomials, this algorithm computes a finite set of polynomials called a Gröbner basis, which enjoys good properties for symbolic computation. By using Gröbner bases, various mathematical objects such as ideals

and modules in ring theory, which are too complicated to calculate by hand, have become computable concretely in a symbolic manner. As symbolic computation does not involve any approximations and can be performed while including unknown parameters , there have been vigorously applied to systems theory and optimization theory.

One of the most popular applications of Gröbner bases lies in so-called elimination theory, which explores systematic for polynomial equations methods to eliminate variables from simultaneous polynomial equations like Gaussian elimination for simultaneous linear equations. An early application of elimination theory to optimization can be seen in for example [41], where a first-order necessary optimality conditions such as KKT conditions and Fritz-John (FJ) conditions are transformed into a sort of triangular form that is suitable to find solutions by using Gröbner bases. Safey El Din [13] proposed a symbolic algorithm for finding the global supremum of a polynomial, where all generalized critical values are computed by elimination theory and then the supremum of them is identified. In systems and control, Walter et al. [42] utilized elimination theory to compute switching surfaces for time-optimal control. Ohtsuka [17] proposed a recursive elimination algorithm for FHOCPs of discrete-time rational systems, which decouples the ELEs of an FHOCP into a finite number of sets of polynomial equations by using elimination theory. Menini and Tornambe [18] characterized all the embeddings of a polynomial system as an elimination ideal for constructing high-gain observers. Such tools from elimination theory, especially those used in [17], are also used in Chapters 3 and 4 of this thesis.

In algebraic geometry, the mathematics of polynomials and its zeros, various geometric notions are associated with algebraic notions and can also be manipulated by the symbolic computation of Gröbner bases. In fact, variable elimination from polynomial equations, which is introduced above, is also interpreted as projection of their solution set. This interpretation is used in [17] to keep the property of mathematical expressions to be necessary optimality conditions during symbolic computation. In Chapter 6, two geometric notions of a tangent cone and projective space are manipulated to perform a limit operation exactly by symbolic computation.

Gröbner bases have also been used for computing a module, which is a generalization of the notion of vector space. In [43], for continuous-time polynomial systems, a class of stabilizing state feedback laws are characterized by using the syzygy module. Yuno and Ohtsuka [44] proposed algorithms to construct two types of static output-feedback laws characterized by modules over a ring, one of which renders a given algebraic set invariant and the other realize a given vector field on a given algebraic

set. Moreover, in [45], they derived a constructive sufficient condition for the existence of a dynamic state-feedback compensator that renders a given semi-algebraic set invariant and is constructed by symbolic computation of modules. On the other hand, in optimization theory, the holonomic gradient descent, which is a variation of gradient descent based on the theory of D-modules in algebraic analysis, is proposed by Nakayama et al. [46] for estimating parameters of a certain probability density function. Its generalized version called the holonomic gradient method (HGM) is also proposed for efficient evaluation of normalizing constants in statistics, and its general problem setting would allow us to use it for various purposes of control and estimation theory. In Chapter 5, we use the HGM to exactly perform the Bayesian filter for nonlinear systems.

1.3 Outline and contribution

This thesis proposes several symbolic-numeric methods for control, estimation, and optimization. Chapter 2 introduces some preliminaries on optimization theory and optimal control and estimation theory. Chapters 3 and 4 propose a symbolic computation method called a recursive elimination method for two types of problems: the FHOCP with terminal constraints and the FHOEP. Chapter 5 proposes an efficient Bayesian filtering method by utilizing symbolic computation on the ring of differential operators. In Chapter 6, we derive a necessary optimality condition for polynomial optimization problems, which can be described as polynomial equations and does not require any CQ. Chapter 7 concludes this thesis and discusses the future work. Some basic notations used in this thesis are included in Appendix A, and Appendix B gives the proofs of auxiliary lemmas and theorems appeared in Chapters 3 and 6. In the remaining part of this section, we briefly introduce the contributions of each chapter.

Chapter 2 — Preliminaries This chapter is devoted to introducing the basics of optimization theory and optimal control and estimation theory, which are used in the following chapters. First, we define the fundamental concepts of optimization theory and introduce some basic theorems in Section 2.1. These concepts and theorems are not only the basis of all the problems discussed in this thesis but also the main subject of Chapter 6. Next, in Section 2.2, the FHOCP and its extended version with terminal constraints are formulated, which is the problem considered in Chapter 3. Finally, we formulate two types of optimal estimation problems in Section 2.3. The first one is the FHOEP, which is the dual concept of the FHOCP in estimation theory

and considered in Chapter 4. The second one is the Bayesian filtering problem, a probabilistic method for state estimation. This problem is considered in Chapter 5

Chapter 3 – Recursive elimination method for finite-horizon optimal control problems with terminal constraints This chapter proposes a recursive elimination method for FHOCPs with terminal constraints under the assumption that all the functions that appeared in the problems are rational or algebraic functions. This method decouples the ELEs into sets of algebraic equations, where each equation set involves only the variables at a single time step. This decoupling is performed by symbolic computation with the concept of elimination ideals, a mathematical tool from commutative algebra. The structure of decoupled equation sets is especially suitable for MPC since the initial optimal input can be computed from the equation set involving the initial state and input. Moreover, we provide sufficient conditions such that the decoupled equation sets locally give a unique optimal feedback control law at each time step. Two numerical examples are provided to illustrate the proposed method and sufficient conditions.

Chapter 4 – Recursive elimination method for moving horizon estimation This chapter proposes a recursive elimination method for FHOEPs of discrete-time nonlinear systems with non-Gaussian noise. FHOEPs are formulated as joint maximum a posteriori (MAP) estimation problems of the state trajectory on the horizon. By utilizing elimination ideals, we eliminate the past state variables from the first-order necessary optimality conditions of the joint MAP estimation problems. By virtue of the recursive structure of FHOEPs, this variable elimination is performed recursively over the discrete-time steps from the oldest time step to the current. Consequently, we obtain an implicit function representation of the state estimate as the function of observed outputs over the horizon. MHE is performed by solving the implicit function representation repeatedly. Numerical examples show the efficiency of the proposed method.

Chapter 5 – Bayesian state estimation via the holonomic gradient method This chapter proposes a symbolic-numeric solution method for Bayesian filtering problems of nonlinear stochastic systems. In this method, though the posterior PDFs of the state are approximated by Gaussian distributions, their mean and variance are computed exactly in the sense that there is no approximation in the one-step estimation process. The current mean and variance are considered as the functions of the

previous mean and variance and observed output, and partial differential equations (PDEs) satisfied by the functions are computed off-line by symbolic computation. In the on-line part, the solution of the derived PDEs is efficiently evaluated for given previous estimates and observed output at every sampling time by using the HGM. A numerical example is provided to show the efficiency of the proposed method compared to the EKF, UKF, and PF.

Chapter 6 – Limit operation in projective space for constructing necessary optimality condition of polynomial optimization problem This chapter proposes a necessary optimality condition for optimization problems of polynomial functions with constraints described by polynomial equations. The proposed condition is more general than the KKT conditions in the sense that it does not require any constraint qualifications. An algorithm to derive the condition is also proposed by using symbolic computation nevertheless the condition itself is based on the penalty function method, which is a classical numerical solution method for constrained optimization problems. Hence, the output of the algorithm is a set of polynomial equations, which is more suitable for computers to find candidates of optimal solutions than other geometric necessary conditions. Two illustrative examples are provided to clarify the methodology and to demonstrate the cases where some local minimizers do not satisfy the KKT conditions.

Chapter 2

Preliminaries

In this chapter, we introduce the basics of optimization theory, optimal control problem, and optimal estimation problem, which will be used in the following chapters.

This chapter is organized as follows. Section 2.1 provides a basic introduction of optimization problem particularly focusing on the concept of optimality conditions, which is discussed in Chapter 6. Section 2.2 is devoted to formulate the finite-horizon optimal control problem (FHOCP) and FHOCP with terminal constraints, which is considered in Chapter 3. In Section 2.3, the optimal estimation problem is formulated in two ways: the finite-horizon optimal estimation problem (FHOEP), which is the subject of Chapter 4, and the Bayesian filtering problem considered in Chapter 5.

2.1 Optimization problem

Consider the following optimization problem:

$$\begin{aligned} &\min_{x} f(x) \\ &\text{s. t. } g(x) = 0, \end{aligned} \tag{2.1}$$

where $x \in \mathbf{R}^n$ is the decision variable and $f\colon \mathbf{R}^n \to \mathbf{R}$ and $g\colon \mathbf{R}^n \to \mathbf{R}^m$ are the cost function and the equality constraint function, respectively. Note that for an inequality constraint $h(x) \leq 0$, we can reformulate it as an equality constraint $\tilde{g}(x, s) = 0$ by introducing a slack variable $s \in \mathbf{R}$ as follows:

$$\tilde{g}(x, s) = h(x) + s^2 = 0. \tag{2.2}$$

Hence, the inequality constraints can be also considered in the setting of optimization problem (2.1). For the details of reformulation (2.2), see [8, 47, 48].

In what follows, several basic definitions are introduced along the line of [2, 8].

Definition 2.1 (Feasible point and set)**.** The point $x \in \mathbf{R}^n$ is said to be *feasible* for the optimization problem (2.1) if $g(x) = 0$ holds. The *feasible set* $\mathcal{X}$ is defined as the subset of all the points that satisfy the constraints, that is,

$$\mathcal{X} \coloneqq \{x \in \mathbf{R}^n \mid g(x) = 0\}.$$

Definition 2.2 (Tangent cone)**.** The *tangent cone* of $\mathcal{X}$ at $x \in \mathcal{X}$ is defined by

$$T(x) \coloneqq \left\{ d \in \mathbf{R}^n \mid \exists \{x_k\} \subset \mathcal{X}, t_k \downarrow 0 \text{ s. t. } x_k \to x \text{ and } \frac{x_k - x}{t_k} \to d \right\}.$$

Remark 2.1. The tangent cone defined above is slightly different from the tangent cone in algebraic geometry defined in subsection A.2.4. We use the former only in this section and use the latter in Chapter 6.

Definition 2.3 (Linearized tangent cone)**.** The *linearized tangent cone* of $\mathcal{X}$ at $\tilde{x} \in \mathcal{X}$ is defined by

$$L(\tilde{x}) \coloneqq \left\{ d \in \mathbf{R}^n \mid \frac{\partial g(\tilde{x})}{\partial x} d = 0 \right\}.$$

Definition 2.4 (Dual of cone)**.** The *dual cone* $\mathcal{C}'$ of a cone $\mathcal{C}$ is defined by

$$\mathcal{C}' \coloneqq \left\{ d \in \mathbf{R}^n \mid d^\top x \geq 0 \ \forall x \in \mathcal{C} \right\}.$$

Definition 2.5 (Local minimizer)**.** The feasible point $x^* \in \mathcal{X}$ is said to be a *local minimizer* if, for a neighborhood $\mathcal{N} \ni x^*$,

$$f(x^*) \leq f(x) \quad \forall x \in \mathcal{N} \cap \mathcal{X}$$

holds. The value of the cost function $f(x^*)$ is said to be a *local minimum*.

Definition 2.6 (Global minimizer)**.** The feasible point $\hat{x} \in \mathcal{X}$ is said to be a *global minimizer* if

$$f(\hat{x}) \leq f(x) \quad \forall x \in \mathcal{X}$$

The value $f(\hat{x})$ is said to be the *global minimum*.

In solving the optimization problem (2.1), necessary optimality conditions are useful both from theoretic and algorithmic viewpoints. Let us begin with the following classical result.

Theorem 2.1 (Fritz John conditions)**.** Let x^* be a local minimizer of optimization problem (2.1). Assume that functions f and g are continuously differentiable. Then, there exists a scalar λ_0 and a vector $\lambda \in \mathbf{R}^m$ such that λ_0 and all the components of λ are not all zero and

$$\lambda_0 \frac{\partial f}{\partial x}(x^*) + \lambda^\top \frac{\partial g}{\partial x}(x^*) = 0 \tag{2.3}$$

holds.

When $\lambda_0 = 0$ in FJ conditions, the first term in the left-hand side of (2.3) vanish, which means that FJ conditions lose all the information of the cost function $f(x)$. λ_0 can be chosen not to be zero if the local minimizer x^* satisfy certain additional conditions. Such conditions, which guarantee the existence of non-zero λ_0 at a local minimizer x^*, are called constraint qualifications (CQs).

Definition 2.7 (Linear independence constraint qualification)**.** For a feasible point $x \in \mathcal{X}$, we say the *linearly independence constraint qualification (LICQ)* holds at x if all the row vectors of $\partial g/\partial x$ are linearly independent at x.

Definition 2.8 (Guignard constraint qualification)**.** For a feasible point $x \in \mathcal{X}$, we say the *Guignard constraint qualification (GCQ)* holds at x if $L'(x) = T'(x)$ holds.

Theorem 2.2 (Karush-Kuhn-Tucker conditions)**.** Let x^* be a local minimizer of the optimization problem (2.1). Assume that functions f and g are continuously differentiable and that a CQ holds at the point x^*. Then, there exists a non-zero vector $\lambda \in \mathbf{R}^m$ such that

$$\frac{\partial f}{\partial x}(x^*) + \lambda^\top \frac{\partial g}{\partial x}(x^*) = 0 \tag{2.4}$$

holds. The vector λ is known as the *Lagrange multiplier* corresponding to the constraint $g(x) = 0$.

Theorem 2.3. For the local minimizer x^*, there exists the Lagrange multiplier λ if and only if the GCQ holds at x^*, that is, the GCQ is the necessary and sufficient condition for the KKT conditions to be necessary optimality conditions.

The FJ and KKT conditions are described as equations. By introducing the penalty function, we can obtain another necessary optimality condition, which does not require any CQs but is much less conservative than the FJ conditions.

Definition 2.9 (Penalty function)**.** The *penalty function* $P(x; r)$ for the optimization problem (2.1) is defined as follows:

$$P(x; r) \coloneqq f(x) + r g^\top(x) g(x), \tag{2.5}$$

where $r \in [0, \infty)$ is a *penalty parameter*.

Theorem 2.4. Let $r_{[0:\infty]}$ be a monotonically increasing sequence such that $r_k \to \infty$ as $k \to \infty$ and $\hat{x}_k$ be a global minimizer of the penalty function $P(x; r_k)$. Then, every convergence point of the sequence $\hat{x}_{[0:\infty]}$ is a global minimizer of the optimization problem (2.1).

2.2 Optimal control problem

Consider the discrete-time nonlinear system

$$x_{k+1} = f_k(x_k, u_k), \tag{2.6}$$

where $k \in \mathbf{Z}_+$ denotes the discrete time step, $x_k \in \mathbf{R}^n$ and $u_k \in \mathbf{R}^m$ are the state and input, respectively, and $f_k \colon \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}^n$ is some given nonlinear function. For this system, the finite-horizon optimal control problem (FHOCP) is defined as follows:

$$\begin{aligned} \min_{u_{[0:N-1]}} \ & \phi(x_N) + \sum_{k=0}^{N-1} L_k(x_k, u_k) \\ \text{s. t. } & x_{k+1} = f_k(x_k, u_k) \quad \text{for } k = 1, \dots, N-1, \\ & x_0 = \bar{x}, \end{aligned} \tag{2.7}$$

where $N \in \mathbf{Z}_+$ is the length of the horizon, $\bar{x} \in \mathbf{R}^n$ is a given initial state, and functions $\phi \colon \mathbf{R}^n \to \mathbf{R}$ and $L_k \colon \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}$ are the terminal cost and stage cost, respectively. The KKT conditions for the FHOCP inherit its recursive structure over k; the KKT conditions are derived as follows:

$$x_{k+1} = \nabla_p H_k(x_k, u_k, p_{k+1}), \tag{2.8}$$

$$p_k = \nabla_x H_k(x_k, u_k, p_{k+1}), \tag{2.9}$$

$$0 = \nabla_u H_k(x_k, u_k, p_{k+1}), \tag{2.10}$$

$$p_N = \nabla_x \phi(x_N), \tag{2.11}$$

where $H_k(x_k, u_k, p_{k+1}) \coloneqq L_k(x_k, u_k) + p_{k+1}^\top f_k(x_k, u_k)$ is called the Hamiltonian of the FHOCP (2.7). The KKT conditions (2.8)–(2.11) are also called the Euler-Lagrange equations (ELEs). The FHOCP, as an optimization problem, satisfies the LICQ at any feasible point, and thus the KKT conditions or the ELEs are necessary optimality conditions.

The FHOCP is often solved as the subproblem of nonlinear model predictive control (NMPC), where the FHOCP is solved at every sampling time and only the

initial optimal input is used as the actual input of the system. To guarantee the stability of NMPC, terminal constraints are often imposed to the FHOCP, that is,

$$\begin{aligned} \min_{u_{[0:N-1]}} \ & \phi(x_N) + \sum_{k=0}^{N-1} L_k(x_k, u_k) \\ \text{s. t. } & x_{k+1} = f_k(x_k, u_k) \quad \text{for } k = 0, \dots, N-1, \\ & x_0 = \bar{x}, \\ & \psi(x_N) = 0, \end{aligned} \tag{2.12}$$

where $\psi\colon \mathbf{R}^n \to \mathbf{R}^d$ is the terminal constraint function. For the FHOCP with terminal constraints, the ELEs are almost the same as the normal FHOCP; only (2.11) is replaced by

$$p_N = \nabla_x \phi(x_N) + \nabla_x \psi(x_N)\nu, \tag{2.13}$$

where $\nu \in \mathbf{R}^d$ is the Lagrange multiplier corresponding to the terminal constraints. Note that in the case with terminal constraints the LICQ may not hold, which means the ELEs (2.8)–(2.10) and (2.13) do not serve necessary optimality conditions for the FHOCP with terminal constraints (2.12).

2.3 Optimal estimation problem

Consider the discrete-time nonlinear stochastic system

$$x_{k+1} = f_k(x_k, u_k) + w_k, \tag{2.14}$$

$$y_k = h_k(x_k) + v_k, \tag{2.15}$$

where symbols k, x_k, u_k, and f_k are the same as those in (2.6), while y_k denotes the output of the system, w_k and v_k denote the system and observation noises, respectively, and $h_k\colon \mathbf{R}^n \to \mathbf{R}^l$ is some given nonlinear function. The distributions of the noises w_k and v_k are given as their probability density functions (PDFs) denoted by $p_w(w_k)$ and $p_v(v_k)$, respectively. Throughout this thesis, the inputs are regarded as a time-varying parameter that does not depend on either the state or output in the context of the state estimation

In practical situations, what we can actually observe is only the sequence of the outputs, while many of the control methods require the current state. Hence, we have to estimate the current state from the observed output sequence. This leads us to formulate the optimal estimation problem, where an estimator is designed to minimize or maximize some performance index for a given output sequence. In this

thesis, we consider two types of optimal estimation problems: the modal trajectory estimation and the Bayesian filtering.

2.3.1 Finite-horizon optimal estimation problem

For the stochastic system, consider the following optimization problem:

$$\begin{aligned} &\max_{x_{[0:N]}} p(x_{[0:N]} \mid y_{[0:N]}) \\ \text{s. t. } &x_{k+1} = f_k(x_k, u_k) + w_k \quad \text{for } k = 1, \dots, N-1, \\ &y_k = h_k(x_k) + v_k, \\ &w_k \sim p_w(w_k), \\ &v_k \sim p_v(v_k), \\ &x_0 \sim p_0(x_0) \end{aligned} \tag{2.16}$$

where $N \in \mathbf{N}$ denotes the length of the horizon, $p_0(x_0)$ is the PDF of the initial state on the horizon, and $p(x_{[0:N]} \mid y_{[0:N]})$ denotes the joint PDF of the state trajectory $x_{[0:N]}$ conditionally on the observed output sequence $y_{[0:N]}$. The PDF $p(x_{[0:N]} \mid y_{[0:N]})$ is the joint posterior PDF in the Bayesian framework, and thus the optimal solution of (2.16) is called the joint *maximum a posteriori* (MAP) estimate. In this thesis, problem (2.16) is also referred to as an FHOEP as the counterpart of the FHOCP in the state estimation. The solution is also called the *modal trajectory* especially in the state estimation theory of dynamical systems [26].

In this problem setting, the constrained optimization problem (2.16) can be reformulated as an unconstrained optimization problem. First, according to the Bayes' rule, the conditional joint PDF can be written as

$$p(x_{[0:N]} \mid y_{[0:N]}) = \frac{p(x_{[0:N]}, y_{[0:N]})}{p(y_{[0:N]})}.$$

The denominator on the right-hand side is the joint PDF of the output sequence $y_{[0:N]}$ marginalized over all state trajectories and thus is independent of $x_{[0:N]}$. Therefore, the maximization (2.16) is equivalent to another maximization:

$$\max_{x_{[0:N]}} p(x_{[0:N]}, y_{[0:N]}). \tag{2.17}$$

The state trajectory $x_{[0:N]}$ is Markovian due to the constraint of the state equation (2.14), and similarly, the output y_k at a time step k only depends on the state x_k at the same time step due to the observation equation (2.15). Therefore, the

joint PDF $p(x_{[0:N]}, y_{[0:N]})$ can be rewritten as the product of the transition PDFs $p(x_k \mid x_{k-1})$, observation PDFs $p(y_k \mid x_k)$, and PDF of the initial state $p_0(x_0)$:

$$p(x_{[0:N]}, y_{[0:N]}) = p(y_0 \mid x_0)p(x_0) \prod_{k=1}^{N} p(y_k \mid x_k)p(x_k \mid x_{k-1}). \tag{2.18}$$

Hence, by taking the logarithms on both sides, maximization (2.17) is reformulated as minimization of the sum of logarithms:

$$\begin{aligned} -\ln p(x_{[0:N]}, y_{[0:N]}) = &-\{\ln p_0(x_0) + \ln p(y_0 \mid x_0)\} \\ &- \sum_{k=1}^{N} \{\ln p(x_k \mid x_{k-1}) + \ln p(y_k \mid x_k)\}. \end{aligned} \tag{2.19}$$

From the rule of transformation of PDFs, all the conditional PDFs in (2.19) are rewritten as

$$p(x_k \mid x_{k-1}) = p_w(w_{k-1}) \left| \det \frac{\partial x_k}{\partial w_{k-1}} \right|^{-1} = p_w(w_{k-1}), \tag{2.20}$$

and similarly,

$$p(y_k \mid x_k) = p_v(v_k) \left| \det \frac{\partial y_k}{\partial v_k} \right|^{-1} = p_v(v_k), \tag{2.21}$$

where the derivatives $\partial x_k / \partial w_{k-1}$ and $\partial y_k / \partial x_k$ are both identity matrices owing to (2.14) and (2.15), and thus the absolute values of their determinants are 1. In the end, by substituting the state and observation equations into (2.20) and (2.21), respectively, and then substituting these equations into the objective function (2.19), the joint MAP estimation problem (2.16) is reformulated as the following unconstrained nonlinear optimization problem:

$$\begin{aligned} \min_{x_{[0:N]}} \; &-\Big\{\ln p_0(x_0) + \ln p_v\left(y_0 - h_0(x_0)\right)\Big\} \\ &- \sum_{k=1}^{N} \Big\{\ln p_w\left(x_k - f_{k-1}(x_{k-1}, u_{k-1})\right) + \ln p_v\left(y_k - h_k(x_k)\right)\Big\}. \end{aligned} \tag{2.22}$$

2.3.2 Bayesian filtering

In this subsection, we introduce another type of optimal estimation problem called the Bayesian filtering problem. Let us consider the conditional PDF $p(x_k \mid y_{[0:k]})$, that is, the PDF of the current state x_k conditionally on the sequence of observed outputs $y_{[0:k]}$. This conditional PDF, which is particularly called the *posterior PDF* in Bayesian framework, has all the statistic information of the current state. The

optimal estimator is thus designed so that it minimizes the Bayes' risk, which is defined by the posterior PDF and a loss function that describes what is optimal.

By virtue of the recursive structure of dynamical systems, the posterior PDF $p(x_{k+1} \mid y_{[0:k+1]})$ can be computed from $p(x_k \mid y_{[0:k]})$ recursively as follows:

$$p(x_{k+1} \mid y_{[0:k]}) = \int_{\mathbf{R}^n} p(x_{k+1} \mid x_k) p(x_k \mid y_{[0:k]}) dx_k, \tag{2.23}$$

$$p(x_{k+1} \mid y_{[0:k+1]}) = \frac{p(y_{k+1} \mid x_{k+1}) p(x_{k+1} \mid y_{[0:k]})}{\int_{\mathbf{R}^n} p(y_{k+1} \mid x_{k+1}) p(x_{k+1} \mid y_{[0:k]}) dx_{k+1}}, \tag{2.24}$$

where the PDFs $p(x_{k+1} \mid x_k)$ and $p(y_{k+1} \mid x_{k+1})$ are obtained from (2.20) and (2.21), respectively. To obtain a point estimate from the posterior PDF, we consider the minimization of the Bayes' risk

$$E\left[R(x_k, \hat{x}_k) \mid y_{[0:k]}\right] \coloneqq \int_{\mathbf{R}^n} R(x_k, \hat{x}_k) p(x_k \mid y_{[0:k]}) dx_k, \tag{2.25}$$

where $\hat{x}_k \in \mathbf{R}^n$ denotes the estimate of the true state x_k, and

$$R(x_k, \hat{x}_k) \colon \mathbf{R}^{2n} \to \mathbf{R}$$

is the loss function, which describes the criterion of optimality.

In the end, the one-step estimation in the Bayesian filtering problem can be described as follows.

$$\begin{aligned} \min_{\hat{x}_k} \; & E\left[R(x_k, \hat{x}_k) \mid y_{[0:k]}\right] \\ \text{s. t. } & x_k = f_{k-1}(x_{k-1}, u_{k-1}) + w_{k-1}, \\ & y_k = h_k(x_k) + v_k, \\ & x_{k-1} \sim p(x_{k-1} \mid y_{[0:k-1]}), \end{aligned} \tag{2.26}$$

Note that this problem setting only involves the variables at current and previous time steps explicitly and depends on the past outputs $y_{[0:k-1]}$ only through the previous posterior PDF $p(x_{k-1} \mid y_{[0:k-1]})$. Therefore, for the simplicity of notations, we omit the past outputs and denote the current variables by the plain symbols, e.g. x_k by x, and denote the previous variables by the plain symbols with superscript $^-$, e.g. x_{k-1} by x^-.

There are many choices for the loss function $R(x_k, \hat{x}_k)$: the zero-one loss function, absolute error, squared error, and so on. In particular, the squared error $R(x_k, \hat{x}_k) = \|x_k - \hat{x}_k\|^2$ is the most popular one, which yields the *minimum mean squared error* (MMSE) estimate. For a large class of loss functions and posterior PDFs, the conditional mean $\hat{x}_k = E[x_k \mid y_{[0:k]}]$ is identical to the MMSE estimate, i.e., the optimal solution of problem (2.26) with the squared-error loss function, as stated in the following theorems.

Theorem 2.5. [26] For the quadratic loss function $R(x_k, \hat{x}_k) = (x_k - \hat{x}_k)^\top Q(x_k - \hat{x}_k)$, where Q is a positive semidefinite matrix, the conditional mean is the MMSE estimate regardless of the posterior PDF $p(x_k \mid y_{[0:k]})$.

Theorem 2.6. [27] The conditional mean is the MMSE estimate if the posterior PDF is symmetric with respect to the conditional mean and the loss function is symmetric and convex.

Throughout this thesis, we fix the loss function as the squared error and focus on the computation of the MMSE estimate, i.e., the conditional mean.

Chapter 3

Recursive Elimination Method for Finite-Horizon Optimal Control Problems with Terminal Constraints

3.1 Introduction

Finite-horizon optimal control problems (FHOCPs) are one of the most important problems in systems and control and have been studied not only as problems interesting by themselves but also as the subproblems of nonlinear model predictive control (NMPC). In particular, the FHOCPs with terminal constraints are useful to guarantee the asymptotic stability of the origin or other subsets of the state space in NMPC [49]. The FHOCPs with terminal constraints can be solved analytically under strong assumptions [23, 50] such as the linearity of the system, quadratic form of the cost function, and so on. However, in general, the FHOCPs cannot be solved analytically and are computationally demanding even when solved numerically.

By assuming certain algebraic properties of FHOCPs, we can apply mathematical tools from computer algebra to these problems. Computer algebra provides various concepts and algorithms of symbolic computation, which can be utilized to reduce the computational burdens of numerical solution methods. Fotiou et al. [51] formulated polynomial FHOCPs, whose nonlinearity is described in terms of polynomial functions, as parametric optimization problems on the initial state and solved the Karush-Kuhn-Tucker (KKT) conditions efficiently by introducing the concept of zero-dimensional ideals in commutative algebra. They also proposed a method to solve the parametric optimization problems by applying cylindrical algebraic decomposition (CAD), where the global optimal solution can be obtained if it exists. Iwane

et al. [52] combined CAD with dynamic programming and proposed a method to compute the value function and optimal control law at each time instant by using CAD. In contrast to these related approaches, in this chapter, we consider rational FHOCPs, where all nonlinearity is described by rational or algebraic functions. Rational FHOCPs with terminal constraints can formulate a wider range of optimal control problems than polynomial FHOCPs can.

Ohtsuka [17] proposed a method to decouple the Euler-Lagrange equations (ELEs) into finite number of sets of algebraic equations by using the concept of elimination ideals in commutative algebra. Each equation set involves only the variables at a single time instant, and this structure saves the computational cost to solve them numerically. Ohtsuka also provided sufficient conditions for the existence and uniqueness of optimal feedback laws in the form of algebraic functions. These sufficient conditions guarantee the properties of optimal solutions not only for the given initial state but also for the other initial states in some neighborhood of the given one. Moreover, by utilizing the concept of zero-dimensional ideals, it is shown that the unique optimal feedback law can be characterized as a point of an algebraic set in the algebraic closure of the rational functions in the state.

In the case with terminal constraints, the Lagrange multipliers associated with terminal constraints are introduced as additional variables. These additional variables are intrinsic to the problem; in contrast to state or costate variables, they cannot be explicitly expressed by the initial state and inputs. The sufficient conditions proposed in [17], which do not consider any terminal constraints or corresponding Lagrange multipliers, cannot be applied to problems with terminal constraints. We therefore extend the sufficient conditions to guarantee the existence of the Lagrange multipliers by using results from sensitivity analysis.

In this chapter, first, we introduce a recursive elimination method for solving FHOCPs with terminal constraints where all nonlinear functions are rational or algebraic functions. Next, we provide sufficient conditions for optimality of the problems. These conditions are less conservative than the classical sufficient conditions [23, 50]. Moreover, by applying the concept of zero-dimensional ideals to the outputs of the recursive elimination method, we characterize the optimal feedback laws in the form of algebraic functions.

3.2 Problem Formulation

Consider the FHOCP with terminal constraints (2.12). We rewrite the statement of the problem here for convenience;

$$\min_{u_{[0:N-1]}} \phi(x_N) + \sum_{k=0}^{N-1} L_k(x_k, u_k) \tag{3.1}$$

$$\text{s. t. } x_{k+1} = f_k(x_k, u_k), \text{for } k = 0, \ldots, N-1 \tag{3.2}$$

$$x_0 = \bar{x}, \tag{3.3}$$

$$\psi(x_N) = 0, \tag{3.4}$$

where $N \in \mathbf{Z}_+$ denotes an optimization horizon of the problem, $x_k \in \mathbf{R}^n$ and $u_k \in \mathbf{R}^m$ denote the state and input of a dynamical system at each time step $k = 0, \ldots, N$, and $\bar{x}$ denotes a given initial state. The scalar-valued functions $\phi\colon \mathbf{R}^n \to \mathbf{R}$ and $L_k\colon \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}$ denote the terminal cost and stage costs, respectively, and their sum is the cost function that will be minimized. Each equation (3.2) is a state equation of the system and consists of a vector-valued function $f_k\colon \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}^n$. Equation (3.4) defines the set of terminal constraints, and $\psi\colon \mathbf{R}^n \to \mathbf{R}^l$ is a vector-valued function. We assume that the components of f_k, $\nabla_x L_k$, and $\nabla_u L_k$ are rational functions and that the components of ψ and $\nabla_x \phi$ are algebraic functions. The definition of algebraic functions is as follows.

Definition 3.1. An analytic function $\rho\colon U \to \mathbf{R}$ defined on an open set $U \subset \mathbf{R}^n$ is said to be an *algebraic function* if a nonzero polynomial $\Phi(y, Y) \in \mathbf{R}[y, Y]$ exists such that $\Phi(y, \rho(y)) = 0$ holds for all $y \in U$. The polynomial $\Phi(y, Y)$ can be regarded as an element of $\mathbf{R}(y)[Y]$ instead of $\mathbf{R}[y, Y]$, and $\rho(y)$ is a root of $\Phi(y, Y) \in \mathbf{R}(y)[Y]$. Therefore, $\rho(y)$ is an element of $\overline{\mathbf{R}(y)}$. Note that each component of the derivative $\partial\rho(y)/\partial y$ is also an algebraic function.

Remark 3.1. Note that for the stage cost $L_k(x_k, u_k)$ only its *derivatives* are assumed to be rational functions. Hence, the stage cost can include, for example, a logarithmic function, which itself is not a rational function but its derivative is.

Our algebraic approach is based on the ELEs. To simplify their description, we first define the discrete-time Hamiltonian at time step k as

$$H_k(x_k, u_k, p_{k+1}) := L_k(x_k, u_k) + p_{k+1}^\top f_k(x_k, u_k), \tag{3.5}$$

where $p_k \in \mathbf{R}^n$ $(k = 0, \ldots, N)$ denote the costates. By using the discrete-time Hamiltonian, the ELEs corresponding to FHOCP (3.1)–(3.4) for $k = 0, \ldots, N-1$ can be described as follows:

$$x_{k+1} = f_k(x_k, u_k), \tag{3.6}$$

$$p_k = \nabla_x H_k(x_k, u_k, p_{k+1}), \tag{3.7}$$

$$\nabla_u H_k(x_k, u_k, p_{k+1}) = 0, \tag{3.8}$$

$$p_N = \nabla_x \phi(x_N) + \nabla_x \psi(x_N)\nu, \tag{3.9}$$

$$\psi(x_N) = 0, \tag{3.10}$$

where $\nu \in \mathbf{R}^l$ denotes a vector consisting of the Lagrange multipliers associated with the terminal constraints. If a constraint qualification (CQ) such as the linear independence constraint qualification (Definition 2.7) holds for terminal constraint (3.4), the ELEs are the necessary optimality conditions of the FHOCP for local optimality [25]. We assume that a CQ for the terminal constraint holds for all the feasible point and there exists at least one optimal solution that satisfies the ELEs. The solutions of the ELEs can be then regarded as candidates of the optimal solutions.

To apply mathematical tools from algebraic geometry to the ELEs, they have to be converted into algebraic equations. Equations (3.6)–(3.8) are converted in [17], and equation (3.9) can be readily converted in the same manner as described in [17]. For the completeness of this thesis, we purposely show the conversion of (3.6)–(3.9). First, we rewrite rational functions in (3.6)–(3.8) as follows:

$$f_k(x_k, u_k) = D_{xk}^{-1}(x_k, u_k) n_{xk}(x_k, u_k), \tag{3.11}$$

$$\nabla_x H_k(x_k, u_k, p_{x+1}) = D_{pk}^{-1}(x_k, u_k) n_{pk}(x_k, u_k, p_{k+1}), \tag{3.12}$$

$$\nabla_u H_k(x_k, u_k, p_{x+1}) = D_{uk}^{-1}(x_k, u_k) n_{uk}(x_k, u_k, p_{k+1}), \tag{3.13}$$

where $n_{xk} \in \mathbf{R}[x_k, u_k]^n$, $n_{pk} \in \mathbf{R}[x_k, u_k, p_{k+1}]^n$, and $n_{uk} \in \mathbf{R}[x_k, u_k, p_{k+1}]^m$ are the vectors of polynomials consisting of the numerators of f_k, $\nabla_x H_k$, and $\nabla_u H_k$, respectively, and $D_{xk} \in \mathbf{R}[x_k, u_k]^{n \times n}$, $D_{pk} \in \mathbf{R}[x_k, u_k]^{n \times n}$, and $D_{uk} \in \mathbf{R}[x_k, u_k]^{m \times m}$ are diagonal matrices of polynomials whose diagonal components are the denominators of f_k, $\nabla_x H_k$, and $\nabla_u H_k$, respectively. To prevent all the denominators from vanishing, we additionally consider the algebraic equations

$$1 - z_k \overline{d}_k(x_k, u_k) = 0, \tag{3.14}$$

where $d_k \in \mathbf{R}$ is an additional scalar variable and

$$\overline{d} \coloneqq (\det D_{xk})(\det D_{pk})(\det D_{uk})$$

is a polynomial. Since all the matrices D_{xk}, D_{pk}, and D_{uk} are diagonal, their determinants can be obtained just by computing the products of their diagonal components. Note that if (3.14) has a solution, all of $\det D_{xk}$, $\det D_{pk}$, and $\det D_{uk}$ are not zero, which means any denominator in (3.6)–(3.8) does not vanish at the solution; in other words, all the denominators do not vanish as long as (3.14) holds.

Next, since the right-hand side of (3.9) is a vector of algebraic functions, there exists a vector of polynomials $\rho_N(x_N, \nu, p_N) \in \mathbf{R}[x_N, \nu, p_N]^n$ such that

$$\rho_N(x_N, \nu, \nabla_x \phi + \nabla_x \psi \nu) = 0$$

holds for all $\nu \in \mathbf{R}^l$ and all x_N in the domains of $\nabla_x \phi$ and $\nabla_x \psi$. Finally, equation (3.10) remains to be converted. For each component $\psi_i \in \overline{\mathbf{R}(x_N)}$ of ψ, a polynomial $\Psi_i(x_N, z_i) \in \mathbf{R}[x_n, z_i]$ exists that satisfies $\Psi_i(x_N, \psi_i(x_N)) = 0$ for all points in the domain of ψ_i. The polynomial Ψ_i can be written in the following form.

$$\Psi_i(x_N, z_i) = z_i^{\alpha_i} d_{\psi i}(x_N, z_i) + \rho_{\psi i}(x_N),$$

where α_i is the minimum degree of z_i in $\Psi_i - \rho_{\psi i}$, and $d_{\psi i}$ and $\rho_{\psi i}$ are appropriate nonzero polynomials. If $z_i = \psi_i(x_N) = 0$ holds, then $\Psi_i(x_N, \psi(x_N)) = 0$ also holds from the definition of Ψ_i and, consequently,

$$\rho_{\psi i}(x_N) = 0. \tag{3.15}$$

Note that equation (3.15) is a necessary condition for the terminal constraint $\psi_i(x_N) = 0$ to hold. Therefore, we recast equation (3.10) into the form $\rho_\psi(x_N) = 0$ where $\mathbf{R}[x_N]^l \ni \rho_\psi(x_N) := [\rho_{\psi 1}(x_N) \cdots \rho_{\psi l}(x_N)]^\top$ by allowing some invalid solutions to appear.

Now, we obtain the following set of algebraic equations from the ELEs.

$$D_{xk}(x_k, u_k) x_{k+1} - n_{xk}(x_k, u_k) = 0, \tag{3.16}$$
$$D_{pk}(x_k, u_k) p_k - n_{pk}(x_k, u_k, p_{k+1}) = 0, \tag{3.17}$$
$$n_{uk}(x_k, u_k, p_{k+1}) = 0, \tag{3.18}$$
$$\rho_N(x_N, \nu, p_N) = 0, \tag{3.19}$$
$$\rho_\psi(x_N) = 0, \tag{3.20}$$
$$1 - z_k \overline{d}_k(x_k, u_k) = 0. \tag{3.21}$$

Equations (3.16)–(3.20) correspond to equations (3.6)–(3.10), and equation (3.21) means that all denominators in the ELEs must not vanish.

Even though equations (3.16)–(3.21) are algebraic equations, it is still difficult to solve them; one of the difficulties is that most of them depend on the variables at different time steps. To get rid of this inconvenient structure, we decouple (3.16)–(3.21) into sets of algebraic equations, where each equation set involves only the variables at a single time step and the Lagrange multiplier associated with the terminal constraint. Note that we have to ensure that the decoupled equations are satisfied by the optimal solutions, that is, the decoupled equations are necessary optimality conditions. This decoupling can be done by utilizing the mathematics of polynomials, i.e., commutative algebra and algebraic geometry.

3.3 Recursive Elimination Method for FHOCPs with terminal constraints

In this section, we introduce the recursive elimination method for decoupling the set of algebraic equations (3.16)–(3.21). Each decoupled set involves the variables at time step k, i.e., x_k, u_k, and p_k and the Lagrange multiplier ν, so that they can be solved independently when x_k is specified. Ohtsuka proposed such a method for rational FHOCP without terminal constraints and demonstrated its efficiency [17]. Here, we extend it to deal with the FHOCP with terminal constraints.

For the FHOCP with terminal constraints, the recursive elimination method is derived as Algorithm 1, which yields Theorem 3.1 and Corollary 3.1.

Theorem 3.1. Denote the optimal solution of FHOCP (3.1)–(3.4) by $\hat{x}_{[0:N]}$, $\hat{u}_{[0:N-1]}$, $\hat{p}_{[0:N]}$, and $\hat{\nu}$. Then, for the ideals J_k $(k = 0, \ldots, N)$ in Algorithm 1, the following

Algorithm 1 Recursive Elimination Method for FHOCP with Terminal Constraints

Input: Algebraic equations (3.16)–(3.21) for FHOCP with terminal constraints
Output: Sets of algebraic equations $F_k = 0$ $(k = 0, \ldots, N)$

1: Let F_N be set of polynomials consisting of left-hand sides of (3.19) and (3.20), and let $J_N := \langle F_N \rangle \subset \mathbf{R}[x_N, \nu, p_N], k := N - 1$
2: **while** $k \geq 0$ **do**
3: $\quad \bar{I}_k := \langle D_{xk} x_{k+1} - n_{xk}, D_{pk} p_k - n_{pk}, n_{uk}, 1 - z_k \overline{d}_k, F_{k+1} \rangle$
4: $\quad J_k := \bar{I}_k \cap \mathbf{R}[x_k, u_k, p_k, \nu]$
5: $\quad$ Let F_k be generators of ideal J_k, that is, $J_k = \langle F_k \rangle$
6: $\quad k \leftarrow k - 1$
7: **end while**

statements hold.

$$(\hat{x}_N, \hat{\nu}, \hat{p}_N) \in \mathcal{V}(J_N), \tag{3.22}$$

$$(\hat{x}_k, \hat{u}_k, \hat{p}_k, \hat{\nu}) \in \mathcal{V}(J_k) \quad (k = 0, \ldots, N-1). \tag{3.23}$$

That is,

$$F_N(\hat{x}_N, \hat{\nu}, \hat{p}_N) = 0, \tag{3.24}$$

$$F_k(\hat{x}_k, \hat{u}_k, \hat{p}_k, \hat{\nu}) = 0 \quad (k = 0, \ldots, N-1). \tag{3.25}$$

Proof. The proof is by induction. First, at $k = N$, the ideal J_N is generated by the left-hand sides of equations (3.19) and (3.20); thus, statement (3.22) holds. Suppose that $(\hat{x}_k, \hat{p}_k, \hat{u}_k, \hat{\nu}) \in \mathcal{V}(J_k)$ at time step k. From the definition of $\bar{I}_{k-1}$, we have

$$(\hat{x}_{k-1}, \hat{p}_{k-1}, \hat{u}_{k-1}, \hat{z}_{k-1}, \hat{x}_k, \hat{p}_k, \hat{u}_k, \hat{\nu}) \in \mathcal{V}(\bar{I}_{k-1}) \tag{3.26}$$

where $\hat{z}_{k-1} = 1/\bar{d}(\hat{x}_{k-1}, \hat{u}_{k-1})$. By projecting $\mathcal{V}(\bar{I}_{k-1})$ onto the $(x_{k-1}, p_{k-1}, u_{k-1}, \nu)$-space and by applying Lemma A.2,

$$(\hat{x}_{k-1}, \hat{p}_{k-1}, \hat{u}_{k-1}\hat{\nu}) \in \pi_{(x_{k-1}, p_{k-1}, u_{k-1}, \nu)}(\mathcal{V}(\bar{I}_{k-1})) \subset \mathcal{V}(J_{k-1}) \tag{3.27}$$

is obtained, and the proof is completed by induction. □

Corollary 3.1. Denote the optimal solution of FHOCP (3.1)–(3.4) by $\hat{x}_{[0:N]}$, $\hat{u}_{[0:N-1]}$, $\hat{p}_{[0:N]}$, and $\hat{\nu}$, and define ideals I_k, K_k, and W_k and their generators G_k^I, G_k^K, and G_k^W as

$$I_k = \langle G_k^I \rangle := J_k \cap \mathbf{R}[x_k, p_k],$$
$$K_k = \langle G_k^K \rangle := J_k \cap \mathbf{R}[x_k, u_k],$$
$$W_k = \langle G_k^W \rangle := J_k \cap \mathbf{R}[x_k, \nu].$$

Then, the following statements hold.

$$(\hat{x}_k, \hat{p}_k) \in \mathcal{V}(I_k),\ (\hat{x}_k, \hat{u}_k) \in \mathcal{V}(K_k),\ (\hat{x}_k, \hat{\nu}) \in \mathcal{V}(W_k).$$

Proof. Corollary follows readily from Theorem 3.1 and Lemma A.2. □

Theorem 3.1 shows that all sets of polynomials F_k must vanish at the optimal solutions; that is, the solutions of equations $F_k = 0$ can be regarded as candidates of the optimal solutions. Moreover, Corollary 3.1 shows that generators of ideal K_k vanish at the optimal values $\hat{x}_k$ and $\hat{u}_k$, and thus it indicates that the set of equations

$G_k^K(x_k, u_k) = 0$ is an implicit representation of the optimal feedback control law at time step k. Unlike the case without terminal constraints, we have to determine the value of an additional variable ν, that is, the Lagrange multiplier associated with the terminal constraint. The value of ν can be determined only when the value of the initial state $x_0 = \bar{x}$ is given, and thus ν can be implicitly represented by x_0. In Algorithm 1, ν is transferred from the polynomial set F_N to F_0, and we obtain the implicit representation of ν by x_0 as the set of algebraic equations $G_0^W = 0$ in Corollary 3.1.

By solving the algebraic equations $G_k^K(\tilde{x}_k, u_k) = 0$ for a given state value $\tilde{x}_k$, we can obtain the candidate values of the optimal input $\hat{u}_k$; they include the local optimal solutions if exist. In the following discussion, a candidate of the optimal value $\hat{y}$ of variable y is denoted by $\tilde{y}$. When equations $G_k^K(\tilde{x}_k, u_k) = 0$ have solutions $\tilde{u}_k^a$ and $\tilde{u}_k^b$, they may yield two different values of the next candidate state $\tilde{x}_{k+1}^a$ and $\tilde{x}_{k+1}^b$ from the state equation (3.2). These candidate state values in turn yields two different sets of equations $G_{k+1}^K(\tilde{x}_{k+1}^a, u_{k+1}) = 0$ and $G_{k+1}^K(\tilde{x}_{k+1}^b, u_{k+1}) = 0$, which may have different sets of candidate values of $\tilde{u}_{k+1}$. Consequently, the sequence of candidate inputs $\tilde{u}_{[0:N-1]}$ so to speak *branches out* at the moment of solving $G_k^K(\tilde{x}_k, u_k) = 0$ for every $k = 0, \ldots, N-1$. Therefore, a tree structure is suitable for expressing the candidate values, where its root is associated with the given initial value $\tilde{x}_0 = \bar{x}$ and each of its nodes is associated with a pair of candidate input $\tilde{u}_k$ and corresponding next state $\tilde{x}_{k+1}$.

Moreover, we can compute the value of the stage cost corresponding to a specific pair of candidates $\tilde{x}_k$ and $\tilde{u}_k$, and we can obtain a candidate value of the cost function corresponding to a sequence of candidate inputs by accumulating the candidate stage costs and the terminal cost. Eventually, each leaf of the tree has the candidate value of the cost function corresponding to the sequence of candidate inputs associated with its ancestors. Therefore, the global optimal solution, if it exists, can be obtained by finding the leaf having the minimum value of the cost function and collecting the candidate inputs associated with its ancestors.

In the following discussion, $T_{\bar{x}}$ denotes the tree associated with a given initial state $\bar{x}$, and for a node $v \in T_{\bar{x}}$, $l(v)$, $x(v)$, and $u(v)$ denote the accumulated stage cost, candidate state, and candidate input associated with the node, respectively. The procedure described in the previous paragraphs is then summarized as Algorithm 2. In this algorithm, the function CalcCands computes $\tilde{u}_k$ and $\tilde{x}_{k+1}$ from $x(v)$ for a given node v, and if the third argument k does not equal $N-1$, the function calls itself recursively. Therefore, by feeding the algorithm with the sequence of ideals $K_{[0:N-1]}$,

Algorithm 2 Recursive Computation of Candidate Values

Input: Sequence of ideals $K_{[0:N-1]}$, node v, time instance k
Output: Tree that has node v as its root

function CalcCands($K_{[0:N-1]}$, v, k)
  Let $\tilde{U}$ be $\mathcal{V}(G_k^K(x(v), u_k))$, where $\langle G_k^K \rangle = K_k \in K_{[0:N-1]}$
  for each $\tilde{u}_k \in \tilde{U}$ **do**
    if $\forall i \in \{1, \ldots, n\}, \bar{d}(x(v), \tilde{u}_k) \neq 0$ **then**
      $\tilde{x}_{k+1} := f_k(x(v), \tilde{u}_k)$
      Add new node v' as child of v
      $x(v') \leftarrow \tilde{x}_{k+1}$
      $u(v') \leftarrow \tilde{u}_k$
      if $k \neq N-1$ **then**
        $l(v') \leftarrow l(v) + L_k(x(v), \tilde{u}_k)$
        CalcCands($K_{[0:N-1]}$, v', $k+1$)
      else
        $l(v') \leftarrow l(v) + L_k(x(v), \tilde{u}_{N-1}) + \phi(\tilde{x}_N)$
      end if
    end if
  end for
  return Tree that has root v
end function

a root v_{root} such that $x(v_{\mathrm{root}}) = \bar{x}$, and the time instance $k = 0$, the algorithm yields $T_{\bar{x}}$ if the total number of candidate values is finite.

Note that the sequences of candidate inputs obtained by Algorithm 2 may include invalid solutions that do not satisfy the ELEs. Moreover, the solutions of the ELEs may include the one that is not locally optimal. Therefore, in the next section, we utilize the second-order sufficient conditions for local optimality to find locally optimal solutions from the candidates.

3.4 Sufficient Conditions for Optimality

Here, we introduce the second-order sufficient conditions for guaranteeing the local optimality of the solutions obtained by the recursive elimination method. These conditions are based on the sufficient conditions for local optimality in nonlinear programming [53], and their applicable range is wider than those of well-known sufficient conditions in FHOCPs with terminal constraints in [23, 50]. Moreover, the presented sufficient conditions also guarantee the uniqueness of the optimal solution in some neighborhood of a given initial state.

First, to simplify the notation, we introduce an auxiliary notation $a_{[s:t]}(x_{[s:t]})$ that denotes the sequence of the values $\{a_s(x_s), \ldots, a_t(x_t)\}$ for a sequence of functions $a_{[s:t]}(x)$ and a sequence of points $x_{[s:t]}$ having the same length. Moreover, we also introduce the matrix-valued functions:

$$Z_{ab}(S_{k+1}, k) := \frac{\partial^2 H_k}{\partial a \partial b} + \left(\frac{\partial f_k}{\partial a}\right)^\top S_{k+1} \frac{\partial f_k}{\partial b}, \tag{3.28}$$

where S_k are $n \times n$ matrices for $k = 0, \ldots, N$, symbols a and b can be replaced by symbols x and u, and $\partial/\partial a$ denotes the derivative with respect to the symbol replacing a. For example,

$$Z_{ux}(S_{k+1}, k) = \frac{\partial^2 H_k}{\partial u \partial x} + \left(\frac{\partial f_k}{\partial u}\right)^\top S_{k+1} \frac{\partial f_k}{\partial x}. \tag{3.29}$$

Using these matrix-valued functions, we make the following assumption in order to state the sufficient conditions.

Assumption 3.1. Sequences of states $\hat{x}_{[0:N]}$, inputs $\hat{u}_{[0:N-1]}$, $\hat{p}_{[k:N]}$, and $\hat{\nu}$ exist that satisfy ELEs (3.6)–(3.10), and the following matrix inequalities hold for $k = 0, \ldots, N-1$.

$$Z_{uu}(S_{k+1}, k) > 0, \tag{3.30}$$

where matrices $S_k \in \mathbf{R}^{n \times n}$ in the function Z_{uu} are determined by the recurrence formula:

$$S_k = Z_{xx}(S_{k+1}, k) - Z_{ux}^\top(S_{k+1}, k) Z_{uu}^{-1}(S_{k+1}, k) Z_{ux}(S_{k+1}, k) \tag{3.31}$$

and the boundary condition:

$$S_N = \frac{\partial^2 \phi}{\partial x^2} + \nu^\top \frac{\partial^2 \psi}{\partial x^2}. \tag{3.32}$$

In this assumption, the arguments of the partial derivatives of H_k, f_k, ϕ, and ψ are parts of sequences $\hat{x}_{[0:N]}$, $\hat{u}_{[0:N-1]}$, and $\hat{p}_{[0:N]}$ and $\hat{\nu}$.

From the viewpoint of nonlinear programming, it can be shown that ELEs (3.6)–(3.10) and matrix inequalities (3.30) are sufficient conditions for local optimality; this is a straightforward consequence of Lemma B.1 in Section B.1. On the basis of the sufficient conditions for local optimality, the existence and uniqueness of local optimal feedback laws can then be stated as the following theorem.

Theorem 3.2. Suppose that Assumption 3.1 holds and that terminal constraint (3.4) satisfies the linear independence constraint qualification (LICQ). Then, for the sequences $\hat{u}_{[0:N-1]}$ and $\hat{p}_{[0:N]}$ and the vector $\hat{\nu}$ whose existence is assumed in Assumption 3.1, unique sequences of differentiable functions $u^*_{[0:N-1]}(x_{[0:N-1]})$, $p^*_{[0:N]}(x_{[0:N]})$, and $\nu^*_{[0:N]}(x_{[0:N]})$ exist that are defined on some neighborhood of $\hat{x}_k$, and they satisfy $u^*_k(\hat{x}_k) = \hat{u}_k$, $p^*_k(\hat{x}_k) = \hat{p}_k$, and $\nu^*_k(\hat{x}_k) = \hat{\nu}$. Furthermore, some neighborhood of the given initial state $\hat{x}_0 = \bar{x}$ exists such that, for any initial state x_0^{CL} in the neighborhood, the closed-loop trajectory $x^{CL}_{[0:N]}$ generated by the state equation (3.2) and the feedback law $u^*_k(x_k^{CL})$, the sequence of costates given by $p^*_k\left(x_k^{CL}\right)$, and the Lagrange multipliers given by $\nu^*_k(x_k^{CL})$ satisfy the ELEs and inequalities (3.30). In other words, the set of differentiable functions $u^*_{[0:N-1]}(x^{CL}_{[0:N-1]})$, $p^*_{[0:N]}(x^{CL}_{[0:N]})$, and $\nu^*_{[0:N]}(x^{CL}_{[0:N]})$ gives a local optimal solution for x_0^{CL} in some neighborhood of $\bar{x}$.

Proof. For $l = 0, \ldots, N-1$, the subsequences $\hat{x}_{[l:N]}$, $\hat{u}_{[l:N-1]}$, $\hat{p}_{[l:N]}$, and $\hat{\nu}$ satisfy the ELEs and inequalities (3.30). From Lemma B.1 in Section B.1, a unique set of differentiable functions $\mathbf{x}_l^{[l+1:N]}(x_l)$, $\mathbf{u}_l^{[l:N-1]}(x_l)$, $\mathbf{p}_l^{[l:N]}(x_l)$, and $\boldsymbol{\nu}_l(x_l)$ exists that satisfy $\mathbf{x}_l^k(\hat{x}_l) = \hat{x}_k$, $\mathbf{u}_l^k(\hat{x}_l) = \hat{u}_k$, $\mathbf{p}_l^k(\hat{x}_l) = \hat{p}_k$, $\boldsymbol{\nu}_l(x_l) = \hat{\nu}$, the ELEs, and inequalities (3.30) for any x_l in some neighborhood of $\hat{x}_l$. We define $\mathbf{x}_l^l(x_l) := x_l$; then, the uniqueness of these functions implies that

$$\mathbf{x}_{l_1}^k(\mathbf{x}_l^{l_1}(x_l)) = \mathbf{x}_{l_2}^k(\mathbf{x}_l^{l_2}(x_l)), \tag{3.33}$$

$$\mathbf{u}_{l_1}^k(\mathbf{x}_l^{l_1}(x_l)) = \mathbf{u}_{l_2}^k(\mathbf{x}_l^{l_2}(x_l)), \tag{3.34}$$

$$\mathbf{p}_{l_1}^k(\mathbf{x}_l^{l_1}(x_l)) = \mathbf{p}_{l_2}^k(\mathbf{x}_l^{l_2}(x_l)), \tag{3.35}$$

$$\boldsymbol{\nu}_{l_1}(\mathbf{x}_l^{l_1}(x_l)) = \boldsymbol{\nu}_{l_2}(\mathbf{x}_l^{l_2}(x_l)), \tag{3.36}$$

hold for any $l_1, l_2 \in \{l, \ldots, k\}$ and for any x_l in some neighborhood of $\hat{x}_l$.

Now, let us define $u^*_k(x_k) := \mathbf{u}_k^k(x_k)$ and $f_k^{CL}(x_k) := f_k\left(x_k, u^*_k(x_k)\right)$ for $k = 0, \ldots, N-1$. Since $\mathbf{x}_k^{k+1}(x_k)$ maps $\hat{x}_k$ onto $\hat{x}_{k+1}$ and

$$\hat{x}_{k+1} = f_k\left(\hat{x}_k, u^*_k(\hat{x}_k)\right) = f_k^{CL}(\hat{x}_k),$$

the uniqueness of $\mathbf{x}_k^{k+1}(x_k)$ implies $\mathbf{x}_k^{k+1}(x_k) = f_k^{CL}(x_k)$ for $k = 0, \ldots, N-1$ and for any x_k in some neighborhood of $\hat{x}_k$. Accordingly, we obtain the following equations for $k = 0, \ldots, N-1$ and for any x_0 in some neighborhood of $\hat{x}_0 = \bar{x}$.

$$\mathbf{x}_0^k(x_0) = \mathbf{x}_{k-1}^k \circ \cdots \circ \mathbf{x}_0^1(x_0) \tag{3.37}$$

$$= f_{k-1}^{CL} \circ \cdots \circ f_0^{CL}(x_0). \tag{3.38}$$

That is, sequence $\mathbf{x}_0^{[0:N]}(x_0)$ is the closed-loop trajectory given by the sequence of feedback control laws $u^*_{[0:N-1]}(x)$, and thus $x_k^{CL} = \mathbf{x}_0^k(x_0)$. Similarly, we define $p_k^*(x_k) := \mathbf{p}_k^k(x_k)$ and $\nu_k^*(x_k) := \boldsymbol{\nu}_k(x_k)$; then,

$$u_k^*(x_k^{CL}) = \mathbf{u}_k^k(\mathbf{x}_0^k(x_0)) = \mathbf{u}_0^k(x_0), \tag{3.39}$$

$$p_k^*(x_k^{CL}) = \mathbf{p}_k^k(\mathbf{x}_0^k(x_0)) = \mathbf{p}_0^k(x_0), \tag{3.40}$$

$$\nu_k^*(x_k^{CL}) = \boldsymbol{\nu}_k(\mathbf{x}_0^k(x_0)) = \boldsymbol{\nu}_0(x_0), \tag{3.41}$$

hold from the definitions of u_k^*, p_k^*, and ν_k^* and equations (3.34)–(3.36). Therefore, the closed-loop trajectory $x^{CL}_{[0:N]}$ and corresponding inputs $u_k^*(x_k^{CL})$, costates $p_k^*(x_k^{CL})$, and Lagrange multiplier $\nu_k^*(x_k^{CL})$ are identical to the sequences $\mathbf{x}_0^k(x_0)$, $\mathbf{u}_0^k(x_0)$, and $\mathbf{p}_0^k(x_0)$ and the Lagrange multiplier $\boldsymbol{\nu}_0(x_0)$, which satisfy the ELEs and inequalities (3.30). □

Note that Theorem 3.2 is not a straightforward extension of the result in [17] because of the existence of an additional variable ν. The result in [17] corresponding to Theorem 3.2 is based on sufficient conditions for optimality of FHOCPs without constraints, and these conditions cannot be applied to FHOCPs with terminal constraints. For such constrained FHOCPs, sufficient conditions for optimality have already been proposed [23, 50]. However, they assume that ν is explicitly represented by each state x_k, whence the applicable range of their conditions is limited. We provide less conservative conditions (Lemma B.1) by getting rid of that assumption and prove Theorem 3.2 by using the lemma.

We can find local optimal solutions from the sequences of candidate inputs obtained by Algorithm 2 by checking whether each candidate satisfies the ELEs and inequalities (3.30). The uniqueness and differentiability of the optimal solution in some neighborhood also enables us to track the optimal solution from that corresponding to one state value to that corresponding to another; we can find a solution of $G_k^K(x_k, u_k) = 0$ with $x_k \neq \hat{x}_k$ from the optimal solutions $\hat{x}_k$ and $\hat{u}_k$ by using numerical computations such as Newton's method. Moreover, we can characterize the optimal feedback laws, costates, and Lagrange multiplier in Theorem 3.2 as algebraic functions of the state by making assumptions on the ideals obtained in Corollary 3.1, which is the topic of the next section.

3.5 Existence of Algebraic Solutions

When $u_k^*(x_k)$, $p_k^*(x_k)$, and $\nu_k^*(x_k)$ are algebraic functions of x_k, each of their components can be regarded as a *point* in $\overline{\mathbf{R}(x_k)}$. Therefore, $u_k^*(x_k)$ can be characterized as

a point of the algebraic set in $\overline{\mathbf{R}(x_k)}^m$ defined by polynomials in $\mathbf{R}(x_k)[u_k]$ instead of $\mathbf{R}[x_k, u_k]$, which leads to the extension $K_k^e \subset \mathbf{R}(x_k)[u_k]$ of the ideal $K_k \subset \mathbf{R}[x_k, u_k]$; the same argument applies to p_k^* and ν_k^*. In particular, if the extension K_k^e is zero-dimensional, a basis of its radical can be computed [54]. Zero-dimensional radical ideals have good properties stated in the following lemma [17].

Lemma 3.1. If $J \subset \mathbf{K}(X)[Y]$ is a zero-dimensional radical ideal, it has exactly n generators $g_1, \ldots, g_n \in \mathbf{K}(X)[Y]$, and $\det[\partial g_i(Y)/\partial Y_j] \neq 0$ for every $Y \in \mathcal{V}(J)$.

Lemma 3.1 shows that a set of generators $g_1, \ldots, g_m$ exists such that the equations $g_i = 0$ $(i = 1, \ldots, m)$ form a *square system of polynomial equations* in the variables Y whose Jacobian at each $Y \in \mathcal{V}(J)$ is a nonzero algebraic function of X; that is, the Jacobian does not vanish in an open and dense subset of X-space. Moreover, under additional mild assumptions, the Gröbner basis of a zero-dimensional radical ideal has a good structure for computing the generators in Lemma 3.1, which is known as the Shape Lemma [55].

Theorem 3.3. Suppose that Assumption 3.1 holds, that the terminal constraint (3.4) satisfies the linear independence constraint qualification, and that the extensions of the ideals $K_k^e \subset \mathbf{R}(x_k)[u_k]$, $I_k^e \subset \mathbf{R}(x_k)[p_k]$, and $W_k^e \subset \mathbf{R}(x_k)[\nu]$ are zero-dimensional ideals for $k = 0, \ldots, N-1$. Then, the optimal feedback laws $u_k^*(x_k)$, costates $p_k^*(x_k)$, and Lagrange multiplier $\nu_k^*(x_k)$, whose existences are guaranteed by Theorem 3.2, are algebraic functions, and $u_k^* \in \mathcal{V}(K_k^e)$, $p_k^* \in \mathcal{V}(I_k^e)$, and $\nu_k^* \in \mathcal{V}(W_k^e)$. Moreover, a set of generators of each ideal exists that form a square system of polynomial equations, and its Jacobian matrix is nonsingular for x_k in an open and dense subset of some neighborhood of $\hat{x}_k$.

Proof. Theorem 3.2 shows that the set of $u_k^*(\tilde{x}_k)$, $p_k^*(\tilde{x}_k)$, and $\nu_k^*(\tilde{x}_k)$ is an optimal solution for $\tilde{x}_k$ in some neighborhood of $\hat{x_k}$. Therefore, Corollary 3.1 implies that $(\tilde{x}_k, u_k^*(\tilde{x}_k)) \in \mathcal{V}(K_k)$, $(\tilde{x}_k, p_k^*(\tilde{x}_k)) \in \mathcal{V}(I_k)$, and $(\tilde{x}_k, \nu_k^*(\tilde{x}_k)) \in \mathcal{V}(W_k)$. Since K_k^e at each time instant k is supposed to be a zero-dimensional ideal, a minimal polynomial $h_i(x_k, u_{ki}) \in \mathbf{R}(x_k)[u_{ki}]$ for each $i = 1, \ldots, m$ exists that vanishes on $\mathcal{V}(K_k^e)$. Note that, from the definition of extensions of ideals, we can choose h_i from K_k instead of K_k^e by multiplying some polynomial in $\mathbf{R}[x_k]$. Therefore, $(\tilde{x}_k, u_k^*(\tilde{x}_k)) \in \mathcal{V}(K_k)$ for all $\tilde{x}_k$ in some neighborhood of $\hat{x}_k$ implies that $h_i\left(x_k, u_{ki}^*(x_k)\right) = 0$ in the neighborhood, and thus, each component $u_{ki}^*(x_k)$ is an algebraic function of x_k. Moreover, $(x_k, u_k^*(x_k)) \in \mathcal{V}(K_k)$ implies that every polynomial in K_k vanishes at $u_k^* \in \overline{\mathbf{R}(x_k)}^m$, and thus, all polynomials in K_k^e also vanish at u_k^*, which implies $u_k^* \in \mathcal{V}(K_k^e)$.

Now, Lemma 3.1 guarantees that the radical $\sqrt{K_k^e}$ has exactly m generators as $\sqrt{K_k^e} = \langle \bar{G}_k^K \rangle$ such that $\det\left[\partial \bar{G}_k^K / \partial u_k^*\right] \in \overline{\mathbf{R}(x_k)}$ is a nonzero algebraic function of x_k, which implies that $\partial \bar{G}_k^K(x_k, u_k^*(x_k))/\partial u_k \in \mathbf{R}^{m\times m}$ is nonsingular in an open and dense subset of some neighborhood of $\hat{x}_k$. The same argument can also be applied to p_k^* and ν_k^*, thus completing the proof. □

Theorem 3.3 shows that we can guarantee the differentiability and nonsingularity of the algebraic state feedback laws in an open and dense subset of some neighborhood of $\hat{x}_k$ by using the notion of zero-dimensional ideals. When we find $u_k^*(\tilde{x}_k)$ for $\tilde{x}_k$ in the neighborhood, nonsingularity helps us to solve equations $\bar{G}_k^K(\tilde{x}_k, u_k) = 0$ numerically by using, for example, Newton's method or the continuation method. Moreover, we can use the minimal polynomials $h_i(x_k, u_{ki})$ of u_{ki} with respect to K_k^e to compute $u_k^*(x_k)$. In this case, we can compute $u_{ki}^*(\tilde{x}_k)$ for $\tilde{x}_k$ by solving the univariate algebraic equation $h_i(\tilde{x}_k, u_{ki}) = 0$. Note that the computation of $u_{ki}^*(\tilde{x}_k)$ can be performed independently for each $i \in \{1, \ldots, m\}$.

3.6 Numerical Examples

3.6.1 Illustrative Example

Let us consider the following FHOCP with a terminal constraint.

$$\min_{u_1,\ldots,u_{N-1}} \quad \sum_{k=0}^{N-1} \frac{1}{2} u_k^2, \tag{3.42}$$

subject to

$$\begin{cases} x_{k+1,1} = x_{k,2} \\ x_{k+1,2} = -x_{k,1} + \dfrac{x_{k,1} u_k}{1 + x_{k,1}}, \end{cases} \tag{3.43}$$

$$x_0 = \bar{x}, \tag{3.44}$$

$$x_N = 0. \tag{3.45}$$

For the optimization horizon $N = 4$, we can obtain the polynomials F_4 and F_3 as

$$F_4 = \{x_{41}, x_{42}, p_{41} - \nu_1, p_{42} - \nu_2\},$$
$$F_3 = \{x_{32}, p_{32} - \nu_1, p_{31}x_{31} - x_{31}^2 - x_{31}, u_3^2 - u_3 - x_{31}^2 - x_{31}, \cdots\}.$$

Some of the polynomials in F_3 and all of the polynomials F_2, F_1, and F_0 are omitted because of space limitations. The ideals $K_k \subset \mathbf{R}[x_k, u_k]$ are obtained from Corol-

lary 3.1 as

$$
\begin{aligned}
K_3 &= \langle x_{32}, u_3 x_{31} - x_{31}^2 - x_{31}, u_3^2 - x_{31}^2 - u_3 - x_{31} \rangle, \\
K_2 &= \langle u_2 x_{21} - x_{21}^2 - x_{21}, u_2^2 - x_{21}^2 - u_2 - x_{21} \rangle, \\
K_1 &= \langle 2u_1^2 x_{11}^2 - 3u_1 x_{11}^3 + x_{11}^4 + \cdots \rangle, \\
K_0 &= \langle 2u_0^2 x_{01}^2 - 3u_0 x_{01}^3 + x_{01}^4 + \cdots \rangle.
\end{aligned}
$$

By using Algorithm 2, we obtain 36 candidate solutions for a given initial state $\bar{x} = [-2.0, 3.0]^\top$, and four of them satisfy the ELEs.

Figure 3.1 shows the state trajectories T_1, T_2, T_3, and T_4 given by these four sequences. The values of the cost functions corresponding to T_1, T_2, T_3, and T_4 are 1.78, 8.50, 8.90, and 2.18. Since the dimension of the input is 1, the matrices $Z_{uu}(S_{k+1}, k)$ in inequalities (3.30) are scalars. Table 3.1 shows the corresponding values of $Z_{uu}(S_{k+1}, k)$ for each trajectory. All of the values of Z_{uu} for T_1 are positive; hence, the sequence of candidate inputs giving T_1 is a locally optimal solution.

The ideals K_1 and K_0 are generated by only one polynomial in $\mathbf{R}[x_k, u_k]$, and their extensions are also generated by these polynomials. Therefore, from Lemma A.1, K_1^e and K_0^e are zero-dimensional ideals. Moreover, generators of these ideals are both square-free, which implies that these ideals are also radical. Then, the optimal state feedback laws can be obtained as the roots of these generators, and we can calculate the optimal feedback laws explicitly because the degrees of the generators are at most 2. The following algebraic feedback laws are the roots of the generators of K_1^e and

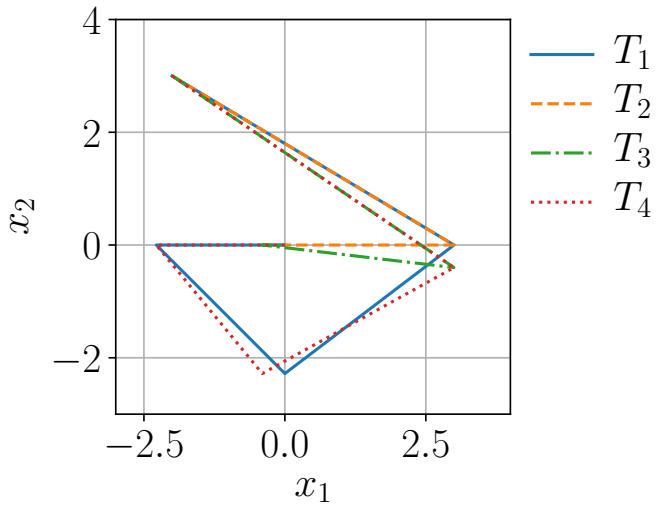


Figure 3.1: State trajectories given by candidate inputs.

Table 3.1: Values of $Z_{uu}(S_{k+1}, k)$

	$k=0$	$k=1$	$k=2$	$k=3$
T_1	3.84	0.892	1.00	1.00
T_2	16.0	−15.0	1.00	1.00
T_3	−157	−18.6	1.00	1.00
T_4	−56.0	1.12	1.00	1.00
T_1'	4.37	0.902	1.00	1.00

K_0^e and give T_1.

$$u_0^*(x_0) = x_{01} + 1, \tag{3.46}$$

$$u_1^*(x_1) = \frac{x_{11}(x_{11}^2 - 1)}{2x_{11}^2 + 2x_{11} + 1}, \tag{3.47}$$

Note that the other inputs u_2 and u_3 are uniquely determined from only state equation (3.43) and terminal constraint (3.45). Note also that these optimal state feedback laws are calculated without any approximations. Theorem 3.2 guarantees that, for all x_0 in some neighborhood of $\bar{x} = [-2, 3]^\top$, these algebraic feedback laws are also optimal. Indeed, for an initial state $\bar{x}' = [-2.4, 3.3]^\top \simeq \bar{x}$, we obtain the trajectory T_1' that corresponds to the values of $Z_{uu}(S_{k+1}, k)$ in Table 3.1, which implies that the sequence of inputs giving T_1' is also a locally optimal solution.

3.6.2 Nonlinear Model Predictive Control Application

In practice, FHOCPs are often accompanied by NMPC, in which an FHOCP is solved repeatedly and only the initial optimal input is utilized as the actual input. By solving the set of algebraic equations $G_0^K(x_0, u_0) = 0$ obtained by Algorithm 1, we can obtain candidates of the initial optimal input immediately from the given initial state. Moreover, in the case of NMPC with a short enough sampling period, the state variation within the sampling interval should be small. Therefore, from Theorem 3.2, the solution of each FHOCP possibly retains local optimality.

As a practical example, let us examine stabilization of an inverted pendulum by NMPC. We define the continuous-time equation of motion of the pendulum as

$$\ddot{\theta}(t) + \dot{\theta}(t) - \sin\theta(t) = u(t), \tag{3.48}$$

where $t \in \mathbf{R}_+ = [0, \infty)$ denotes continuous time, $\theta(t) \in \mathbf{R}$ denotes the angle subtended by the pendulum and the vertical axis, and $u(t) \in \mathbf{R}$ denotes the controllable

external torque. To apply the proposed method, we replace $\sin\theta$ in (3.48) with $(\theta - 7/60\theta^3)/(1 + 1/20\theta^2)$ by using Padé approximation, which often gives a better approximation of a function than one given by a truncated Taylor series and may still work outside of the convergence region of the Taylor series [56]. By using the forward difference approximation with a sampling period of $\Delta t = 0.05$, the discrete-time model can be obtained as

$$\begin{cases} x_{k+1,1} = x_{k,1} + x_{k,2}\Delta t \\ x_{k+1,2} = x_{k,2} + \left(-x_{k,2} + \dfrac{x_{k,1} - \frac{7}{60}x_{k,1}^3}{1 + \frac{1}{20}x_{k,1}^2} + u_k \right) \Delta t, \end{cases} \tag{3.49}$$

where $x_{k,1}$ and $x_{k,2}$ denote the angle and angular velocity, respectively, and u_k denotes the controllable external torque. The control objective is to regulate the state to the origin and reduce control burdens. Hence, the stage costs and terminal constraint are defined as

$$L_k(x_k, u_k) = \frac{1}{2}u_k^2, \quad \psi(x_N) = x_N = 0. \tag{3.50}$$

Since this terminal constraint specifies the terminal state, the terminal cost is omitted ($\phi(x_N) \equiv 0$).

For $N = 2$, G_0^K consists of only one polynomial:

$$\kappa_0(x_0, u_0) = 3(x_{01}^2 + 20)u_0 + 1193x_{01}^3 + 117x_{01}^2x_{02} + 24060x_{01} + 2340x_{02}. \tag{3.51}$$

Since $\kappa_0(x_0, u_0)$ is linear in u_0, the root of $\kappa_0(x_0, u_0)$ as a polynomial of u_0 can be computed symbolically, and this root is an explicit form of the feedback control law at $k = 0$. Moreover, under the problem setting of this example, both the scalars $Z_{uu}(S_1, 0)$ and $Z_{uu}(S_2, 1)$ in inequalities (3.30) are equal to one irrespective of the initial state, which means that the obtained feedback control law is locally optimal for an arbitrary initial state by Theorem 3.2.

Figure 3.2 shows the trajectories of the system (3.48) with the feedback control law defined by $\kappa_0(x(t), u(t)) = 0$ and the free response without the control law. Each arrow in the figure shows the direction that each trajectory goes toward. The initial state is set to $x_0 = [1.0\ 0.0]^\top$. In the controlled case, the state and control input are sampled and computed with a sampling period of 0.01 time units; the sampling period of NMPC is smaller than that of the discretization. It is readily seen that the state of the controlled system is regulated to the origin.

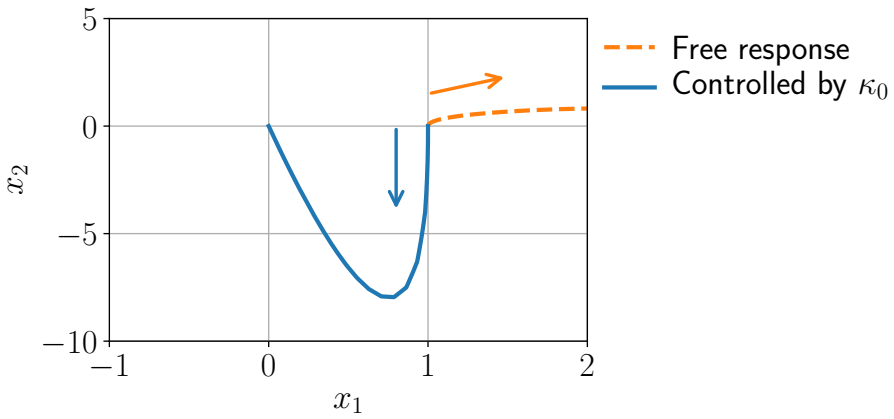


Figure 3.2: Trajectories of system (3.48) derived from controlled system (solid line) and its free response (dashed line).

3.7 Summary

We proposed a recursive elimination method for rational FHOCPs with terminal constraints. By applying the concept of elimination ideals from commutative algebra, the method decouples the ELEs into sets of algebraic equations, where each equation set involves the Lagrange multiplier associated with the terminal constraint and the variables at a single time instant. We also proposed an algorithm to solve the sets of algebraic equations and obtain candidates of local optimal solutions. Sufficient conditions for optimality are provided, which guarantee not only the optimality but also the uniqueness of the optimal solution in some neighborhood of the given initial state. By checking whether the presented sufficient conditions are satisfied or not for each candidate, we can select local optimal solutions from the candidates. Moreover, we also provided sufficient conditions for the existence of the optimal solutions in the form of algebraic functions of the states. By utilizing the concept of zero-dimensional ideals and some additional assumptions, we can guarantee the nonsingularity of the optimal solutions in some neighborhood, which is a useful property for numerical computations. In the future, we plan to establish sufficient conditions for optimality using only the information at the initial time instant, i.e., the value of the initial state, which is suitable for model predictive control.

Chapter 4

Recursive Elimination Method for Moving Horizon Estimation

4.1 Introduction

After the great success of the Kalman filter (KF) [4,5] in both theoretical and practical aspects, the study of optimal filtering has been developed to consider the problem setting of nonlinear systems with non-Gaussian noise, which is one of the most important problem settings in control engineering [57–59].

The most popular result in the optimal filtering theory for nonlinear systems is the extended Kalman filter (EKF) [26], which uses linear approximations to extend the concept of the KF to nonlinear cases. Its algorithm is simple and analogous to that of the KF, which not only makes it easy for us to imagine how the EKF works but also leads to a small computational cost. However, the use of linear approximations in its algorithm limits the accuracy of the EKF so that it can have surprisingly bad performance in some practical applications with nonlinear systems and non-Gaussian noise [60]. Another extension of the KF is the unscented Kalman filter (UKF) [28]. This filtering method uses the unscented transformation to approximate the distribution distorted by nonlinear systems with the Gaussian distribution whose mean and covariance are the same as those of the distorted one. Therefore, the UKF often provides more accurate estimates than the EKF (see, e.g., [61]). However, both EKF and UKF require the assumption that the system and measurement noises are Gaussian.

In some applications, such an assumption does not hold. Chen and Hu [62] have proposed a new Kalman-like filtering method for nonlinear systems, which does not require any statistical information of noise but only its bounds. They constructed an upper bound of estimation error by using a Taylor series approximation and computed

the gain of the filter that minimizes the upper bound. In some numerical examples, this new filtering method outperformed the EKF, UKF, and cubature KF (CKF) with inaccurate covariances of system and measurement noises. It is often difficult to certify that the noises are Gaussian, and this method is promising for such cases. However, the bounds of noise are sometimes unavailable when, for example, the noise has an impulsive character that produces deviations of high amplitude with short durations more often than the Gaussian distribution does [63]. It is known that radar and sonar noise [64] and disturbances caused by air turbulence [65] have the impulsive character, which has motivated several researchers to model such impulsive noise by using non-Gaussian distributions [63, 64, 66, 67]. With this in mind, it is still important to study optimal filtering methods for nonlinear systems with non-Gaussian noise even if we assume that their statistical properties or distributions are known.

As a popular approach for nonlinear filtering with non-Gaussian noise, the particle filter (PF) was first proposed by Kitagawa [31,32] and Gordon et al. [33]. This filter uses the Monte Carlo method to approximate the posterior distribution of the state, and thus Gaussian assumptions for noise distributions are not required. Although the PF with enough particles yields more accurate estimates than those derived from the EKF and UKF, it usually has a high computational cost, which can reach an unacceptable level for real-time purposes.

Another promising approach is called moving horizon estimation (MHE). In MHE, an estimation of the finite length of state trajectory is performed at every sampling time by using the same length of measurement sequence, and the length of the estimated state trajectory is called the horizon. While the EKF, UKF, and other Kalman-like filters estimate the current state by using the fixed previous estimate, MHE reoptimizes the past estimates on the horizon and estimates the current state by using those reoptimized past estimates. Indeed, it has been reported that this reoptimization property improves the current state estimate in some examples [68]. The estimation problem at each sampling time is formulated as a nonlinear optimization problem, which is usually computationally demanding to solve while the sampling interval is limited. Therefore, it is important to reduce its computational time.

In most of the efficient algorithms for MHE [69–72], the objective function minimized in each optimization problem is assumed to be quadratic, which is equivalent to the Gaussian assumption for noise distributions. For non-Gaussian cases, several methods have been developed to approximate the distributions of non-Gaussian noise by using simple distributions such as Gaussian or uniform distributions [25].

Monin [57] proposed an MHE algorithm for non-Gaussian cases by approximating the non-Gaussian distributions with the Gaussian mixture defined by the Max operator. Although these approximating methods can deal with a wide range of non-Gaussian noise, the computational complexity grows combinatorially with increasing the number of simple distributions used in the approximation. Hence, it gets more difficult to reduce the computational time for solving each optimization problem when we consider non-Gaussian cases.

To tackle such difficulties of MHE with nonlinear systems and non-Gaussian noise, we fucus on the similarity of the MHE procedure to model predictive control (MPC) in nonlinear control theory. Indeed, if we choose the joint maximum a posteriori (MAP) estimation as the optimality criterion, the derived optimization problem has a structure similar to that of an optimal control problem with a fixed horizon [23,25,73]. Hence, the same technique as introduced in Chapter 3 would also be applied to solving the joint MAP estimation.

Specifically, the analogy between MHE and MPC has led us to propose a recursive elimination method for optimal filtering problems. In the joint MAP estimation, the state trajectory on the horizon is computed to maximize the joint probability density of all the states conditionally on the measurements. However, in the MHE procedure, only the estimate of the current state is used as a solution for the filtering problem at each time step, which means there is no need to compute the other past estimates. Hence, by using the off-line variable elimination technique in the joint MAP estimation, we can eliminate the past state variables from the filtering problem to reduce the on-line computational time.

The rest of this chapter is organized as follows. In Section 4.2, we formulate the problems dealt with in this chapter, derive the stationary conditions, and convert them into algebraic equations by introducing additional variables. Section 4.3 is devoted to introducing a recursive elimination method for optimal filtering problems. In Section 4.4, an MHE algorithm that uses the implicit function representation derived from the recursive elimination is proposed. In Section 4.5, a numerical example is provided to compare the proposed MHE algorithm with other state estimation methods and show the efficiency of the proposed method. Section 4.6 summarize this chapter.

4.2 Problem Formulation

Consider the finite-horizon optimal estimation problem (FHOEP) (2.16).

$$x^*_{[0:N]} := \arg\max_{x_{[0:N]}} p(x_{[0:N]} \mid y_{[0:N]}) \tag{4.1}$$

$$\text{s. t.} \quad x_k = f_{k-1}(x_{k-1}, u_{k-1}) + w_{k-1}, \tag{4.2}$$

$$y_k = h_k(x_k) + v_k, \tag{4.3}$$

$$w_k \sim p_w(w_k), \tag{4.4}$$

$$v_k \sim p_v(v_k), \tag{4.5}$$

$$x_0 \sim p_0(x_0, \mu), \tag{4.6}$$

where $\mu \in \mathbf{R}^l$, which is not included in (2.16), is a vector of parameters that describes the time evolution of the PDF of the initial state on the horizon; in the MHE context, this PDF corresponds to the so-called arrival cost [69]. In this problem, although functions f_k, h_k, p_w, and p_v are assumed to be given, p_0 is what we have to choose. Typically, p_0 is chosen as Gaussian with the mean of the past estimate, which corresponds to the quadratic arrival cost [69, 71]. In this case, μ is a parameter denoting the past estimate. For a more reasonable design method, we can take into account the state equation (4.2) and the PDF of system noise (4.4); p_0 can be defined as

$$p_0(x_0, \mu) := p_w(x_0 - f_0(\mu_1, \mu_2)),$$

where $\mu = \begin{bmatrix} \mu_1^\top & \mu_2^\top \end{bmatrix}^\top \in \mathbf{R}^{n+m}$ is the parameter representing the state and input at $N+1$ real-time steps ago. This would be a better choice than the typical one if a Gaussian distribution cannot approximate the PDF of system noise p_w.

As stated in Section 2.3, FHOEP (4.1)–(4.6) can be reformulated into an unconstrained nonlinear optimization problem

$$\min_{x_{[0:N]}} \; -\Big\{\ln p_0(x_0, \mu) + \ln p_v\left(y_0 - h_0(x_0)\right)\Big\} - \sum_{k=1}^{N} \Big\{\ln p_w\left(x_k - f_{k-1}(x_{k-1}, u_{k-1})\right) + \ln p_v\left(y_k - h_k(x_k)\right)\Big\}. \tag{4.7}$$

By differentiating the objective function with respect to the state trajectory, the

stationary conditions for the minimization problem (4.7) can be obtained as follows:

$$-\frac{1}{p_0}\frac{\partial p_0}{\partial x_0}+\frac{1}{p_v}\frac{\partial p_v}{\partial v_0}\bigg|_{v_0=y_0-h_0}\frac{\partial h_0}{\partial x_0}+\frac{1}{p_w}\frac{\partial p_w}{\partial w_0}\bigg|_{w_0=x_1-f_0}\frac{\partial f_0}{\partial x_0}=0, \tag{4.8}$$

$$\begin{aligned}-\frac{1}{p_w}\frac{\partial p_w}{\partial w_{k-1}}\bigg|_{w_{k-1}=x_k-f_{k-1}}&+\frac{1}{p_v}\frac{\partial p_v}{\partial v_k}\bigg|_{v_k=y_k-h_k}\frac{\partial h_k}{\partial x_k}\\&+\frac{1}{p_w}\frac{\partial p_w}{\partial w_k}\bigg|_{w_k=x_{k+1}-f_k}\frac{\partial f_k}{\partial x_k}=0\qquad(k=1,\dots,N-1),\end{aligned} \tag{4.9}$$

$$-\frac{1}{p_w}\frac{\partial p_w}{\partial w_{N-1}}\bigg|_{w_{N-1}=x_N-f_{N-1}}+\frac{1}{p_v}\frac{\partial p_v}{\partial v_N}\bigg|_{v_N=y_N-h_N}\frac{\partial h_N}{\partial x_N}=0, \tag{4.10}$$

where the arguments of all functions are omitted for simplicity. We assume that the unconstrained problem (4.7) has at least one local minimizer $x^*_{[0:N]}\in\mathbf{R}^{n\times(N+1)}$ with a finite minimum value of the cost function, and thus (4.8)–(4.10) are necessary conditions for optimality. Note that when the parameter μ and the histories of inputs $u_{[0:N-1]}$ and measurements $y_{[0:N]}$ are given, (4.8)–(4.10) can be viewed as a two-point boundary value problem for the sequence of tuples (x_k,x_{k-1}) $(k=1,\dots,N)$ because (4.8) and (4.10) can be regarded as the initial and the terminal conditions, respectively.

Now, we assume that functions $f_k:\mathbf{R}^n\times\mathbf{R}^m\to\mathbf{R}^p$ and $h_k:\mathbf{R}^n\to\mathbf{R}^p$ in the state and observation equations (4.2) and (4.3), respectively, are vectors of rational functions. In addition, we also assume that the derivatives of logarithms

$$\begin{aligned}\frac{\partial}{\partial w_k}\{\log p_w(w_k)\}&=\frac{1}{p_w(w_k)}\frac{\partial p_w}{\partial w_k}(w_k),\\ \frac{\partial}{\partial v_k}\{\log p_v(v_k)\}&=\frac{1}{p_v(v_k)}\frac{\partial p_v}{\partial v_k}(v_k),\\ \frac{\partial}{\partial x_0}\{\log p_0(x_0,\mu)\}&=\frac{1}{p_0(x_0,\mu)}\frac{\partial p_0}{\partial x_0}(x_0,\mu)\end{aligned} \tag{4.11}$$

are described by rational functions. Under these assumptions, all the left-hand sides of (4.8) and (4.10) consist of rational functions.

Remark 4.1. The assumption that the derivatives (4.11) consist of rational functions is not so restrictive. For example, if p_w is described as an exponential of a rational function g,

$$p_w(w_k)=\exp(g(w_k)),$$

then its logarithm g and its derivative of the logarithm $\partial g/\partial w_k$ are both rational functions. Moreover, even if p_w itself is a rational function g, its derivative of the logarithm $\partial(\log g)/\partial w_k=(\partial g/\partial w_k)/g$ is a rational function though its logarithm $\log g$ is not.

Remark 4.2. Although the unconstrained problem (4.7) is considered in this chapter, equality and inequality constraints can be taken into account if they consist of rational functions. In this case, the stationary conditions (4.8)–(4.10) are replaced by the Karush-Kuhn-Tucker (KKT) conditions, and thus we can apply the following discussions and algorithms by using some minor modifications to deal with the Lagrange multipliers associated with the additional constraints. Note, however, that a global minimizer of the constrained optimization problem may no longer be interpreted as a MAP estimation, that is, a maximizer of the conditional joint PDF (4.1). This is because, for example, a state constraint can destroy the independence of the noise sequence $w_{[0:\infty]}$, which we assume throughout this chapter (see [74] for details). As an example of constraints that are compatible with the MAP estimation interpretation, inequality constraints independent of the state such as box constraints $\underline{w_i} < w_{k,i} < \overline{w_i}$ $(i = 1, \ldots, n)$ can be considered for either or both w and v.

To apply the algebraic geometry techniques used in [17] and Chapter 3, we next convert these equations into algebraic equations. This conversion can be performed for each time step k by canceling the denominators of the n equations at the time step; first, let us select (4.8), which consists of an n-dimensional vector-valued rational function and can be rewritten as follows:

$$D_0^{-1}\left(x_0, x_1, \mu, u_0, y_0\right) n_0\left(x_0, x_1, \mu, u_0, y_0\right) = 0, \tag{4.12}$$

where n_0 is a vector of polynomials whose i-th component $n_{0,i}$ is the numerator of the left-hand side of the i-th equation in (4.8) and D_0 is a diagonal matrix-valued function whose i-th diagonal element is the denominator polynomial corresponding to $n_{0,i}$. It is obvious that the solution of (4.8) satisfies a set of n algebraic equations

$$n_0\left(x_0, x_1, \mu, u_0, y_0\right) = 0. \tag{4.13}$$

To prevent all denominators from vanishing, consider an algebraic equation

$$1 - d_0 \det D_0\left(x_0, x_1, \mu, u_0, y_0\right) = 0, \tag{4.14}$$

where $d_0 \in \mathbf{R}$ is an additional scalar variable. Note that if (4.14) has a solution, the value of $\det D_0$ is given as $\det D_0 = 1/d_0 \neq 0$, which means any denominator included in (4.8) does not vanish at the solution; in other words, all the denominators do not vanish as long as (4.14) holds. Consequently, (4.8) is equivalent to $(n + 1)$ algebraic equations, namely, (4.13) and (4.14). Note that if $\det D_0$ is strictly positive or negative on its domain, (4.14) always has a solution and can be omitted.

The process of the conversion from (4.8) to the algebraic equations (4.13) and (4.14) can also be performed for (4.9) and (4.10) in the same way. In the following discussion, we denote a vector of polynomials $[n_0^\top \;\; 1 - d_0 \det D_0]^\top$ by F_0 and derive the vectors of polynomials F_k $(k = 1, \ldots, N-1)$ and F_N from (4.9) and (4.10), respectively, in the same way. In the end, all equations (4.8)–(4.10) can be equivalently converted into the following equations:

$$F_0(x_0, x_1; \mu, u_0, y_0, d_0) = 0, \tag{4.15}$$

$$F_k(x_{[k-1:k+1]}; u_{k-1}, u_k, y_k, d_k) = 0 \qquad (k = 1, \ldots, N-1), \tag{4.16}$$

$$F_N(x_{N-1}, x_N; u_{N-1}, y_N, d_N) = 0. \tag{4.17}$$

Equations (4.15)–(4.17) are still necessary optimality conditions because they are equivalent to the original stationary conditions (4.8)–(4.10). By solving these equations for given parameter μ and histories of inputs $u_{[0:N-1]}$ and outputs $y_{[0:N]}$, we can obtain the candidates of the optimal solution of the MAP estimation problem (4.1)–(4.6). However, it is still difficult and time-consuming to solve (4.15)–(4.17) because they include all the state variables on the horizon and are coupled by these variables. Recall that, in the MHE procedure, just the terminal estimate x_N^* is used as the estimate of the current state, which means we do not need to compute the past estimates $x_{[0:N-1]}^*$ on-line. By using algebraic geometry techniques, we can eliminate the past estimates from (4.15)–(4.17) off-line without any approximation and obtain equations only in the terminal state x_N and other parameters μ, $u_{[0:N-1]}$, and $y_{[0:N]}$, which leads to a reduction of the computational cost.

4.3 Recursive Elimination Method for FHOEPs

If the state and observation equations (4.2)–(4.3) are linear and the PDFs of w_k and v_k are Gaussian, (4.15)–(4.17) (which are exactly the same as (4.8)–(4.10) in linear cases) are explicitly solvable and have a unique solution, which corresponds to the KF [75]. On the other hand, in nonlinear cases, these equations are no longer explicitly solvable, and we have to rely on numerical methods such as Newton's method to solve them. Generally speaking, nonlinear equations become harder to solve as the number of variables increases, which means decreasing the number of variables can reduce the computational cost for solving the equations; indeed, in the numerical examples introduced in [17] and Chapter 3, computational costs are reduced by decreasing the numbers of variables. Elimination theory [76] in algebraic geometry is one of the

most powerful tools to decrease the number of variables, and it can eliminate some of the variables from nonlinear equations symbolically. It is also shown by several numerical examples in [17] and Chapter 3 that this off-line variable elimination leads to a reduction of the on-line computational cost.

In MHE, as mentioned before, we only need to compute the terminal estimate x_N^*, and the other estimates $x_{[0:N-1]}^*$ are introduced just for connecting x_N^* to the past measurements $y_{[0:N-1]}$. Therefore, by eliminating the estimates $x_{[0:N-1]}^*$ from (4.15)–(4.17), we can reduce the on-line computational cost to solve the MAP estimation problem (4.1). In this section, we introduce the recursive elimination method for (4.15)–(4.17).

By using elimination ideals introduced in Section A.2, we can eliminate the variables $x_{[0:N-1]}$ from (4.15)–(4.17). However, it is known that the computational complexity of a Gröbner basis is, in the worst case, doubly exponential in the number of variables [77, 78]. Therefore, the number of variables that are handled at the same time during the elimination process needs to be reduced. Now, recall that each set of equations at time step k in (4.16) includes only the state at the time step x_k and its neighbors x_{k-1} and x_{k+1}. This adjacently coupled structure of (4.16) allows us to perform the elimination process recursively as Algorithm 3.

Remark 4.3. Due to the expensive computational complexity of a Gröbner basis, the complexity of Algorithm 3 mainly results from the computations of elimination ideals and can be quite high. However, computation of each elimination ideal J_k from I_k in Algorithm 3 involves much fewer variables than in all equations (4.15)–(4.17), which means that using Algorithm 3 is more efficient than directly eliminating the state history $x_{[0:N-1]}$ from the whole set of equations (4.15)–(4.17) all at once.

Algorithm 3 Recursive Elimination Method for Joint MAP Filtering Problem

Input: Algebraic equations (4.15)–(4.17) for joint MAP estimation
Output: Algebraic equations $G_N(x_N, d_N; \mu, u_{[0:N-1]}, y_{[0:N]}) = 0$

Let J_0 be ideal generated by set of polynomials $F_0 \subset \mathbf{R}[x_0, x_1, \mu, u_0, y_0, d_0]$, $k := 1$
while $k < N$ **do**
    $I_k := J_{k-1} + \langle F_k \rangle \subset \mathbf{R}[x_{[k-1:k+1]}, \mu, u_{[0:k-1]}, y_{[0:k]}, d_{[k-1:k]}]$
    $J_k := I_k \cap \mathbf{R}[x_k, x_{k+1}, \mu, u_{[0:k-1]}, y_{[0:k]}, d_k]$
    $k \leftarrow k + 1$
end while
$I_N := J_{N-1} + \langle F_N \rangle \subset \mathbf{R}[x_{N-1}, x_N, \mu, u_{[0:N-1]}, y_{[0:N]}, d_{[N-1:N]}]$
$J_N := I_N \cap \mathbf{R}[x_N, \mu, u_{[0:N-1]}, y_{[0:N]}, d_N]$
Let G_N be a set of generators of J_N, that is, $J_N = \langle G_N \rangle$

From Lemma A.2, we can derive a relationship between the algebraic set defined by J_N in Algorithm 3 and the solution set of (4.15)–(4.17); by letting

$$G_N(x_N; \mu, u_{[0:N-1]}, y_{[0:N]}, d_N) \subset \mathbf{R}[x_N, \mu, u_{[0:N-1]}, y_{[0:N]}, d_N]$$

be a set of generators of J_N, this relationship is stated as follows.

Lemma 4.1. Let sequence $x^*_{[0:N]}$ be the joint MAP trajectory with respect to a certain parameter $\bar{\mu}$ and histories of inputs $\bar{u}_{[0:N-1]}$ and measurements $\bar{y}_{[0:N]}$. Moreover, let $\bar{d}_{[0:N]}$ be a sequence that satisfies (4.15)–(4.17) with $x^*_{[0:N]}$, $\bar{\mu}$, $\bar{u}_{[0:N-1]}$, and $\bar{y}_{[0:N]}$. Then, x^*_N, $\bar{\mu}$, $\bar{u}_{[0:N-1]}$, $\bar{y}_{[0:N]}$, and $\bar{d}_N$ satisfy

$$G_N(x^*_N; \bar{\mu}, \bar{u}_{[0:N-1]}, \bar{y}_{[0:N]}, \bar{d}_N) = 0. \tag{4.18}$$

This lemma states that, for any given $\bar{\mu}$, $\bar{u}_{[0:N-1]}$, and $\bar{y}_{[0:N]}$, (4.18) can be considered as a necessary condition for optimality of x_N. This means that, roughly speaking, (4.18) can be viewed as an implicit function representation of the optimal estimate x^*_N as a function of the parameter μ and the histories of inputs $u_{[0:N-1]}$ and measurements $y_{[0:N]}$.

4.4 Application to Moving Horizon Estimation

In the following discussion in this chapter, symbol j denotes a real-time step, and k denotes a time step over the horizon of the joint MAP estimation problem (4.1)–(4.6). Furthermore, the estimate at time step k over the horizon at real-time step j is denoted by $x^*_{k;j}$ if necessary.

If we have $\mu = \mu(j)$ and the histories of inputs $u_{[0:N-1];j}$ and outputs $y_{[0:N];j}$ at real-time step j, we can obtain a candidate of the optimal estimate $x^*_{N;j}$ by substituting those data into (4.18) and then solving it. In general, (4.18) is nonlinear in x_N and d_N and has more than one solution, which means the obtained candidate is not necessarily optimal. However, finding the global estimate is extremely difficult and, even when possible, computationally demanding. Therefore, we need to find a better solution in a certain sense by considering the trade-off between optimality and computational cost.

One approach is solving (4.18) with several initial guesses that are different from each other. These initial guesses would yield several candidates for the optimal estimate, and then we can choose the best one. Note that, in general, the indeterminates of (4.18) are x_N and d_N, which means we have to give initial guesses not only for x_N

but also for d_N. If we have an initial guess x_{init}, then the corresponding initial guess d_{init} can be given from (4.14) at $k = N$ as

$$d_{\text{init}} = 1/\det D_N(x^*_{N;j-1}, x_{\text{init}}, u_{N-1}, y_N), \tag{4.19}$$

where $x^*_{N;j-1}$ is the optimal estimate at the previous real-time step. To determine which candidate is better, we compare the value of the PDF of x_N conditionally on x_{N-1} and y_N, which is obtained as follows:

$$\begin{aligned} p(x_N \mid x_{N-1}, y_N) &= \frac{p(x_N, y_N \mid x_{N-1})}{p(y_N)} \\ &= \frac{p(y_N \mid x_N)p(x_N \mid x_{N-1})}{p(y_N)} \propto p(y_N \mid x_N)p(x_N \mid x_{N-1}). \end{aligned} \tag{4.20}$$

When $x_{N-1} = x^*_{N-1}$, x_N maximizing the PDF (4.20) among the solutions of $G_N = 0$ is the optimal estimate x^*_N. In MHE, the previous optimal estimate $x^*_{N;j-1}$ can be regarded as a good estimate of $x^*_{N-1;j}$. Therefore, we can approximate $p(x_{N;j} \mid x_{N-1;j})$ in (4.20) by $p(x_{N;j} \mid x_{N;j-1})$. In the end, by taking the logarithm, we can determine the optimal solution by computing and comparing the values of the following function:

$$\begin{aligned} \log p(x_{N;j} \mid x_{N-1;j}, y_{N;j}) \cong \log p_v\left(y_{N;j} - h_N(x_{N;j})\right) \\ + \log p_w\left(x_{N;j} - f_{N-1}(x_{N;j-1}, u_{N-1;j})\right) + c, \end{aligned} \tag{4.21}$$

where c is a constant derived from $p(y_N)$ and then neglected in the comparison. Finally, the algorithm to perform MHE using the derived equation $G_N = 0$ is summarized as Algorithm 4.

Remark 4.4. The stability of Algorithm 4, in the sense that the error between the estimate and the true state at each time step is finite, is quite difficult to investigate because we consider nonlinear systems and non-Gaussian noise and do not assume the boundedness of noise. In particular, the boundedness of noise is usually assumed to guarantee the stability and convergence of algorithms for MHE [69–72]. This boundedness can be taken into account in Algorithm 4 by a slight modification of the candidate evaluation (lines 6–9). The estimates of noise can be computed from a candidate $\tilde{x}$, the previous estimate, and the observed output via the state equation (4.2) and observation equation (4.3). The candidate yielding invalid noise estimates can then be discarded. Although this modification would be useful to guarantee the stability or convergence of Algorithm 4, the proof is still part of our future work.

Algorithm 4 Moving Horizon Estimation Using Implicit Function Representation

Input: Equation $G_N(x_N; \mu, u_{[0:N-1]}, y_{[0:N]}, d_N) = 0$, parameter $\mu(j)$ as function of real-time step j, histories of inputs $u_{[0:N-1];j}$ and $y_{[0:N];j}$, and set of initial guesses X_{init}

Output: Optimal estimate $x^*_{N;j}$ at real-time step j

$l_{\max} \leftarrow -\infty$
for each $x_{\text{init}} \in X_{\text{init}}$ **do**
    Compute initial guess d_{init} from equation (4.19)
    Compute solution $\tilde{x}$ and $\tilde{d}$ of $G_N = 0$ with initial guess x_{init} and d_{init} by substituting $\mu(j)$, $u_{[0:N-1];j}$, and $y_{[0:N];j}$
    Compute value of PDF $l(\tilde{x})$ of $\tilde{x}$ by substituting (4.21)
    if $l(\tilde{x}) > l_{\max}$ **then**
        $l_{\max} \leftarrow l(\tilde{x})$
        $x^*_{N;j} \leftarrow \tilde{x}$
    end if
end for
return $x^*_{N;j}$

4.5 Numerical Example

A numerical example is provided to show the efficiency of the proposed method. Algorithm 3 is implemented using `Maple` on a PC (Intel Core i7-8550U 1.80 GHz, RAM: 8 GB), and Algorithm 4 and other algorithms such as the UKF, and PF are implemented using `Python` on the same PC.

Consider the following one-dimensional nonlinear system:

$$x_{j+1} = \frac{1}{100}x_j^2 + u_j + w_j, \tag{4.22}$$

$$y_j = x_j + v_j. \tag{4.23}$$

The system disturbance w_j is sampled from a Gaussian distribution $\mathcal{N}(0, 3)$, while the measurement noise v_j is sampled from the standard Cauchy distribution, whose PDF is defined as

$$p_v(v_j) = \frac{1}{\pi\left(1 + v_j^2\right)}. \tag{4.24}$$

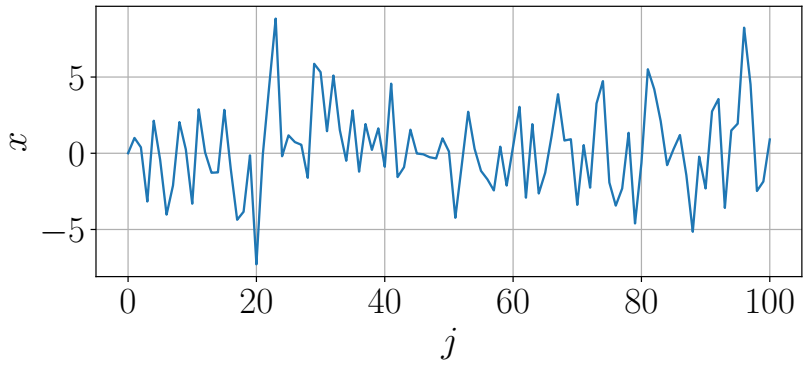
The Cauchy distribution is a heavy-tailed distribution and does not have a mean and variance. It is known that such heavy-tailed distributions can model impulsive noise such as atmospheric and underwater acoustic noise, which are the dominant sources of the noise observed in radar and sonar applications [64]. Hence, the state estimator for dynamic systems with Cauchy noise has been intensively studied these days [63, 67]. Note that the denominator in the PDF (4.24) never vanishes for any

value of v_j, which means we can omit all equations (4.14) in this case. The input u_j is defined as a function of j:

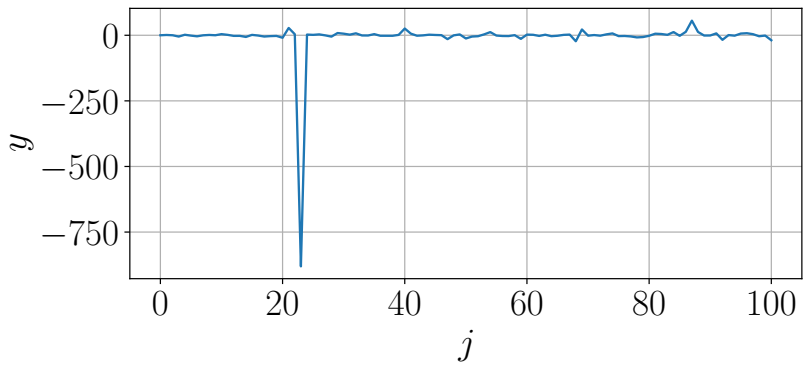
$$u_j = \cos(0.6j),$$

while it is treated as a symbol u_j during the off-line variable elimination. Figure 4.1 shows a realization of the state trajectory of the system and its output sequence. Several impulsive deviations can be observed in the output sequence (especially at $j = 23$) due to the Cauchy distribution.

The proposed method is applied to this estimation problem with the settings as follows. The length of the horizon is set to $N = 1$, and we define the PDF of the initial state as the Gaussian $\mathcal{N}(\mu, 3)$, where the value of μ is given as the previous estimate $x^*_{j-1} = x^*_{1;j-1}$. Algorithm 3 yields $G_1(x_1, \mu, u_0, y_0, y_1)$ consisting of a polynomial of degree 16. The computational time for this off-line variable elimination is 38.7 s. By substituting the measurements $y_{0;j}$ and $y_{1;j}$, input $u_{0;j} = \cos(0.6(j-1))$, and the parameter $\mu(j) = x^*_{j-1}$ into the equation $G_1 = 0$, we can obtain a candidate of the current estimate x^*_j for an initial guess. The set of initial guesses X_{init} in Algorithm 4



(a) State trajectory.



(b) Output.

Figure 4.1: Example of state trajectory of system (4.22) and its outputs derived from (4.23).

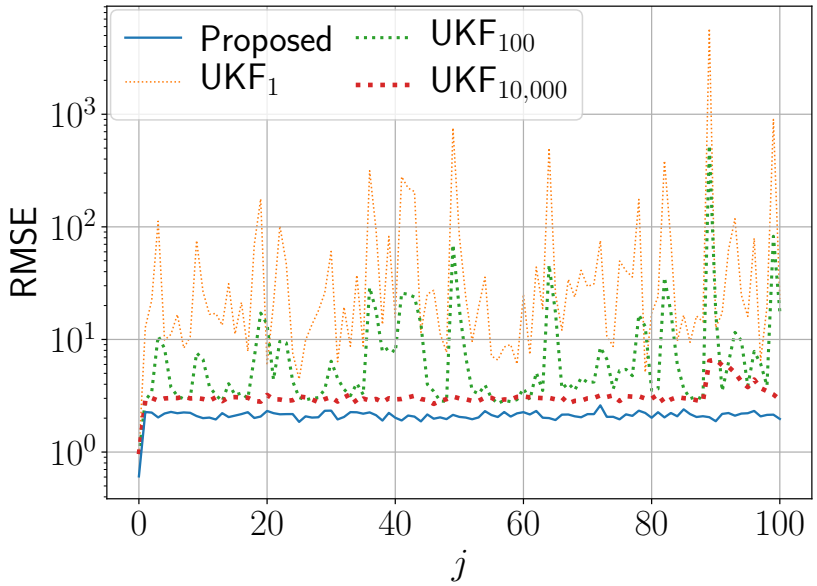
is defined as

$$X_{\text{init}} := \left\{ x^*_{j-1}, x^*_{j-1} \pm 2, x^*_{j-1} \pm 4, x^*_{j-1} \pm 6 \right\}.$$

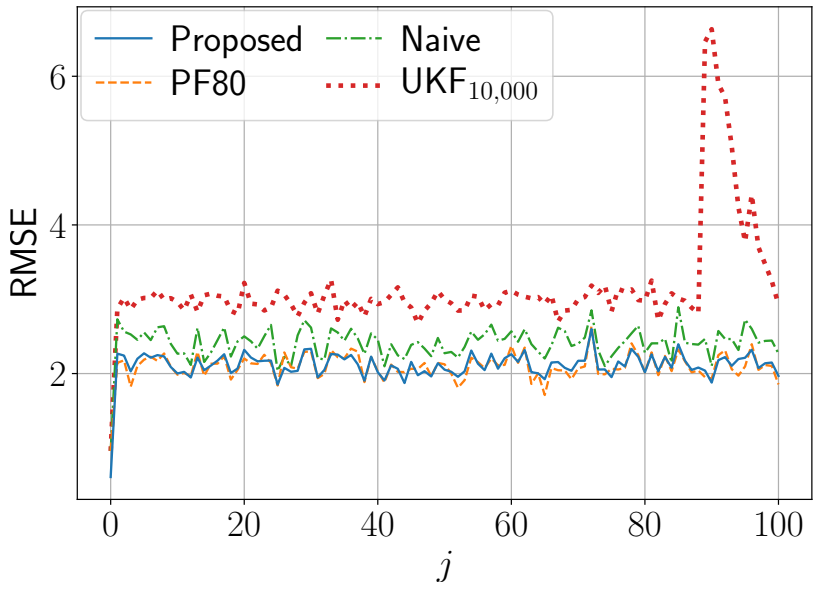
Remark 4.5. Due to the high computational cost of symbolic computation in Algorithm 3, the length of the horizon for this example is limited to $N = 1$. The computational cost of a Gröbner basis highly and complicatedly depends on the input polynomials and the order of variables used in the computation [76]. Hence, an order of variables that can reduce the computational cost of Algorithm 3 or a class of problems to which the algorithm can be efficiently applied could be investigated in future work.

On the other hand, for comparison, we also performed the UKF and PF to solve the same state estimation problem. The UKF requires the variance of observation noise, which cannot be defined for the Cauchy distribution. We used several different variances of 1, 100, and 10,000 for the UKF to show how the variance affects its performance. The parameter κ of the UKF is set to 0, which, after several trials with different κ, turned out not to affect the performance in this example. For the PF, the number of particles is set to 80, which shows the same performance as the proposed method in the root mean squared error (RMSE). Although the EKF is also one of the most popular methods, to which the proposed method is compared, it is omitted in this example because the system (4.22) has no linear term, which yields an inaccurate linear-approximation model in the EKF and results in large estimation errors. Moreover, to show how the off-line variable elimination by Algorithm 3 contributes to the performance of the proposed method, we also perform MHE by solving the stationary conditions (4.8)–(4.10) directly. Since the stationary conditions contain indeterminates of x_0 and x_1, a pair of initial guesses are needed to solve (4.8)–(4.10). We use the Cartesian product $X_{\text{init}} \times X_{\text{init}}$ for the set of initial guesses, obtain several optimal estimate candidates from those initial guesses, and select the best one with the same criterion as that of the proposed method. Hereafter, this method of solving the stationary conditions directly is referred to as the naive MHE method.

For 300 realizations of the system (4.22)–(4.23), state estimation is carried out by using the four methods mentioned before: the proposed method, naive MHE, UKF, and PF. Figure 4.2 shows the RMSEs between the true state and the estimates averaged for all realizations at each time step. It can be seen that the proposed method, PF, and the naive MHE method have almost the same performance and outperform the UKFs. Although, as can be seen in Fig. 4.2a, the RMSEs of UKFs become smaller as the measurement variance increases, they cannot reach the RMSEs



(a) RMSEs derived from proposed method (solid) and UKF with measurement variance of 1 (thin, dotted), 100 (medium, dotted), and 10,000 (thick, dotted).



(b) RMSEs derived from proposed method (solid), PF (dashed), naive MHE method (dash-dotted), and UKF with measurement variance of 10,000 (dotted).

Figure 4.2: Comparison of RMSEs for proposed method, UKF, PF, and naive MHE method.

of the other three methods, which is indicated in Fig. 4.3. This figure shows the estimated trajectories that were derived from the proposed method, PF, naive MHE method, and UKF with a measurement variance of 10,000 for the same realization as

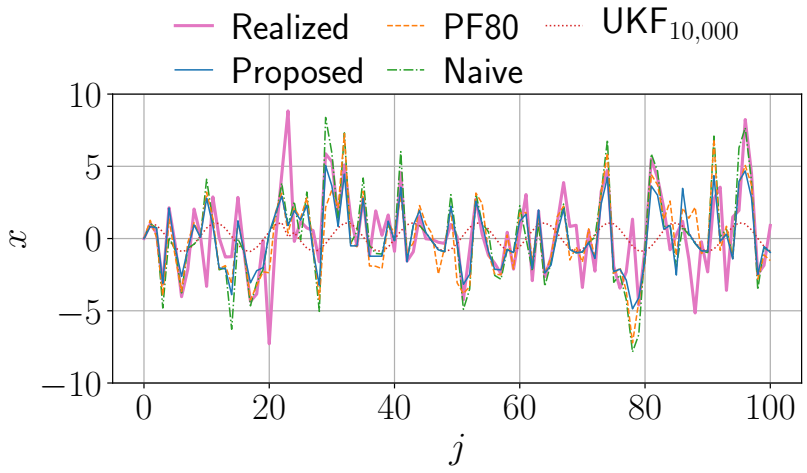


Figure 4.3: Estimated trajectories for certain realization (thick, solid) derived from proposed method (thin, solid), PF (dashed), naive MHE method (dash-dotted), and UKF with measurement variance of 10,000 (dotted).

in Fig. 4.1. All methods except for the UKF yield almost the same trajectories that are close to the realized one, while the trajectory derived from the UKF depicts a sinusoidal curve. This is because the UKF ignores all measurements due to the quite large measurement variance, which indicates that the performance of UKF cannot be improved by increasing the measurement variance further.

To show the efficiency of the proposed method, we next compare the computational times for each method. Figure 4.4 shows the boxplots of computational times for all time steps and realizations. Although the UKF is fastest, its performance in RMSE is much worse than those of other methods as mentioned above. On the other hand, the slowest is the naive MHE method, whose median of computational times is about six times larger than that of the PF. The proposed method is almost 30% faster than the PF when comparing the median of computational times in spite of their comparable RMSEs, which shows the efficiency of the proposed method.

Remark 4.6. Note that the PF is performed on a single thread in this example and thus can be accelerated by exploiting the parallel computation. Note also that lines 3–5 of Algorithm 4 can also be parallelized over all initial guesses in X_{init}, which will result in further acceleration of the proposed method.

Remark 4.7. The maximum computational time for the proposed method is about 0.03 s, which must be shorter than the period of measurements to execute Algorithm 4 in real time. Although this maximum may be undesirable for some applications, it can be shortened by limiting the number of iterations in the numerical algorithm to solve the equation $G_N = 0$ (line 4 of Algorithm 4).

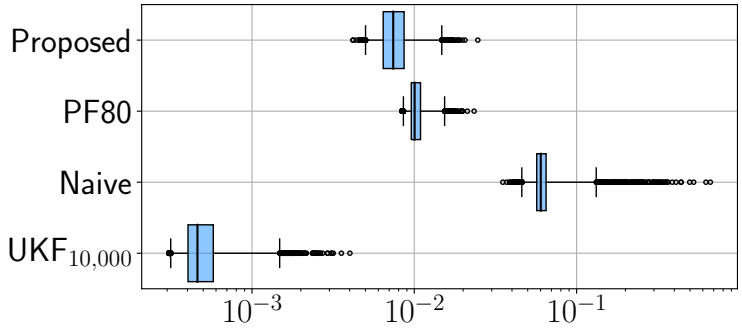


Figure 4.4: Computational times for proposed method, PF, UKF, and naive MHE method. Edges of whiskers are set to 0.5th and 99.5th percentiles.

4.6 Summary

In this chapter, we proposed a recursive elimination method for the joint MAP estimation of a class of nonlinear systems, where the stationary conditions are written as equations consisting of rational functions. First, the estimation problem is reformulated as an unconstrained nonlinear optimization problem, and then the stationary conditions are derived. Next, the stationary conditions are converted into algebraic equations by introducing additional variables to deal with rational functions in those conditions. By applying the concept of elimination ideals from commutative algebra, the proposed method eliminates the past state variables from the algebraic equations and obtains an implicit function representation of the optimal estimate as a function of several parameters including the measurement sequence. An MHE algorithm using this implicit representation is proposed, whose efficiency compared to other estimation methods including the particle filter is shown in a numerical example.

In the field of state estimation, some properties of the estimation error such as the convergence to zero for noise-free cases and the stability for bounded-noise cases are often investigated. For future work, the proof of such convergence and stability will be considered. For another direction of further studies, new numerical algorithms oriented to solve the implicit function representation derived from the proposed method can be investigated.

Chapter 5

State Estimation via Holonomic Gradient Method

5.1 Introduction

In the framework of moving horizon estimation, the optimality of estimates is considered only over a finite horizon, in other words, the optimal solution is optimal only for the outputs observed over the horizon and may not be optimal for all the past observed outputs. To take into account all the past outputs, we need to solve the Bayesian filtering problem, where all the information of the past outputs is involved in the probability density function (PDF) of the state conditionally on them and used to compute the optimal estimate.

In the Bayesian estimation framework, the posterior distribution is estimated instead of a point estimate. The Bayesian estimate yields much more information (e.g., the confidence of the estimate, derived as the variance of the posterior distribution) than the point estimate does, which can be used to control a system, for example. Because the posterior distribution is a function on the state space, however, rather than a point in that space, finding the distribution is equivalent to finding a *functional* that maps the previous posterior distribution to the current one, which is often impossible and computationally demanding even when possible.

The EKF and UKF overcome this difficulty by approximating the following four PDFs by Gaussian PDFs: (i) the posterior PDF of the previous state $p(x^-)$, (ii) the prior PDF of the current state $p(x \mid u)$, (iii) the predictive PDF of the output $p(y \mid u)$, and (iv) the posterior PDF of the current state $p(x \mid u, y)$; here, x^- and x denote the previous and current states, respectively, u denotes a known input, and y denotes the observed output. Because a Gaussian distribution is completely described by its mean and variance, with the approximations for $p(x^-)$ and $p(x \mid u, y)$, the computation of

the posterior PDF can be reduced to evaluating a finite number of *functions* that map u, y, and the mean and variance of x^- to those of x. Moreover, with the other approximations for $p(x \mid u)$ and $p(y \mid u)$, those functions can be explicitly obtained by composition of simple arithmetic operations.

Note that the substantial simplification from the computation of a functional to that of the functions can be achieved with only the approximations for $p(x^-)$ and $p(x \mid u, y)$. Therefore, if we could evaluate these functions efficiently without the other two approximations, the filtering problem would be solved more accurately, considering the nonlinearities of the system dynamics and observation process. The main obstacle to achieving such evaluation is the integration of nonlinear functions with parameters accompanied by marginalization and computation of expectations.

In this chapter, we assume that all integrands appearing in the computation of the posterior PDF are *holonomic functions*. Roughly speaking, a holonomic function is a function defined by a set of partial differential equations (PDEs) having certain good properties (see Definition A.41 for details), and almost all the functions appearing in systems theory belong to this class of functions. It is interesting that, for any holonomic function, we can compute *a set of PDEs satisfied by its integral* instead of the integral itself. By using this property, for a wide class of nonlinear systems and non-Gaussian noise, we can obtain a set of PDEs satisfied by the mean and variance of the posterior PDF as functions of the data, i.e., the mean and variance of the previous state, known input, and observed output. Note that the PDEs can be computed symbolically, which means that we can compute them offline. Moreover, the set of PDEs satisfied by the mean and variance can be converted to a Pfaffian system: a set of PDEs for a certain vector-valued function that can be solved efficiently. Therefore, if we compute the PDEs and convert them into a Pfaffian system offline, then we can efficiently evaluate the mean and variance for any given data online.

5.2 Problem Setting

Consider the following discrete-time system with system and observation noises:

$$x = f(x^-, u) + w, \tag{5.1}$$

$$y = h(x) + v, \tag{5.2}$$

where $x^-, x \in \mathbf{R}^n$ are the previous and current states, respectively, $u \in \mathbf{R}^m$ is the known input, and $y \in \mathbf{R}^r$ is the current output. The functions $f : \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}^n$

and $h : \mathbf{R}^n \to \mathbf{R}^r$ are analytic functions that describe the system dynamics and measurement, respectively. The random vectors $w \in \mathbf{R}^n$ and $v \in \mathbf{R}^r$ are the system and observation noises, respectively, and are assumed to be independent and identically distributed. We also assume that the PDFs of w and v (denoted respectively by $p_w(w)$ and $p_v(v)$) are given and defined over $\mathbf{R}^n$ and $\mathbf{R}^r$, respectively.

In the Bayesian framework, the statistical property of the current state can be completely described by the posterior PDF, which can be written by the Bayes rule as

$$p(x \mid u, y) = \frac{p(y, x \mid u)}{p(y \mid u)}, \tag{5.3}$$

where $p(y, x \mid u)$ is the joint PDF of the output and state, and $p(y \mid u)$ is the marginal PDF of the output. Assuming the posterior PDF of the previous state, denoted by $p(x^- \mid \mu^-, \Sigma^-)$, to be Gaussian, the PDFs in equation (5.3) can be written as follows.

$$p(y, x \mid u, \mu^-, \Sigma^-) = \int_{\mathbf{R}^n} p(y \mid x) p(x \mid x^-, u) p(x^- \mid \mu^-, \Sigma^-) dx^-, \tag{5.4}$$

and

$$p(y \mid u, \mu^-, \Sigma^-) = \int_{\mathbf{R}^n} p(y, x \mid u, \mu^-, \Sigma^-) dx, \tag{5.5}$$

where $p(x \mid x^-, u)$ and $p(y \mid x)$ are the transition and observation PDFs, respectively. Under our problem setting, the transition and observation PDFs can be described by using p_v and p_w:

$$p(x \mid x^-, u) = p_w(x - f(x^-, u)), \tag{5.6}$$

$$p(y \mid x) = p_v(y - h(x)), \tag{5.7}$$

where (5.7) and (5.6) derive from the rule of transformation of PDFs [26]. Consequently, the posterior PDF (5.3) can be written as a fraction of integrals (5.4) and (5.5).

Now, consider the approximation of $p(x \mid u, y, \mu^-, \Sigma^-)$ by a certain Gaussian PDF that has the same mean and variance as those of the original posterior PDF. If we can compute the first and second moments of $p(x \mid u, y, \mu^-, \Sigma^-)$, then we can readily obtain the approximating Gaussian: the mean is equal to the first moment of the PDF, and the variance can be derived from the first and second moments. The first moment of $p(x \mid u, y, \mu^-, \Sigma^-)$ is defined as the expectation of x:

$$E[x] = \int_{\mathbf{R}^n} x p(x \mid u, y, \mu^-, \Sigma^-) dx = \frac{\int_{\mathbf{R}^n} x p(y, x \mid u, \mu^-, \Sigma^-) dx}{p(y \mid u, \mu^-, \Sigma^-)}, \tag{5.8}$$

where the second equality is obtained from (5.3). Similarly, the second moment is defined as

$$E\left[xx^\top\right] = \frac{\int_{\mathbf{R}^n}\left(xx^\top\right)p(y,x \mid u,\mu^-,\Sigma^-)dx}{p(y \mid u,\mu^-,\Sigma^-)}. \tag{5.9}$$

The numerators in (5.8) and (5.9) define the following vector- and matrix-valued functions:

$$\Phi^{(1)} = \left[\phi_1^{(1)} \ \cdots \ \phi_n^{(1)}\right]^\top \coloneqq \int_{\mathbf{R}^n} xp(y,x \mid u,\mu^-,\Sigma^-)dx, \tag{5.10}$$

$$\Phi^{(2)} = \left\{\phi_{ij}^{(2)}\right\} \coloneqq \int_{\mathbf{R}^n}\left(xx^\top\right)p(y,x \mid u,\mu^-,\Sigma^-)dx, \tag{5.11}$$

which map $(u,y,\mu^-,\Sigma^-) \in \mathbf{R}^m \times \mathbf{R}^r \times \mathbf{R}^n \times \mathrm{PD}(n)$ to $\Phi^{(1)}(u,y,\mu^-,\Sigma^-) \in \mathbf{R}^n$ and $\Phi^{(2)}(u,y,\mu^-,\Sigma^-) \in \mathrm{PD}(n)$, respectively. The output PDF, which is equal to the denominator of (5.8) and (5.9), can also be regarded as a scalar-valued function:

$$\psi(u,y,\mu^-,\Sigma^-) \coloneqq \int_{\mathbf{R}^n} p(y,x \mid u,\mu^-,\Sigma^-)dx. \tag{5.12}$$

If we can evaluate the functions $\Phi^{(1)}$, $\Phi^{(2)}$, and ψ for given data u, y, μ^-, and Σ^-, then we can compute the first and second moments of the posterior PDF as in (5.8) and (5.9). The first moment μ is a conditional mean of the current state with respect to the current measurement, which is known as a *minimum variance estimate*. Moreover, by computing the mean μ and variance Σ of the posterior PDF of the current state, we can use them as parts of the data at the next step. Consequently, the filtering problem of the system expressed by (5.1) and (5.2) is reduced to the problem of how to evaluate $(n+n(n+1)/2+1)$ scalar-valued functions $\phi_i^{(1)}$ $(i=1,\ldots,n)$, $\phi_{ij}^{(2)}$ $(i,j \in \{1,\ldots,n\})$, and ψ. Throughout this chapter, we make the following assumption.

Assumption 5.1. All integrands in definitions (5.10)–(5.12) are holonomic functions.

5.3 Computation of Linear PDEs

In the previous section, we obtained three functions of the data u, y, μ^-, and Σ^-, which can be described by the integrals of multiplications of given functions. In general, the integral of a nonlinear function does not have a closed form, and thus, we have to rely on numerical integration algorithms such as the Monte Carlo method to evaluate the integral. For a certain wide class of functions called holonomic functions, however, we can efficiently evaluate their integralsby using a symbolic numeric method called the holonomic gradient method (HGM) [46]. Roughly speaking, a holonomic

function is characterized by a finite set of linear PDEs and a finite number of boundary values. This characterization enables us to translate the calculations in holonomic functions, such as multiplication and integration, to manipulations of differential operators. By applying this translation, we can explicitly derive the set of linear PDEs satisfied by the integral of a holonomic function, instead of the integral itself.

An analytic function $\alpha(X)$ can be certified as a holonomic function by finding a certain type of differential operators; if we can find $l_i \in \mathbf{R}(X)\langle\partial_i\rangle$ such that $l_i \bullet \alpha = 0$ for all $i = 1, \ldots, n$, then it follows from Theorem A.7 that the ideal $\langle l_1, \ldots, l_n\rangle$ is zero-dimensional, which indicates that α is a holonomic function.

Example 5.1. An n-dimensional Gaussian distribution

$$\mathcal{N}(X \mid \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n|\Sigma|}} \exp\left(-\frac{1}{2}(X-\mu)^\top \Sigma^{-1}(X-\mu)\right)$$

is annihilated by the following n differential operators:

$$[l_1 \ \cdots \ l_n]^\top \coloneqq \partial_X + \Sigma^{-1}(X-\mu).$$

Because $l_i \in \mathbf{R}(X)\langle\partial_i\rangle$ for all $i = 1, \ldots, n$, the ideal $\langle l_1, \ldots, l_n\rangle$ is zero-dimensional, so $\mathcal{N}$ is a holonomic function of X for fixed μ and Σ. Moreover, by differentiating $\mathcal{N}$ with respect to each component of μ and Σ, we can readily show that $\mathcal{N}$ is also a holonomic function of X, μ, and Σ.

Remark 5.1. The class of holonomic functions contains most kinds of functions emerging in both theoretical and practical problems of systems theory, including polynomials, rational functions, trigonometric functions, exponentials, logarithms, and their sums, products, and compositions under mild assumptions [79].

The most important feature of holonomic functions is the closure property, as stated in the following theorem.

Theorem 5.1. [80] For holonomic functions $\alpha(X)$ and $\beta(X)$ in $X = [X_1 \ \cdots \ X_n]^\top$, the following hold.

(i) The product $\alpha \cdot \beta$ is also a holonomic function.

(ii) Assume that $\alpha = \alpha(X_1, \ldots, X_n)$ is infinitely differentiable on $\mathbf{R}^n$ and rapidly decreasing with respect to X_n, meaning that

$$\lim_{X_n \to \infty} X_n^s \partial_n^t \alpha(X) = 0 \quad (s, t \in \mathbf{Z}_{\geq 0})$$

for any $[X_1 \ \cdots \ X_{n-1}]^\top \in \mathbf{R}^{n-1}$. Then, the integral

$$\int_{-\infty}^{\infty} \alpha(X) dX_n \tag{5.13}$$

is also a holonomic function of $[X_1 \ \cdots \ X_{n-1}]^\top$.

As mentioned before, the integral of a nonlinear function can rarely be derived in closed form. If the integrand is a holonomic function, however, the zero-dimensional ideal annihilating the integral can *always* be algorithmically derived from that annihilating the integrand. We refer to [80] for the theorems introduced below, which define the multiplication and integration of holonomic functions in terms of holonomic ideals.

Theorem 5.2. (Multiplication) Let $\alpha(X)$ and $\beta(X)$ be holonomic functions, which are solutions of holonomic ideals I_α and I_β, respectively. By replacing the indeterminates of β and I_β with Y, define an ideal

$$I_{\alpha\otimes\beta} \coloneqq \{a_1 l_\alpha + a_2 l_\beta \mid l_\alpha \in I_\alpha; l_\beta \in I_\beta; a_1, a_2 \in \mathcal{D}_{2n}\} \subset \mathcal{D}_{2n} = \mathbf{R}[X, Y]\langle \partial_X, \partial_Y \rangle.$$

This ideal can be regarded as an ideal of $\mathbf{R}[X, Z]\langle \partial_X, \partial_Z \rangle$ by applying a coordinate transformation $Z = X - Y$. Then, the product $\alpha(X)\beta(X)$ is a solution of the restriction of $I_{\alpha\otimes\beta}$ with respect to Z.

Theorem 5.3. (Integration) Let $\alpha(X)$ be a holonomic function of variables $X = [X_1 \ \cdots \ X_n]^\top$, which is a solution of a holonomic ideal $I \subset \mathcal{D}_n = \mathbf{R}[X]\langle \partial_X \rangle$. Moreover, assume that α is infinitely differentiable on $\mathbf{R}^n$ and rapidly decreasing with respect to X_n. Let $\mathcal{D}_{n-1}$ be $\mathbf{R}[X_1, \ldots, X_{n-1}]\langle \partial_1, \ldots, \partial_{n-1} \rangle$. Then, the integral

$$\int_{-\infty}^{\infty} \alpha(X) dX_n$$

is a solution of the holonomic ideal $(I + \partial_n \mathcal{D}_n) \cap \mathcal{D}_{n-1}$.

The bases of both ideals for the product and integral can be computed by computing the restriction of the holonomic ideals [81], and the algorithms to compute them are given in [80]. Consequently, Theorems 5.2 and 5.3, accompanied by the closure property (Theorem 5.1), guarantee that the multiplication and integration of holonomic functions can *always* be performed recursively *in terms of holonomic ideals*. The algorithms use Gröbner bases in the Weyl algebra $\mathcal{D}_n$, which are certain bases of ideals that have good properties and can be computed by many computer algebra systems (CASs) such as `Risa/Asir` [82], `Macaulay2` [83], and `SINGULAR` [84].

As a result, the sets of linear PDEs satisfied by ψ and all the components of $\Phi^{(1)}$ and $\Phi^{(2)}$ can be symbolically computed from those satisfied by $p(y \mid x)$, $p(x \mid x^-, u)$, and $p(x^- \mid \mu^-, \Sigma^-)$, as summarized by the following theorem.

Theorem 5.4. Suppose that $p(y \mid x) = p_v(y - h(x))$ and $p(x \mid x^-, u) = p_w(x - f(x^-, u))$ are both holonomic functions and are solutions of holonomic ideals I^o and I^s, respectively. Let I^- be the holonomic ideal that annihilates the Gaussian PDF $p(x^- \mid \mu^-, \Sigma^-)$ with mean μ^- and variance Σ^-. Then, all the functions $\phi_i^{(1)}$ $(i = 1, \ldots, n)$, $\phi_{ij}^{(2)}$ $(i, j \in \{1, \ldots, n\})$, and ψ are holonomic. Moreover, if bases of I^o, I^s, and I^- are given, then a basis of the holonomic ideal annihilating each function $\phi_i^{(1)}$, $\phi_{ij}^{(2)}$, and ψ can be computed symbolically from the given bases.

Proof. Because the joint PDF $p(y, x \mid u, \mu^-, \Sigma^-)$ is defined by multiplying $p(y \mid x)$, $p(x \mid x^-, u)$, and $p(x^- \mid \mu^-, \Sigma^-)$ and integrating the product over $x^- \in \mathbf{R}^n$ as in (5.4), Theorems 1 and 2 guarantee that the joint PDF is a holonomic function and annihilated by a certain holonomic ideal I^{joint}. Hence, a basis of I^{joint} can be computed from those of ideals I^o, I^s, and I^-.

First, for the vector-valued function $\Phi^{(1)}$, the i-th component $\phi_i^{(1)}$ is defined by integrating the product of the joint PDF and x_i. Note that x_i is a holonomic function annihilated by a holonomic ideal $J_i \coloneqq \langle \partial_i^2, x_i\partial_i - 1\rangle$. Therefore, $\phi_i^{(1)}$ is also a holonomic function annihilated by a holonomic ideal, which can be computed from I^{joint} and J_i for all $i = 1, \ldots, n$. Next, for the matrix-valued function $\Phi^{(2)}$, the (i, j)-component $\phi_{ij}^{(2)}$ is a holonomic function, and the corresponding holonomic ideal can be computed from I^{joint} and a holonomic ideal annihilating the function $x_i x_j$:

$$J_{ij} \coloneqq \begin{cases} \langle \partial_i^3, x_i^2\partial_i^2 - 2, x_i\partial_i - 2\rangle & (i = j) \\ \langle \partial_i^2, \partial_j^2, x_i\partial_i - 1, x_j\partial_j - 1\rangle & (i \neq j) \end{cases},$$

for all $i, j \in \{1, \ldots, n\}$. Finally, the scalar-valued function ψ is obviously a holonomic function, because it is the integral of a holonomic function (i.e., the joint PDF) over $x \in \mathbf{R}^n$. Thus, the corresponding holonomic ideal can be computed from I^{joint}. □

5.4 Pfaffian Systems and Evaluation of Functions $\Phi^{(1)}$, $\Phi^{(2)}$, and ψ

Theorem 5.4 guarantees that we can symbolically compute the bases of holonomic ideals annihilating the functions $\phi_i^{(1)}$ $(i = 1, \ldots, n)$, $\phi_{ij}^{(2)}$ $(i, j \in \{1, \ldots, n\})$, and ψ. By solving these linear PDEs for certain boundary conditions, we can evaluate

the functions for any given data u, y, μ^-, and Σ^-. In general, it is quite difficult and computationally demanding to solve PDEs even in linear cases. The correspondence between zero-dimensional ideals and holonomic functions, however, also gives a method to evaluate them efficiently almost everywhere in their domains of definition, which is named the HGM. In detail, the set of linear PDEs can be reduced to a nonlinear ordinary differential equation (ODE) for a vector-valued function, of which a certain component corresponds to the solution of the original PDEs. Once the ODE is obtained, we can evaluate its solution by numerically integrating it from a certain initial point to the point at which we want to evaluate the solution, namely, the holonomic function satisfying the original PDEs.

To introduce the evaluation method for holonomic functions, we first define a *Pfaffian system* for a holonomic ideal. In the following discussion, $I_i^{(1)}$, $I_{ij}^{(2)}$, and I^ψ denote the holonomic ideals annihilating functions $\phi_i^{(1)}$, $\phi_{ij}^{(2)}$, and ψ, respectively, and X denotes a vector $\left[(\mu^-)^\top \ \mathrm{vech}(\Sigma^-)^\top \ u^\top \ y^\top\right]^\top \in \mathbf{R}^N$, where $N = n + n(n+1)/2 + m + r$. The ideals $I_i^{(1)}$, $I_{ij}^{(2)}$, and I^ψ are then subsets of the Weyl algebra $\mathcal{D} = \mathbf{R}[X]\langle\partial\rangle$.

For a holonomic ideal $I \subset \mathcal{D}$, a zero-dimensional ideal $\mathcal{R}I \subset \mathcal{R} = \mathbf{R}(X)\langle\partial\rangle$ can be defined by Theorem A.7. Definition A.40 indicates that the quotient ring $\mathcal{R}/\mathcal{R}I$ is a finite-dimensional vector space over $\mathbf{R}(X)$. Specifically, from the Macaulay's theorem A.2, there exists a finite set of multi-index vectors $\{d_0, d_1, \ldots, d_s\} \subset \mathbf{Z}^n_{\geq 0}$, including $d_0 = \mathbf{0}$, such that $\{[\partial^{d_0}], [\partial^{d_1}], \ldots, [\partial^{d_s}]\} \subset \mathcal{R}/I$ is a basis of $\mathcal{R}/I$.

This theorem indicates that an equivalence class $[\partial_i \cdot \partial^{d_j}] \in \mathcal{R}/I$ can be written as a linear combination of $\{[\partial^{d_0}], [\partial^{d_1}], \ldots, [\partial^{d_s}]\}$ over $\mathbf{R}(X)$:

$$[\partial_i \cdot \partial^{d_j}] = \sum_{k=1}^{s} a_{jk}^i(X)[\partial^{d_k}] \quad \left(a_{jk}^i(X) \in \mathbf{R}(X)\right), \tag{5.14}$$

where $i = 1, \ldots, n$ and $j = 1, \ldots, s$. Now, for any solution α of I, consider the actions of $l_1, l_2 \in [l] \in \mathcal{R}/I$ on α. By the definition of the equivalence class, there exists $\tilde{l} \in I$ such that $l_1 = l_2 + \tilde{l}$, so $l_1 \bullet \alpha = (l_2 + \tilde{l}) \bullet \alpha = l_2 \bullet \alpha + 0 = l_2 \bullet \alpha$. Therefore, the action of the equivalence class $[l]$ on the solution α is well-defined, and we obtain

$$(\partial_i \cdot \partial^{d_j}) \bullet \alpha = \sum_{k=1}^{s} a_{jk}^i(X)\partial^{d_k} \bullet \alpha, \tag{5.15}$$

from (5.14). By letting $A_i(X) \in \mathbf{R}(X)^{s\times s}$ be a matrix-valued function whose (j,k)-component is $a_{jk}^i(X)$, the equations defined by (5.15) can be summarized as

$$\partial_i \bullet Q(X) = A_i(X)Q(X) \quad (i = 1, \ldots, n), \tag{5.16}$$

where $Q(X) = [\partial^{d_0} \bullet \alpha \ \partial^{d_1} \bullet \alpha \ \cdots \ \partial^{d_s} \bullet \alpha]^\top$. Note that the first component $\{Q(X)\}_1 = \partial^{d_0} \bullet \alpha$ is equal to $\alpha(X)$, because $d_0 = \mathbf{0}$. The set of first order PDEs defined by (5.16) is called the *Pfaffian system* for a vector-valued function $Q(X)$. For example, because $I_i^{(1)}$ is holonomic, there exists a set of multi-index vectors $\{d_0, d_1, \ldots, d_s\} \subset \mathbf{Z}^n_{\geq 0}$ so that the Pfaffian system for a vector-valued function $Q_i^{(1)}(X) = \left[\partial^{d_0} \bullet \phi_i^{(1)} \ \partial^{d_1} \bullet \phi_i^{(1)} \ \cdots \ \partial^{d_s} \bullet \phi_i^{(1)}\right]^\top$ can be defined. For $I_{ij}^{(2)}$ and I^ψ, such sets of multi-index vectors also exist, and so do the corresponding vector valued-functions $Q_{ij}^{(2)}$ and Q^ψ.

If the value of function Q is given at a certain starting point $X = X_s$, then we can evaluate it at a target point $X = X_t$ by integrating the Pfaffian system (5.16). Specifically, for a given pair of starting and target points (X_s, X_t), we can define an injective vector-valued function $X(t) : [0, 1] \to \mathbf{R}^n$ such that $X(0) = X_s$ and $X(1) = X_t$, i.e., a curve from X_s to X_t in $\mathbf{R}^n$. The Pfaffian system is then reduced to an ODE for the vector-valued function $Q(X(t))$:

$$\begin{aligned} \frac{d}{dt}Q(X(t)) &= [\partial_1 \bullet Q \ \cdots \ \partial_n \bullet Q] \frac{dX}{dt} \\ &= [A_1 \bullet Q \ \cdots \ A_n \bullet Q] \frac{dX}{dt}, \end{aligned} \tag{5.17}$$

which can be solved with respect to a given initial condition $Q(X(0)) = Q(X_s)$ by any numerical integration method such as the Runge-Kutta method. Consequently, the value $\alpha(X_t)$ is obtained from the first component of $Q(X_t) = Q(X(1))$.

Note that the integration process to solve a Pfaffian system fails if there exists $\tilde{t} \in [0, 1]$ such that the denominator of an $A_i(X(\tilde{t}))$'s component in the Pfaffian system vanishes. Strictly speaking, the integration cannot be performed if the curve $X(t)$ passes through the zero set of the least common multiple of all denominators appearing in the $A_i(X)$ in the Pfaffian system. This zero set is called the *singular locus* of the Pfaffian system. Because the singular locus is a zero set of a certain polynomial, i.e., a closed subset of $\mathbf{R}^N$ of measure zero, for any starting point X_s that is not included in the singular locus, there exists a neighborhood of X_s that is mutually disjoint from the singular locus. From the connected component of the neighborhood including X_s, we can choose any point as the target X_t and construct an integration path $X(t)$. We can compute the defining equations of the singular locus from $A_i(X)$ $(i = 1, \ldots, n)$ and use it to determine one or several starting points for the integration, as will be shown in subsection 5.5.3. Finally, Algorithm 5 summarizes the whole algorithm for state estimation using ODE (5.17).

Algorithm 5 State Estimation Using Pfaffian Systems

Input: Mean μ^- and variance Σ^- of PDF for previous state, known input u, observed current output y, Pfaffian systems (5.16) for all functions $\phi_i^{(1)}$, $\phi_{ij}^{(2)}$, and ψ, starting point X_s

Output: Failure or estimates of mean μ and variance Σ of PDF for current state

1: Compute $Q_i^{(1)}(X_s)$ $(i = 1, \dots, n)$, $Q_{ij}^{(2)}(X_s)$ $(i, j \in \{1, \dots, n\})$, and $Q^{\psi}(X_s)$
2: Set $X_t \leftarrow \left[(\mu^-)^\top \ \mathrm{vech}(\Sigma^-)^\top \ u^\top \ y^\top\right]^\top$
3: Define $X(t)$ so that $X(0) = X_s$ and $X(1) = X_t$ (e.g., $X(t) \coloneqq (1-t)X_s + tX_t$)
4: **if** $\exists \tilde{t} \in [0, 1]$ such that $X(\tilde{t})$ is included in singular locus of Pfaffian systems **then**
5: **return** algorithm has failed
6: **end if**
7: Integrate ODE (5.17) from $X(0) = X_s$ to $X(1) = X_t$ for all functions $Q_i^{(1)}$, $Q_{ij}^{(2)}$, and Q^{ψ} by numerical integration
8: Set $\phi_i^{(1)} \leftarrow \left\{Q_i^{(1)}(X_t)\right\}_1$ $(i = 1, \dots, n)$, $\phi_{ij}^{(2)} \leftarrow \left\{Q_{ij}^{(2)}(X_t)\right\}_1$ $(i, j \in \{1, \dots, n\})$, and $\psi \leftarrow \left\{Q^{\psi}(X_t)\right\}_1$
9: **return** $\mu = \left[\phi_1^{(1)}/\psi \ \cdots \ \phi_n^{(1)}/\psi\right]^\top$ and $\Sigma = \left\{\phi_{ij}^{(2)}/\psi\right\} - \mu\mu^\top$

Remark 5.2. We should emphasize that the outputs of Algorithm 5 are exact in the sense that if the prior distribution is identical to the Gaussian $p(x^- \mid \mu^-, \Sigma^-)$, then the algorithm does not include any approximations except for the discretization in the numerical integration at line 7. This is the difference from other existing methods such as the EKF, UKF, and particle filter (PF): in the EKF and UKF, the prediction and output PDFs are approximated up to the second-order moment, and in the PF, all the PDFs are approximated by a Monte Carlo method.

5.5 Numerical Example

5.5.1 Problem setting and computation of Pfaffian systems

In this section, we provide a numerical example to show the efficiency of the proposed method. Consider the following scalar system with Gaussian noises:

$$x = \frac{4}{5}x^- + u + w, \tag{5.18}$$

$$y = \frac{2x}{1 + x^2} + v, \tag{5.19}$$

where w and v are noises having a standard Gaussian distribution $\mathcal{N}(0, 1)$. We define the PDF of the previous state x^- as

$$p(x^- \mid \mu^-, \Lambda^-) = \sqrt{\frac{\Lambda^-}{2\pi}} \exp\left(-\frac{1}{2}\Lambda^-(x^- - \mu^-)^2\right), \tag{5.20}$$

where the parameter $\Lambda^- = (\Sigma^-)^{-1} \in \mathbf{R}$ is introduced to obtain a basis of the holonomic ideal I^-.

First, we need the holonomic ideals I^-, I^s, and I^o in Theorem 5.4, that is, the sets of linear PDEs satisfied by the PDFs $p(x^- \mid \mu^-, \Lambda^-)$, $p(x \mid x^-, u)$, and $p(y \mid x)$. By differentiating $p(x^- \mid \mu^-, \Lambda^-)$ with respect to all the variables x^-, μ^-, and Λ^-, we obtain the following differential operators:

$$\begin{array}{l} \partial_{x^-} + \Lambda^-(x^- - \mu^-), \quad \partial_{\mu^-} + \Lambda^-(x^- - \mu^-), \\ 2\Lambda^-\partial_{\Lambda^-} + 1 - \Lambda^-(x^- - \mu^-)^2, \end{array} \tag{5.21}$$

which annihilate $p(x^- \mid \mu^-, \Lambda^-)$ and generate a holonomic ideal I^-. On the other hand, the differential operators annihilating $p(x \mid x^-, u)$ and $p(y \mid x)$ can be derived from the differential operators $\partial_w + w$ and $\partial_v + v$ annihilating the standard Gaussian PDF, via several changes of variables and differential operators:

$$\begin{aligned} w &= x - \frac{4}{5}x^- - u, & v &= y - \frac{2x}{1+x^2}, \\ \partial_w = \partial_x &= -\frac{5}{4}\partial_{x^-} = -\partial_u, & \partial_v = \partial_y &= -\frac{1}{\partial_x \bullet \left(\frac{2x}{1+x^2}\right)}\partial_x. \end{aligned}$$

The ideals I^s and I^o are then obtained as follows.

$$\begin{aligned} I^s = \big\langle & -25\partial_{x^-} + 20x - 16x^- - 20u, \\ & 5\partial_x + 5x - 4x^- - 5u, -5\partial_u + 5x - 4x^- - 5u \big\rangle, \end{aligned} \tag{5.22}$$

$$I^o = \big\langle (1+x^2)^3\partial_x + 2(x^4-1)y - 4x(x^2-1), (1+x^2)\partial_y + (1+x^2)y - 2x \big\rangle. \tag{5.23}$$

Because both of these ideals are holonomic, Theorem 5.4 guarantees that the functions $\Phi^{(1)}$, $\Phi^{(2)}$, and ψ defined in (5.10)–(5.12) are holonomic. For this example, both $\Phi^{(1)}$ and $\Phi^{(2)}$ are scalar-valued, so we denote $\Phi^{(1)}$ and $\Phi^{(2)}$ by $\phi^{(1)}$ and $\phi^{(2)}$, respectively, and we omit the subscripts for the holonomic ideals $I^{(1)}$ and $I^{(2)}$ and vector-valued functions $Q^{(1)}$ and $Q^{(2)}$ that correspond to $\phi^{(1)}$ and $\phi^{(2)}$, respectively. To compute the bases of holonomic ideals $I^{(1)}$, $I^{(2)}$, and I^{ψ}, we use `Risa/Asir` [82] and its library `nk_restriction`. This library has functions to compute the holonomic ideals for multiplication and integration (Theorems 5.2 and 5.3), and they can be used to compute the bases of the holonomic ideals. As an example of a result, the basis of $I^{(1)}$ consists of 18 differential operators in $\mathcal{D}_4 = \mathbf{R}[\mu^-, \Lambda^-, u, y]\langle \partial_{\mu^-}, \partial_{\Lambda^-}, \partial_u, \partial_y \rangle$.

For simplicity of notation, we denote the vector of variables $[\mu^- \ \Lambda^- \ u \ y]^\top$ by $X = [X_1 \ X_2 \ X_3 \ X_4]^\top$ and fix $\mathcal{D}_4 = \mathbf{R}[X]\langle\partial\rangle$ and $\mathcal{R}_4 = \mathbf{R}(X)\langle\partial\rangle$. The Pfaffian systems for all ideals $I^{(1)}$, $I^{(2)}$, and I^{ψ} are computed from their bases by using

the library `yang` implemented on `Risa/Asir`. In this example, the set of multi-index vectors in Theorem A.2 is common for all the ideals, and thus, all the $\mathbf{R}(X)$-vector spaces $\mathcal{R}_4/I^{(1)}$, $\mathcal{R}_4/I^{(2)}$, and $\mathcal{R}_4/I^{\psi}$ are spanned by the equivalence classes of $\{1, \partial_2, \partial_3, \partial_4, \partial_2\partial_3, \partial_3\partial_4, \partial_4^2\}$. For example, by applying these differential operators to the function $\phi^{(1)}$, we obtain a vector-valued function

$$Q^{(1)} : \mathbf{R}^4 \ni X \mapsto [\phi^{(1)}(X)\ \partial_2\phi^{(1)}(X)\ \partial_3\phi^{(1)}(X) \\ \partial_4\phi^{(1)}(X)\ \partial_2\partial_3\phi^{(1)}(X)\ \partial_3\partial_4\phi^{(1)}(X)\ \partial_4^2\phi^{(1)}(X)]^\top \in \mathbf{R}^7.$$

This vector-valued function corresponds to $Q(X)$ in the definition of a Pfaffian system (5.16). Finally, we have the following three Pfaffian systems:

$$\partial_i \bullet Q^{(1)}(X) = A_i^{(1)}(X)Q^{(1)}(X), \quad (i = 1, \ldots, 4), \tag{5.24}$$

$$\partial_i \bullet Q^{(2)}(X) = A_i^{(2)}(X)Q^{(2)}(X), \quad (i = 1, \ldots, 4), \tag{5.25}$$

$$\partial_i \bullet Q^{\psi}(X) = A_i^{\psi}(X)Q^{\psi}(X), \quad (i = 1, \ldots, 4), \tag{5.26}$$

which correspond to $\phi^{(1)}$, $\phi^{(2)}$, and ψ, respectively. All the coefficient matrices $A_i^{(1)}$, $A_i^{(2)}$, and A_i^{ψ} are matrix-valued functions from $\mathbf{R}^4$ to $\mathbf{R}^{7\times 7}$. All the computations to obtain the Pfaffian systems from the bases of ideals I^-, I^s, and I^o were performed on a PC (Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz; RAM: 16 GB) and took less than 3 seconds in total.

By integrating the Pfaffian systems (5.24)–(5.26) from a certain starting point X_s to a target point X_t, we can obtain the values of functions $\phi^{(1)}(X_t)$, $\phi^{(2)}(X_t)$, and $\psi(X_t)$ as the first components of the corresponding vectors. Consequently, the mean μ and variance Σ of the posterior PDF can be obtained as $\mu = \phi^{(1)}(X_t)/\psi(X_t), \quad \Sigma = \phi^{(2)}(X_t)/\psi(X_t) - \mu^2$. Note that, for this example, Pfaffian system (5.24) has a singular locus defined by the following equation:

$$(50X_2 + 32)X_4 + 25X_2X_3 + 20X_1X_2 = 0, \tag{5.27}$$

when $X_2 = \Lambda^- > 0$. Hence, the terminal points should be selected so that none of the integration paths intersect the singular locus.

5.5.2 Computation of mean and variance

To show that Algorithm 5 yields the exact mean and variance, we computed the algorithm for several data inputs. We fixed the starting point as $X_s = [1.0\ 1.0\ 0.0\ 0.0]^\top$ and selected the target points as $X_t = [1.0\ 1.0\ \tilde{u}\ \tilde{y}]^\top$, where $\tilde{u}$ and $\tilde{y}$ were each chosen from $\{1.0, 1.5, 2.0, 2.5, 3.0\}$ so that the number of target points was 25 in total. The

integration path was defined as $X(t) = (1-t)X_s + tX_t$ for each target point. The initial vectors for $Q^{(1)}$, $Q^{(2)}$, and Q^{ψ} at X_s and the true values of μ and Σ at the target points were computed by using `Maple`. To solve the Pfaffian systems, the Adams-Bashforth-Moulton predictor-corrector method of order 4 (ABM4) was implemented in `Python`, and the Runge-Kutta method (RK4) was used to initialize the first three steps. Because the evaluation of the coefficients $A_i^{(1)}$, $A_i^{(2)}$, and A_i^{ψ} in (5.24)–(5.26) has a high computational cost, the evaluations were implemented using `Cython`. Each integration path was discretized so that the discretization interval $\|X(t+\Delta t) - X(t)\|$ was equal to 0.2. Hence, a target point more distant from the starting point required a larger number of iterations and more computational time.

For comparison, we also computed μ and Σ by the EKF, UKF, and PF, which were also implemented in `Python`. Specifically, the one-step estimations of the EKF, UKF, and PF were performed for the previous estimate μ^- and variance $\Sigma^- = (\Lambda^-)^{-1}$ by using each pair of the input $\tilde{u}$ and measurement $\tilde{y}$. Figure 5.1 shows the errors of the means and variances computed by the proposed method, EKF, UKF, and PF with $1,000$ particles. We can see that the proposed method yielded more accurate values of the mean and variance than the other methods did.

To show the efficiency of the proposed method, the numerical integrations were also performed by using `Maple` with three significant digits, which approximately corresponds to the accuracy of the proposed method. Table 5.1 compares the computational times of all the methods. The proposed method was faster than the PF and numerical integration performed by `Maple`, which indicates its efficiency. Although the EKF and UKF were much faster, the proposed method is still useful for applications requiring high estimation accuracy with a long time constant.

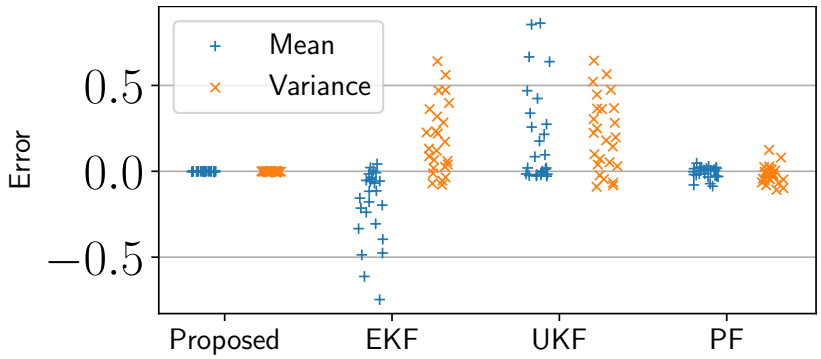


Figure 5.1: Errors of means and variances computed by proposed method, EKF, UKF, and PF. Each dot corresponds to each target point (i.e., pair of input and measurement).

Table 5.1: Minimum, average, and maximum computational times [s] of proposed method, EKF, UKF, PF, and Maple for all pairs of input and measurement.

	Minimum	Average	Maximum
Proposed	2.6×10^{-3}	3.9×10^{-3}	5.5×10^{-3}
EKF	7.0×10^{-5}	8.2×10^{-5}	1.8×10^{-4}
UKF	2.0×10^{-4}	2.3×10^{-4}	4.7×10^{-4}
PF	9.8×10^{-2}	1.1×10^{-1}	1.3×10^{-1}
Maple	1.6×10^{-2}	3.1×10^{-2}	9.4×10^{-2}

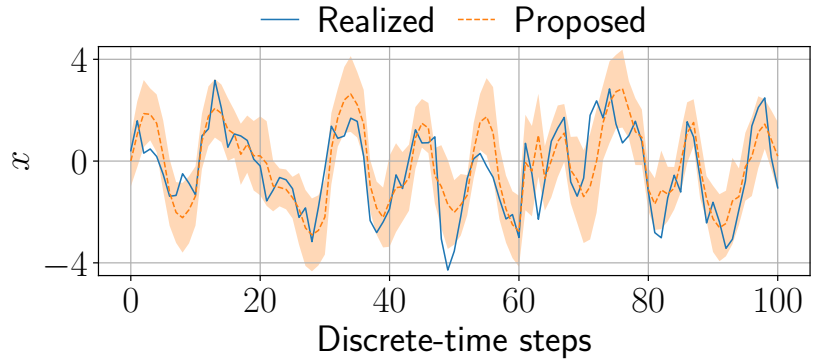
5.5.3 State estimation example

Finally, for the same problem setting as above, we performed a state estimation procedure by using the proposed method. The singular locus defined by equation (5.27) divides the Euclidian space $\mathbf{R}^4$ into two regions: $C_+ := \{X \in \mathbf{R}^4 \mid \text{left-hand side (l.h.s.) of (5.27)} > 0\}$ and $C_- := \{X \in \mathbf{R}^4 \mid \text{l.h.s. of (5.27)} < 0\}$. Therefore, we chose the starting points as $X_{s+} = [1.0, 1.0, 0.0, 0.0]^\top \in C_+$ and $X_{s-} = [1.0, 1.0, -1.0, -1.0]^\top \in C_-$, and we used `Maple` to precompute their corresponding initial vectors. The region to which a given X belongs is determined by substituting it for the l.h.s. of (5.27), and the integration path is defined as in line 3 of Algorithm 5 by setting $X_s = X_{+(-)}$ and $X_t = X$. By repeatedly running Algorithm 5, the mean and variance of the current state is estimated from the previous estimated mean and variance, the previous input, and the observed current output. Figure 5.2 shows a realization of the system in (5.18) and (5.19) and the mean μ and variance Σ estimated by the proposed method.

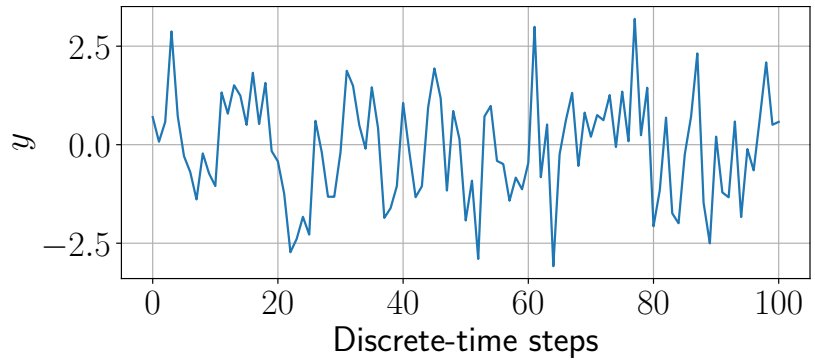
For comparison, the proposed method, EKF, UKF, and PF with 30 particles were performed for 300 realizations. The number of particles for the PF was determined so that its computational time was approximately equal to that of the proposed method. Figure 5.3 shows the root mean square errors (RMSEs) of all the methods averaged over all realizations at each time step. We can see that the EKF had the largest RMSE at every time step, and that the proposed method slightly outperformed the UKF and PF.

As another performance measure besides the RMSE, the negative log-likelihood (NLL) [85] was used to compare all the methods except for the EKF. The NLL is defined by using the estimated mean μ and variance Σ in the following way:

$$\mathrm{NLL}(\mu, \Sigma) := \frac{1}{2}\log|\Sigma| + \frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu) + \frac{n}{2}\log(2\pi), \tag{5.28}$$



(a) Trajectories of realized state (solid) and estimates (dashed). Area between $\mu \pm \sqrt{|\Sigma|}$ is filled



(b) Trajectory of observed output

Figure 5.2: Trajectories for realization of system in (5.18) and (5.19)

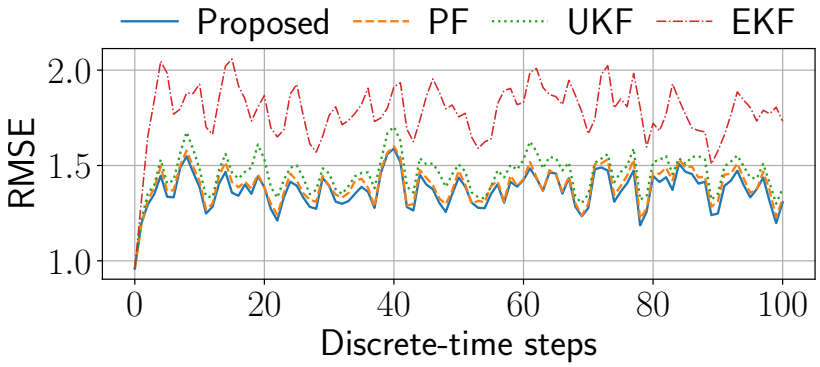


Figure 5.3: RMSE for proposed method (solid), PF (dashed), UKF (dotted), and EKF (dash-dotted)

where x is the true current state. Because the third term in definition (5.28) is a constant, it has no effect on the comparison and is thus omitted hereafter. Figure 5.4 shows the resulting NLLs. The proposed method obviously outperformed the UKF and PF. Roughly speaking, the NLL can evaluate the confidence of an estimate via

the first term in definition (5.28), as well as the error of the estimate in the second term. Therefore, the proposed method yields a good estimate in the sense of Bayesian estimation rather than in the sense of point estimation.

Lastly, Fig. 5.5 summarizes the computational times for all the methods. The range of computational times for the proposed method is much wider than that for the PF, although the means are approximately the same. This is because the computational time for one-step estimation with the proposed method depends on the distance between a given starting point X_s and the target point X_t at which we want to evaluate the mean and variance. Therefore, by preparing more starting points and corresponding initial vectors scattered in the space of X, the computational time would be further reduced.

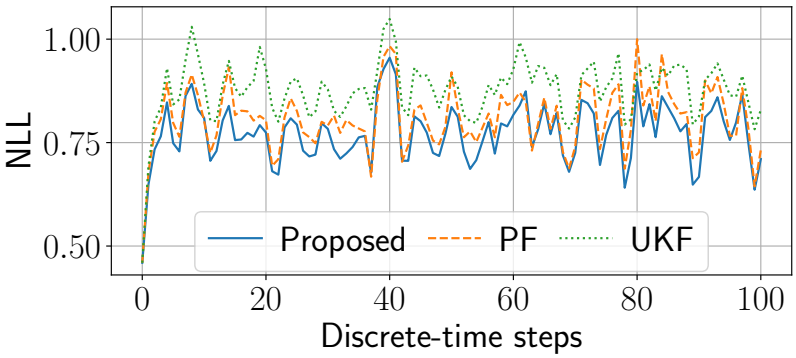


Figure 5.4: NLL for proposed method (solid), PF (dashed), and UKF (dotted)

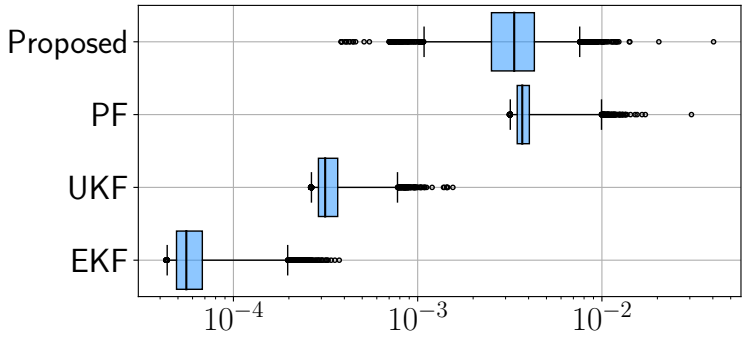


Figure 5.5: Computational times [s] of proposed method, PF, UKF, and EKF. Edges of whiskers indicate 0.5th and 99.5th percentiles

5.6 Summary

We have proposed a Bayesian filtering method that exactly computes the mean and variance of the posterior PDF. By assuming that all the integrands are holonomic functions, the integrations to compute the posterior PDF can be performed by symbolic computation of PDEs without introducing any approximations. The obtained PDEs can be converted to a Pfaffian system, i.e., a set of first-order PDEs for a vector-valued function. The Pfaffian system can be further converted to a set of ODEs by specifying an integration path; then, the mean and variance can be obtained efficiently by numerical integration methods such as RK4 and ABM4.

For future work, we will study the choice of integration path by considering the singular loci of Pfaffian systems. As another direction, rather than approximation of the posterior PDF by a Gaussian, the higher-order moments of the posterior PDF can be computed and propagated to the next time step, which will improve the estimation accuracy.

Chapter 6

Limit Operation in Projective Space for Constructing Necessary Optimality Condition of Polynomial Optimization Problem

6.1 Introduction

Optimality conditions are fundamental to nonlinear programming (NLP). They theoretically play an essential role in the analysis of optimization problems and practically provide important tools to solve optimization problems. A variety of optimality conditions have been investigated for optimization problems with both equality and inequality constraints [2, 8, 86, 87]. Among these, the Karush-Kuhn-Tucker (KKT) conditions are most popular and have become the basis of many numerical algorithms for solving NLP problems, such as Newton's method, the interior point method, and the augmented Lagrangian method [9, 88].

The KKT conditions cannot play the role of necessary optimality conditions on their own; they need an additional condition on minimizers called a constraint qualification (CQ). For example, the first-order necessary condition for optimality consists of the KKT conditions and the linear independence CQ (LICQ), where it is assumed that the gradients of all equality constraints and active inequality constraints are linearly independent at minimizers. There are various CQs, including the Slater CQ (SCQ), Mangasarian-Fromovitz CQ (MFCQ), Abadie's CQ (ACQ), and Guignard's CQ (GCQ) [2, 36]. GCQ is in a sense the weakest CQ that ensures the KKT conditions are necessary optimality conditions [37]. In other words, a local minimizer that violates the GCQ cannot be obtained by solving the KKT conditions.

Let us give a simple example in which the KKT conditions are no longer necessary conditions for optimality. Consider an NLP problem that minimizes a cost function x_1+x_2 under equality constraints $(x_1-1)^2+x_2^2-1=0$ and $(10x_1-8)^2+(5x_2)^2-64=0$. The feasible set of this problem consists of three points, which are shown along with the contours of the cost function located at them in Fig. 6.1. Obviously, this problem

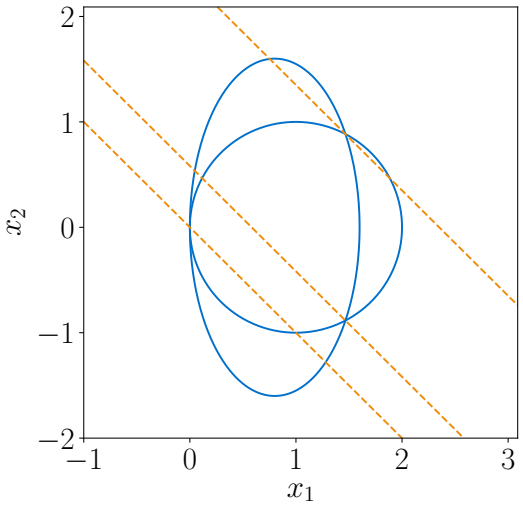


Figure 6.1: Feasible set (intersections of solid curves) and contours of cost function (dashed lines).

has a global minimizer at $(x_1, x_2) = (0, 0)$. However, at this minimizer, the constraints violate the GCQ, and hence the minimizer does not satisfy the KKT conditions. For the following discussion, we say that a minimizer is *non-KKT type* if it does not satisfy the KKT conditions. Note that there are many optimization methods based on the KKT conditions (such as the augmented Lagrangian method) and that none of them can find non-KKT type minimizers.

As necessary optimality conditions for all local minimizers, including non-KKT type ones, Andreani et al. [38] proposed the approximate-KKT (AKKT) conditions. Roughly speaking, these conditions claim that each local minimizer is the convergence point of a sequence consisting of points that approximately satisfy the KKT conditions. Such optimality conditions described by the existence of a convergent sequence are called *sequential optimality conditions* [38] and are the subject of intensive investigation these days [38, 86, 87, 89]. Although AKKT conditions are especially useful for giving a termination criterion in numerical optimization algorithms [38, 90], they are difficult to verify for a given candidate. Indeed, to determine whether the given candidate can be a minimizer or not, we have to show *the existence of a sequence*,

which is more difficult than, for example, showing whether or not the KKT conditions are satisfied at the candidate. From this viewpoint, it is important to develop *a necessary condition described by equations like the KKT conditions but satisfied even by non-KKT type minimizers.*

To find such a new necessary optimality condition, we first consider the quadratic penalty function method, which is a conventional method to relax a constrained NLP into an unconstrained one. With this method, if global minimizers of the penalty function converge to a point as a penalty parameter goes to infinity, the convergence point is a global minimizer of the original NLP. This convergence property does not assume any CQ, so the property is valid even for non-KKT type global minimizers. This is why we select the quadratic penalty function method as the starting point for finding a new necessary optimality condition. Indeed, it has previously been proven by using a certain penalty function method that the AKKT conditions are valid for all minimizers [38]. More precisely, that proof guarantees the existence of a sequence for each local minimizer that converges to the minimizer and that consists of stationary points of a certain penalty function. This is the basis of our proposed condition.

The stationary points of a penalty function can be viewed as functions of the penalty parameter defined by an implicit function representation consisting of the stationary conditions. This leads us to consider the limit operation of the penalty parameter to infinity in the stationary conditions. Note that just substituting infinity for the penalty parameter is meaningless because this ends up neglecting the terms from the cost function in the stationary conditions. Therefore, we utilize symbolic computation to perform the limit operation of the penalty parameter precisely. First, we extend the equations obtained from the stationary conditions defined on a Euclidian space to those defined on a projective space, which can be regarded as a union of a Euclidian space and its points at infinity. Next, we make some changes of variables through the projective space and transform the limit operation to infinity into the limit operation to zero. Finally, we use a tangent cone to precisely perform the limit operation to zero and to obtain the new necessary condition by using symbolic computation techniques from algebraic geometry.

Note that the Fritz-John (FJ) conditions [2] are also satisfied by all local minimizers and described by equations. However, the points that satisfy the FJ conditions often include an infinite number of points that are neither locally optimal nor even KKT. In fact, we can make all feasible points satisfy the FJ conditions by replacing an equality constraint with two inequality constraints or by adding a redundant in-

equality constraint [91]. In Section 6.5, we show a case where the new condition has finite solutions even though the FJ conditions have infinite ones.

In this chapter, we consider nonlinear constrained optimization problems (COPs) defined as

$$\begin{aligned} &\min_{x} f(x) \\ &\text{s. t. } g(x) = 0, \end{aligned} \tag{6.1}$$

where $x \in \mathbf{R}^n$ is a vector of indeterminates, $f : \mathbf{R}^n \to \mathbf{R}$ is a cost function, and $g : \mathbf{R}^n \to \mathbf{R}^m$ is a vector-valued function describing the constraints. We assume that functions f and g consist of polynomials to utilize symbolic computation techniques from algebraic geometry. We also assume that the feasible set

$$\mathcal{X} := \{x \in \mathbf{R}^n \mid g(x) = 0\}$$

is nonempty throughout this chapter. Note that although formulation (6.1) includes only equality constraints, it can also handle inequality constraints by introducing slack variables. Specifically, by introducing a slack variable $s \in \mathbf{R}$, an inequality constraint $\tilde{g}(x) \leq 0$ is converted into an equality constraint $g(x, s) := \tilde{g}(x) + s^2 = 0$. Therefore, we can solve a COP with the equality constraint $g(x, s) = 0$ and indeterminates $[x^\top \ s]^\top \in \mathbf{R}^{n+1}$ instead of the original COP with inequality. It is well-known that the converted COP is equivalent to the original one in terms of minimizers, and this conversion technique has been used by many researchers from the 1960s [47] onward [8,48]. For COPs (6.1), we provide a new necessary optimality condition with no CQs that is satisfied by all local minimizers.

This chapter is organized as follows. Section 6.2 introduces the quadratic penalty function method and demonstrates the existence of a stationary point sequence that converges to each minimizer of the original COP. In Section 6.3, we show how the limit of a penalty parameter to infinity can be reduced to the limit of an additional parameter to the origin by utilizing a projective space. This transformation of a problem enables us to precisely compute the limit points of the trajectories that the stationary points of a penalty function move along. After that, in Section 6.4, we propose an algorithm to symbolically obtain the equations holding at the limit points by means of a tangent cone. In Section 6.5, examples are provided to illustrate the methodology. We summarize this chapter in Section 6.6 with brief mention of future work.

6.2 Penalty Function Method and its Convergence Property

The quadratic penalty function $P(x;r)$ for COP (6.1) is defined as

$$P(x;r) := f(x) + rg^\top(x)g(x), \tag{6.2}$$

where $r \in \mathbf{R}$ is a penalty parameter. It is well-known that the convergence points of the global minimizers of $P(x;r)$, if any, are also the global minimizers of the original COP [8, 9]. As a more practical property of the quadratic penalty function method, such convergence property for local minimizers has been also established. Let X^* be an isolated compact set of local minimizers of COP (6.1) that corresponds to a certain local minimum value. It is proven that there exists a sequence of local minimizers of $P(x;r)$ that converges to a *certain* point of X^* [92]. As a straightforward consequence of this convergence property, if X^* consists of only one minimizer, the existence of the sequence converging to the minimizer is guaranteed. However, if this is not the case, that is, if X^* consists of non-isolated minimizers, the existence of such a sequence is not guaranteed for *each* minimizer of X^*.

As necessary optimality conditions for all minimizers, the AKKT conditions were proposed in [38]. Roughly speaking, AKKT conditions state that, for any local minimizer, there exists a sequence of points converging to the local minimizer, where all points in the sequence approximately satisfy the KKT conditions. In other words, the existence of such a sequence is a necessary condition for all minimizers including even non-isolated ones. This is proved by using the convergence property of the Internal-External Penalty method in [92], and the proof gives a viewpoint on such a sequence not from the KKT conditions but from the penalty function method, which is the basis of our result.

Let us define a localized penalty function, which is localized to a specific local minimizer $\hat{x}$:

$$P^{\mathrm{loc}}(x;r,\hat{x}) := f(x) + \|x-\hat{x}\|^2 + rg^\top(x)g(x). \tag{6.3}$$

The reason we call it localized is as follows. For any local minimizer $\hat{x}$, there exists a real number $\varepsilon > 0$ such that a minimization problem restricted on a ball $\mathcal{B}_\varepsilon(\hat{x}) := \{x \in \mathbf{R}^n \mid \|x-\hat{x}\| \leq \varepsilon\} \subset \mathbf{R}^n$, as

$$\begin{aligned} \min_{x \in \mathcal{B}_\varepsilon(\hat{x})} \ & f(x) + \|x-\hat{x}\|^2 \\ \text{s. t. } & g(x) = 0, \end{aligned} \tag{6.4}$$

has a *unique* local minimizer $x = \hat{x}$, and $P^{\mathrm{loc}}(x; r, \hat{x})$ is the penalty function for this minimization problem localized to $\hat{x}$. By using this localized penalty function, a necessary optimality condition described by the existence of a stationary point sequence can be derived as follows.

Theorem 6.1. Let $r_{[1:\infty]}$ be a sequence monotonically going to infinity as k does, $\hat{x}$ be a minimizer of the original COP (6.1), and $P^{\mathrm{loc}}(x; r, \hat{x})$ be a function defined as (6.3) with domain of definition $\mathbf{R}^n$. Then, there exists an integer $k^* > 0$ such that there exists a sequence of stationary points $x_{[k^*:\infty]}$ satisfying

$$P_x^{\mathrm{loc}}(x_k; r_k, \hat{x}) = 0 \tag{6.5}$$

and converging to $\hat{x}$ as k goes to infinity. In other words, the existence of such a sequence $x_{[k^*:\infty]}$ is a necessary condition for $\hat{x}$ to be a local minimizer.

For the proof of Theorem 6.1, see Section B.2

Remark 6.1. Since the sequence $x_{[k^*:\infty]}$ in Theorem 6.1 is also an infinite sequence, if we replace its subscripts appropriately, we can assume $k^* = 1$ without loss of generality.

Remark 6.2. Note that although constraint $x \in \mathcal{B}_\varepsilon(\hat{x})$ is imposed in the localized minimization problem (6.4), this is no longer assumed in Theorem 6.1. Due to the lacking this constraint, equation (6.5) can have a solution $\tilde{x}_k$ that does not belong to $\mathcal{B}_\varepsilon(\hat{x})$ and hence does not converge to $\hat{x}$. However, this fact has nothing to do with the existence of the sequence $x_{[1:\infty]}$ introduced in the proof, and Theorem 6.1 still gives a necessary optimality condition to be satisfied by every local minimizer $\hat{x}$.

The stationary condition $P_x^{\mathrm{loc}}(x; r, \hat{x}) = 0$ is written as

$$P_x^{\mathrm{loc}}(x; r, \hat{x}) = f_x(x) + 2(x - \hat{x}) + r g_x(x) g(x) = 0. \tag{6.6}$$

When we try to use Theorem 6.1 to find a minimizer of COP (6.1), we have to solve equation (6.6) to find the sequence of stationary points $x_{[1:\infty]}$. However, equation (6.6) is defined using the minimizer $\hat{x}$, which we are seeking now. This deadlock can be avoided by regarding the minimizer $\hat{x}$ as an additional variable $y \in \mathbf{R}^n$, that is, regarding equation (6.6) as an equation defined on $\mathbf{R}^{2n+1}$:

$$P_x^{\mathrm{loc}}(x; r, y) = f_x(x) + 2(x - y) + r g_x(x) g(x) = 0, \tag{6.7}$$

where $x = [x_1 \ \cdots \ x_n]^\top$ and $y = [y_1 \ \cdots \ y_n]^\top$. By computing the convergence points of sequences $x_{[1:\infty]}$ for all $y \in \mathbf{R}^n$ and selecting pairs (x_∞, y) such that $x_\infty = y$ holds,

we can find candidates of minimizers that satisfy the necessary optimality condition stated as Theorem 6.1. For all $y \in \mathbf{R}^n$ admitting the sequences $r_{[1:\infty]}$ and $x_{[1:\infty]}$ that satisfy equation (6.7) for every k, symbolic computation enables us to compute a set of equations satisfied by every limit x_∞. The derived equations can be solved without iteratively increasing the penalty parameter, unlike other numerical algorithms for the penalty function method.

6.3 Limit Points in Projective Space

In this section, we utilize the concepts of projective space, homogenization, and dehomogenization to deal with the limit operation of the penalty parameter symbolically. When we regard equation (6.7) as an equation defined on $\mathbf{R}^{2n+1}$, the convergence points of its solutions as r goes to infinity lie in the hyperplane at infinity of $\mathbf{R}^{2n+1}$. In this section, we fix certain sequences ($r_{[1:\infty]}$ and $x_{[1:\infty]}$) such that r_k monotonically goes to infinity as k does, x_k converges to a point $x_\infty \in \mathbf{R}^n$, and all points $\begin{bmatrix} x_k^\top & r_k \end{bmatrix}^\top \in \mathbf{R}^{n+1}$ satisfy equation (6.7) for a fixed $y \in \mathbf{R}^n$. Moreover, for the following discussion, we denote the set consisting of all the components of $P_x^{\mathrm{loc}}(x; r, y)$ by F, that is,

$$F := \left\{ \left[P_x^{\mathrm{loc}}(x; r, y)\right]_1, \ldots, \left[P_x^{\mathrm{loc}}(x; r, y)\right]_n \right\} \subset \mathbf{R}[x, y, r], \tag{6.8}$$

where $\left[P_x^{\mathrm{loc}}(x; r, y)\right]_i$ is the i-th component of $P_x^{\mathrm{loc}}(x; r, y)$.

Let us consider a projective space $\mathbf{P}^{2n+1}$ and identify $\mathbf{R}^{2n+1}$ with an open subset,

$$U_0 := \{[X_0 : \cdots : X_{2n+1}] \in \mathbf{P}^{2n+1} \mid X_0 \neq 0\}, \tag{6.9}$$

by a homeomorphism $\phi_0 : \mathbf{R}^{2n+1} \to U_0$ that sends $[x_1 \ \cdots \ x_n \ y_1 \ \cdots \ y_n \ r]^\top$ to $[1 : x_1 : \cdots : x_n : y_1 : \cdots : y_n : r]$. As mentioned in subsection A.2.2, the other open subset,

$$U_{2n+1} := \{[X_0 : \cdots : X_{2n+1}] \in \mathbf{P}^{2n+1} \mid X_{2n+1} \neq 0\}, \tag{6.10}$$

includes almost all points of the hyperplane at infinity H_0 of U_0; indeed, all the limit points we need. This is because the set difference

$$H_0 \setminus U_{2n+1} = \{[X_0 : \cdots : X_{2n+1}] \in \mathbf{P}^{2n+1} \mid X_0 = X_{2n+1} = 0\}$$

only includes the points at infinity of U_0 where the $(2n+1)$-coordinate, which corresponds to the penalty parameter r, is exactly zero.

There is also a homeomorphism $\phi_{2n+1} : \mathbf{R}^{2n+1} \to U_{2n+1}$ that sends a point $[\rho\ \xi_1\ \cdots\ \xi_n\ \eta_1\ \cdots\ \eta_n]^\top$ to the equivalent class $[\rho : \xi_1 : \cdots : \xi_n : \eta_1 : \cdots : \eta_n : 1]$. To construct a mapping from x-y-r space to ξ-η-ρ space, consider two open subsets:

$$D_r := \left\{ \begin{bmatrix} x^\top & y^\top & r \end{bmatrix}^\top \in \mathbf{R}^{2n+1} \mid r \neq 0 \right\} \subset \mathbf{R}^{2n+1},$$
$$D_\rho := \left\{ \begin{bmatrix} \rho & \xi^\top & \eta^\top \end{bmatrix}^\top \in \mathbf{R}^{2n+1} \mid \rho \neq 0 \right\} \subset \mathbf{R}^{2n+1},$$

and the restrictions of ϕ_0 and ϕ_{2n+1}^{-1}:

$$\phi_0|_{D_r} :\ D_r \ni \begin{bmatrix} x^\top & y^\top & r \end{bmatrix}^\top \mapsto [1 : x_1 : \cdots : x_n : y_1 : \cdots : y_n : r] \in U_0 \cap U_{2n+1},$$
$$\phi_{2n+1}^{-1}|_{U_0 \cap U_{2n+1}} :\ U_0 \cap U_{2n+1} \ni [X_0 : \cdots : X_{2n+1}] \mapsto \begin{bmatrix} \dfrac{X_0}{X_{2n+1}} & \cdots & \dfrac{X_{2n}}{X_{2n+1}} \end{bmatrix}^\top \in D_\rho.$$

Then, a composite mapping $\Phi := \left(\phi_{2n+1}^{-1}|_{U_0 \cap U_{2n+1}}\right) \circ \left(\phi_0|_{D_r}\right)$ is a homeomorphism between D_r and D_ρ defined by

$$\begin{bmatrix} x^\top & y^\top & r \end{bmatrix}^\top \mapsto \begin{bmatrix} \rho & \xi^\top & \eta^\top \end{bmatrix}^\top = \Phi\left(\begin{bmatrix} x^\top & y^\top & r \end{bmatrix}^\top\right) = \frac{1}{r}\begin{bmatrix} 1 & x^\top & y^\top \end{bmatrix}^\top. \tag{6.11}$$

Note that both closures $\overline{D_r}$ and $\overline{D_\rho}$ are equal to $\mathbf{R}^{2n+1}$. Through the homeomorphism Φ, we obtain the triplet of sequences $\rho_{[1:\infty]}$, $\xi_{[1:\infty]}$, and $\eta_{[1:\infty]}$ as the image of sequences $r_{[1:\infty]}$ and $x_{[1:\infty]}$ under Φ with a fixed y.

The sequences $r_{[1:\infty]}$ and $x_{[1:\infty]}$ satisfy equation (6.7) for a parameter y, or in other words, every point $[x_k^\top\ y^\top\ r_k]^\top \in \mathbf{R}^{2n+1}$ belongs to $\mathcal{V}(F)$. This indicates there exists an equation satisfied by $\rho_{[1:\infty]}$, $\xi_{[1:\infty]}$, and $\eta_{[1:\infty]}$ for all $k = 1, \ldots, \infty$, and such equation can be obtained by homogenization and dehomogenization defined in subsection A.2.3, as explained below. Let us define a set of homogeneous polynomials $F^{\mathrm{hom}} \subset \mathbf{R}[X_0, \ldots, X_{2n+1}]$ as

$$F^{\mathrm{hom}} := \left\{ f^{\mathrm{hom}} \in \mathbf{R}[X_0, \ldots, X_{2n+1}] \mid f \in F \right\}, \tag{6.12}$$

where $f^{\mathrm{hom}}(X_0, \ldots, X_{2n+1})$ is the homogenization of $f(x_1, \ldots, x_n, y_1, \ldots, y_n, r) \in F$ of total degree d, that is,

$$f^{\mathrm{hom}}(X_0, \ldots, X_{2n+1}) := X_0^d f\left(\frac{X_1}{X_0}, \ldots, \frac{X_{2n+1}}{X_0}\right). \tag{6.13}$$

Its dehomogenizations for index $2n+1$ yield the other polynomial set $\mathcal{F} \subset \mathbf{R}[\rho, \xi, \eta]$:

$$\mathcal{F} := F^{\mathrm{hom}}|_{X_0=\rho, X_1=\xi_1, \ldots, X_n=\xi_n, X_{n+1}=\eta_1, \ldots, X_{2n}=\eta_n, X_{2n+1}=1}. \tag{6.14}$$

The sequences $\rho_{[1:\infty]}$, $\xi_{[1:\infty]}$, and $\eta_{[1:\infty]}$ satisfy the algebraic equations

$$(f^{\mathrm{hom}})^{\mathrm{deh}}(\rho, \xi_1, \ldots, \xi_n, \eta_1, \ldots, \eta_n) = 0 \qquad \left(\forall (f^{\mathrm{hom}})^{\mathrm{deh}} \in \mathcal{F}\right)$$

because, for all $[x^\top\ y^\top\ r]^\top \in \mathcal{V}(F) \cap D_r$, we have

$$\begin{aligned}(f^{\mathrm{hom}})^{\mathrm{deh}} \circ \Phi\left(\begin{bmatrix}x^\top & y^\top & r\end{bmatrix}^\top\right) &= (f^{\mathrm{hom}})^{\mathrm{deh}}(r^{-1}, r^{-1}x_1, \ldots, r^{-1}x_n, r^{-1}y_1, \ldots, r^{-1}y_n) \\ &= f^{\mathrm{hom}}(r^{-1}, r^{-1}x_1, \ldots, r^{-1}x_n, r^{-1}y_1, \ldots, r^{-1}y_n, 1) \\ &= r^{-d} f(x_1, \ldots, x_n, y_1, \ldots, y_n, r) \\ &= 0. \end{aligned} \tag{6.15}$$

Now, since the homeomorphism Φ and its inverse Φ^{-1} are both continuous, the following proposition is trivially obtained.

Proposition 6.1. Let the pair of sequences $\rho_{[1:\infty]}$ and $\xi_{[1:\infty]}$ be a part of the image of sequences $r_{[1:\infty]}$ and $x_{[1:\infty]}$ under the homeomorphism Φ with a fixed y. Then,

$$\frac{\eta_k}{\rho_k} = y \tag{6.16}$$

holds for all k. Moreover, for the limit x_∞, sequences $r_{[1:\infty]}$ and $x_{[1:\infty]}$ satisfy

$$\lim_{k\to\infty} r_k = \infty, \tag{6.17}$$

$$\lim_{k\to\infty} x_k = x_\infty \tag{6.18}$$

if and only if $\{\rho_k\}_{k=1}^\infty$ and $\{\xi_k\}_{k=1}^\infty$ satisfy

$$\lim_{k\to\infty} \rho_k = 0, \tag{6.19}$$

$$\lim_{k\to\infty} \frac{\xi_k}{\rho_k} = x_\infty. \tag{6.20}$$

Proposition 6.1 shows that we can obtain the limit of the sequence $\lim_{k\to\infty} x_k$ as the limit of fractions $\lim_{k\to\infty} \xi_k/\rho_k$. Moreover, from

$$\lim_{k\to\infty} \xi_k = \left(\lim_{k\to\infty} \rho_k\right)\left(\lim_{k\to\infty} \frac{\xi_k}{\rho_k}\right) = 0, \quad \lim_{k\to\infty} \eta_k = \left(\lim_{k\to\infty} \rho_k\right)\left(\lim_{k\to\infty} \frac{\eta_k}{\rho_k}\right) = 0,$$

the points $[\rho_k\ \xi_k^\top\ \eta_k^\top]^\top \in \mathbf{R}^{2n+1}$ converge to the origin as $k \to \infty$, which implies that $\mathcal{V}(\mathcal{F})$ contains the origin as a closed subset of the closure $\overline{D_\rho} = \mathbf{R}^{2n+1}$. This indicates that, to compute the limit points in ξ-η-ρ space, we need only the local information about this algebraic set at the origin.

Remark 6.3. Note that Proposition 6.1 and the following discussion also indicate that $\mathcal{V}(\mathcal{F})$ includes the origin if and only if the convergence point x_∞ ($\|x_\infty\| < \infty$) exists. Moreover, Theorem 6.1 guarantees the existence of such a convergence point if at least one local minimizer exists. Therefore, $\mathcal{V}(\mathcal{F})$ includes the origin and thus all the mathematical tools introduced in subsection A.2.4 can be applied whenever COP (6.1) has at least one local minimizer.

Remark 6.4. The whole discussion of this section can be conceptually illustrated as Fig. 6.2. Let y be fixed, and then η is uniquely determined by ρ and y as in (6.16) under Φ. Hence, for simplicity, only the variables x, r, ξ, and ρ are illustrated in Fig. 6.2.

Figure 6.2 shows three spaces: two Euclidean spaces of dimension $n+1$ represented by the vertical and horizontal planes and the Euclidean space of dimension $n+2$ where the former two spaces are embedded as the linear submanifolds $\{[X_0, \ldots, X_{n+1}] \in \mathbf{R}^{n+2} \mid X_0 = 1\}$ and $\{[X_0, \ldots, X_{n+1}] \in \mathbf{R}^{n+2} \mid X_{n+1} = 1\}$, respectively. The vertical plane represents x-r space, while the horizontal one represents ξ-ρ space. As stated in subsection A.2.2, the set of all lines in $\mathbf{R}^{n+2}$ passing through the origin $O_{\mathbf{R}^{n+2}}$ is identical to a projective space $\mathbf{P}^{n+1}$. It is readily seen that for each point of x-r space (blue circles), there exists a unique point of ξ-ρ space (yellow circles) that lies in the line in $\mathbf{R}^{n+2}$ passing through the origin (red lines), or equivalently the "point" of $\mathbf{P}^{n+1}$, that the point of x-r space lies in; this correspondence is what the homeomorphism Φ represents.

The thick blue curve on the vertical plane shows the solution set of $F(x, y, r) = 0$ with a fixed y, while the thick yellow curve on the horizontal plane shows the solution set of $\mathcal{F}(\xi, \eta, \rho) = 0$ with $\eta = y\rho$ (see (6.16)). Recall that every point (x_k, r_k) consisting of $r_{[1:\infty]}$ and $x_{[1:\infty]}$ lies in the blue curve, and every point (ξ_k, ρ_k) consisting

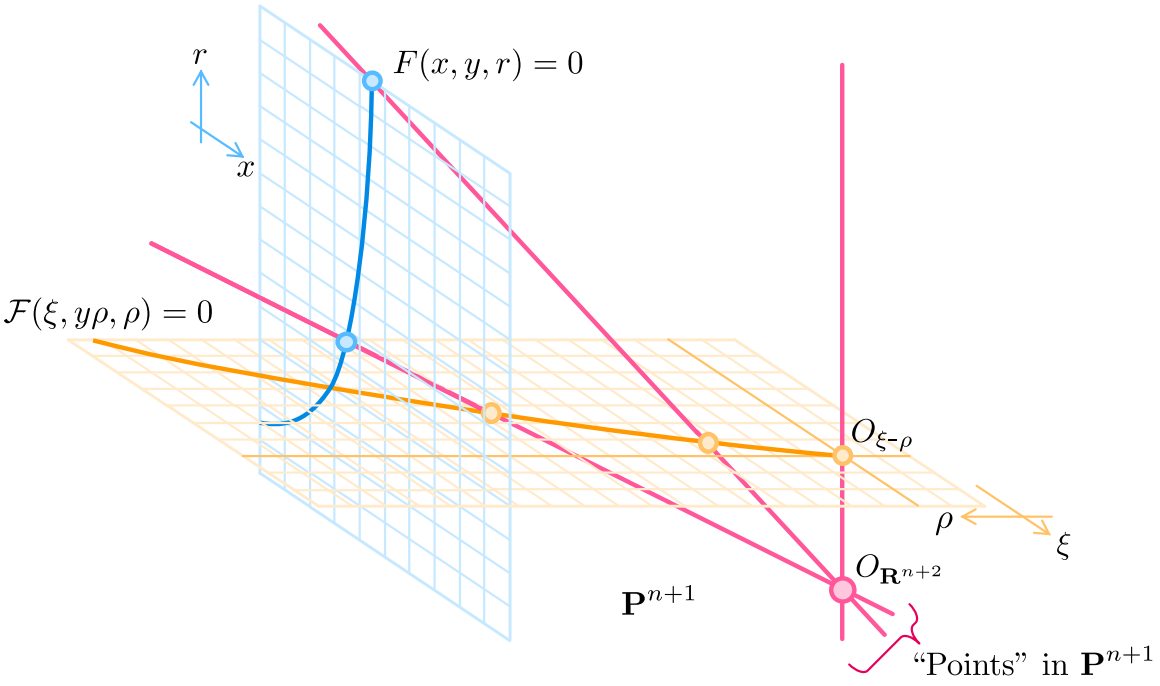


Figure 6.2: Illustration of Proposition 6.1

of $\rho_{[1:\infty]}$ and $\xi_{[1:\infty]}$ lies in the yellow curve. Therefore, conditions (6.17) and (6.18) says that the sequence of points $(x_{[1:\infty]}, r_{[1:\infty]})$ asymptotically approaches to a vertical line whose x-coordinate is x_∞ as k goes to infinity. The corresponding "point" of $\mathbf{P}^{n+1}$ (red lines) moves simultaneously and, in the end, reach to the line parallel to the vertical plane and passing through $O_{\mathbf{R}^{n+2}}$ and $O_{\xi\text{-}\rho}$, the origin of ξ-ρ space. Hence, the sequence $(\xi_{[1:\infty]}, \rho_{[1:\infty]})$ on the yellow curve converges to the origin $O_{\xi\text{-}\rho}$, and the direction the sequence approaches from corresponds to the limit x_∞ (conditions (6.19) and (6.20)).

6.4 Computation of Limit Points and New Necessary Condition for Optimality

To focus on the origin in the algebraic set $\mathcal{V}(\mathcal{F}) \subset \overline{D_\rho} = \mathbf{R}^{2n+1}$, it is useful to consider a tangent cone $C_0(\mathcal{V}(\mathcal{F})) \subset \mathbf{R}^{2n+1}$ at the origin. From Lemma A.4, the tangent cone can be seen as the set of lines that approximates $\mathcal{V}(\mathcal{F})$ in a neighborhood of the origin. Note that the limit of fractions $[x_\infty^\top \ y^\top]^\top = \lim_{k\to\infty}[\xi_k^\top \ \eta_k^\top]^\top/\rho_k$ can be seen as the gradient of such a line with respect to ρ at the origin. This consideration leads us to relate the limit $\lim_{k\to\infty}[\xi_k^\top \ \eta_k^\top]^\top/\rho_k$ with the gradient of each line in $C_0(\mathcal{V}(\mathcal{F}))$ at the origin, as stated in the following lemma.

Lemma 6.1. Let $r_{[1:\infty]}$ be a sequence monotonically tending to infinity and $x_{[1:\infty]}$ be a sequence such that each pair (x_k, r_k) satisfies equation (6.7) with a fixed y. Suppose that $x_{[1:\infty]}$ has a convergence point x_∞. Then, $[1 \ x_\infty^\top \ y^\top]^\top \in C_0(\mathcal{V}(\mathcal{F}))$ holds.

Proof. Let us define sequences $\rho_{[1:\infty]}$, $\xi_{[1:\infty]}$, and $\eta_{[1:\infty]}$ as in Proposition 6.1. From the preceding discussion of the proposition, all polynomials in $\mathcal{F}$ vanish at each triplet (ρ_k, ξ_k, η_k) because its preimage (x_k, y, r_k) under Φ satisfies equation (6.7). Moreover, as mentioned in the proposition and the discussion following it,

$$\lim_{k\to\infty} \left[\rho_k \ \xi_k^\top \ \eta_k^\top\right]^\top = 0 \in \mathbf{R}^{2n+1}$$

holds if $\lim_{k\to\infty} r_k = \infty$ and $\lim_{k\to\infty} x_k = x_\infty$ hold. Since algebraic sets are closed, this convergence implies 0 is an element of $\mathcal{V}(\mathcal{F}) \subset \mathbf{R}^{2n+1}$, and thus we can define the sequence of secant lines $L_{[1:\infty]}$ as those through 0 and the points $q_k := [\rho_k \ \xi_k^\top \ \eta_k^\top]^\top$. Let us parametrize each line L_k by $\{tv_k \mid v_k = q_k/\rho_k, t \in \mathbf{R}\}$; then, Proposition 6.1 readily shows that $L_{[1:\infty]}$ converges to the line:

$$L := \left\{tv \mid v = \left[1 \ x_\infty^\top \ y^\top\right]^\top, t \in \mathbf{R}\right\}$$

as $k \to \infty$. Therefore, from Lemma A.4, $[1 \ x_\infty^\top \ y^\top]^\top \in L \subset C_0(\mathcal{V}(\mathcal{F}))$ holds. □

This lemma shows that the intersection of the tangent cone and a hyperplane

$$C_0(\mathcal{V}(\mathcal{F})) \cap \left\{ \left[\rho\ \xi^\top\ \eta^\top\right]^\top \in \mathbf{R}^{2n+1} \;\middle|\; \rho = 1 \right\}$$

includes all the points of the form $[1\ x_\infty^\top\ y^\top]^\top$ where x_∞ is a limit point, as r goes to infinity, of the sequences $x_{[1:\infty]}$ whose elements satisfy equation (6.7) with a fixed y. In other words, if we have a set of l polynomials $G = \{G_1, \ldots, G_l\} \subset \mathbf{R}[\rho, \xi, \eta]$ defining the tangent cone $C_0(\mathcal{V}(\mathcal{F}))$, $G_i(1, x_\infty, y) = 0$ $(i = 1, \ldots, l)$ holds for every pair of x_∞ and y satisfying the assumptions of Lemma 6.1. Moreover, if $y = \hat{x}$ attains a minimum of COP (6.1), Theorem 6.1 guarantees the existence of a sequence $x_{[1:\infty]}$ that converges to $x_\infty = \hat{x}$. This implies that every minimizer $\hat{x}$ satisfies equation $G_i(1, \hat{x}, \hat{x}) = 0$ $(i = 1, \ldots, l)$. Finally, the whole process to obtain the polynomial set

$$\mathcal{G} := \{G_i(1, x, x) \in \mathbf{R}[x] \mid G_i \in G\}$$

from COP (6.1) can be summarized as Algorithm 6, and by using $\mathcal{G}$, the new necessary condition for optimality can be stated as Theorem 6.2.

Theorem 6.2. Let $\mathcal{G} = \{\mathcal{G}_1, \ldots, \mathcal{G}_l\} \subset \mathbf{R}[x]$ be the set of equations obtained by Algorithm 6. Suppose that $\hat{x}$ is a minimizer of COP (6.1). Then, $\hat{x}$ satisfies the equations

$$\mathcal{G}_i(\hat{x}) = 0 \quad \forall i = 1, \ldots, l. \tag{6.21}$$

Proof. Let $\hat{x}$ be a minimizer of COP (6.1). For a sequence $r_{[1:\infty]}$ monotonically tending to infinity as k does, Theorem 6.1 guarantees that the existence of a sequence $x_{[1:\infty]}$ converging to $\hat{x}$ whose elements satisfy equation $P_x^{\mathrm{loc}}(x_k; r_k, \hat{x}) = 0$ for every k. Then, Lemma 6.1 indicates that $[1\ \hat{x}^\top\ \hat{x}^\top]^\top$ is a point of $C_0(\mathcal{V}(\mathcal{F}))$. Now, Let

Algorithm 6 Symbolic Computation of $\mathcal{G}$

Input: COP (6.1)
Output: Set of polynomial $\mathcal{G} \subset \mathbf{R}[x]$

1: Compute polynomial set $F \subset \mathbf{R}[x, y, r]$ in (6.8) by differentiating localized penalty function $P^{\mathrm{loc}}(x; r, y) = f(x) + \|x - y\|^2 + r g^\top(x) g(x)$
2: Compute $\mathcal{F} \subset \mathbf{R}[\rho, \xi, \eta]$ by computing homogenization as in equation (6.12) and dehomogenization as in equation (6.14)
3: Compute set of generators $G \subset \mathbf{R}[\rho, \xi, \eta]$ of ideal defining tangent cone $C_0(\mathcal{V}(\mathcal{F}))$ from polynomial set $\mathcal{F}$, for example, by using Gröbner basis described in Lemma A.3
4: Define $\mathcal{G}$ as $G|_{\rho=1, \xi=x, \eta=x}$

$G = \{G_1, \ldots, G_l\} \subset \mathbf{R}[\rho, \xi, \eta]$ be a set of generators of an ideal defining $C_0(\mathcal{V}(\mathcal{F}))$, that is, $C_0(\mathcal{V}(\mathcal{F})) = \mathcal{V}(G)$ holds. Since $[1\ \hat{x}^\top\ \hat{x}^\top]^\top \in C_0(\mathcal{V}(\mathcal{F}))$,

$$\mathcal{G}_i(\hat{x}) = G_i(1, \hat{x}, \hat{x}) = 0$$

holds for all $i = 1, \ldots, l$, which completes the proof. □

6.5 Numerical Examples

This section is devoted to two numerical examples. The first one demonstrates the proposed methodology and clarifies the relationships among the penalty function method, homogenization, and the tangent cone. The second one demonstrates how the proposed method can find non-KKT type global minimizers.

Example 1 (Illustrative example) Let us consider the following COP:

$$\begin{aligned} \min_x\ & \frac{1}{2}(x_1^2 + x_2^2) \\ \text{s. t. } & (x_1 - 5)^2 - (x_2 - 4)^3 = 0, \end{aligned} \tag{6.22}$$

where $x = [x_1\ x_2]^\top \in \mathbf{R}^2$ are indeterminates. Figure 6.3 shows the feasible set and contours of the cost function. For this problem, the penalty function of the localized COP (6.4) is obtained as

$$P^{\mathrm{loc}}(x; r, y) = \frac{1}{2}(x_1^2 + x_2^2) + (x_1 - y_1)^2 + (x_2 - y_2)^2 + r\left\{(x_1 - 5)^2 - (x_2 - 4)^3\right\}^2, \tag{6.23}$$

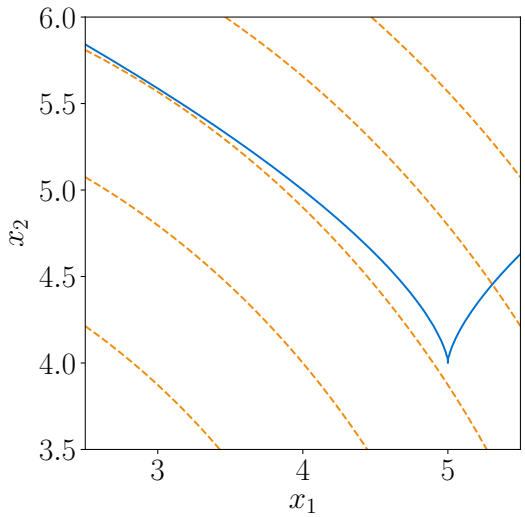


Figure 6.3: Feasible set (solid lines) and contours of cost function (dashed lines).

where $r \in \mathbf{R}$ is a penalty parameter and $y = [y_1\ y_2]^\top$ are additional variables representing the coordinates of a minimizer. The stationary conditions $P_x^{\mathrm{loc}}(x; r, y) = 0$ are the following polynomial equations:

$$\begin{aligned} &x_1 + 2(x_1 - y_1) + 4r\left((x_1-5)^2 - (x_2-4)^3\right)(x_1-5) = 0, \\ &x_2 + 2(x_2 - y_2) - 6r\left((x_1-5)^2 - (x_2-4)^3\right)(x_2-4)^2 = 0. \end{aligned} \tag{6.24}$$

Hence, the polynomial set $F \subset \mathbf{R}[x_1, x_2, y_1, y_2, r]$ in Section 6.3 consists of the left-hand sides of equations (6.24).

By replacing x_1, x_2, y_1, y_2, and r with X_1/X_0, X_2/X_0, X_3/X_0, X_4/X_0, and X_5/X_0, respectively, the homogenizations of F are obtained as

$$X_0^4 X_1 + 2X_0^4(X_1 - X_3) + 4X_5\left(X_0(X_1 - 5X_0)^2 - (X_2 - 4X_0)^3\right)(X_1 - 5X_0), \tag{6.25}$$

$$X_0^5 X_2 + 2X_0^5(X_2 - X_4) - 6X_5\left(X_0(X_1 - 5X_0)^2 - (X_2 - 4X_0)^3\right)(X_2 - 4X_0)^2, \tag{6.26}$$

where $[X_0 : X_1 : X_2 : X_3 : X_4 : X_5] \in \mathbf{P}^5$ is a homogeneous coordinate. The dehomogenizations of polynomials (6.25) and (6.26), denoted as $\mathcal{F}$ in Section 6.4, are as follows:

$$\begin{aligned} &\rho^4\xi_1 + 2\rho^4(\xi_1 - \eta_1) + 4\left(\rho(\xi_1 - 5\rho)^2 - (\xi_2 - 4\rho)^3\right)(\xi_1 - 5\rho), \\ &\rho^5\xi_2 + 2\rho^5(\xi_2 - \eta_2) - 6\left(\rho(\xi_1 - 5\rho)^2 - (\xi_2 - 4\rho)^3\right)(\xi_2 - 4\rho)^2, \end{aligned} \tag{6.27}$$

where $\rho = X_0/X_5$, $\xi_1 = X_1/X_5$, $\xi_2 = X_2/X_5$, $\eta_1 = X_3/X_5$, and $\eta_2 = X_4/X_5$.

From polynomials (6.27), we obtain $G \subset \mathbf{R}[\rho, \xi_1, \xi_2, \eta_1, \eta_2]$, which define the tangent cone $C_0(\mathcal{V}(\mathcal{F}))$, as a set of five polynomials with total degrees 5, 6, 7, 9, and 10. By substituting $\rho = 1$, $\xi = \eta = x$ to the polynomial set G, we have the polynomial set $\mathcal{G}$ consisting of five polynomials:

$$3x_1{x_2}^2 - 22x_1x_2 + 48x_1 - 10x_2, \tag{6.28}$$

$$\begin{aligned} 4x_1{x_2}^3 - 4{x_1}^3 - 48x_1{x_2}^2 - 20{x_2}^3 + 60{x_1}^2 \\ + 192x_1x_2 + 240{x_2}^2 - 556x_1 - 960x_2 + 1780, \end{aligned} \tag{6.29}$$

$$\begin{aligned} 6{x_2}^5 - 6{x_1}^2{x_2}^2 - 120{x_2}^4 + 48{x_1}^2x_2 + 60x_1{x_2}^2 + 960{x_2}^3 \\ - 96{x_1}^2 - 480x_1x_2 - 3990{x_2}^2 + 960x_1 + 8880x_2 - 8544, \end{aligned} \tag{6.30}$$

$$\begin{aligned} 72x_1{x_2}^3 - 108{x_1}^3 - 864x_1{x_2}^2 - 540{x_2}^3 \\ + 1620{x_1}^2 + 3376x_1x_2 + 6600{x_2}^2 - 12324x_1 - 26480x_2 + 48060, \end{aligned} \tag{6.31}$$

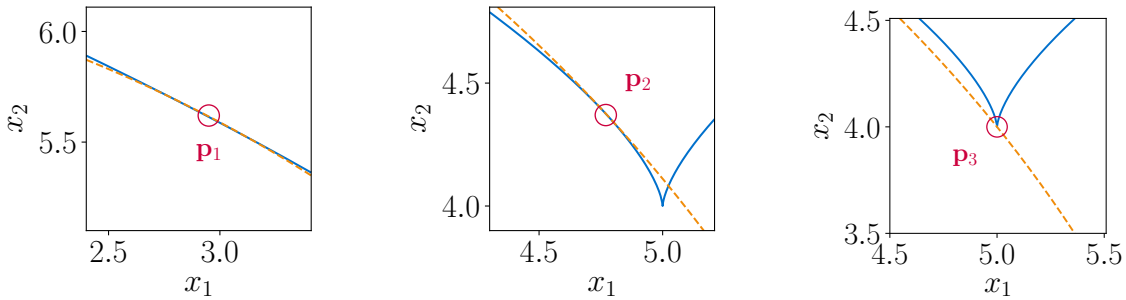
and

$$108{x_1}^4 - 48{x_1}^2{x_2}^2 - 1620{x_1}^3 + 592{x_1}^2x_2 + 6180{x_1}^2 - 2000x_1x_2 + 1800{x_2}^2 - 1980x_1 - 9600x_2. \quad (6.32)$$

We can obtain three candidates of minimizers (listed in Table 6.1) by solving equations (6.28)–(6.32). Theorem 6.2 guarantees that these candidates include all minimizers and, consequently, global minimizers. Figure 6.4 shows the obtained candidates and contours of the cost function corresponding to those candidates. In Fig. 6.4, the contours of the cost function are tangent to the feasible set at the candidate points $\mathbf{p}_1$ and $\mathbf{p}_2$, which are typical situations on the KKT points. In fact, $\mathbf{p}_1$ is a global minimizer, and $\mathbf{p}_2$ is a local maximizer. On the other hand, at $\mathbf{p}_3$, the contour of the cost function does not seem to be tangent to the feasible set; indeed, $\mathbf{p}_3$ has no Lagrange multipliers, so the KKT conditions do not hold. However, as shown in Fig. 6.4(c), $\mathbf{p}_3$ is obviously a local minimizer because any feasible point in its neighborhood lies in the area where the value of the cost function is larger than $f(\mathbf{p}_3)$.

Table 6.1: Candidates derived from proposed necessary condition and corresponding values of cost function.

	$\mathbf{p}_1$	$\mathbf{p}_2$	$\mathbf{p}_3$
x_1	2.95	4.77	5.00
x_2	5.62	4.37	4.00
$f(\mathbf{p}_i)$	20.1	20.9	20.5



(a) Candidate $\mathbf{p}_1$ (b) Candidate $\mathbf{p}_2$ (c) Candidate $\mathbf{p}_3$

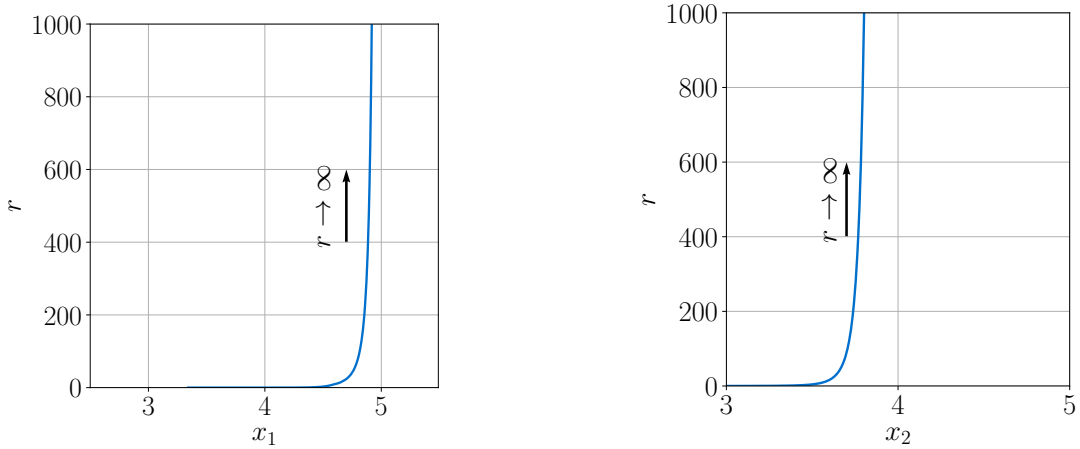
Figure 6.4: Candidates (circles) in feasible set (solid lines) and corresponding contours of cost function (dashed lines). Labels beside each point correspond to labels in Table 6.1.

To illustrate Theorem 6.1 and Proposition 6.1, let us fix y to $[5\ 4]^\top$, a minimizer of COP (6.22). If we substitute $y = [5\ 4]^\top$ into equations (6.24), Theorem 6.1 guarantees the existence of a stationary point sequence (or trajectory for continuously varying r) that converges to the minimizer, that is, $x_\infty = y = [5\ 4]^\top$. Figure 6.5 shows the solution trajectory of equations (6.24) with $y = [5\ 4]^\top$ projected onto the x_1-r and x_2-r planes. We can see that the trajectory approaches the minimizer $[x_1\ x_2]^\top = [5\ 4]^\top$. However, x_2 still has a non-negligible error for $r = 1000$, which means the common solution method (where r is fixed to a number assumed to be sufficiently large) ends up with the wrong solution.

For Proposition 6.1, equation (6.16) shows that

$$\begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} = \rho \begin{bmatrix} 5 \\ 4 \end{bmatrix} \tag{6.33}$$

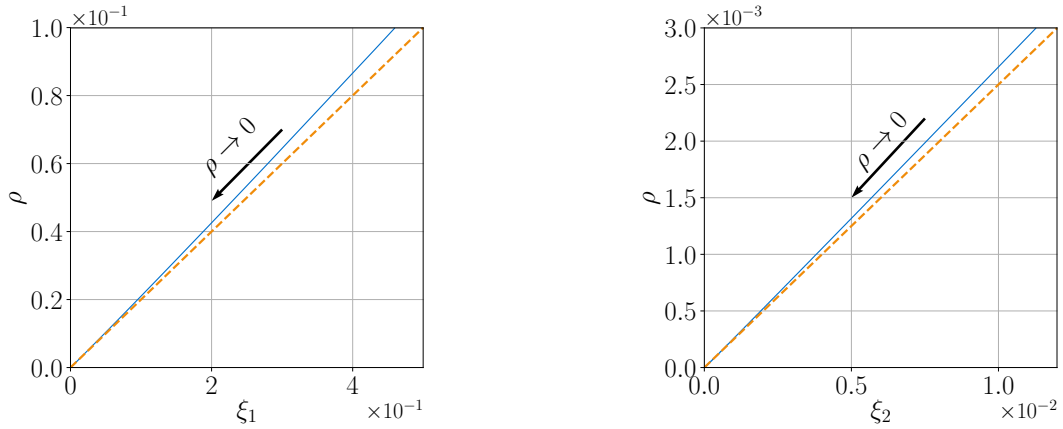
holds for $y = [5\ 4]^\top$. Substituting equation (6.33) into polynomials (6.27), the algebraic set $\mathcal{V}(\mathcal{F})$ is obtained as the solid curves shown in Fig. 6.6. We can regard the algebraic set $\mathcal{V}(\mathcal{F})$ as a trajectory of ξ with respect to ρ, and it is readily observed that the trajectory converges to the origin as $\rho \to 0$, as mentioned in the discussion following Proposition 6.1. Moreover, the tangent of the trajectory at the origin (dashed lines in Fig. 6.6) has the gradient $[5\ 4]^\top$ at the origin, which is the consequence of Proposition 6.1 stated as equations (6.19) and (6.20).



(a) Trajectory of x_1 with respect to r

(b) Trajectory of x_2 with respect to r

Figure 6.5: Trajectory of x with respect to r satisfying equation (6.24) for $y = [5\ 4]^\top$, which is projected onto x_1-r and x_2-r planes.



(a) Trajectory of ξ_1 with respect to ρ

(b) Trajectory of ξ_2 with respect to ρ

Figure 6.6: Trajectory satisfying equation (6.27) $= 0$ (solid lines) and its tangent cone at origin (dashed lines) for $y = [5\ 4]^\top$, which is projected onto ξ_1-ρ and ξ_2-ρ planes.

Example 2 (Case with global minimizers violating KKT conditions) Let us consider the following COP with three indeterminates and a constraint:

$$\begin{aligned} &\min_x \frac{1}{2}\|x\|^2 \\ &\text{s. t. } (x_2 - x_1^2 + 2)^2 - (x_3 - 1)^3 = 0, \end{aligned} \tag{6.34}$$

where $x = [x_1\ x_2\ x_3]^\top \in \mathbf{R}^3$. The penalty function of the COP is obtained as

$$\frac{1}{2}x_1^2 + \frac{1}{2}x_2^2 + \frac{1}{2}x_3^2 + r\left\{(x_2 - x_1^2 + 2)^2 - (x_3 - 1)^3\right\}^2.$$

Figure 6.7 shows the feasible set $\mathcal{X}$ of the COP, where all points in the intersection $\mathcal{X} \cap \{x \mid x_3 = 1\}$ are singular and form a parabola on a plane of $x_3 = 1$. As shown, the curve of singularities $\mathcal{X} \cap \{x \mid x_3 = 1\}$ is like "the bottom of a ravine" and is defined by equations $x_3 = 1$ and $x_2 = x_1^2 - 1$. The bottom of the ravine lies above the origin, and thus the feasible point that is closest to the origin would lie on the bottom. In other words, the global minimizer would be included in the curve of singularities. If this is the case, these points cannot be KKT points because, for all points in the curve, the derivatives of the constraint function vanish, whereas those of the cost function do not vanish, which indicates the nonexistence of Lagrange multipliers. Therefore, the Lagrange multiplier method or other methods based on KKT conditions or assuming the existence of Lagrange multipliers cannot find the global minimizers.

For this problem, the proposed method yields a set of 14 polynomials of the highest degree seven as $\mathcal{G}$. These equations have five solutions $\mathbf{p}_1, \cdots, \mathbf{p}_5$, which are listed in Table 6.2. As shown in Fig. 6.8, the proposed algorithm yields a set of candidates

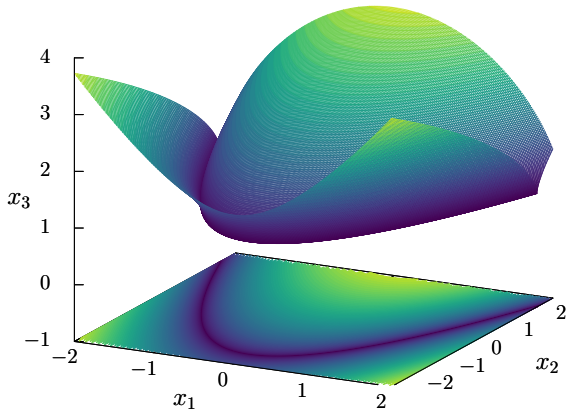


Figure 6.7: Feasible set of COP (6.34). Heat map on x_1-x_2 plane shows projection of feasible set, whose color corresponds to x_3-coordinate of projected points.

Table 6.2: Candidates derived by proposed algorithm and corresponding values of cost function.

	$\mathbf{p}_1$	$\mathbf{p}_2$	$\mathbf{p}_3$	$\mathbf{p}_4$	$\mathbf{p}_5$
x_1	0.00	0.00	0.00	1.22	−1.22
x_2	−1.35	−1.94	−2.00	−0.50	−0.50
x_3	1.75	1.16	1.00	1.00	1.00
$f(p_i)$	2.44	2.55	2.50	1.38	1.38

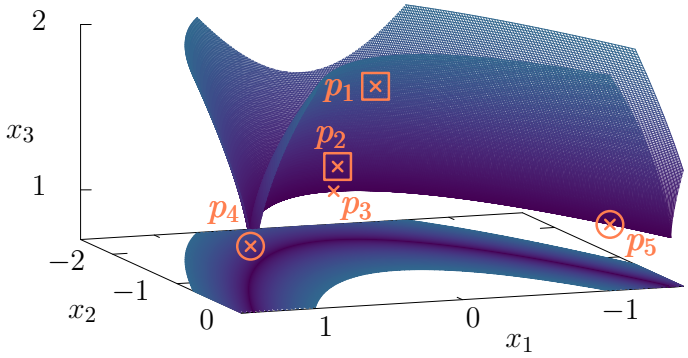


Figure 6.8: Candidates of minimizer obtained by proposed method (cross), KKT points (open square), and global minimizers (open circle). Labels beside each point correspond to labels in Table 6.2.

that includes all KKT points ($\mathbf{p}_1$ and $\mathbf{p}_2$) and some other non-KKT points on the singular curve ($\mathbf{p}_3$, $\mathbf{p}_4$, and $\mathbf{p}_5$). In Table 6.2, points $\mathbf{p}_4$ and $\mathbf{p}_5$ attain the minimum among the candidates, and thus they are the non-KKT type global minimizers.

Note again that there are no Lagrange multipliers corresponding to these global minimizers. Indeed, the derivative of the corresponding Lagrangian with respect to

x_3 is

$$x_3 - 3\lambda(x_3 - 1)^2,$$

where $\lambda \in \mathbf{R}$ is the Lagrange multiplier corresponding to the constraint of COP (6.34). It is obvious that this derivative cannot be zero if $x_3 = 1$ holds; in other words, when a point is included in the singular curve $\mathcal{X} \cap \{x \mid x_3 = 1\}$.

Remark 6.5. For the first example, the FJ conditions yield the same three points as in Table 6.1. However, for the second example, the FJ conditions are satisfied by all the points of $\mathcal{X} \cap \{x \mid x_3 = 1\}$, which includes an infinite number of points neither locally optimal nor KKT. This indicates that the proposed condition is less conservative and can yield a significantly smaller number of candidates than the FJ conditions do.

6.6 Summary

We have proposed a new necessary optimality condition for polynomial optimization problems with polynomial constraints. The proposed necessary condition is satisfied by all minimizers and thus does not require any constraint qualifications. First, a sequential optimality condition based on the quadratic penalty function, which is described by the existence of a certain sequence converging to a minimizer as the penalty parameter tends to infinity, is introduced. By considering a projective space, the limit operation of the penalty parameter is symbolically performed as a computation of the tangent cone at the origin. The set of polynomials, which vanish at all the points satisfying the sequential optimality condition, is obtained. Two numerical examples are provided to illustrate the methodology and demonstrate that the proposed necessary condition can be satisfied even by non-KKT type minimizers.

One direction of further study is to generalize the problem settings. For instance, the algorithm should be readily applicable to parametric optimization problems. We will also extend the functions appearing in the problem to include more general functions than polynomials, as long as they can be homogenized in some sense.

It is worth mentioning that, in the proposed algorithm, the iterative computations for solving stationary conditions and updating the penalty parameter are reduced to solving the equations defining the tangent cone only once. This reduction can be applied to any equation with parameters if some convergence property of their solutions is guaranteed, for instance, as mentioned in Theorem 6.1. Therefore, the proposed method can be applied to a broader class of problems beyond optimization problems.

Chapter 7

Conclusions

7.1 Summary

In this thesis, we proposed symbolic-numeric methods for nonlinear optimal control, optimal estimation, and optimization. Under some algebraic assumptions on the problem settings, various mathematical notions and corresponding symbolic algorithms from commutative algebra, algebraic geometry, and the theory of D-modules are utilized. In Chapters 3 and 4, off-line symbolic algorithms called recursive elimination methods are proposed to derive small-sized problems, and on-line numerical algorithms are also proposed to solve them. By dealing with the main difficulties such as coupling of variables off-line by symbolic computation, the on-line computation was performed efficiently. In Chapter 5, the integrations accompanied by the computation of posterior mean and variance are carried out off-line by means of partial differential equations (PDEs). The solution of the resulting PDEs is then efficiently evaluated by using a symbolic-numeric method called the holonomic gradient method. In Chapter 6, we proposed a derivation algorithm of necessary optimality conditions that are satisfied by all optimal solutions and do not require any constraint qualifications. The proposed algorithm is based on the penalty function method, which is a classical numerical solution method for constrained optimization problems. The procedure of the penalty function method was first interpreted as a geometric procedure including limit operation of a parameter, and then it was further interpreted as a symbolic algorithm by means of algebraic geometry. By the proposed symbolic algorithm, we can compute the polynomial equations exactly satisfied by all the optimal solutions, which are less conservative than Fritz John (FJ) conditions as shown in a numerical example.

7.2 Discussion and future work

From the following two perspectives, we give suggestions for the future work on the basis of the results in this thesis.

7.2.1 Theoretical perspective

In Chapters 3 and 4, the outputs of the recursive elimination methods are just necessary optimality conditions. Although sufficient optimality conditions are provided in Chapter 3, they require derivatives of several functions evaluated on a candidate trajectory, which are computationally demanding to compute. Therefore, in the future, numerical algorithms that are based on the outputs of recursive elimination methods and guaranteed to converge to a local optimal solution could be studied.

The theoretically essential part of the method proposed in Chapter 5 is the exact evaluation of the first and second moments (or equivalently mean and variance) of the state distribution. The higher-order moments are also holonomic functions of the data, and thus their evaluations can be carried out similarly to the first and second ones. Therefore, a direction of future work is to approximate the state distribution by considering higher-order moments and using more general distributions such as skew Gaussian distribution or distributions in exponential family, which would yield more accurate estimates. Another direction is the application to finite-horizon nonlinear stochastic optimal control problems. In these problems, it is also important to evaluate the mean and variance of the state distribution on the horizon. Hence, the same techniques as those used in Chapter 5, especially the holonomic gradient method, would be useful to solve the problems efficiently.

The existing necessary optimality conditions that can be described by equations and inequalities [93–95] tend to be generalizations of the Karush-Kuhn-Tucker (KKT) conditions or the FJ conditions and based on the theory of Lagrange multipliers. On the other hand, the necessary condition proposed in Chapter 6 is based on a necessary condition derived by the penalty function method and thus free from the existence of Lagrange multipliers. Moreover, the proposed condition does not use any inequality or the notion of cones in convex geometry, which is the essential difference from the existing necessary conditions. Moreover, to the best of the author's knowledge, only the FJ conditions are the necessary conditions that can be described by equations and inequalities and do not require any constraint qualifications. At least, the example provided in the second paragraph of section 6.5 shows that the proposed condition is less conservative than the FJ conditions. Hence, exploring necessary optimality

conditions from the perspective of algebraic geometry would be a promising direction, and the relationships between the proposed condition and other geometric or KKT-like conditions should be studied.

7.2.2 Computational perspective

Contrary to geometric and analytic approaches, all the symbolic algorithms introduced and proposed in this thesis are exact and guaranteed to halt in a finite number of computations. However, the practical computational times of the proposed methods still suffer from the high computational cost of Gröbner bases, which limits the class of problems that the proposed methods can be practically applied to. It is remained to clarify a class of problems for which the computations of Gröbner bases in the proposed methods can be easily performed or avoided. One possibility to broaden the class of problems is the choice of the term ordering in each computation of a Gröbner basis. There are still some degrees of freedom for the choice of term orderings, and it is known that the change of term ordering can often lead to a drastic reduction of computations [76]. Hence, exploring appropriate term orderings is also a part of future work. Another possibility is exploiting numerical approximations in symbolic computation. Numerical algorithms to obtain symbolic outputs have been studied in recent years [96–99], and many useful notions such as approximate greatest common divisors [100], homotopy continuation [101], and border bases [102] have been introduced with the development of their computation algorithms. Although the exactness of the outputs is no longer guaranteed when we rely on numerical computation, the advantage of computing the outputs including undetermined parameters would be still available. These approaches can thus be applied to get rid of the high computational costs for computing Gröbner bases.

On the other hand, the algorithms in the on-line part introduced in Chapters 3–5 may also suffer from high computational cost or equivalently from the limitations on computational time. Even though the numerical examples show the efficiency of the proposed methods in the on-line part, it sometimes declines because the off-line part may yield large-scale polynomials having many terms and large coefficients, which induce ineffective computations in the on-line part. For example, when evaluating a large-scale polynomial having many terms up to a large total degree, it is clearly inefficient to evaluate each monomial independently since a monomial of a higher degree may be evaluated as the product of the monomials of lower degrees that are already evaluated. From this perspective, it could be studied to employ special

expressions of large-scale polynomials oriented to evaluation such as those derived by the multivariate Horner scheme [103].

Appendix A

Mathematical Preliminaries

A.1 Ring theory

This section is devoted to introducing fundamental notions of ring theory for the sake of completeness. We refer to [54, 76, 104, 106] for most of the following definitions, propositions, and theorems.

A.1.1 Basic definitions

Definition A.1 (Group)**.** Let G be a set and "+" be a binary operation that associates each pair of $a, b \in G$ with $a + b$. Then G is called a *group* if the binary operation satisfies the following laws:

(i) $(a + b) + c = a + (b + c)$ for all $a, b, c \in G$ (associative law);

(ii) there exists $0 \in G$ such that $a + 0 = 0 + a = a$ for all $a \in G$ (neutral element);

(iii) for each $a \in G$, there exists $-a \in G$ such that $a+(-a) = (-a)+a = 0$ (inverse).

A group additionally satisfying the following condition is called an *abelian group*:

(iv) $a + b = b + a$ for all $a, b \in G$ (commutative law).

Definition A.2 (Ring)**.** Let R be a set and "+" and "$\cdot$", called *addition* and *multiplication*, be binary operations that associate each pair of $a, b \in R$ with $a + b$ and $a \cdot b$, respectively. Then R is called a *ring* if the two binary operations satisfy the following laws:

(i) R is an abelian group under addition;

(ii) $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in R$ (associative law of multiplication);

(iii) $c \cdot (a + b) = c \cdot a + c \cdot b, (a + b) \cdot c = a \cdot c + b \cdot c$ for all $a, b, c \in R$ (distributive laws);

(iv) there exists $1 \in R$ such that $a \cdot 1 = 1 \cdot a = a$ for all $a \in R$ (neutral element of multiplication).

The neutral element for addition 0 is called *zero element* or *zero* of the ring R, while that for multiplication 1 is called *unit element* or *one* of R.

Throughout this thesis, we always abbreviate $a \cdot b$ to ab unless otherwise noted.

Definition A.3 (Commutative and non-commutative ring)**.** A ring R is called a *commutative ring* if it additionally satisfies the following law:

(viii) $ab = ba$ for all $a, b \in R$ (commutative law of multiplication),

otherwise called a *non-commutative ring*.

Example A.1 (Zero ring)**.** The ring consisting of a single element is called the *zero ring* or *trivial ring*. Let $\{0\}$ denotes this ring since the single element must be 0 from the definition of ring.

Definition A.4 (Field)**.** A commutative ring K, not the zero ring, is called a *field* if the set of all non-zero elements in K is a group under multiplication.

Example A.2 (Field of real numbers)**.** The set of real numbers $\mathbf{R}$ is a field with its usual addition and multiplication.

Example A.3 (Field of rational functions)**.** For a field K, the set of all rational functions

$$K(X) \coloneqq \left\{ \frac{f}{g} \mid f, g \in K[X], g \neq 0 \right\}$$

is a field with its usual addition and multiplication.

Definition A.5 (Ideal)**.** For a ring R, not necessarily commutative, a subset $I \subseteq R$ is called a *left ideal* if it satisfies the following conditions:

(i) $0 \in I$;

(ii) if $a, b \in I$, then $a + b \in I$;

(iii) if $a \in I$ and $r \in R$, then $ra \in I$.

The subset I is instead called a *right ideal* if condition (iii) is replaced by

(iii') if $a \in I$ and $r \in R$, then $ar \in I$.

A left ideal that is also a right ideal is called a *two-sided ideal* or simply *ideal*. In particular, all ideals of a commutative ring are two-sided ideals.

Example A.4. For any ring R, the singleton subset $\{0\} \subset R$ is a two-sided ideal, which is called the *zero ideal* and denoted by 0.

From the definition of a commutative ring, all ideals of a commutative ring are two-sided ideals. In this subsection, a left ideal is also referred to as an ideal if it is a subset of a non-commutative ring.

Definition A.6. For a ring R, let $\{r_1, \ldots, r_d\}$ be a finite number of elements of R. Then the *ideal generated by* $r_1, \ldots, r_d$ is the set of all linear combinations

$$I = \langle r_1, \ldots, r_d \rangle \coloneqq \{a_1 r_1 + \cdots + a_d r_d \mid a_1, \ldots, a_d \in R\}.$$

A finite set that generates an ideal I is called a *basis* of I, and each element a basis is called a *generator* of I.

Definition A.7 (Noetherian ring)**.** A ring R is called *left(right) noetherian* if all left(right) ideals of R are finitely generated. A left noetherian ring that is also a right noetherian ring is called *two-sided noetherian* or simply *noetherian*. In particular, a commutative ring is always two-sided notherian if it is left noetherian.

Definition A.8 (Equivalence class)**.** [107] A binary relation $\sim$ on a set S is called an *equivalence relation* if it has the following properties:

(i) $a \sim a$ (reflectivity);

(ii) if $a \sim b$, then $b \sim a$ (symmetry);

(iii) if $a \sim b$ and $b \sim c$, then $a \sim c$ (transitivity).

For an element $a \in S$, the *equivalence class* of a under $\sim$ is defined as subset

$$[a] \coloneqq \{s \in S \mid a \sim s\}.$$

Proposition A.1. Let I be an ideal of a ring R. Then a binary relation $\sim$ on R defined as

$$a \sim b \overset{\text{def}}{=} a - b \in I$$

is an equivalence relation on R and called *congruence modulo* I.

Definition A.9 (Quotient ring)**.** Let I be an ideal of a ring R. The set of all the equivalence classes under congruence modulo I, denoted by R/I, is a ring with the addition and multiplication defined as

$$[a] + [b] \coloneqq [a + b],$$
$$[a][b] \coloneqq [ab],$$

and called the *quotient ring* of R modulo I.

Finally, we define the dimension of a commutative ring and its ideals.

Definition A.10 (Prime ideal)**.** For a commutative ring R, an ideal $I \subset R$ is called *prime* if

$$fg \in I \Rightarrow \text{either } f \in I \text{ or } g \in I$$

holds for any two elements $f, g \in R$.

Definition A.11 (Krull dimension of commutative ring)**.** The *Krull dimension* of a commutative ring R, denoted by $\dim R$, is the supremum of the length of the chains of prime ideals $p_0 \subsetneq p_1 \subsetneq \cdots \subsetneq p_n$ in R.

Definition A.12 (Krull dimension of ideal of commutative ring)**.** For a commutative ring R, the *Krull dimension of an ideal* $I \subset R$ is the Krull dimension of the quotient ring R/I.

A.1.2 Polynomial ring

In this subsection, K denotes a field of characteristic zero.

Definition A.13 (Monomial)**.** For variables $X \coloneqq (X_1, \ldots, X_n)$ and a tuple of non-negative integers $\alpha \coloneqq (\alpha_1, \ldots, \alpha_n) \in \mathbf{Z}^n_{\geq 0}$, a *monomial* is a product of the form

$$X^\alpha \coloneqq X_1^{\alpha_1} \cdot X_2^{\alpha_2} \cdots X_n^{\alpha_n}.$$

The vector α is called the *multi-index* of this monomial. The *total degree* of a monomial X^α, denoted by $|\alpha|$, is the sum $\alpha_1 + \cdots + \alpha_n$.

Definition A.14 (Polynomial)**.** A *polynomial* f in variables X with coefficients in K is a finite linear combination of monomials:

$$f = \sum_\alpha a_\alpha X^\alpha \qquad (a_\alpha \in K),$$

where the summation is over a finite subset of $\mathbf{Z}_{\geq 0}$. The *total degree* of f is the maximum of $|\alpha|$ over the finite subset.

One of the main subjects in algebraic geometry is a commutative ring called a polynomial ring defined as follows.

Definition A.15 (Polynomial ring)**.** For a field K, $K[X_1, \ldots, X_n]$ denotes the set of all polynomials in variables $X_1, \ldots, X_n$ with coefficients in K. This set is a commutative ring with its usual addition and multiplication, which is called a *polynomial ring* in the variables $X_1, \ldots, X_n$ over K.

In this subsection, X denotes a vector $[X_1, \ldots, X_n]^\top$, and a polynomial ring $K[X_1, \ldots, X_n]$ is abbreviated to $K[X]$.

Theorem A.1 (Hilbert's basis theorem)**.** Let K be a field. The polynomial ring $K[X]$ is noetherian, that is, all ideals of the ring are finitely generated.

Definition A.16 (Monomial order)**.** A *monomial order* $\prec$ on $K[X]$ is a relation on $\mathbf{Z}_{\geq 0}^n$, or equivalently on the set of all monomials in $K[X]$, satisfying the following conditions:

(i) $\prec$ is a total order on $\mathbf{Z}_{\geq 0}$;

(ii) if $\alpha \prec \beta$ and $\gamma \in \mathbf{Z}_{\geq 0}$, then $\alpha + \gamma \prec \beta + \gamma$;

(iii) $0 \prec \alpha$ for all $\alpha \in \mathbf{Z}_{\geq 0}$.

Definition A.17 (Initial ideal)**.** Let I be an ideal of $K[X]$ with $I \neq 0$ and $\prec$ be a monomial order on $K[X]$. For a polynomial $f \in K[X]$, $\mathrm{in}_\prec(f)$ denotes the largest monomial of f with respect to $\prec$ and is called the *initial monomial*. Then the ideal generated by monomials $\{\mathrm{in}_\prec(f) \mid f \in I\}$ is called the *initial ideal* of I and denoted by $\mathrm{in}_\prec(I)$, that is,

$$\mathrm{in}_\prec(I) \coloneqq \langle \{\mathrm{in}_\prec(f) \mid f \in I\} \rangle.$$

Definition A.18 (Gröbner basis)**.** Fix a monomial order $\prec$ on $K[X]$. For an ideal I of $K[X]$, a *Gröbner basis* with respect to $\prec$ is a basis of I consisting of finite generators $\{g_1, \ldots, g_m\} \subset K[X]$ such that the initial monomials $\{\mathrm{in}_\prec(g_1), \ldots, \mathrm{in}_\prec(g_m)\}$ generates the initial ideal $\mathrm{in}_\prec(I)$ of I.

For an ideal $I \subset K[X]$ with $I \neq K[X]$, the quotient ring $K[X]/I$ can be regarded as a vector space over K with its addition and the scalar multiplication defined as

$$a[f] \coloneqq [af] \qquad (a \in K, [f] \in K[X]/I).$$

Theorem A.2 (Macaulay's Theorem)**.** Fix a monomial order $\prec$ on $K[X]$, let $I \subset K[X]$ be an ideal with $I \neq K[X]$, and let $\mathrm{in}_{\prec}(I)$ be the initial ideal of I with respect to $\prec$. Then, the set of monomials $\{x^{\alpha} \mid x^{\alpha} \notin \mathrm{in}_{\prec}(I)\}$ is a basis of the vector space $K[X]/I$ over K.

Definition A.19 (Zero-dimensional ideal)**.** An ideal $I \subset K[X]$ is called *zero-dimensional* if the quotient ring $K[X]/I$ is a finite-dimensional vector space over K, or equivalently the set of monomials $\{x^{\alpha} \mid x^{\alpha} \notin \mathrm{in}_{\prec}(I)\}$ is a finite set.

Zero-dimensional ideals are characterized by the following lemma [54].

Lemma A.1. An ideal $I \subset K[X]$ is zero-dimensional if and only if a nonzero polynomial $h_i \in K[X_i]$ exists for each $i = 1, \ldots, n$ such that $I \cap K[X_i] = \langle h_i \rangle$ holds.

The polynomials h_i $(i = 1, \ldots, m)$ are called *minimal polynomials* of X_i with respect to I and can be computed from generators of I by using Gröbner bases.

Definition A.20 (Radical ideal)**.** For an ideal $I \subset K[X]$, the set of polynomials:

$$\sqrt{I} \coloneqq \{g \in K[X] \mid \exists s \in \mathbf{N} \text{ s.t. } g^s \in I\} \tag{A.1}$$

is also an ideal called the *radical* of I. I is called a *radical ideal* when $I = \sqrt{I}$ holds. Obviously, $\mathcal{I}(J) = \mathcal{V}\left(\sqrt{I}\right)$ holds.

Definition A.21 (Extension of ideal)**.** For an ideal $I \subset \mathbf{R}[X, Y]$ with $X = [X_1 \cdots X_n]^{\top}$ and $Y = [Y_1 \cdots Y_m]^{\top}$, the *extension* of the ideal I to $\mathbf{R}(X)[Y]$ is the ideal I^e defined as

$$I^e \coloneqq \{b_1 g_1 + \cdots + b_s g_s \mid b_1, \ldots, b_s \in \mathbf{R}(X)[Y], g_1, \ldots, g_s \in I,\ s \in \mathbf{N}\}. \tag{A.2}$$

A.2 Algebraic Geometry

This section is devoted to introducing the notions in algebraic geometry. We refer to [54, 76, 108] for most of the definitions and lemmas here.

A.2.1 Elimination theory

Definition A.22 (Algebraically closed field)**.** For a field K, a nonconstant polynomial $f \in K[X] = K[X_1]$ is called *irreducible* if it has the property that whenever $f = gh$ for some polynomials $g, h \in K[X]$, then either g or h is a constant. The field K is called *algebraically closed* if every irreducible polynomial in $K[X]$ is of degree one.

Example A.5 (Field of complex numbers)**.** The set of complex numbers $\mathbf{C}$ is an algebraically closed field because, from the *Fundamental Theorem of Algebra*, every polynomial $f \in \mathbf{C}[X]$ of degree d can be written as

$$f(X) = c(X - a_1)^{\alpha_1}(X - a_2)^{\alpha_2} \cdots (X - a_s)^{\alpha_s}$$

where $c, a_1, \ldots, a_s \in \mathbf{C}$ and $\alpha_1, \ldots, \alpha_s \in \mathbf{Z}_{\geq 0}$ are such that $a_1, \ldots, a_s$ are pairwise distinct and $\alpha_1 + \cdots + \alpha_s = d$, which readily indicates that every irreducible polynomial in $\mathbf{C}[X]$ is of the form $c(X - a_1)$.

Definition A.23 (Algebraic closure)**.** For a field K, there exists a unique algebraic extension $\overline{K}$ of K that is algebraically closed, which is called the *algebraic closure* of K.

For the details of algebraic extensions of fields, see [109].

Definition A.24 (Algebraic set)**.** For an ideal $I \subset K[X] = K[X_1, \ldots, X_n]$, the *algebraic set* defined by I is the subset:

$$\mathcal{V}(I) \coloneqq \{X \in \overline{K}^n \mid f(X) = 0 \text{ for all } f \in I\} \subset \overline{K}^n.$$

The algebraic set $\mathcal{V}(I)$ is also denoted by $\mathcal{V}(f_1, \ldots, f_m)$ if I is generated by $f_1, \ldots, f_m \in K[X]$.

Definition A.25. For an algebraic set $V \subset K^n$, the set of polynomials:

$$\mathcal{I}(V) \coloneqq \{f \in K[X] \mid f(\tilde{X}) = 0 \text{ for all } \tilde{X} \in V\}$$

is an ideal of $K[X]$. This ideal is called an *ideal of* V.

Definition A.26 (Elimination ideal)**.** For an ideal $I \subset \mathbf{R}[X, Y]$ with $X = [X_1 \ \cdots \ X_n]^\top$ and $Y = [Y_1 \ \cdots \ Y_m]^\top$, the intersection $I \cap \mathbf{R}[Y] \subset \mathbf{R}[Y]$ is also an ideal and is called the *elimination ideal* of I with respect to the variable X.

The algebraic sets $\mathcal{V}(I)$ and $\mathcal{V}(I \cap \mathbf{R}[Y])$ are related by the following lemma [76].

Lemma A.2. For an ideal $I \subset \mathbf{R}[X, Y]$,

$$\pi_Y(\mathcal{V}(I)) \subset \mathcal{V}(I \cap \mathbf{R}[Y])$$

holds, where $\pi_Y \colon \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}^m$ defined by $(X, Y) \mapsto Y$ is the projection of $\mathcal{V}(I)$ onto the Y-space, i.e., $\pi_Y(\mathcal{V}(I)) = \{Y \in \mathbf{R}^m \mid \exists X \in \mathbf{R}^n \ s.t. \ (X, Y) \in \mathcal{V}(I)\}$.

This lemma explains how the computation of elimination ideals corresponds to the variable elimination from multivariate algebraic equations; indeed, the algebraic sets $\mathcal{V}(I)$ and $\mathcal{V}(I \cap \mathbf{R}[Y])$ can also be viewed as the root sets of generators of I and $I \cap \mathbf{R}[Y]$, respectively. In other words, Lemma A.2 guarantees that, by computing generators of $I \cap \mathbf{R}[Y]$, we can obtain a set of polynomials involving only the variable Y, whose roots contain the Y-coordinates of all the roots of generators of I.

A Gröbner basis of the elimination ideal $I \cap \mathbf{R}[Y]$ can be computed from generators of I.

Definition A.27 (Elimination order)**.** A monomial order $\prec$ on $\mathbf{R}[X, Y]$ is called an *elimination order with respect to* X if it has the following property: for a monomial $X^\alpha Y^\beta$ with multi-index $(\alpha, \beta) \in \mathbf{Z}_{\geq 0}^{n+m}$ with $\alpha \neq 0$,

$$(0, \beta') \prec (\alpha, \beta) \text{ or equivalently } Y^{\beta'} \prec X^\alpha Y^\beta$$

holds for all $\beta' \in \mathbf{Z}_{\geq 0}^m$.

Proposition A.2 (Gröbner bases of elimination ideals)**.** Let $\prec$ be an elimination order on $\mathbf{R}[X, Y]$ with respect to X. For an ideal $I \subset \mathbf{R}[X, Y]$, let $G \subset \mathbf{R}[X, Y]$ be a Gröbner basis for I with respect to $\prec$. Then, a subset of G that do not involve any component of X, that is, the intersection $G \cap \mathbf{R}[Y]$ is a Gröbner basis for the elimination ideal $I \cap \mathbf{R}[Y]$ with respect to X.

A.2.2 Projective Space

In projective geometry, we define the equivalence class

$$[L] \coloneqq \{\text{all lines parallel to a given line } L \subset \mathbf{R}^n\}$$

as *a point at infinity* of $\mathbf{R}^n$. Note that this definition of points at infinity is a straightforward extension of the characterization of points in $\mathbf{R}^n$; namely, a point in $\mathbf{R}^n$ can be characterized by a set of all lines, any two of which cross each other at that point. This definition of points at infinity enables us to integrate $\mathbf{R}^n$ and points at infinity into a projective space $\mathbf{P}^n$, where it is no longer necessary to distinguish points of $\mathbf{R}^n$ from those at infinity.

Definition A.28 (Projective Space)**.** An *n-dimensional projective space* over $\mathbf{R}$ denoted by $\mathbf{P}^n$ is the set of equivalence classes of the equivalence relation $\sim$ on $\mathbf{R}^{n+1} \setminus \{\mathbf{0}\}$, where $\sim$ is defined for $\mathbf{x}, \mathbf{y} \in \mathbf{R}^{n+1} \setminus \{\mathbf{0}\}$ as

$$\mathbf{x} \sim \mathbf{y} \overset{def}{\iff} \exists \lambda \in \mathbf{R} \setminus \{0\} \text{ s.t. } \mathbf{x} = \lambda \mathbf{y}. \tag{A.3}$$

An equivalence class $p \in \mathbf{P}^n$ is also denoted by $[X_0 : \cdots : X_n]$ with any point $[X_0 \ \cdots \ X_n]^\top \in p$, and this representation is called a *homogeneous coordinate* of p.

From the definition (A.3), it is obvious that for each equivalent class $p \in \mathbf{P}^n$, there exists a unique line passing through the origin of $\mathbf{R}^{n+1}$ and includes every point in p. Therefore, we can identify a projective space $\mathbf{P}^n$ with the set of all lines in $\mathbf{R}^{n+1}$ passing through its origin. Note that a projective space is a topological space with the quotient topology induced from the natural topology of $\mathbf{R}^{n+1} \setminus \{\mathbf{0}\}$ by the quotient mapping $\mathbf{R}^{n+1} \setminus \{\mathbf{0}\} \ni [X_0 \ \cdots \ X_n]^\top \mapsto p = [X_0 : \cdots : X_n] \in \mathbf{P}^n$.

To show that a projective space $\mathbf{P}^n$ is the union of a Euclidian space $\mathbf{R}^n$ and a set of all points at infinity of $\mathbf{R}^n$, let us consider an open subset U_0 defined as

$$U_0 \coloneqq \{[X_0 : \cdots : X_n] \in \mathbf{P}^n \mid X_0 \neq 0\} \subset \mathbf{P}^n. \tag{A.4}$$

This open subset can be identified with $\mathbf{R}^n$ because there is a homeomorphism $\phi_0 : \mathbf{R}^n \to U_0$ that maps $[x_1 \cdots x_n]^\top \in \mathbf{R}^n$ to $[1 : x_1 : \cdots : x_n] \in U_0$; the inverse mapping ϕ_0^{-1} can be defined as

$$\phi_0^{-1} : U_0 \ni [X_0 : \cdots : X_n] = [1 : X_1/X_0 : \cdots : X_n/X_0] \mapsto [X_1/X_0 \ \cdots \ X_n/X_0]^\top \in \mathbf{R}^n.$$

For the complement $H_0 \coloneqq \mathbf{P}^n \setminus U_0$, each point $[0 : X_1 : \cdots : X_n] \in H_0$ uniquely determines a line L through the origin by $L = \{\mathbf{x} = [x_1 \ \cdots \ x_n]^\top \in \mathbf{R}^n \mid i \in \{1, \ldots, n\}, t \in \mathbf{R}, x_i = tX_i\}$. We can consider the equivalence class consisting of all lines parallel to L, which defines a point at infinity, and thus there is a bijective correspondence between all points of H_0 and those at infinity of $\mathbf{R}^n \cong U_0$. From this viewpoint, H_0 is called the *hyperplane at infinity* of U_0.

Note that there are many other pairs of subsets that have the same property as (U_0, H_0). Indeed, for each $i = 1, \ldots, n$, we can define subsets U_i and H_i as

$$U_i \coloneqq \{[X_0 : \cdots : X_n] \in \mathbf{P}^n \mid X_i \neq 0\}, \tag{A.5}$$

$$H_i \coloneqq \mathbf{P}^n \setminus U_i. \tag{A.6}$$

For each pair, we can identify U_i with $\mathbf{R}^n$ by a homeomorphism $\phi_i : \mathbf{R}^n \to U_i$ defined in the same way as U_0, and under this identification, H_i can be identified with the set of all points at infinity of U_i. The set of n pairs $\{(U_i, \phi_i)\}_{i=1}^n$ can be regarded as an *atlas* of the projective space $\mathbf{P}^n$ when we treat the space as a manifold, and each pair (U_i, ϕ_i) is called a *chart* of $\mathbf{P}^n$.

It can be easily shown that the intersection

$$U_i \cap H_j = \{[X_0 : \cdots : X_n] \in \mathbf{P}^n \mid X_i \neq 0, X_j = 0\}$$

is dense in H_j if and only if $i \neq j$ holds, which indicates each subset U_i contains almost all points at infinity of the other subsets U_j. This means that, by changing the chart from U_j to U_i $(j \neq i)$, we can reduce any computations in the hyperplane at infinity H_j to computations of finite values in $U_i \cong \mathbf{R}^n$.

A.2.3 Homogenization and Dehomogenization

Definition A.29 (Homogenization of Polynomial)**.** For a real coefficient polynomial $f(x_1, \ldots, x_n) \in \mathbf{R}[x_1, \ldots, x_n]$, the *homogenization of* f is the homogeneous polynomial $f^{\mathrm{hom}} \in \mathbf{R}[X_0, \ldots, X_n]$ defined as

$$f^{\mathrm{hom}}(X_0, \ldots, X_n) \coloneqq X_0^d \cdot f\left(\frac{X_1}{X_0}, \ldots, \frac{X_n}{X_0}\right), \tag{A.7}$$

where d is the total degree of f.

Note that a homogenization f^{hom} of degree d is a homogeneous function of degree d because

$$f^{\mathrm{hom}}(\varepsilon X_0, \ldots, \varepsilon X_n) = \varepsilon^d f^{\mathrm{hom}}(X_0, \ldots, X_n)$$

holds. Therefore, if a homogeneous coordinate of $p \in \mathbf{P}^n$ satisfies $f^{\mathrm{hom}} = 0$, all homogeneous coordinates of p also satisfy the same equation, and thus it makes sense to consider the subset of $\mathbf{P}^n$ where f^{hom} vanishes. We also call this subset an algebraic set defined by f^{hom} and denote it by $\mathcal{V}(f^{\mathrm{hom}})$. The relationship between the algebraic set of the homogenization f^{hom} and that of the original polynomial f is

$$\mathcal{V}(f) = \mathcal{V}(f^{\mathrm{hom}}) \cap U_0,$$

which indicates $\mathcal{V}(f^{\mathrm{hom}}) \subset \mathbf{P}^n$ is an extension of $\mathcal{V}(f) \subset \mathbf{R}^n \cong U_0$.

Conversely, we can define a polynomial defined on U_0 corresponding to any homogeneous polynomial defined on $\mathbf{P}^n$, which is called *dehomogenization*.

Definition A.30 (Dehomogenization of Homogeneous Polynomial)**.** For a homogeneous polynomial $g(X_0, \ldots, X_n) \in \mathbf{R}[X_0, \ldots, X_n]$, the *dehomogenization of* g is a polynomial $g^{\mathrm{deh}} \in \mathbf{R}[x_1, \ldots, x_n]$ defined as

$$g^{\mathrm{deh}}(x_1, \ldots, x_n) \coloneqq g(1, x_1, \ldots, x_n). \tag{A.8}$$

Note that we can consider the dehomogenization of g for any index $i = 0, \ldots, n$ in the same way, although for simplicity we do not indicate i explicitly, and i is specified in accordance with the context.

It is readily known that there is also the relationship

$$\mathcal{V}(g^{\mathrm{deh}}) = \mathcal{V}(g) \cap U_0.$$

Note that, in general, a homogeneous polynomial f and the homogenization of its dehomogenization $(f^{\mathrm{deh}})^{\mathrm{hom}}$ may be different from each other. For example, a homogeneous polynomial

$$g(X_0, X_1, X_2) = X_0^3 X_1^2 + X_0^2 X_2^3$$

is different from

$$(g^{\mathrm{deh}})^{\mathrm{hom}}(X_0, X_1, X_2) = X_0 X_1^2 + X_2^3.$$

Remark A.1. Although $X_0 = 1$ is substituted into g in (A.8), this definition is consistent with the definition of a point in $\mathbf{P}^n$; for any homogeneous coordinate $[X_0 : \cdots : X_n]$ of a point $p \in \mathbf{P}^n$ (where X_0 is not necessarily equal to one), we can obtain the other homogeneous coordinate $[1 : X_1/X_0 : \cdots : X_n/X_0]$, which represents the same point p. From this viewpoint, dehomogenization corresponds to the changes of variables $x_i = X_i/X_0$ for $i = 1, \ldots, n$. Indeed, applying changes of variables $X_i = x_i X_0$ to $g(X_0, \ldots, X_n)$ of degree d, we obtain

$$g(X_0, \ldots, X_n) = X_0^d g^{\mathrm{deh}}(x_1, \ldots, x_n),$$

which indicates that $g = 0$ is equivalent to $g^{\mathrm{deh}} = 0$ unless X_0 is equal to zero. This equation is the same as the definition of homogenization (A.7) except that d is the total degree of g but not of g^{deh}.

A.2.4 Tangent Cone

Throughout this subsection, an algebraic set V is assumed to include the origin $\mathbf{0}$.

The definition of a tangent cone is as follows [76].

Definition A.31 (Tangent Cone)**.** For an algebraic set $V = \mathcal{V}(f_1, \ldots, f_s) \subset \mathbf{R}^n$, its *tangent cone* at $\mathbf{0} \in V \subset \mathbf{R}^n$ is an algebraic set $C_{\mathbf{0}}(V) \subset \mathbf{R}^n$ defined as

$$C_{\mathbf{0}}(V) \coloneqq \mathcal{V}(\{f^{\mathrm{min}} \mid f \in \mathcal{I}(V)\}), \tag{A.9}$$

where f^{min} denotes the homogeneous component of the lowest degree in f, i.e., the sum of all the terms of f whose degrees are equal to the lowest degree of f.

Note that this definition indicates the set of polynomials defining a tangent cone consists of homogeneous polynomials. As can be seen in the above definition, a tangent cone is an algebraic set, which means there exists an ideal $J = \langle g_1, \ldots, g_t \rangle \subset \mathbf{R}[x_1, \ldots, x_n]$ that defines the tangent cone. There are various algorithms to compute generators of J [54, 98, 102, 108], and most of them use a Gröbner basis, which is a set of generators that has good properties for symbolic computations (see [76] for details). The following lemma gives us one of the methods to compute generators of J by using a Gröbner basis with respect to an elimination order [76] for X_0.

Lemma A.3 (The Tangent Cone Algorithm [102, 108])**.** Consider an ideal $I \subset \mathbf{R}[x_1, \ldots, x_n]$ generated by polynomials $f_1, \ldots, f_s$ and assume the algebraic set $\mathcal{V}(I) \subset \mathbf{R}^n$ includes the origin $\mathbf{0}$. Let $\{G_1, \ldots, G_t\}$ be a Gröbner basis of an ideal generated by polynomials $f_1^{\mathrm{hom}}, \ldots, f_s^{\mathrm{hom}}$ with respect to an elimination order for X_0. Then, for the dehomogenizations $\{G_1^{\mathrm{deh}}, \ldots, G_t^{\mathrm{deh}}\} \subset \mathbf{R}[x_1, \ldots, x_n]$, the following equation holds:

$$C_{\mathbf{0}}(\mathcal{V}(I)) = \mathcal{V}\left((G_1^{\mathrm{deh}})^{\min}, \ldots, (G_t^{\mathrm{deh}})^{\min}\right), \tag{A.10}$$

where $(G_i^{\mathrm{deh}})^{\min}$ denotes the homogeneous component of the lowest degree in G_i^{deh}.

Although Definition A.31 consists of only algebraic statements, it is also useful to clarify the characterization of a tangent cone from the geometric viewpoint. Before that, let us introduce the following definition.

Definition A.32. We say that a sequence of lines $L_{[1:\infty]} \subset \mathbf{R}^n$ through $\mathbf{0}$ converges to a line L also through $\mathbf{0}$ if, for a given parametrization $L = \{tv \in \mathbf{R}^n \mid v \in \mathbf{R}^n, t \in \mathbf{R}\}$, there exist parametrizations of $L_k = \{tv_k \in \mathbf{R}^n \mid v_k \in \mathbf{R}^n, t \in \mathbf{R}\}$ such that $\lim_{k\to\infty} v_k = v$ holds.

By using this definition, we can show the characterization of a tangent cone as Lemma A.4.

Lemma A.4. A line $L \subset \mathbf{R}^n$ is called a *secant line* of an algebraic set $V \subset \mathbf{R}^n$ if the intersection of L and V consists of more than two points. Let $L_\infty \subset \mathbf{R}^n$ be a line through $\mathbf{0} \in V$. L_∞ is then a subset of the tangent cone $C_{\mathbf{0}}(V)$ if and only if there exists a sequence $q_{[1:\infty]} \subset V \setminus \{\mathbf{0}\}$ such that $\lim_{k\to\infty} q_k = \mathbf{0}$ holds and the sequence of secant lines $\{L_k \subset \mathbf{R}^n \mid \mathbf{0}, q_k \in L_k \cap V\}_{k=1}^{\infty}$ converges to the line L_∞ as $k \to \infty$.

Remark A.2. Note that the notion of tangent cones in algebraic geometry, which is defined in the above definition, is slight different from the tangent cone in convex geometry defined in Section 2.1. Specifically, a tangent cone in convex geometry

consists of all the *half-lines* that emanates from a certain point while that in algebraic geometry consists of all the *lines* that pass through the point. In this section and Chapter 6, we refer to a tangent cone in algebraic geometry simply as a tangent cone.

A.3 Theory of D-modules

This section is devoted to introducing the notions in the theory of D-modules. Although the main mathematical objects in the theory of D-modules are obviously modules over the Weyl algebra, namely, D-modules, it is enough for the purpose of this thesis to introduce the notion of ideals in the Weyl algebra rather than modules. We refer to [81, 105, 106, 110] for most of the following definitions, lemmas, propositions, and theorems.

A.3.1 Rings of differential operators

Definition A.33 (Differential operators)**.** For a field K of characteristic zero, let ∂_i $(i = 1, \ldots, n)$ be a linear mapping of $K[X] = K[X_1, \ldots, X_n]$ defined on $f(X) \in K[X]$ by $\partial_i \bullet f(X) \coloneqq \partial f(X)/\partial X_i$. Furthermore, define the action of a polynomial $a(X) \in K[X]$ on $f(X) \in K[X]$ as a linear mapping by the usual multiplication on $K[X]$, that is, $a(X) \bullet f(X) \coloneqq a(X)f(X)$. For a multi-index $\alpha = [\alpha_1 \ \cdots \ \alpha_n]^\top \in \mathbf{Z}_{\geq 0}$, ∂^α denotes the composition of linear mappings $\partial_1^{\alpha_1} \bullet \cdots \bullet \partial_n^{\alpha_n}$. A *differential operator with coefficients in* $K[X]$ is a finite sum of the form

$$\sum_\alpha a_\alpha(X)\partial^\alpha \qquad (a_\alpha(X) \in K[X]). \tag{A.11}$$

Hereafter, $K[X]\langle\partial\rangle$ denotes the set of all differential operators with coefficients in $K[X]$, where $\partial \coloneqq [\partial_1 \ \cdots \ \partial_n]^\top$.

Definition A.34 (Total symbol)**.** The *total symbol* of a differential operator $l = \sum_\alpha a_\alpha(X)\partial^\alpha$ is a polynomial in $2n$ indeterminates:

$$\Psi(l) \coloneqq \sum_\alpha a_\alpha(X)\xi^\alpha,$$

where $\xi = [\xi_1 \ \cdots \ \xi_n]^\top$ is a vector of additional indeterminates.

Although a composition of differential operators $\partial_i \bullet a(X)$ $(a(X) \in K[X])$ is not in the form of (A.11), it acts on any polynomial $f(X) \in K[X]$ in the same way as a differential operator $a(X)\partial_i + \partial a(X)/\partial X_i$ since

$$\partial_i \bullet (a(X) \bullet f(X)) = a(X)\frac{\partial f(X)}{\partial X_i} + \frac{\partial a(X)}{\partial X_i}f(X) = \left(a(X)\partial_i + \frac{\partial a(X)}{\partial X_i}\right) \bullet f(X)$$

holds. Hence, $\partial_i \bullet a(X)$ defines the same linear mapping of $K[X]$ as $a(X)\partial_i + \partial a(X)/\partial X_i$ defines, which is in the form of (A.11). We can define a unique expression of such linear mappings as a differential operator as follows.

Definition A.35 (Canonical form)**.** A differential operator is said to be in *canonical form* if it is expressed in the form of (A.11).

Lemma A.5 (Uniqueness of canonical form)**.** The canonical form of a differential operator is uniquely determined by the linear mapping of $K[X]$ that the differential operator defines. In other words, a differential operator is the zero mapping as a linear mapping of $K[X]$ if and only if $a_\alpha = 0$ hold for all $\alpha \in \mathbf{Z}_{\geq 0}$ in its canonical form.

Example A.6. A linear mapping $\partial_i \bullet a(X)$ is uniquely expressed in its canonical form $a(X)\partial_i + \partial a(X)/\partial X_i$.

Lemma A.6 (Leibnitz rule)**.** The composition of any two differential operators $l_1, l_2 \in K[X]\langle\partial\rangle$ is also an element of $K[X]\langle\partial\rangle$, that is, $l_1 \bullet l_2$ can be also expressed in canonical form. Moreover, its total symbol is given as

$$\Psi(l_1 \bullet l_2) = \sum_{\alpha \in \mathbf{Z}_{\geq 0}} \frac{1}{\alpha!} \frac{\partial^{|\alpha|}\Psi(l_1)}{\partial \xi^\alpha} \frac{\partial^{|\alpha|}\Psi(l_2)}{\partial x^\alpha}, \tag{A.12}$$

where $\alpha! \coloneqq \alpha_1! \cdots \alpha_n!$.

From Lemma A.6, the composition of differential operators can be regarded as a multiplication on $K[X]\langle\partial\rangle$. Moreover, this multiplication is non-commutative according to (A.12), and thus $K[X]\langle\partial\rangle$ is a non-commutative ring.

Definition A.36 (Weyl algebra)**.** The set $K[X]\langle\partial\rangle$ is a non-commutative ring with its usual addition and the multiplication defined by the composition of differential operators. This non-commutative ring is called the *n-dimensional Weyl algebra* and denoted by $\mathcal{D}_n$. We omit the subscript n if it is not necessary to specify its dimension.

All of the definitions and lemmas introduced so far are also valid when the polynomial ring $K[X]$ is replaced with the field of rational functions $K(X)$, and then we obtain another non-commutative ring of differential operators as follows.

Definition A.37 (Ring of differential operators)**.** Let $K(X)\langle\partial\rangle$ be the set of all differential operators with coefficients in $K(X)$. This set is a non-commutative ring with the same addition and multiplication as those of Weyl algebra and denoted by $\mathcal{R}_n$. We omit the subscript n if it is not necessary to specify its dimension.

Theorem A.3. The Weyl algebra $\mathcal{D}$ is left noetherian.

Theorem A.4. The ring of differential operators $\mathcal{R}$ is left noetherian.

From the above two theorems, all left ideals of $\mathcal{D}$ and $\mathcal{R}$ are finitely generated. For the following discussion, we refer to a left ideal of $\mathcal{D}$ or $\mathcal{R}$ just as an ideal of $\mathcal{D}$ or $\mathcal{R}$ unless otherwise noted.

A.3.2 Ideals of $\mathcal{R}$ and holonomic functions

The non-commutative ring $\mathcal{R} = \mathbf{R}(X)\langle\partial\rangle$ can be associated with a (commutative) polynomial ring $\mathbf{R}(X)[\xi]$ by considering total symbols. Hence, almost all notions in subsection A.1.2 can be extended to the case of $\mathcal{R}$.

Definition A.38 (Term order on $\mathcal{R}$)**.** For any monomial order $\prec$ on $\mathbf{R}(X)[\xi]$, a *term order* on $\mathcal{R}$ is defined as

$$a(X)\partial^\alpha \prec b(X)\partial^\beta \stackrel{\text{def}}{=} \xi^\alpha \prec \xi^\beta,$$

where $a(X), b(X) \in \mathbf{R}(X)$.

Following the terminology of [106, 110], we refer to the order on $\mathcal{R}$ defined above as a term order instead of a monomial order although, in the case of $\mathcal{R}$, it is exactly the same as a monomial order on the polynomial ring $\mathbf{R}(X)[\xi]$.

Definition A.39 (initial term and initial ideal of $\mathcal{R}$)**.** The *initial term* of a differential operator $l \in \mathcal{R}$ with respect to a term order $\prec$ is

$$\mathrm{in}_\prec(l) \coloneqq a_\alpha(X)\xi^\alpha \in \mathbf{R}(X)[\xi],$$

where α is the largest multi-index with respect to $\prec$. For an ideal of $\mathcal{R}$, its *initial ideal* $\mathrm{in}_\prec(I)$ is defined as an ideal of $\mathbf{R}(X)[\xi]$ as follows:

$$\mathrm{in}_\prec(I) \coloneqq \langle\{\mathrm{in}_\prec(l) \mid l \in I\}\rangle \subset \mathbf{R}(X)[\xi].$$

By using term orders and initial ideals defined above, Gröbner bases (Definition A.18), Macaulay's theorem (Theorem A.2), and zero-dimensional ideals (Definition A.19) for the non-commutative ring $\mathcal{R}$ can be similarly defined.

Definition A.40 (Zero-dimensional ideal of $\mathcal{R}$)**.** Let I be an ideal in $\mathcal{R}$. We call I a *zero-dimensional* ideal in $\mathcal{R}$ if the quotient ring $\mathcal{R}/I$ is a finite-dimensional vector space over $\mathbf{R}(X)$.

Lemma A.7. An ideal $I \subset \mathcal{R}_n = \mathbf{R}(X)\langle\partial\rangle$ is zero-dimensional if and only if $I \cap \mathbf{R}(X)\langle\partial_i\rangle \neq \{0\}$ $(i = 1, \ldots, n)$.

For an ideal $I \subset \mathcal{R}_n$ and a smooth function $f(X) = f(X_1, \ldots, X_n)$, we say that $f(X)$ is a *solution of* I or that I *annihilates* $f(X)$ if $l \bullet f = 0$ holds for all $l \in I$.

Definition A.41 (Holonomic function)**.** Let $f(X)$ be a holomorphic function at a point $X = a \in \mathbf{C}^n$. Then, $f(X)$ is called a *holonomic function* if there exists a zero-dimensional ideal of $\mathcal{R}_n$ that annihilates $f(X)$.

The domain of definition of a holonomic function can be extended as stated in the following theorem.

Theorem A.5. Let $f(X)$ be a holonomic function. There exists a polynomial p such that the function f can be analytically continued to the universal covering space of $\mathbf{C}^n \setminus \mathcal{V}(p)$.

A.3.3 Ideals of $\mathcal{D}$ and holonomic ideals

Contrary to the case of $\mathcal{R}$, the definition of Gröbner bases for $\mathcal{D}$ is quite complicated.

Definition A.42 (Weight vector and order of differential operator)**.** Let $u, v \in \mathbf{R}^n$ be two vectors that satisfy $u_i + v_i \geq 0$ for all $i = 1, \ldots, n$, and the pair (u, v) is called a *weight vector*. Let $l \in \mathcal{D}$ be a differential operator in canonical form:

$$l = \sum_{\beta} a_\beta(X)\partial^\beta = \sum_{(\alpha,\beta)} a_{\alpha,\beta} X^\alpha \partial^\beta, \tag{A.13}$$

where $a_\beta(X) = \sum_\alpha a_{\alpha,\beta} X^\alpha$ for each β. Then, the (u, v)-*order* is defined as

$$\mathrm{ord}_{(u,v)}(l) \coloneqq \max\left\{u \cdot \alpha + v \cdot \beta \mid \alpha, \beta \in \mathbf{Z}^n_{\geq 0}, a_{\alpha,\beta} \neq 0\right\},$$

where $(\cdot)$ denotes the inner product.

Definition A.43 (Associated graded ring)**.** For any weight vector (u, v), by rearranging the variables, we can assume that there exists $m \in \{0, \ldots, n\}$ such that the weight vector satisfies

$$\begin{cases} u_i + v_i = 0 & (1 \leq i \leq m) \\ u_i + v_i > 0 & (m+1 \leq i \leq n) \end{cases}.$$

The *associated graded ring* $\mathrm{gr}_{(u,v)}(\mathcal{D}_n)$ of $\mathcal{D}_n$ with respect to a weight vector (u, v) is the non-commutative ring

$$\mathrm{gr}_{(u,v)}(\mathcal{D}_n) \coloneqq \mathcal{D}_m[X_{m+1}, \ldots, X_n, \xi_{m+1}, \ldots, \xi_n],$$

where $\mathcal{D}_m$ denotes the m-dimensional Weyl algebra in the variables $X_1, \ldots, X_m$.

Example A.7. The associated graded ring $\mathrm{gr}_{(u,v)}(\mathcal{D}_n)$ is identical to the polynomial ring in $2n$ variables $\mathbf{R}[X, \xi]$ when $u_i + v_i > 0$ holds for all $i = 1, \ldots, n$.

Definition A.44 (Initial form)**.** For a differential operator $l \in \mathcal{D}_n$ in canonical form of (A.13), let E be the set of pairs of multi-indices (α, β) such that $a_{\alpha,\beta} \neq 0$. The *initial form* of l with respect to $(u, v) \in \mathbf{R}^{2n}$ is defined as

$$\mathrm{in}_{(u,v)}(l) \coloneqq \sum_{\substack{(\alpha,\beta)\in E \\ u\cdot\alpha+v\cdot\beta=m}} a_{\alpha,\beta} \prod_{\{i|u_i+v_i>0\}} X_i^{\alpha_i}\xi^{\beta_i} \prod_{\{i|u_i+v_i=0\}} X_i^{\alpha_i}\partial^{\beta_i} \in \mathrm{gr}_{(u,v)}(\mathcal{D}_n).$$

Proposition A.3. For an ideal $I \subset \mathcal{D}_n$ and a weight vector $(u, v) \in \mathbf{R}^{2n}$, the set of all finite linear combinations of initial forms $\{\mathrm{in}_{(u,v)}(l) \mid l \in I\}$, denoted by $\mathrm{in}_{(u,v)}(I)$, is an ideal of the associated graded ring $\mathrm{gr}_{(u,v)}(\mathcal{D}_n)$.

Definition A.45 (Initial ideal of $\mathcal{D}$)**.** For an ideal $I \subset \mathcal{D}_n$, the ideal $\mathrm{in}_{(u,v)}(I) \subset \mathrm{gr}_{(u,v)}(\mathcal{D}_n)$ defined above is called the *initial ideal* of I with respect to a weight vector $(u, v) \in \mathbf{R}^{2n}$.

Definition A.46 (Gröbner basis with respect to weight vector)**.** For an ideal $I \subset \mathcal{D}_n$ and a weight vector $(u, v) \in \mathbf{R}^{2n}$, a finite subset $G \subset I$ is a *Gröbner basis* of I with respect to (u, v) if it generates I and the initial forms $\{\mathrm{in}_{(u,v)}(g) \mid g \in G\} \subset \mathrm{gr}_\cdot(u, v)(\mathcal{D}_n)$ generates the initial ideal $\mathrm{in}_{(u,v)}(I) \subset \mathrm{gr}_\cdot(u, v)(\mathcal{D}_n)$.

In the end, we introduce holonomic ideals, which are the ideals of $\mathcal{D}$ corresponding to the zero-dimensional ideals of $\mathcal{R}$.

Definition A.47 (Holonomic ideal)**.** Let $\mathbf{0}$ and $\mathbf{1}$ be two vectors in $\mathbf{R}^n$ whose components are all 0 and 1, respectively. An ideal $I \subset \mathcal{D}_n$ is called *holonomic* if the Krull dimension of its initial ideal with respect to a weight vector $(\mathbf{0}, \mathbf{1})$ is n, that is,

$$\dim \mathrm{in}_{(\mathbf{0},\mathbf{1})}(I) = n.$$

Theorem A.6. For a zero-dimensional ideal $I \subset \mathcal{R}$, the intersection $I \cap \mathcal{D}$ is a *holonomic ideal* in $\mathcal{D}$.

Theorem A.7. Let I be a holonomic ideal in $\mathcal{D}$, and let $\mathcal{R}I$ denote the following set of differential operators:

$$\mathcal{R}I \coloneqq \{a_1 l_1 + \cdots + a_s l_s \mid a_1, \ldots, a_s \in \mathcal{R}; l_1, \ldots, l_S \in I; s \in \mathbf{Z}_{\geq 0}\}.$$

Then, $\mathcal{R}I$ is a zero-dimensional ideal in $\mathcal{R}$.

Appendix B

Proofs of Lemmas and Theorems

B.1 Sufficient optimality conditions for FHOCPs with terminal constraints

We provide a proof of the following lemma, which is the basis of the sufficient conditions for the existence and uniqueness of local optimal feedback laws, which are introduced in Section 3.4 of Chapter 3.

Lemma B.1. For FHOCP (3.1)–(3.4), suppose that Assumption 3.1 holds and the terminal constraint (3.4) satisfies the linear independence constraint qualification. Then, for the sequences $\hat{u}_{[0:N-1]}$ and $\hat{p}_{[0:N]}$ and the vector $\hat{\nu}$ whose existence is assumed in Assumption 3.1, there exists a unique set of differentiable functions $\mathbf{x}_{[0:N]}(x_0)$, $\mathbf{u}_{[0:N-1]}(x_0)$, $\mathbf{p}_{[0:N]}(x_0)$, and $\boldsymbol{\nu}_{[0:N]}(x_0)$ that satisfy $\mathbf{x}_k(\hat{x}_0) = \hat{x}_k$, $\mathbf{u}_k(\hat{x}_0) = \hat{u}_k$, $\mathbf{p}_k(\hat{x}_0) = \hat{p}_k$, $\boldsymbol{\nu}_k(\hat{x}_0) = \hat{\nu}$, the ELEs, and matrix inequalities (3.30) for all x_0 in some neighborhood of $\hat{x}_0$.

Now, let us consider the Lagrangian function of the FHOCP, defined as

$$\begin{aligned}\bar{J} :=& \phi(x_N) + \nu^\top \psi(x_N) \\ &+ \sum_{k=0}^{N-1} \left[H_k(x_k, u_k, p_{k+1}) - p_{k+1}^\top x_{k+1} \right]. \end{aligned} \tag{B.1}$$

and consider the second-order terms in its Taylor expansion at the point $\hat{x}_{[0:N]}$, $\hat{u}_{[0:N-1]}$, $\hat{p}_{[0:N]}$, and $\hat{\nu}$:

$$\begin{aligned} d^2\bar{J} =& \frac{1}{2} dx_N^\top \left[\frac{\partial^2 \phi}{\partial x^2} + \hat{\nu}^\top \frac{\partial^2 \psi}{\partial x^2} \right] dx_N \\ &+ \frac{1}{2} \sum_{k=0}^{N-1} \begin{bmatrix} dx_k \\ du_k \end{bmatrix}^\top \begin{bmatrix} \frac{\partial^2 H_k}{\partial x^2} & \frac{\partial^2 H_k}{\partial x \partial u} \\ \frac{\partial^2 H_k}{\partial u \partial x} & \frac{\partial^2 H_k}{\partial u^2} \end{bmatrix} \begin{bmatrix} dx_k \\ du_k \end{bmatrix}, \end{aligned} \tag{B.2}$$

where dx_k and du_k are infinitesimal changes from $\hat{x}_k$ and $\hat{u}_k$, respectively. The sequence $dx_{[0:N]}$ is defined by the sequence $du_{[0:N-1]}$ and the following linearized system:

$$dx_{k+1} = \frac{\partial f_k}{\partial x} dx_k + \frac{\partial f_k}{\partial u} du_k, \tag{B.3}$$

$$dx_0 = 0, \tag{B.4}$$

$$\frac{\partial \psi}{\partial x} dx_N = 0, \tag{B.5}$$

where the partial derivatives of f_k and ψ in equation (B.3)–(B.5) are evaluated at $\hat{x}_{[0:N]}$ and $\hat{u}_{[0:N-1]}$. Lemma B.1 can be deduced from a classical result in sensitivity analysis [53] if a real number $\alpha > 0$ exists such that

$$d^2 \bar{J} \geq \alpha \left[\sum_{k=0}^{N-1} \left(\|du_k\|^2 + \|dx_{k+1}\|^2 \right) \right] \tag{B.6}$$

holds for any feasible sequences of infinitesimal changes $dx_{[0:N]}$ and $du_{[0:N-1]}$. Therefore, we will prove that the premises in Lemma B.1 imply the positivity of $d^2\bar{J}$ under constraints (B.3)–(B.5). To do so, we will make a change of variables:

$$\eta := \Gamma_N y_N, \tag{B.7}$$

where $y_N := [du_0^\top \cdots du_{N-1}^\top \; dx_1^\top \cdots dx_N^\top]^\top \in \mathbf{R}^{N(n+m)}$ and $\Gamma_N \in \mathbf{R}^{Nm \times N(n+m)}$ is defined as

$$\Gamma_N := \begin{bmatrix} \mathbf{Z}_{uu} & \begin{bmatrix} O_{m\times(N-1)n} & O_{m\times n} \\ \mathbf{Z}_{ux} & O_{(N-1)m\times n} \end{bmatrix} \end{bmatrix} \tag{B.8}$$

and $\mathbf{Z}_{uu}$ and $\mathbf{Z}_{ux}$ are

$$\mathbf{Z}_{uu} := \text{block-diag}\left[Z_{uu}^0, \ldots, Z_{uu}^{N-1} \right] \in \mathbf{R}^{Nm \times Nm},$$

$$\mathbf{Z}_{ux} := \text{block-diag}[Z_{ux}^1, \ldots, Z_{ux}^{N-1}] \in \mathbf{R}^{(N-1)m \times (N-1)n},$$

for the matrices Z_{uu}^k, Z_{ux}^k $(k = 0, \ldots, N-1)$ in Assumption 3.1, and $O_{l_1 \times l_2}$ is the $l_1 \times l_2$ zero matrix.

To prove Lemma B.1 by using the change of variables (B.7), we introduce the following proposition. To simplify the notation, we abbreviate $Z_{ab}(S_{k+1}, k)$ to Z_{ab}^k.

Proposition B.1. Suppose that Assumption 3.1 holds. For any sequence $du_{[0:N-1]} \subset \mathbf{R}^m$ and sequence $dx_{[0:N]} \subset \mathbf{R}^n$ defined by the linearized system (B.3)–(B.5), the following statement holds.

$$\Gamma_N y_N = 0 \Leftrightarrow y_N = 0. \tag{B.9}$$

Proof. Sufficiency ($\Leftarrow$) is trivial; thus, we will prove only necessity ($\Rightarrow$). The proof is by induction for the optimization horizon N. First, for $N = 1$, the matrix Γ_1 is obtained as

$$\Gamma_1 = \begin{bmatrix} Z_{uu}^0 & 0 \end{bmatrix}, \tag{B.10}$$

and $y_1 := [du_0^\top \quad dx_1^\top]^\top$. From the inequality $Z_{uu}^0 > 0$ in Assumption 3.1, $\Gamma_1 y_1 = 0$ implies $du_0 = 0$, which also implies $dx_1 = 0$ from the linearized state equation (B.3) with $dx_0 = 0$.

Next, suppose $y_{N'} = [du_0^\top \cdots du_{N'-1}^\top \; dx_1^\top \cdots dx_{N'}^\top]^\top = 0$. Then,

$$y_{N'+1} = [0 \cdots 0 \; du_{N'} \; 0 \cdots 0 \; dx_{N'+1}]$$

holds, and thus $\Gamma_{N'+1} y_{N'+1} = 0$ implies

$$\Gamma_{N'+1} y_{N'+1} = \begin{bmatrix} 0 \\ Z_{uu}^{N'} du_{N'} \end{bmatrix} = 0. \tag{B.11}$$

From the inequality $Z_{uu}^k > 0$ in Assumption 3.1, $Z_{uu}^{N'} du_{N'} = 0$ implies $du_{N'} = 0$. Accordingly, the linearized state equation (B.3) with $dx_{N'} = 0$ also implies $dx_{N'+1} = 0$. Therefore, the proof is completed by induction. □

Now, we prove Lemma B.1 using Proposition B.1.

Proof. Consider the following quantity that is identical to zero for any sequence $du_{[0:N-1]}$ and the sequence $dx_{[0:N]}$ defined by (B.3)–(B.5):

$$\frac{1}{2} \sum_{k=0}^{N-1} \left[dx_{k+1}^\top S_{k+1} \left(\frac{\partial f_k}{\partial x} dx_k + \frac{\partial f_k}{\partial u} du_k - dx_{k+1} \right) \right], \tag{B.12}$$

where the matrices $S_k \in \mathbf{R}^{n \times n}$ are defined in Assumption 3.1. By completing the square, the sum of $d^2\bar{J}$ and the quantity (B.12) can be written as a quadratic form:

$$\begin{aligned} d^2\bar{J} &= \frac{1}{2} \eta^\top \mathbf{Z}_{uu}^{-1} \eta \\ &+ \frac{1}{2} \sum_{k=0}^{N-1} dx_k^\top \left[Z_{xx}^k - \left(Z_{ux}^k \right)^\top \left(Z_{uu}^k \right)^{-1} Z_{ux}^k - S_k \right] dx_k \\ &+ \frac{1}{2} dx_N^\top \left[\frac{\partial^2 \phi}{\partial x^2} + \hat{\nu}^\top \frac{\partial^2 \psi}{\partial x^2} - S_N \right] dx_N \\ &+ dx_0^\top \left[Z_{xx}^0 - \left(Z_{ux}^0 \right)^\top \left(Z_{uu}^0 \right)^{-1} Z_{ux}^0 \right] dx_0, \end{aligned} \tag{B.13}$$

where $\eta = \Gamma_N y_N$ and $y_N = [du_0^\top \cdots du_{N-1}^\top \; dx_1^\top \cdots dx_N^\top]^\top$. From (3.31) and (3.32), the second and third terms on the right-hand side of (B.13) vanish, and from (B.4), the fourth term also vanishes.

Now, Assumption 3.1 guarantees that $\mathbf{Z}_{uu} > 0$, which implies the existence of a real number $\beta > 0$ such that

$$d^2\bar{J} = \frac{1}{2}\eta^\top \mathbf{Z}_{uu}^{-1}\eta \geq \beta\|\eta\|^2.$$

From Proposition B.1, a real number $\gamma > 0$ exists such that

$$\gamma = \min_{\|y_N\|=1}\|\Gamma_N y_N\|^2$$

holds or, equivalently, $\|\Gamma_N y_N\|^2 \geq \gamma\|y_N\|^2$ holds for any y_N defined by equations (B.3)–(B.5). Therefore,

$$d^2\bar{J} \geq \beta\|\eta\|^2 \geq \beta\gamma\|y_N\|^2 = \alpha\left(\sum_{k=0}^{N-1}\left[\|du_k\|^2 + \|dx_{k+1}\|^2\right]\right)$$

holds for $\alpha := \beta\gamma > 0$. The theorem thus follows from a classical result of sensitivity analysis [53], the proof completes. □

B.2 Necessary condition derived from the quadratic penalty function method

In this section, we provide a proof of Theorem 6.1. This proof is a part of the proof of Theorem 2.1 in [38] and is included here for the completeness of this thesis.

Proof. Let ε be such that the localized minimization problem (6.4) has a unique local minimizer and $r_{[1:\infty]}$ be a sequence of positive real numbers monotonically going to infinity as k does. Since $\hat{x}$ is an interior point of $\mathcal{B}_\varepsilon$, there exists an open subset $U \subset \mathcal{B}_\varepsilon$ including $\hat{x}$. According to the generally accepted result for the quadratic penalty function method [8], there exists a global minimizer sequence $x_{[1:\infty]} \subset \mathcal{B}_\varepsilon$ of the penalty function (6.3) that converges to the global minimizer $\hat{x} \in \mathcal{B}_\varepsilon$ of the localized COP (6.4). In particular, there exists a sufficiently large k^* such that $x_{[k^*:\infty]} \subset U$ holds. In the problem settings of Chapter 6, the penalty function (6.3) is continuous, and thus every x_k for all $k \geq k^*$ satisfies the stationary condition (6.5). □

Bibliography

[1] D. Bertsekas, *Convex Optimization Theory*, Athena Scientific, 2009.

[2] M. Fukushima, *Fundamentals of Nonlinear Optimization, (in Japanese)*, Asakura Publishing, 2001.

[3] R. E. Kalman, "Contributions to the theory of optimal control," *Boletín de la Sociedad Matemática Mexicana*, vol. 5, pp. 102–119, 1960.

[4] R. E. Kalman, "A new approach to linear filtering and prediction problems," *ASME Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[5] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Journal of Basic Engineering*, vol. 83, no. 1, pp. 95–108, 1961.

[6] T. Kailath, *Linear Systems*, Prentice-Hall, 1980.

[7] B. D. O. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*, Dover Publications, 2007.

[8] D. Bertsekas, *Nonlinear Programming*, Athena Scientific, 3rd edition, 2016.

[9] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, 2nd edition, 2006.

[10] A. Ishidori, *Nonlinear Control Systems*, Springer-Verlag, 3rd edition, 2000.

[11] P. A. Parrilo and B. Sturmfels, "Minimizing polynomial functions," *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, vol. 60, pp. 1–16, 2001.

[12] J. B. Lasserre, "Global optimization with polynomials and the problem of moments," *SIAM Journal on Optimization*, vol. 11, no. 3, pp. 796–817, 2001.

[13] M. Safey El Din, "Computing the global optimum of a multivariate polynomial over the reals," *Proceedings of the Twenty-First International Symposium on Symbolic and Algebraic Computation*, pp. 71–78, 2008.

[14] A. Greuet, F. Guo, M. Safey El Din, and L. Zhi, "Global optimization of polynomials restricted to a smooth variety using sums of squares," *Journal of Symbolic Computation*, vol. 47, no. 5, pp. 503–518, 2012.

[15] T. Ohtsuka, "Solutions to the Hamilton-Jacobi equation with algebraic gradients," *IEEE Transactions on Automatic Control*, vol. 56, no. 8, pp. 1874–1885, 2011.

[16] Y. Kawano and T. Ohtsuka, "Observability at an initial state for polynomial systems," *Automatica*, vol. 49, no. 5, pp. 1126–1136, 2013.

[17] T. Ohtsuka, "A recursive elimination method for finite-horizon optimal control problems of discrete-time rational systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 11, pp. 3081–3086, 2014.

[18] L. Menini and A. Tornambè, "On the use of algebraic geometry for the design of high-gain observers for continuous-time polynomial systems," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 19, pp. 43–48, 2014.

[19] T. Yuno and T. Ohtsuka, "A sufficient condition for the stability of discrete-time systems with state-dependent coefficient matrices," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, 2014.

[20] L. Menini, C. Possieri, and A. Tornambè, "A symbolic algorithm to compute immersions of polynomial systems into linear ones up to an output injection," *Journal of Symbolic Computation*, vol. 99, pp. 1–20, 2020.

[21] R. E. Bellman, *Dynamic Programming*, Princeton University Press, 1957.

[22] V. G. Pontryagin, Lev S. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, *The Mathematical Theory of Optimal Processes*, Interscience Publishers, 1962.

[23] J. A. E. Bryson and Y.-C. Ho, *Applied Optimal Control*, John Wiley & Sons, 1st edition, 1975.

[24] N. Sakamoto, "Analysis of the Hamilton-Jacobi equation in nonlinear control theory by symplectic geometry," *SIAM Journal on Control and Optimization*, vol. 40, no. 6, pp. 1924–1937, 2002.

[25] G. Goodwin, M. M. Seron, and J. A. de Doná, *Constrained Control and Estimation: An Optimisation Approach*, Springer-Verlag, 2006.

[26] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, Elsevier Science, 1970.

[27] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, Prentice-Hall, 1979.

[28] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," *SPIE*, vol. 3068, 1997.

[29] K. Ito and K. Xiong, "Gaussian filters for nonlinear filtering problems," *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, 2000.

[30] I. Arasaratnam, S. Haykin, and R. J. Elliott, "Discrete-time nonlinear filtering algorithms using Gauss-Hermite quadrature," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 953–977, 2007.

[31] G. Kitagawa, "Monte Carlo filtering and smoothing method for non-Gaussian nonlinear state space model," *Institute of Statistical Mathematics Research Memorandum*, vol. 462, 1993.

[32] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, 1996.

[33] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings F - Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.

[34] W. Karush, "Minima of functions of several variables with inequalities as side constraints," M. S. Thesis, Unicersity of Chicago, 1939.

[35] H. W. Kuhn and A. W. Tucker, "Nonlinear programming," J. Neyman ed. *the Second Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, 1951.

[36] D. W. Peterson, "A review of constraint qualifications in finite-dimensional spaces," *SIAM Review*, vol. 15, no. 3, pp. 639–654, 1973.

[37] F. J. Gould and J. W. Tolle, "A necessary and sufficient qualification for constrained optimization," *SIAM Journal on Applied Mathematics*, vol. 20, no. 2, pp. 164–172, 1971.

[38] R. Andreani, G. Haeser, and J. M. Martínez, "On sequential optimality conditions for smooth constrained optimization," *Optimization*, vol. 60, no. 5, pp. 627–641, 2011.

[39] B. Buchberger, "Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensinalen Polynomideal (An Algorithm for Finding the Basis Elements in the Residue Class Ring Modulo a Zero Dimensional Polynomial Ideal)," PhD thesis, University of Innsbruck, 1965.

[40] M. P. Abramson, "Bruno Buchberger's PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal," *Journal of Symbolic Computation*, vol. 41, no. 3–4, pp. 475–511, 2006.

[41] K. Hägglöf, P. O. Lindberg, and L. Svensson, "Computing global minima to polynomial optimization problems using Gröbner bases," *Journal of Global Optimization*, vol. 7, no. 2, pp. 115–125, 1995.

[42] U. Walther, T. T. Georgiou, and A. Tannenbaum, "On the computation of switching surfaces in optimal control: A Gröbner basis approach," *IEEE Transactions on Automatic Control*, vol. 46, no. 4, pp. 534–540, 2001.

[43] L. Menini and A. Tornambè, "Stabilization of polynomial nonlinear systems by algebraic geometry techniques," *IEEE Transactions on Automatic Control*, vol. 60, no. 9, pp. 2482–2487, 2015.

[44] T. Yuno and T. Ohtsuka, "Lie derivative inclusion with polynomial output feedback," *Transactions of the Institute of Systems, Control and Information Engineers*, vol. 28, no. 1, pp. 22–31, 2015.

[45] T. Yuno, E. Zerz, and T. Ohtsuka, "Invariance of a class of semi-algebraic sets for polynomial systems with dynamic compensators," *Automatica*, vol. 122, Article 109243, 2020.

[46] H. Nakayama, K. Nishiyama, M. Noro, K. Ohara, T. Sei, N. Takayama, and A. Takemura, "Holonomic gradient descent and its application to the Fisher-Bingham integral," *Advances in Applied Mathematics*, vol. 47, no. 3, pp. 639–658, 2011.

[47] S. G. Nash, "SUMT (revisited)," *Operations Research*, vol. 46, pp. 763–775, 1998.

[48] E. H. Fukuda and M. Fukushima, "A note on the squared slack variables technique for nonlinear optimization," *Journal of the Operations Research Society of Japan*, vol. 60, no. 3, pp. 262–270, 2017.

[49] P. Scokaert, J. Rawlings, and E. Meadows, "Discrete-time stability with perturbations: application to model predictive control," *Automatica*, vol. 33, no. 3, pp. 463–470, 1997.

[50] F. L. Lewis, D. L. Vrabie, and V. L. Syrmos, *Optimal Control*, John Wiley & Sons, 3rd edition, 2012.

[51] I. A. Fotiou, P. Rostalski, P. A. Parrilo, and M. Morari, "Parametric optimization and optimal control using algebraic geometry methods," *International Journal of Control*, vol. 79, no. 11, pp. 1340–1358, 2006.

[52] H. Iwane, A. Kira, and H. Anai, "Construction of explicit optimal value functions by a symbolic-numeric cylindrical algebraic decomposition," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6885 LNCS, pp. 239–250, 2011.

[53] J. F. Bonnans and A. Shapiro, *Perturbation Analysis of Optimization Problems*, Springer, 2000.

[54] D. A. Cox, J. B. J. Little, and D. O'Shea, *Using Algebraic Geometry*, Springer-Verlag, 2nd edition, 2005.

[55] M. Kreuzer and L. Robbiano, *Computational Commutative Algebra 1*, Springer-Verlag, 2000.

[56] P. R. Graves-Morris, "The numerical calculation of Padé approximants," L. Wuytack ed. *Padé Approximation and its Applications*, pp. 231–245, Springer, 1979.

[57] A. Monin, "Modal trajectory estimation using maximum Gaussian mixture," *IEEE Transactions on Automatic Control*, vol. 58, no. 3, pp. 763–768, 2013.

[58] I. Rusnak, "Maximum likelihood optimal estimator of non-autonomous nonlinear dynamic systems," *Proceedings of European Control Conference*, pp. 909–914, 2015.

[59] X. Luo, Y. Jiao, W. L. Chiou, and S. S. Yau, "A novel suboptimal method for solving polynomial filtering problems," *Automatica*, vol. 62, pp. 26–31, 2015.

[60] S. Thrun, W. Burgard, D. Fox, and R. C. Arkin, *Probabilistic Robotics*, MIT Press, 2005.

[61] J. L. Crassidis and F. L. Markley, "Unscented filtering for spacecraft attitude estimation," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 4, pp. 536–542, 2003.

[62] B. Chen and G. Hu, "Nonlinear state estimation under bounded noises," *Automatica*, vol. 98, pp. 159–168, 2018.

[63] J. H. Fernández, J. L. Speyer, and M. Idan, "Stochastic estimation for two-state linear dynamic systems with additive Cauchy noises," *IEEE Transactions on Automatic Control*, vol. 60, no. 12, pp. 3367–3372, 2015.

[64] E. E. Kuruoğlu, W. J. Fitzgerald, and P. J. Rayner, "Near optimal detection of signals in impulsive noise modeled with a symmetric α-stable distribution," *IEEE Communications Letters*, vol. 2, no. 10, pp. 282–284, 1998.

[65] P. Reeves, "A non-Gaussian turbulence simulation," Technical Report AFFDL-TR-69-67, Air Force Flight Dynamics Laboratory, 1969.

[66] M. Idan and J. L. Speyer, "Cauchy estimation for linear scalar systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 6, pp. 1329–1342, 2010.

[67] M. Idan and J. L. Speyer, "State estimation for linear scalar dynamic systems with additive Cauchy noises: Characteristic function approach," *SIAM Journal on Control and Optimization*, vol. 50, no. 4, pp. 1971–1994, 2012.

[68] V. A. Bavdekar, R. B. Gopaluni, and S. L. Shah, "A comparison of moving horizon and Bayesian state estimators with an application to a pH process," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 46, pp. 160–165, 2013.

[69] C. V. Rao, J. B. Rawlings, and D. Q. Mayne, "Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations," *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 246–258, 2003.

[70] A. Alessandri, M. Baglietto, and G. Battistelli, "Moving-horizon state estimation for nonlinear discrete-time systems: New stability results and approximation schemes," *Automatica*, vol. 44, no. 7, pp. 1753–1765, 2008.

[71] A. Alessandri, M. Baglietto, G. Battistelli, and V. Zavala, "Advances in moving horizon estimation for nonlinear systems," *Proceedings of the IEEE Conference on Decision and Control*, pp. 5681–5688, 2010.

[72] A. Wynn, M. Vukov, and M. Diehl, "Convergence guarantees for moving horizon estimation based on the real-time iteration scheme," *IEEE Transactions on Automatic Control*, vol. 59, no. 8, pp. 2215–2221, 2014.

[73] R. E. Mortensen, "Maximum-likelihood recursive nonlinear filtering," *Journal of Optimization Theory and Applications*, vol. 2, no. 6, pp. 386–394, 1968.

[74] C. V. Rao, "Moving horizon strategies for the constrained monitoring and control of nonlinear discrete-time systems," Ph.D. dissertation, University of Wisconsin-Madison, 2000.

[75] R. E. Larson and J. Peschon, "A dynamic programming approach to trajectory estimation," *IEEE Transactions on Automatic Control*, vol. 11, no. 3, pp. 537–540, 1966.

[76] D. A. Cox, J. Little, and D. O'Shea, *Ideals, Varieties, and Algorithms*, Springer International Publishing, 4th edition, 2015.

[77] K. Kühnle and E. Mayr, "Exponential space computation of Gröbner bases.," *International Symposium on Symbolic and Algebraic Computation*, pp. 63–71, 1996.

[78] S. Gao, "Counting zeros over finite fields using Gröbner bases," Ph.D. dissertation, Carnegie Mellon University, 2009.

[79] M. Kauers, "The holonomic toolkit," C. Schneider and J. Bluemlein eds. *Computer Algebra in Quantum Field Theory, Integration, Summation, and Special Fcuntions*, Springer-Verlag, pp. 119–144, 2013.

[80] T. Oaku, Y. Shiraki, and N. Takayama, "Algebraic algorithms for D-Modules and numerical analysis," *Computer Mathematics (Proceedings of ASCM 2003)*, vol. 10, pp. 23–39, 2003.

[81] M. Saito, B. Sturmfels, and N. Takayama, *Gröbner Deformations of Hypergeometric Differential Equations*, Springer-Verlag, 2000.

[82] M. Noro, N. Takayama, H. Nakayama, K. Nishiyama, and K. Ohara, "Risa/Asir: A computer algebra system," `http://www.math.kobe-u.ac.jp/Asir/asir-ja.html`, 2020.

[83] D. R. Grayson and M. E. Stillman, "Macaulay2, a software system for research in algebraic geometry," `http://www.math.uiuc.edu/Macaulay2/`, 2020.

[84] W. Decker, G.-M. Greuel, G. Pfister, and H. Schönemann, "SINGULAR 4-2-0 —A computer algebra system for polynomial computations," `http://www.singular.uni-kl.de`, 2020.

[85] M. P. Deisenroth, *Efficient Reinforcement Learning Using Gaussian Processes*, KIT Scientific Publishing, 2010.

[86] J. M. Martínez and B. F. Svaiter, "A practical optimality condition without constraint qualifications for nonlinear programming," *Journal of Optimization Theory and Applications*, vol. 118, pp. 117–133, 2003.

[87] R. Andreani, J. M. Martínez, and B. F. Svaiter, "A new sequential optimality condition for constrained optimization and algorithmic consequences," *SIAM Journal on Optimization*, vol. 20, no. 6, pp. 3533–3554, 2010.

[88] M. Bierlaire, *Optimization: Principles and Algorithms*, EPFL Press, 1st edition, 2015.

[89] R. Andreani, N. S. Fazzio, M. L. Schuverdt, and L. D. Secchin, "A sequential optimality condition related to the quasi-normality constraint qualification and its algorithmic consequences," *SIAM Journal on Optimization*, vol. 29, no. 1, pp. 743–766, 2019.

[90] C. Kanzow, D. Steck, and D. Wachsmuth, "An augmented Lagrangian method for optimization problems in Banach spaces," *SIAM Journal on Control and Optimization*, vol. 56, no. 1, pp. 272–291, 2018.

[91] N. Andreasson, A. Evgrafov, and M. Patriksson, *An Introduction to Continuous Optimization: Foundations and Fundamental Algorithms*, Dover Publications, 2020.

[92] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Wiley, 1968.

[93] A. F. Izmailov and M. V. Solodov, "Optimality conditions for irregular inequality-constrained problems," *SIAM Journal on Control and Optimization*, vol. 40, no. 4, pp. 1280–1295, 2002.

[94] E. R. Avakov, A. V. Arutyunov, and A. F. Izmailov, "Necessary conditions for an extremum in a mathematical programming problem," *Proceedings of the Steklov Institute of Mathematics*, vol. 256, no. 1, pp. 2–25, 2007.

[95] O. Brezhneva and A. A. Tret'yakov, "When the Karush-Kuhn-Tucker Theorem Fails: Constraint Qualifications and Higher-Order Optimality Conditions for Degenerate Optimization Problems," *Journal of Optimization Theory and Applications*, vol. 174, no. 2, pp. 367–387, 2017.

[96] L. Robbiano and J. Abbott, *Approximate Commutative Algebra*, Springer-Verlag, 2010.

[97] A. Terui, "GPGCD: An iterative method for calculating approximate GCD of univariate polynomials," *Theoretical Computer Science*, vol. 479, pp. 127–149, 2013.

[98] R. Krone, "Numerical algorithms for dual bases of positive-dimensional ideals," *Journal of Algebra and its Applications*, vol. 12, no. 6, pp. 1–21, 2013.

[99] S. Telen, B. Mourrain, and M. Van Barel, "Truncated normal forms for solving polynomial systems," *ACM Communications in Computer Algebra*, vol. 52, no. 3, pp. 78–81, 2018.

[100] E. Kaltofen, J. P. May, Z. Yang, and L. Zhi, "Approximate factorization of multivariate polynomials using singular value decomposition," *Journal of Symbolic Computation*, vol. 43, no. 5, pp. 359–376, 2008.

[101] A. Leykin, "Numerical algebraic geometry," *The Journal of Software for Algebra and Geometry*, vol. 3, pp. 5–10, 2011.

[102] M. Kreuzer and L. Robbiano, *Computational Commutative Algebra 2*, Springer-Verlag, 1st edition, 2005.

[103] J. M. Peña and T. Sauer, "On the multivariate Horner scheme," *SIAM Journal on Numerical Analysis*, vol. 37, no. 4, pp. 1186–1197, 2000.

[104] P. M. Cohn, *Introduction to Ring Theory*, Springer-Verlag, 2000.

[105] S. C. Coutinho, *A Primer of Algebraic D-Modules*, Cambridge University Press, 1995.

[106] T. Hibi ed. *Gröbner Bases: Statistics and Software Systems*, Springer Japan, 1st edition, 2013.

[107] J. Munkres, *Topology*, Pearson, 2nd edition, 2000.

[108] D. Eisenbud, *Commutative Algebra*, vol. 150, Springer, 2004.

[109] S. Lang, *Algebra*, Springer-Verlag, 3rd edition, 2002.

[110] T. Oaku, *D-modules and Computational Mathematics (in Japanese)*, Asakura Publishing, 2002.

List of Publications

Peer-reviewed journal articles

1 Tomoyuki Iori and Toshiyuki Ohtsuka, "Recursive elimination method in moving horizon estimation for a class of nonlinear systems and non-Gaussian noise," *SICE Journal of Control, Measurement, and System Integration*, vol. 13, no. 6, pp. 282–290, 2020. **Chapter 4**

2 Tomoyuki Iori and Toshiyuki Ohtsuka, "Limit operation in projective space for constructing necessary optimality condition of polynomial optimization problem," *Journal of Operations Research Society of Japan*, vol. 63, no. 4, pp. 114–133, 2020. **Chapter 6**

3 Tomoyuki Iori, Yu Kawano, and Toshiyuki Ohtsuka, "Algebraic approach to nonlinear optimal control problems with terminal constraints: sufficient conditions for existence of algebraic solutions," *SICE Journal of Control, Measurement, and System Integration*, vol. 11, no. 3, pp. 198–206, 2018. **Chapter 3**

Peer-reviewed conference proceedings

1 Tomoyuki Iori and Toshiyuki Ohtsuka, "Symbolic-numeric computation of posterior mean and variance for a class of discrete-time nonlinear stochastic systems," In *Proceedings of 59th Conference on Decision and Control (CDC 2020)*, Jeju Island, Korea (South), pp. 4814–4821, 2020. **Chapter 5**

2 Tomoyuki Iori and Toshiyuki Ohtsuka, "Recursive elimination method for moving horizon estimation of discrete-time polynomial systems," In *Proceedings of 58th Conference on Decision and Control (CDC 2019)*, Nice, France, pp. 1076–1082, 2019. **Chapter 4**

3 Tomoyuki Iori, Yu Kawano, and Toshiyuki Ohtsuka, "Algebraic approach to nonlinear finite-horizon optimal control problems with terminal constraints,"

In *Proceedings of 11th Asian Control Conference (ASCC 2017)*, Gold Coast, Australia, pp. 1–6, 2017. **Chapter 3**

4 Tomoyuki Iori, Yu Kawano, and Toshiyuki Ohtsuka, "Algebraic approach to nonlinear finite-horizon optimal control problems of discrete-time systems with terminal constraints," In *Proceedings of SICE Annual Conference 2017*, Kanazawa, Japan, pp. 220–225, 2017. **Chapter 3**

The contents in Chapter 6 were first published in the following journal paper:
T. Iori and T. Ohtsuka, "Limit operation in projective space for constructing necessary optimality condition of polynomial optimization provlems," *Journal of Operations Research Society of Japan*, vol. 63, no. 4, pp. 114–133, 2020. (DOI: `https://doi.org/10.15807/jorsj.63.114`)