East Asian Journal on Applied Mathematics
doi: 10.4208/eajam.070514.161014a

Vol. 4, No. 4, pp. 345-367
November 2014

Computing Switching Surfaces in Optimal Control Based on Triangular Decomposition

Xiaoliang Li[1,4], Yanli Huang[2,4,*], Zewei Zheng[3] and Wanyou Cheng[1]

[1] *School of Computer Science, Dongguan University of Technology, Dongguan, Guangdong 523808, China.*
[2] *School of Computer Science and Software Engineering, Tianjin Polytechnic University, Tianjin 300387, China.*
[3] *School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China.*
[4] *Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Guangxi University for Nationalities, Nanning 530006, China.*

Received 7 May 2014; Accepted (in revised version) 16 October 2014

Available online 5 November 2014

Abstract. Various algorithms for optimal control require the explicit determination of switching surfaces. However, switching strategies may be very complicated, such that the computation of switching surfaces is quite challenging. General methods are proposed here to compute switching surfaces systematically, based on algebraic computational tools such as triangular decomposition. Our methods are highly complex compared to some widely-used numerical options, but they can be made feasible for real-time applications by moving the computational burden off-line. The tutorial-style presentation is intended to introduce potentially powerful symbolic computation methods to system scientists in particular, and an illustrative example of time-optimal control is given to show the effectiveness and generality of our approach.

AMS subject classifications: 13P15, 13P25

Key words: Optimal control, switching surface, triangular decomposition, semi-algebraic system.

1. Introduction

Optimal control has been used in many areas of modern system science such as aeronautics, astronautics, robotics and power electronics [8, 13, 16, 26, 41]. The control algorithms typically require explicit determination of switching surfaces — surfaces where the sign of the control input changes. However, switching strategies may be very complicated in many practical applications, so the computation of switching surfaces becomes quite challenging. Some special approaches to their computation have been developed [1, 18, 24]. Walther *et*

*Corresponding author. *Email address:* `huangyanlibuaa@gmail.com` (Yanli Huang)

al. [30] introduced tools from computational algebraic geometry for time-optimal control problems, by transforming the computation of switching surfaces into a combinatorial problem. Their main idea is to first compute Gröbner bases of particular polynomial equations deduced from the original system, and then use Sturm's theorem to determine whether the equations have non-negative solutions. Their approach is somewhat non-systematic and may not be feasible for real-time control due to its high complexity, which motivated the development of more general systematic methods for computing switching surfaces presented here. Another purpose of this article is to introduce powerful methods of symbolic computation (coupled with numerical computation) for system scientists.

Our methods are based on triangular decomposition and related algebraic tools. Like Gröbner bases, triangular decomposition is a main elimination approach for solving systems of multivariate polynomial equations. For example, consider the equations

$$\begin{cases} P_1 = x_1^2 + x_2 + x_3 - 1 = 0\,, \\ P_2 = x_2^2 + x_3 + x_1 - 1 = 0\,, \\ P_3 = x_3^2 + x_1 + x_2 - 1 = 0\,. \end{cases}$$

Under the variable ordering $x_1 < x_2 < x_3$, triangular decomposition of the polynomial set $\mathscr{P} = \{P_1, P_2, P_3\}$ results in

$$\begin{aligned} &\mathscr{T}_1 = [x_1^2 + 2x_1 - 1, x_2 - x_1, x_3 - x_1], \quad \mathscr{T}_2 = [x_1 - 1, x_2, x_3]\,, \\ &\mathscr{T}_3 = [x_1, x_2 - 1, x_3], \quad \mathscr{T}_4 = [x_1, x_2, x_3 - 1]\,, \end{aligned}$$

such that the union of the zero sets of $\mathscr{T}_1, \cdots, \mathscr{T}_4$ is identical to the zero set of $\mathscr{P}$. The *triangular sets* $\mathscr{T}_1, \cdots, \mathscr{T}_4$ are of triangular form. Note that the zero set of a triangular set can readily be obtained by successively computing the zeros of its polynomials. In our algorithms, triangular decomposition is thus used as a preprocessing tool in the analysis of solutions of polynomial equations. Formal notation and properties related to triangular decomposition are provided in subsection 3.1 below.

Since triangular decomposition is such a key aspect of our approach, a brief literature overview may be helpful. In considering differential ideals, Ritt [25] introduced the notion of characteristic set, one of the best known concepts for triangular sets. Several decades later, Wu [37, 38] extended Ritt's work by removing irreducibility requirements of characteristic sets, proposing efficient algorithms for decomposing polynomial sets, and successfully applying them to geometric theorem proof. Wu's method was intensively studied and improved by a number of researchers (e.g. [4, 9–11, 21, 34, 35], but the zero set of a characteristic set may be empty. To avoid this degeneracy, Kalkbrener [15] and Yang *et al.* [39] independently introduced the notion of a regular set, and proposed methods for decomposing any algebraic variety into finite components represented by regular sets. Wang [31] proposed another method for triangular decomposition, which is considered to be quite efficient. Other relevant work on the triangular decomposition of polynomial sets is discussed in Refs. [3, 17, 23, 33] (and references therein).

Our work presented here is based on the observation that the problem of computing the switching surfaces of the time-optimal control can be translated into identifying whether there exist (complex and non-negative real) solutions of particular equations, say $\mathscr{F}$ with parameters — cf. Section 2. By applying methods of triangular decomposition, $\mathscr{F}$ can be rendered by a finite number of triangular sets with a simpler form. The projection property of a special class of triangular sets may then be used to directly identify whether $\mathscr{F}$ has complex solutions. However, it is by no means trivial to determine whether or not there exists any non-negative real solution of $\mathscr{F}$. We propose general algorithms to solve this problem, based on a number of powerful tools from real algebraic geometry such as Sturm's theorem, real root isolation and cylindrical algebraic decomposition. The border polynomial notation plays an important role, and its formal definition and relevant property are given in subsection 3.2. All the algorithms presented here have been implemented in Maple 17 based on the Epsilon package[†]. (The source code together with experimental results is available from the authors on request.)

In Section 2, as an illustrative example we consider a simple system consisting of a chain of integrators, and address two (complex and real) versions of the problem related to the computation of optimal control. Section 3 summarises our notation, such as for triangular sets and semi-algebraic systems. In Section 4, the complex version of the problem is solved directly, based on the triangular decomposition of a particular polynomial set. Sections 5 and 6 focus on the resolution of the real version of the problem. In Section 5, we propose an approach to determine the switching strategy of the system given in Section 2; and in Section 6, we modify our algorithms for real-time applications by moving the computational burden off-line. Section 7 presents our concluding remarks.

2. Problem Statement

As previously mentioned, our emphasis is on devising systematic general approaches to compute optimal control for a system consisting of a chain of integrators. For convenience of comparison, we consider the same example as in Ref. [30] — viz. the third-order linear system with saturated control input

$$\dot{X}_3(t) = X_2(t)\,, \quad \dot{X}_2(t) = X_1(t)\,, \quad \dot{X}_1(t) = v(t)\,, \tag{2.1}$$

where the dot denotes differentiation with respect to time t and $|v(t)| \leq 1$. It is well known that minimum-time optimal control for this system leads to "bang-bang" control with at most three switchings. However, although we consider such a very simple example, our methods are completely general and can be extended in a straightforward manner to systems of any order.

The objective is to take the minimum time t_f to drive the system (2.1) from any given initial state $(X_1(0), X_2(0), X_3(0)) = (a, b, c)$ to the origin $(X_1(t_f), X_2(t_f), X_3(t_f)) = (0, 0, 0)$. For simplicity, let us first consider the inverse problem — viz. the minimum time to drive the system from $(X_1(0), X_2(0), X_3(0)) = (0, 0, 0)$ to $(X_1(t_f), X_2(t_f), X_3(t_f)) = (a, b, c)$. The equivalence of these two problems is demonstrated at the end of this section.

[†]Available at http://www-spiral.lip6.fr/∼wang/epsilon/

Suppose that

$$v(t)=\begin{cases} +1\,, & 0\le t<t_1\,,\\ -1\,, & t_1\le t<t_1+t_2\,,\\ +1\,, & t_1+t_2\le t<t_f=t_1+t_2+t_3\,,\end{cases} \tag{2.2}$$

where $t_1, t_2, t_3 \geq 0$ are the length of successive intervals where $v(t)$ stays constant. Then it is easy to prove that

$$\begin{cases} X_1(t_f)=t_1-t_2+t_3\,,\\ X_2(t_f)=\dfrac{t_1^2}{2}+t_1t_2+\dfrac{t_3^2}{2}+t_3t_1-\dfrac{t_2^2}{2}-t_2t_3\,,\\ X_3(t_f)=\dfrac{t_1^3}{6}+\dfrac{t_3^3}{6}+\dfrac{t_1^2t_2}{2}+\dfrac{t_1^2t_3}{2}+\dfrac{t_2^2t_1}{2}+\dfrac{t_3^2t_1}{2}+t_1t_2t_3-\dfrac{t_2^3}{6}-\dfrac{t_2^2t_3}{2}-\dfrac{t_3^2t_2}{2}\,.\end{cases}$$

Note that $X_1(t_f) = a$, $X_2(t_f) = b$ and $X_3(t_f) = c$. (The above relations would be more complicated for the direct problem where the initial state is not at the origin.) We consider the following polynomial equations:

$$\begin{cases} F_1=t_1-t_2+t_3-a=0\,,\\ F_2=\dfrac{t_1^2}{2}+t_1t_2+\dfrac{t_3^2}{2}+t_3t_1-\dfrac{t_2^2}{2}-t_2t_3-b=0\,,\\ F_3=\dfrac{t_1^3}{6}+\dfrac{t_3^3}{6}+\dfrac{t_1^2t_2}{2}+\dfrac{t_1^2t_3}{2}+\dfrac{t_2^2t_1}{2}+\dfrac{t_3^2t_1}{2}+t_1t_2t_3-\dfrac{t_2^3}{6}-\dfrac{t_2^2t_3}{2}-\dfrac{t_3^2t_2}{2}-c=0\,,\end{cases} \tag{2.3}$$

involving the variables t_1, t_2, t_3 and parameters a, b, c.

Remark 2.1. Determining the optimal control v of system (2.1) can be transformed to solving system (2.3). If there exists any real solution such that $t_1 \geq 0, t_2 \geq 0, t_3 \geq 0$, then the optimal control v should be set as in (2.2). There are a number of numerical methods for solving polynomial equations — e.g. Newton–Raphson methods [7] and homotopy continuation methods [29]. Numerical methods may suffer from computational instability and the difficulty in identifying signs of the solutions (particularly near zero) — whereas our approach based on (exact) symbolic computation overcomes these shortcomings, since computing the optimal control v then involves determining the solution signs instead of explicitly solving the above equations.

Both the complex and real version of the problem of solving system (2.3) are now considered — viz.

Problem 1. For any given complex numbers a, b and c, do complex solutions of system (2.3) exist?
and
Problem 2. For any given real numbers a, b and c, does system (2.3) have at least one real solution satisfying $t_1 \geq 0$, $t_2 \geq 0$ and $t_3 \geq 0$?

Remark 2.2. If the answer to Problem 1 is negative, the optimal control v should take the values $-1,+1,-1$, successively. If the answer to Problem 2 is positive, then v should be assumed to be $+1,-1,+1$, and the length of the successive intervals where $v(t)$ stays constant should be the non-negative components t_1, t_2 and t_3 of the solution; otherwise, v should take the values $-1,+1,-1$, successively. Further details are given in Ref. [30].

Suppose that system (2.1) has been driven from the origin to (a,b,c) by setting (2.2). From basic control theory, by reversing the direction of time as well as the sign of control the path is traversed backwards. More precisely, if

$$v(t)=\begin{cases}-1, & 0\le t<t_3\,,\\ +1, & t_3\le t<t_3+t_2\,,\\ -1, & t_3+t_2\le t<t_3+t_2+t_1\,,\end{cases}$$

then system (2.1) moves back from (a,b,c) to $(0,0,0)$. This shows that the direct problem is equivalent to its inverse.

3. Preliminaries

Let us now introduce some notations and properties for the triangular set and semi-algebraic system, for the problems given in the previous section.

3.1. Triangular decomposition methods

In the following, $\mathbb{R}$ and $\mathbb{C}$ denote the real and complex fields, respectively. We use $\mathbb{R}[x_1,\cdots,x_n]$ to denote the multivariate polynomial ring over $\mathbb{R}$ with variables ordered as $x_1<\cdots<x_n$. Let $F\in\mathbb{R}[x_1,\cdots,x_n]$. We call $\mathrm{lv}(F)=\max_<\{x_i\mid \deg(F,x_i)\neq 0, 1\le i\le n\}$ the *leading variable* of F. The leading coefficient of F, viewed as a univariate polynomial in its leading variable, is called the *initial* of F and denoted by $\mathrm{ini}(F)$.

Let $\mathscr{P}$ and $\mathscr{Q}$ be two sets of multivariate polynomials with coefficients in $\mathbb{R}$. We denote by $\mathrm{Zero}(\mathscr{P})$ the set of all common zeros in $\mathbb{C}^n$ of the polynomials in $\mathscr{P}$, and by $\mathrm{Zero}(\mathscr{P}/\mathscr{Q})$ the subset of $\mathrm{Zero}(\mathscr{P})$ with elements that do not annihilate any polynomial in $\mathscr{Q}$.

Definition 3.1 (Triangular Set)**.** An ordered polynomial set $[T_1,\cdots,T_r]\subseteq\mathbb{R}[x_1,\cdots,x_n]\backslash\mathbb{R}$ is called a *triangular set* if the leading variable of T_i is ordered smaller than that of T_j for any $i<j$.

A triangular set $[T_1,\cdots,T_r]$ thus has a simple special structure, and its zeros can easily be obtained by successively solving $T_1=0,\cdots,T_r=0$. However, for a generic triangular set $\mathscr{T}$, it is not guaranteed that the corresponding zero set $\mathrm{Zero}(\mathscr{T}/\mathrm{ini}(\mathscr{T}))$ is non-empty. For example, if

$$\mathscr{T}=\left[x_1^2-u, x_2^2+2x_1x_2+u,(x_1+x_2)x_3+1\right]$$

where $u<x_1<x_2<x_3$, then $\mathrm{Zero}(\mathscr{T}/\mathrm{ini}(\mathscr{T}))=\emptyset$ since any common zero of the first two polynomials annihilates the initial $x+y$ of the last polynomial.

The emptiness of the zero sets of triangular sets would be a problem when counting the real solutions of polynomial equations. To avoid this, triangular sets of other kinds (with better properties) are needed. Such triangular sets include for example *regular sets* [15,33, 39], *simple sets* [32] and *irreducible triangular sets* [36,37]. For instance, the definition of regular set is as follows.

Definition 3.2 (Regular Set). A triangular set $[T_1,\cdots,T_r]$ is said to be *regular* or called a *regular set* if no regular zero of $\mathscr{T}_i$ annihilates the initial of T_{i+1} for all $i=1,\cdots,r-1$, where $\mathscr{T}_i=[T_1,\cdots,T_i]$ and a regular zero of $\mathscr{T}_i$ is a zero of $\mathscr{T}_i$ such that the variables other than the leading variables of $T_1,\cdots,T_i$ are not specific values.

Effective algorithms have been developed by Wu [37,38], Lazard [17], Kalkbrener [15] and others [2,12,14,19,23] to decompose any polynomial set $\mathscr{P}$ into finitely many triangular sets $\mathscr{T}_1,\cdots,\mathscr{T}_k$ with different properties such that

$$\operatorname{Zero}(\mathscr{P})=\bigcup_{i=1}^{k}\operatorname{Zero}(\mathscr{T}_i/\operatorname{ini}(\mathscr{T}_i)), \tag{3.1}$$

where $\operatorname{ini}(\mathscr{T}_i)$ denotes the set of initials of all polynomials in $\mathscr{T}_i$. Wang [31–33] has also proposed efficient algorithms for computing the triangular decomposition as (3.1). His methods are more general, and can be used to decompose a polynomial system $[\mathscr{P},\mathscr{Q}]$ (both $\mathscr{P}$ and $\mathscr{Q}$ are polynomial sets) into finite *triangular systems* $[\mathscr{T}_i,\mathscr{S}_i], i=1,\cdots,k$ such that

$$\operatorname{Zero}(\mathscr{P}/\mathscr{Q})=\bigcup_{i=1}^{k}\operatorname{Zero}(\mathscr{T}_i/\mathscr{S}_i),$$

where $[\mathscr{T}_i,\mathscr{S}_i]$ could be *fine triangular systems* [31], *regular systems* [33], or *simple systems* [32], respectively corresponding to triangular sets, regular sets, or simple sets. Wang's algorithms have been implemented in the Maple package Epsilon, which serves as one of our main computational tools here. Let $\mathscr{P}=[x_1x_2^2+x_3^2, x_1x_3+x_2]$ and $\mathscr{Q}=\{x_1\}$ with $x_1<x_2<x_3$. For example, applying the RegSer function in Epsilon, the polynomial system $[\mathscr{P},\mathscr{Q}]$ may be decomposed into 2 regular systems:

$$\Big[[x_2,x_3],\{x_1,x_1^3+1\}\Big],\quad \Big[[x_1^3+1,x_1x_3+x_2],\{\}\Big]. \tag{3.2}$$

We use $\mathscr{P}_{\le s}$ to denote the subset of $\mathscr{P}$, where only polynomials with leading variable smaller than and equal to x_s are contained. $\operatorname{Proj}_s Z$ stands for the projection of a zero set Z into the subspace $\mathbb{C}^s:=\{(x_1,\cdots,x_s)\,|\,x_1,\cdots,x_s\in\mathbb{C}\}$ — e.g. if $Z=\{(0,1,2),(5,6,7)\}$, then $\operatorname{Proj}_2 Z=\{(0,1),(5,6)\}$. The following proposition indicates that the projection of the zero set of a polynomial system can readily be obtained by computing its regular systems.

Proposition 3.1 (Projection Property [36]). *Suppose that* $[\mathscr{T}_i,\mathscr{S}_i]$, $i=1,\cdots,k$ *are regular systems of the polynomial system* $[\mathscr{P},\mathscr{Q}]$. *Then, for any* $s=1,\cdots,n$,

$$\operatorname{Proj}_s\operatorname{Zero}(\mathscr{P}/\mathscr{Q})=\bigcup_{i=1}^{k}\operatorname{Zero}((\mathscr{T}_i)_{\le s}/(\mathscr{S}_i)_{\le s}).$$

A triangular set in which all polynomials other than the first are linear with respect to their corresponding leading variables is said to be *quasi-linear*. Furthermore, a triangular system $[\mathscr{T},\mathscr{S}]$ is said to be *quasi-linear* if $\mathscr{T}$ is quasi-linear — e.g. the two regular systems in (3.2) are all quasi-linear by definition. A quasi-linear triangular set $\mathscr{T}$ (or triangular system $[\mathscr{T},\mathscr{S}]$) has extremely simple structure, with its zeros easily obtained by analysing the first polynomial in $\mathscr{T}$. The following theorem paves the way for decomposing a polynomial system into a finite number of quasi-linear triangular systems (cf. Ref. [20] for a proof).

Theorem 3.1. *Let $[\mathscr{T},\mathscr{S}]$ be a regular system with $\mathscr{T}=[T_1(\boldsymbol{u},x_1),\cdots,T_r(\boldsymbol{u},x_1,\cdots,x_r)]$, where $x_1,\cdots,x_r$ are respectively the leading variables of $T_1,\cdots,T_r$ and $\boldsymbol{u}$ are its parameters. For a random sequence of integers $c_2,\cdots,c_r$, the probability is 1 that all simple systems (under the same ordering $x_1<\cdots<x_r$) of $[\mathscr{T}^*,\mathscr{S}^*]$ are quasi-linear, where $\mathscr{T}^*$ and $\mathscr{S}^*$ are obtained from $\mathscr{T}$ and $\mathscr{S}$ respectively by replacing x_1 with $x_1+c_2x_2+\cdots+c_rx_r$.*

Triangular systems produced by triangular decomposition are often quasi-linear — cf. (3.2) for example. However, this is not always the case. In practice, to obtain quasi-linear triangular systems of a polynomial system $[\mathscr{P},\mathscr{Q}]$, one may first decompose $[\mathscr{P},\mathscr{Q}]$ into regular systems $[\mathscr{T}_i,\mathscr{S}_i]$, $i=1,\cdots,s$. Then for those regular systems, say $[\mathscr{T}_j,\mathscr{S}_j]$, $j=1,\cdots,t$, that are not quasi-linear, from Theorem 3.1 one may randomly choose $c_{2j},\cdots,c_{rj}$ and decompose $[\mathscr{T}_j^*,\mathscr{S}_j^*]$ into simple systems until the resulting triangular systems are all quasi-linear.

3.2. Semi-algebraic system and border polynomial

Definition 3.3 (Semi-algebraic System)**.** A semi-algebraic system is an equation set of the form

$$\begin{cases} F_1(u_1,\cdots,u_s,x_1,\cdots,x_n)=0\,,\\ \qquad\vdots\\ F_n(u_1,\cdots,u_s,x_1,\cdots,x_n)=0\,,\\ P_1(u_1,\cdots,u_s,x_1,\cdots,x_n)\lesseqgtr 0\,,\\ \qquad\vdots\\ P_r(u_1,\cdots,u_s,x_1,\cdots,x_n)\lesseqgtr 0\,,\end{cases}$$

where F_i and P_j are polynomials over $\mathbb{R}$ with $u_1,\cdots,u_s$ as their parameters and $x_1,\cdots,x_n$ as their variables, and the symbol $\lesseqgtr$ represents $>$, $\geq$, $<$, $\leq$ or $\neq$.

Semi-algebraic systems are often prevalent in practice. Indeed, real solutions of particular semi-algebraic systems can be seen to characterise various problems in science and engineering. For example, Problem 2 in Section 2 can be reduced to determining whether there exists any real solution of $\{F_1=0,F_2=0,F_3=0,t_1\geq 0,t_2\geq 0,t_3\geq 0\}$.

Definition 3.4. Let $A=\sum_{i=0}^{m}a_i\,x^i$ and $B=\sum_{j=0}^{l}b_j\,x^j$ be two univariate polynomial in x, where $a_i, b_j \in \mathbb{C}$ and $a_m, b_l \neq 0$. The determinant

$$\left|\begin{array}{cccccc} a_m & a_{m-1} & \cdots & a_0 & & \\ & \ddots & \ddots & \ddots & \ddots & \\ & & a_m & a_{m-1} & \cdots & a_0 \\ b_l & b_{l-1} & \cdots & b_0 & & \\ & \ddots & \ddots & \ddots & \ddots & \\ & & b_l & b_{l-1} & \cdots & b_0 \end{array}\right| \begin{array}{l} \left.\vphantom{\begin{array}{c} a \\ \ddots \\ a \end{array}}\right\} l \\ \left.\vphantom{\begin{array}{c} b \\ \ddots \\ b \end{array}}\right\} m \end{array}$$

is called the *Sylvester resultant* (or simply the *resultant*) of A and B, and denoted by $\mathrm{Res}(A,B)$. The resultant $\mathrm{Res}(A, \mathrm{d}A/\mathrm{d}x)$ is called the *discriminant* of A and denoted by $\mathrm{Discr}(A)$.

The following two propositions follow from this the definition.

Proposition 3.2 (cf. Ref. [22]). *$A = 0$ and $B = 0$ have common roots in $\mathbb{C}$ if and only if*

$$\mathrm{Res}(A,B) = 0\,.$$

Proposition 3.3 (cf. Ref [22]). *$A = 0$ has multiple roots in $\mathbb{C}$ if and only if* $\mathrm{Discr}(A) = 0$.

For the real solution classification of a semi-algebraic system, a crucial concept is the border polynomial originally introduced by Yang *et al.* [40]. Here we use a simpler notation for border polynomials, suitable for special semi-algebra systems with one single variable (Yang's notation is more general).

Definition 3.5 (Border Polynomial). Consider the semi-algebraic system with only one variable x:

$$\mathbb{S} = \begin{cases} F(\mathbf{u}, x) = 0\,, \\ P_1(\mathbf{u}, x) > 0, \cdots, P_s(\mathbf{u}, x) > 0\,, \end{cases}$$

where $\mathbf{u}$ are parameters and $F(\mathbf{u}, x) = \sum_{i=0}^{m} a_i(\mathbf{u})\, x^i$. Then we call the product

$$a_m(\mathbf{u}) \cdot \mathrm{Discr}(F) \cdot \prod_{i=1}^{s} \mathrm{Res}(F, P_i)$$

the *border polynomial* of $\mathbb{S}$.

Theorem 3.2. *The zeros of the border polynomial of $\mathbb{S}$ divide the parameter space into separated regions. For each region, the number of distinct real solutions of $\mathbb{S}$ is invariant.*

Proof. The number of distinct real solutions of $F = 0$ changes if and only if the leading coefficient $a_m(\mathbf{u})$ or the discriminant $\mathrm{Discr}(F)$ goes from non-zero to zero and vice versa. Suppose that the number of real solutions of $F = 0$ is fixed. If any $\mathrm{Res}(F, P_i)$ goes across zero, then real zeros of F may pass through boundaries of the intervals determined by $P_i > 0$, so the number of real solutions of $\mathbb{S}$ may vary. For any given region, the signs of $a_m(\mathbf{u})$, $\mathrm{Discr}(F)$ and $\mathrm{Res}(F, P_i)$ remain the same, hence the number of distinct real solutions of $\mathbb{S}$ is invariant.

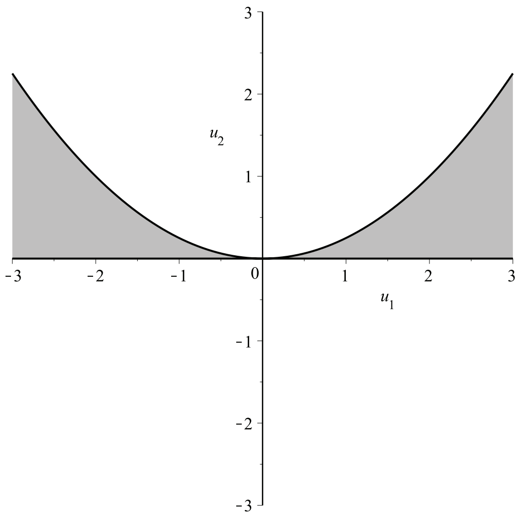

Figure 1: The parameter space for Example 3.1.

Example 3.1. Consider the semi-algebraic system $\{x^2+u_1x+u_2=0,\ x>0\}$. We have that $\mathrm{Discr}(x^2+u_1x+u_2)=u_1^2-4u_2$ and $\mathrm{Res}(x^2+u_1x+u_2,x)=u_2$, so the border polynomial of this system is $u_2(u_1^2-4u_2)$. Its zeros divide the parameters space $\{(u_1,u_2)\,|\,u_1,u_2\in\mathbb{R}\}$ into 4 separate regions as shown in Fig. 1. From Theorem 3.2, the number of distinct real solutions of the considered semi-algebraic system is invariant, which is obvious for this simple example.

4. Solving Problem 1 using Triangular Decomposition

Let $\mathscr{F}=[F_1,F_2,F_3]$ be the set of polynomials in system (2.3). Decomposing $[\mathscr{F},\emptyset]$ into regular systems under the variable ordering $a<b<c<t_1<t_2<t_3$ by the RegSer function in the Epsilon package, we obtain 6 regular systems $[\mathscr{T}_1,\mathscr{S}_1],\cdots,[\mathscr{T}_6,\mathscr{S}_6]$ satisfying

$$\mathrm{Zero}(\mathscr{F})=\bigcup_{i=1}^{6}\mathrm{Zero}(\mathscr{T}_i/\mathscr{S}_i). \tag{4.1}$$

To save space, we give only the first branch $[\mathscr{T}_1,\mathscr{S}_1]=[[T_1,T_2,T_3],\{S_1,S_2\}]$ with

$$\begin{aligned}
&T_1=I_1t_1^4+(48\,a^3-144\,ab+144\,c)t_1^3+(-18\,a^4-72\,b^2+72\,a^2)t_1^2\\
&\qquad +a^6-6\,a^4b-48\,a^3c+36\,a^2b^2+144\,abc-72\,b^3-72\,c^2\,,\\
&T_2=I_2t_2+J_2,\quad T_3=-t_3+t_2-t_1-a,\quad I_1=-36\,a^2+72\,b\,,\\
&I_2=-6\,t_1^2+3\,a^2-6\,b,\quad J_2=(-3\,a^2+6\,b)t_1+2\,a^3-6\,ab+6\,c\,,\\
&S_1=a^2-2\,b,\quad S_2=a^6-6\,a^4b-48\,a^3c+36\,a^2b^2+144\,abc-72\,b^3-72\,c^2\,.
\end{aligned} \tag{4.2}$$

For any given value of the parameters a,b,c such that $S_1\neq 0$, $S_2\neq 0$ or simply $S_1S_2\neq 0$, the initial I_1 of the polynomial T_1 is non-zero. Furthermore, since $\mathrm{Res}(T_1,I_2)=1296\,S_2^2$, the initial I_2 of T_2 is also non-zero if $S_1S_2 \neq 0$ and $T_1 = 0$. Thus provided that $S_1S_2 \neq 0$, $\mathscr{T}_1$ always has complex zeros, which can readily be obtained by solving $T_1 = 0, T_2 = 0$ and $T_3 = 0$ for t_1, t_2 and t_3, respectively. Hence $\mathrm{Zero}(\mathscr{T}_1/\mathscr{S}_1) \neq \emptyset$, and similarly one can prove that $\mathrm{Zero}(\mathscr{T}_i/\mathscr{S}_i) \neq \emptyset$ for $i = 2, \cdots, 6$. We formalise these results in the following proposition:

Proposition 4.1 (Non-emptiness — cf. Ref [36]). *For any regular system $[\mathscr{T}, \mathscr{S}]$, we have* $\mathrm{Zero}(\mathscr{T}/\mathscr{S}) \neq \emptyset$.

To solve Problem 1, we need to project each $\mathrm{Zero}(\mathscr{T}_i/\mathscr{S}_i)$ into the complex parameter space $\mathbb{C}^3 = \{(a, b, c) \,|\, a, b, c \in \mathbb{C}\}$ and check whether or not the projections cover the entire parameter space. Let us use $[\overline{\mathscr{T}_i}, \overline{\mathscr{S}_i}]$ to denote the regular system corresponding to the projection of $\mathrm{Zero}(\mathscr{T}_i/\mathscr{S}_i)$. From the projection property (Proposition 3.1), these $[\overline{\mathscr{T}_i}, \overline{\mathscr{S}_i}]$ are easily obtained — viz.

$$
\begin{aligned}
[\overline{\mathscr{T}_1}, \overline{\mathscr{S}_1}] &= [[\ ], \{S_1, S_2\}], \\
[\overline{\mathscr{T}_2}, \overline{\mathscr{S}_2}] &= [[S_1, U_1], \{\}], \\
[\overline{\mathscr{T}_3}, \overline{\mathscr{S}_3}] &= [[S_1], \{U_2, U_3\}], \\
[\overline{\mathscr{T}_4}, \overline{\mathscr{S}_4}] &= [[S_2], \{S_1\}], \\
[\overline{\mathscr{T}_5}, \overline{\mathscr{S}_5}] &= [[S_2], \{S_1\}], \\
[\overline{\mathscr{T}_6}, \overline{\mathscr{S}_6}] &= [[S_1, U_4], \{\}],
\end{aligned}
\tag{4.3}
$$

with $U_1 = a^3 - 6c$, $U_2 = a^3 - 3ab + 3c$,
$U_3 = a^6 - 6a^4b - 48a^3c + 144abc - 72c^2$, $U_4 = a^6 - 12a^3c + 36c^2$.

We now prove that zeros of the above systems cover the entire parameter space $\mathbb{C}^3$. It is obvious that

$$\mathrm{Zero}(\overline{\mathscr{T}_1}/\overline{\mathscr{S}_1}) \cup \mathrm{Zero}(\overline{\mathscr{T}_4}/\overline{\mathscr{S}_4}) = \mathbb{C}^3 \setminus \mathrm{Zero}([S_1]),$$

so that

$$\mathrm{Zero}(\overline{\mathscr{T}_1}/\overline{\mathscr{S}_1}) \cup \mathrm{Zero}(\overline{\mathscr{T}_4}/\overline{\mathscr{S}_4}) \cup \mathrm{Zero}(\overline{\mathscr{T}_3}/\overline{\mathscr{S}_3}) = \mathbb{C}^3 \setminus \Lambda,$$

where $\Lambda = \mathrm{Zero}([S_1, U_2]) \cup \mathrm{Zero}([S_1, U_3])$. Furthermore, it can be proved from the PolynomialIdeals package in Maple that

$$\langle S_1, U_1 \rangle \cap \langle S_1, U_4 \rangle \subseteq \sqrt{\langle S_1, U_2 \rangle}, \qquad \langle S_1, U_1 \rangle \cap \langle S_1, U_4 \rangle \subseteq \sqrt{\langle S_1, U_3 \rangle},$$

where $\langle \mathscr{P} \rangle$ is the polynomial ideal generated by the polynomial set $\mathscr{P}$ and $\sqrt{\mathscr{I}}$ is the radical of the ideal $\mathscr{I}$. From basic theory of polynomial algebra,

$$
\begin{aligned}
\mathrm{Zero}([S_1, U_2]) &\subseteq \mathrm{Zero}([S_1, U_1]) \cup \mathrm{Zero}([S_1, U_4]) = \mathrm{Zero}(\overline{\mathscr{T}_1}/\overline{\mathscr{S}_1}) \cup \mathrm{Zero}(\overline{\mathscr{T}_6}/\overline{\mathscr{S}_6}), \\
\mathrm{Zero}([S_1, U_3]) &\subseteq \mathrm{Zero}([S_1, U_1]) \cup \mathrm{Zero}([S_1, U_4]) = \mathrm{Zero}(\overline{\mathscr{T}_1}/\overline{\mathscr{S}_1}) \cup \mathrm{Zero}(\overline{\mathscr{T}_6}/\overline{\mathscr{S}_6}).
\end{aligned}
$$

Consequently,

$$\bigcup_{i=1}^{6} \mathrm{Zero}(\overline{\mathscr{T}_i}/\overline{\mathscr{S}_i}) = \mathbb{C}^3$$

such that the system (2.3) has complex solutions for all parameter assignments.

One may observe that, among all branches in (4.1), only the projection of Zero($\mathcal{T}_1/\mathcal{S}_1$) is of the same dimension as the parameter space $\mathbb{C}^3$. We call $[\mathcal{T}_1, \mathcal{S}_1]$ the *main branch* of the regular systems of $\mathcal{F}$.

5. Algorithm for Optimal Control

We now propose a systematic approach to drive an m-order linear system Φ from any given initial state $\bar{\boldsymbol{u}}$ to the origin in minimum time. In Remark 2.1, it is pointed out that by solving particular polynomial equations one can decide how long to keep a certain constant ($+1$ or -1) for the optimal control v. However, discrete time implementations are typically used in engineering. Suppose a sample of the state of a system Φ is taken at small time steps, say every 0.001 seconds. From the algorithm Switch presented in subsection 5.1, we know that $\mathsf{Switch}(\Phi, \bar{\boldsymbol{u}})$ should be the first recommended value for the time-optimal control for driving Φ to the origin, so let $v(t) = \mathsf{Switch}(\Phi, \bar{\boldsymbol{u}})$ for $0 \le t < 0.001$. Under the control of this $v(t)$, the system will move to a new state when $t = 0.001$. By sampling, we get the state and denote it as $\bar{\boldsymbol{u}}_1$. Similarly, on setting $v(t) = \mathsf{Switch}(\Phi, \bar{\boldsymbol{u}}_1)$ for $0.001 \le t < 0.002$, the system will arrive at $\bar{\boldsymbol{u}}_2$. In this way, the system Φ would be driven to the target through the path $\bar{\boldsymbol{u}} \to \bar{\boldsymbol{u}}_1 \to \bar{\boldsymbol{u}}_2 \to \cdots \to \mathbf{0}$, and the whole optimal control $v(t)$ could finally be obtained.

5.1. Switching algorithm

For any vector of numbers $\bar{\boldsymbol{u}} = (\bar{u}_1, \cdots, \bar{u}_m)$, we use $*|_{\bar{u}}$ to denote the result of $*$ specified at $\boldsymbol{u} = \bar{\boldsymbol{u}}$, where $*$ could be a polynomial or a polynomial set. In Algorithm 1 we formalise the steps in computing the recommended present value for the time-optimal control to move a system from its current state to the origin. (The algorithm has the system and its current state as the input.) Our method is similar to counting distinct real solutions of a semi-algebraic system [20], which serves as the main computational tools in analysing the multiplicity of competitive equilibria of semi-algebraic economies.

In Algorithm 1, $\mathsf{Simplify}(\{T_1' = 0, \cdots, T_m' = 0, t_1 \ge 0, \cdots, t_m \ge 0\})$ translates the semi-algebraic system $\{T_1' = 0, \cdots, T_m' = 0, t_1 \ge 0, \cdots, t_m \ge 0\}$ into an equivalent simpler system $\{T_1'(t_1) = 0, A_1(t_1) \ge 0, \cdots, A_m(t_1) \ge 0\}$ with a single variable t_1 — cf. Ref. [20] for further details. The operation $\mathsf{Split}(\{T_1' = 0, A_1 \ge 0, \cdots, A_m \ge 0\})$ returns a set of finitely many semi-algebraic systems with only strict inequalities such that the solution set remains the same. Indeed, if the greatest common divisor of T_1' and A_1 is 1, then $A_1 \ge 0$ can be replaced by $A_1 > 0$; otherwise, the system should be split into two — viz.

$$\begin{cases} \gcd(T_1', A_1) = 0\,, \\ A_2 \ge 0\,, \\ \quad\vdots \\ A_m \ge 0\,, \end{cases} \qquad \begin{cases} T_1'/\gcd(T_1', A_1) = 0\,, \\ A_1 > 0\,, \\ A_2 \ge 0\,, \\ \quad\vdots \\ A_m \ge 0\,. \end{cases}$$

Algorithm 1: $v(0) := \mathsf{Switch}(\Phi, \bar{\boldsymbol{u}})$

Input: Φ — an m-order linear system; $\bar{\boldsymbol{u}}$ — the initial state of Φ.
Output: $v(0)$ — the recommended present value (either $+1$ or -1) for the time-optimal control of driving Φ from $\bar{\boldsymbol{u}}$ to the origin $\mathbf{0}$.

$\mathscr{F}$:= the polynomial set corresponding to Φ with $\boldsymbol{u}$ as its parameters and $t_1, \cdots, t_m$ as its variables (see Section 2 for details);
Let $\boldsymbol{u} < t_1 < \cdots < t_m$ and decompose $\mathscr{F}$ into regular systems $[\mathscr{T}_i, \mathscr{S}_i]$, $i = 1, \cdots, l$;
$[\overline{\mathscr{T}_i}, \overline{\mathscr{S}_i}]$:= the regular system corresponding to $\overline{\mathrm{Proj}_m \mathrm{Zero}(\mathscr{T}_i/\mathscr{S}_i)}$.
Δ := the set of indices i's such that $\bar{\boldsymbol{u}} \in \mathrm{Zero}(\overline{\mathscr{T}_i}/\overline{\mathscr{S}_i})$ for $i \in \{1, \cdots, l\}$;
$\Gamma := \emptyset$;
for $j \in \Delta$ **do**
  Suppose that $\mathscr{T}_j|_{\bar{\boldsymbol{u}}} = [T_1'(t_1), \cdots, T_m'(t_1, \cdots, t_m)]$ is quasi-linear, for otherwise we apply the quasi-linearization technique in Section 3.1;
  $\mathbb{S} := \mathsf{Simplify}(\{T_1' = 0, \cdots, T_m' = 0, t_1 \geq 0, \cdots, t_m \geq 0\})$;
  $\Gamma := \Gamma \cup \mathsf{Split}(\mathbb{S})$;
end
for $\mathbb{U} \in \Gamma$ **do**
  Suppose that $\mathbb{U} = \{F(t_1) = 0, P_1(t_1) > 0, \cdots, P_s(t_1) > 0\}$;
  $S := \mathsf{Iso}(F, \prod_{k=1}^{s} P_k)$;
  C := the open intervals of the complement of S such that $P_k > 0$ for all $k = 1, \cdots, s$;
  if $\mathit{Count}(F, C) > 0$ **then**
    $v(0) := (-1)^m$;
    return;
  end
end
$v(0) := -(-1)^m$;
return;

Proceeding in the same way for the other '$\geq$' inequalities of the above semi-algebraic systems, we obtain finitely many systems of the form

$$\begin{cases} F(t_1) = 0\,, \\ P_1(t_1) > 0\,, \\ \qquad \vdots \\ P_s(t_1) > 0\,, \end{cases}$$

where $\gcd(F, P_k) = 1$ for all $k = 1, \cdots, s$. It is obvious that the solutions of all the semi-algebraic systems eventually obtained are the same as those from $\{T_1'(t_1) = 0, A_1(t_1) \geq 0, \cdots, A_m(t_1) \geq 0\}$. For any univariate polynomials F and G such that $\gcd(F, G) = 1$, the operation $\mathsf{Iso}(F, G)$ isolates the real zeros of G — e,g. using the modified Uspensky algorithm [6]. Thus $\mathsf{Iso}(F, G)$ returns a sequence of closed intervals $[f_1, g_1], \cdots, [f_m, g_m]$ such that

- f_i, g_i are all rational numbers,
- $f_1 \le g_1 < f_2 \le g_2 < \cdots < f_m \le g_m$,
- $[f_i, g_i] \cap [f_j, g_j] = \emptyset$ for $i \ne j$,
- each $[f_i, g_i]$ contains one and only one real zero of G,
- every $[f_i, g_i]$ covers no real zero of F.[‡]

Finally, for any univariate polynomial F and a set C consisting of finite open intervals, the operation $\mathsf{Count}(F, C)$ returns the number of distinct real solutions of F located in C and Sturm's theorem may be invoked.

Proof of Algorithm 1. As pointed out in Section 2, the values for the optimal control v of the system Φ depend upon whether $\mathscr{F}$ has any real solution such that $t_1 \ge 0, \cdots, t_m \ge 0$. It is notable that $\mathscr{F}$ is obtained from the inverse problem (i.e. driving Φ from $\mathbf{0}$ to $\boldsymbol{u}$), hence both the order and the sign of the recommended value sequence of v may need to be reversed. Thus if $c = +1, -1, +1$ in the inverse problem for a third-order system, then the recommended values of v in the direct problem are $-1, +1, -1$; on the other hand, if $c = +1, -1, +1, -1$ in the inverse problem for a fourth-order system, we also have $c = +1, -1, +1, -1$ in the direct problem. In Algorithm 1, we set the present value $v(0) = (-1)^m$ if $\mathscr{F}$ has non-negative solutions, and otherwise we set $v(0) = -(-1)^m$.

We now illustrate how Algorithm 1 essentially identifies whether or not $\mathscr{F}$ has non-negative solutions. Obviously,

$$\mathrm{Zero}(\mathscr{F}) = \bigcup_{i=1}^{l} \mathrm{Zero}(\mathscr{T}_i / \mathscr{S}_i),$$

so that

$$\mathrm{Zero}(\mathscr{F}|_{\bar{u}}) = \bigcup_{j \in \Delta} \mathrm{Zero}(\mathscr{T}_j|_{\bar{u}} / \mathscr{S}_j|_{\bar{u}}),$$

and hence only $\mathscr{T}_j|_{\bar{u}}$, $j \in \Delta$ need to be considered when counting solutions of $\mathscr{F}$. Consider the first **for** loop. Suppose that

$$\begin{aligned} &\mathsf{Simplify}(\{T_1' = 0, \cdots, T_m' = 0, t_1 \ge 0, \cdots, t_m \ge 0\}) \\ &= \{T_1'(t_1) = 0, A_1(t_1) \ge 0, \cdots, A_m(t_1) \ge 0\}. \end{aligned}$$

Then the problem is reduced to determining whether, for each $j \in \Delta$, there exists any real zero of the semi-algebraic system

$$\begin{cases} T_1' = 0, \\ A_1 \ge 0, \\ \quad\vdots \\ A_m \ge 0. \end{cases}$$

[‡]This can be realized because $\gcd(F, G) = 1$.

From the property of the sub-procedure Split, the problem is equivalent to whether any system in Γ has real solutions. Moreover, we have $\gcd(F, \prod_{k=1}^{s} P_k) = 1$ for any $\{F(t_1) = 0, P_1(t_1) > 0, \cdots, P_s(t_1) > 0\} \in \Gamma$. In the second **for** loop, suppose that the result of $S := \mathsf{Iso}(F, \prod_{k=1}^{s} P_k)$ is of the form $[f_1, g_1], \cdots, [f_m, g_m]$. Then the complement of S is

$$(-\infty, f_1)\,, \quad (g_1, f_2)\,, \quad \cdots, \quad (g_m, +\infty)\,.$$

It is obvious that the signs of $P_1, \cdots, P_s$ must be fixed in each of the above open intervals, and can be identified by verifying at a sample point. Consequently, C could readily be obtained and $\mathsf{Count}(F, C)$ counts distinct real solutions of F located in C. If $\mathsf{Count}(F, C) > 0$ for some $\mathbb{U} \in \Gamma$, then $\mathbb{U}$ has real solutions, which proves that $\mathscr{F}$ has at least one real solution such that $t_1 \geq 0, \cdots, t_m \geq 0$. □

5.2. Illustrative case

Consider the third-order system (2.1) in Section 2 and the initial state $\bar{\boldsymbol{u}} = (a, b, c) = (1, 1, 1)$. Since $S_1|_{(1,1,1)} = -1 \neq 0$ and $S_2|_{(1,1,1)} = -17 \neq 0$, we have $(1, 1, 1) \in \mathrm{Zero}(\overline{\mathscr{T}_1}/\overline{\mathscr{S}_1})$. Moreover, it can be verified that $(1, 1, 1) \notin \mathrm{Zero}(\overline{\mathscr{T}_i}/\overline{\mathscr{S}_i})$ for any $i = 2, \cdots, 6$, so from (4.1) we have that $\mathrm{Zero}(\mathscr{F}|_{(1,1,1)}) = \mathrm{Zero}(\mathscr{T}_1|_{(1,1,1)})$. Thus $\Delta = \{1\}$, and it remains to determine whether $\mathscr{T}_1|_{(1,1,1)}$ with non-negative components has at least one real zero. We have $\mathscr{T}_1|_{(1,1,1)} = [T_1', T_2', T_3']$ with

$$\begin{aligned}
&T_1' = 36\,t_1^4 + 48\,t_1^3 - 18\,t_1^2 - 17\,,\\
&T_2' = I t_2 + J, \quad I = -6\,t_1^2 - 3, \quad J = 3\,t_1 + 2\,,\\
&T_3' = -t_3 + t_2 - t_1 + 1\,.
\end{aligned}$$

Other than the first, the polynomials in $\mathscr{T}_1|_{(1,1,1)}$ are linear with respect to their leading variables, such that $\mathscr{T}_1|_{(1,1,1)}$ is quasi-linear.

Let us demonstrate how the sub-procedure $\mathsf{Simplify}(\{T_1' = 0, T_2' = 0, T_3' = 0, t_1 \geq 0, t_2 \geq 0, t_3 \geq 0\})$ works. On solving $T_3' = 0$ for t_3, $T_2' = 0$ for t_2 and then substituting the solutions $t_3 = t_2 - t_1 + 1$, $t_2 = -J/I$ successively into $t_3 \geq 0$ and $t_2 \geq 0$, we obtain $-J/I - t_1 + 1 \geq 0$ and $-J/I \geq 0$, respectively. The problem is thus reduced to determining whether there are real solutions of the following system with only one variable t_1:

$$\begin{cases}
T_1' = 0\,,\\
t_1 \geq 0\,,\\
A/A' \geq 0\,,\\
B/B' \geq 0\,,
\end{cases}$$

where $A = -J$, $A' = I$, $B = -J - t_1 I + I$ and $B' = I$. Polynomials rather than rational functions are preferred in the computation. Since $I(\bar{t}_1) \neq 0$ for any zero $\bar{t}_1$ of T_1' from the definition of a regular system, one could replace the last two inequalities by $A^* = AA' =$

$-IJ \geq 0$ and $B^* = BB' = (-J - t_1 I + I)I \geq 0$, respectively. We arrive at

$$\begin{cases} T_1' = 0\,, \\ t_1 \geq 0\,, \\ A^* \geq 0\,, \\ B^* \geq 0\,. \end{cases}$$

Since $\gcd(T_1', t_1) = 1$, $\gcd(T_1', A^*) = 1$ and $\gcd(T_1', B^*) = 1$, we have $\Gamma = \{\mathbb{U}\}$ where

$$\mathbb{U} = \begin{cases} T_1' = 0\,, \\ t_1 > 0\,, \\ A^* > 0\,, \\ B^* > 0\,. \end{cases}$$

In order to determine whether system $\mathbb{U}$ has real solutions, let us write $S := \mathsf{Iso}(T_1', t_1 A^* B^*)$ to obtain the sorted sequence of intervals

$$[-1, -1/2], \quad [0,0], \quad [3/4, 1]\,.$$

Consequently, the real zeros of T_1' must lie in

$$(-\infty, -1)\,, \quad (-1/2, 0)\,, \quad (0, 3/4)\,, \quad (1, +\infty)\,.$$

Furthermore, in each of these open intervals the signs of t_1, A^* and B^* are invariant, and can be identified by testing at a sample point in each interval. For example, to determine the sign of A^* on $(-\infty, -1)$, we have that $A^*(-2) = -108 < 0$, so A^* is negative at every point in $(-\infty, -1)$. Proceeding in this way for other intervals, we conclude that the inequality constraints $t_1 > 0$, $A^* > 0$ and $B^* > 0$ are satisfied only on $C = (0, 3/4)$. Finally, by computing the Sturm sequence we obtain that $\mathsf{Count}(T_1', C) = 1 > 0$.

In conclusion, for $(a, b, c) = (1, 1, 1)$ the system (2.3) has exactly one non-negative solution, so we should set $v(0) = -1$ as the recommended present value for the time-optimal control to drive system (2.1) from $(1, 1, 1)$ to the origin $(0, 0, 0)$.

6. Moving the Computational Burden Off-line

The switching algorithm presented in Section 5 involves the computation of real root isolation and a Sturm sequence, which may be intractable for large systems and near invalid for real-time control. In this section, we modify the switching algorithm to make it available for real-time applications. The key idea is to divide the computation into two phases — the off-line and the on-line. The computational burden is moved off-line as it is only necessary to verify that particular inequalities are satisfied in the on-line stage, and the computation is extremely fast.

6.1. The off-line phase

The example given in Section 2 serves to demonstrate how to compute control strategies in the off-line phase, and we then formalise the steps in an algorithm.

Let us reconsider the main branch $[\mathscr{T}_1, \mathscr{S}_1] = [[T_1, T_2, T_3], \{S_1, S_2\}]$ for regular systems, where S_1, S_2 are polynomials in a, b, c — cf. (4.2). For any $\bar{a}, \bar{b}, \bar{c} \in \mathbb{R}$ such that $S_1|_{(\bar{a},\bar{b},\bar{c})} \neq 0$ and $S_2|_{(\bar{a},\bar{b},\bar{c})} \neq 0$, from (4.3) we have that $(\bar{a}, \bar{b}, \bar{c}) \notin \mathrm{Zero}(\overline{\mathscr{T}_i}/\overline{\mathscr{S}_i})$ for any $i = 2, \cdots, 6$. Consequently, the zeros of $\mathscr{F}|_{(\bar{a},\bar{b},\bar{c})}$ should be same as those for $\mathscr{T}_1|_{(\bar{a},\bar{b},\bar{c})}$. Since the main branch is already quasi-linear, one can easily solve $T_3 = 0$ for t_3 and $T_2 = 0$ for t_2. Substituting the solutions $t_3 = t_2 - t_1 - a$, $t_2 = -J_2/I_2$ successively into $t_3 \geq 0$ and $t_2 \geq 0$, we obtain $-J_2/I_2 - t_1 - a \geq 0$ and $-J_2/I_2 \geq 0$, respectively. Thus provided that $S_1 S_2 \neq 0$, the problem is reduced to determining the condition on a, b, c such that there exist real zeros of the semi-algebraic system

$$\begin{cases} T_1 = 0\,, \\ t_1 \geq 0\,, \\ C \geq 0\,, \\ D \geq 0\,, \end{cases} \tag{6.1}$$

where $C = -I_2 J_2$, $D = (-J_2 - t_1 I_2 + I_2) I_2$. Moreover, let us suppose that

$$\mathrm{Res}(T_1, t_1) \cdot \mathrm{Res}(T_1, C) \cdot \mathrm{Res}(T_1, D) \neq 0\,.$$

From Proposition 3.2, we know that T_1 has no common zero with t_1, C and D, so system (6.1) can be transformed further to

$$\begin{cases} T_1 = 0\,, \\ t_1 > 0\,, \\ C > 0\,, \\ D > 0\,. \end{cases} \tag{6.2}$$

We now construct the border polynomial of (6.2), and its square free part $B = B_1 B_2 B_3 B_4$, where

$$\begin{aligned}
B_1 &= S_1 = a^2 - 2\,b\,, \\
B_2 &= S_2 = a^6 - 6\,a^4 b - 48\,a^3 c + 36\,a^2 b^2 + 144\,abc - 72\,b^3 - 72\,c^2\,, \\
B_3 &= a^6 + 6\,a^4 b - 48\,a^3 c + 36\,a^2 b^2 - 144\,abc + 72\,b^3 - 72\,c^2\,, \\
B_4 &= a^6 - 6\,a^4 b - 144\,a^3 c + 84\,a^2 b^2 + 432\,abc - 200\,b^3 - 216\,c^2\,,
\end{aligned}$$

such that the zeros of B divide the parameter space $\mathbb{R}^3$ into separated regions. Fig. 2 shows the graphs of $B_1 = 0$, $B_2 = 0$, $B_3 = 0$ and $B_4 = 0$ (shown in blue, green, white and red, respectively.) From Theorem 3.2, the number of real solutions of (6.2) is invariant in a fixed region. Thus one may choose a sample point from each region, which can be done systematically, using for example the cylindrical algebraic decomposition method [5]. The problem then is to determine whether there is any real solution of the result of (6.2)

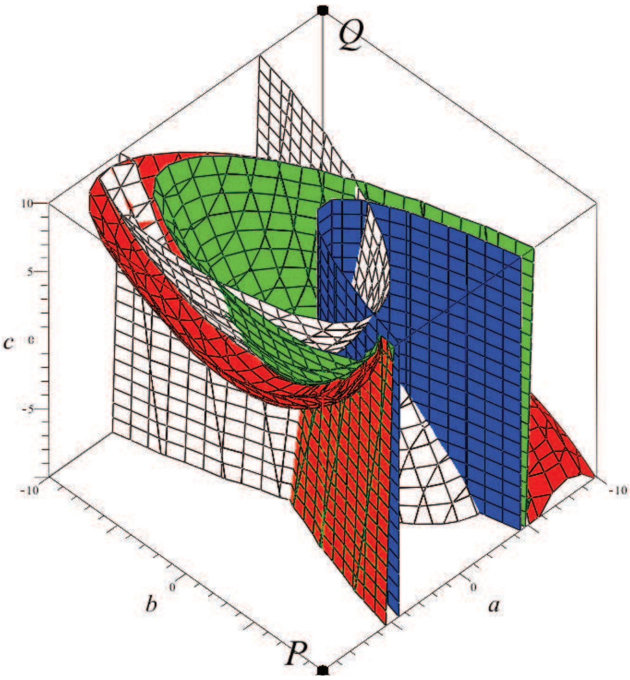

Figure 2: Partitions of the parameter space of (6.2).

specified at these sample points. For example, at $P = (10, 10, -10)$ shown in Fig. 2, the system (6.2) becomes

$$\begin{cases} -2880\, t_1^4 + 32160\, t_1^3 - 115200\, t_1^2 + 1016800 = 0\,, \\ t_1 > 0\,, \\ (-120\, t_1 + 670)(3\, t_1^2 - 120) > 0\,, \\ (-3\, t_1^3 + 30\, t_1^2 - 530)(3\, t_1^2 - 120) > 0\,. \end{cases}$$

On applying the method proposed in Section 5, we know that the above system has at least one real solution, so $v(0)$ should be -1 if $(\bar{a}, \bar{b}, \bar{c})$ in the same region as P. Proceeding further in the same way, the recommended present value $v(0)$ for the optimal control therefore can be obtained for other regions.

To succinctly describe the way to set $v(0)$, we give simple and formal representations of these regions, beginning with utilising the sign of B_i, $i = 1, \cdots, 4$. For example, as

$$B_1|_P = 80 > 0, \quad B_2|_P = 1016800 > 0, \quad B_3|_P = 2648800 > 0, \quad B_4|_P = 2026400 > 0,$$

we expect that the region where P is located might be described by $B_i > 0$, $i = 1, \cdots, 4$. Unfortunately, this is not the case. Consider the point $Q = (-10, -10, 10)$. We also have $B_i|_Q > 0$ for all $i = 1, \cdots, 4$, so the Boolean formula $\bigwedge_{i=1}^{4} B_i > 0$ corresponds to at least two different regions. However, we may introduce an additional polynomial $A_1 = a^3 - 3\, a\, b + 3\, c$ that satisfies $A_1|_P = 670 \geq 0$ and $A_1|_Q = -1270 \leq 0$, so the two regions where points P

and Q lie can be described by

$$\bigwedge_{i=1}^{4} B_i > 0 \wedge A_1 > 0 \quad \text{and} \quad \bigwedge_{i=1}^{4} B_i > 0 \wedge A_1 < 0 \,,$$

respectively. This illustrates that other polynomials could help in the characterisation of different regions. Generally, it would be fairly hard to find these polynomials by hand, but Yang *et al.* [40] have pointed out that they are contained in the so-called *generalised discriminant list* and can be selected by repeated trials. For our example, 4 additional polynomials may be needed — including A_1 above,

$$\begin{aligned} A_2 &= 3\,a^3 - 7\,a\,b + 3\,c \,, \\ A_3 &= 117\,a^8 - 400\,a^6 b + 120\,a^5 c - 496\,a^4 b^2 - 2736\,a^3 bc + 4176\,a^2 b^3 \\ &\quad - 72\,a^2 c^2 + 7296\,a b^2 c - 5808\,b^4 - 3312\,bc^2 \,, \end{aligned}$$

and A_4, a complex polynomial of degree 15 with 27 terms. (We do not give A_4 here, due to limitation in space.)

Given the above preparation, we now describe how to set $v(0)$ for any given $(\bar{a}, \bar{b}, \bar{c})$ such that

$$N = S_1 \cdot S_2 \cdot \mathrm{Res}(T_1, t_1) \cdot \mathrm{Res}(T_1, C) \cdot \mathrm{Res}(T_1, D) \cdot \prod_{i=1}^{4} B_i \neq 0 \,.$$

The optimal control $v(0)$ should be set as -1 if and only if system (2.3) has at least one non-negative solution, or if and only if one of the following Boolean formulae holds:

- $B_1 < 0 \bigwedge B_2 < 0 \bigwedge B_3 < 0 \bigwedge B_4 < 0 \bigwedge A_2 < 0 \bigwedge A_3 < 0 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge A_1 < 0 \bigwedge A_2 < 0 \bigwedge A_3 < 0 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge A_1 < 0 \bigwedge 0 < A_2 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge 0 < B_3 \bigwedge 0 < B_4 \bigwedge A_1 < 0 \bigwedge 0 < A_2 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_4 < 0 \bigwedge A_1 < 0 \bigwedge 0 < A_2 \bigwedge 0 < A_3 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge 0 < B_4 \bigwedge A_1 < 0 \bigwedge A_3 < 0 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge B_2 < 0 \bigwedge B_3 < 0 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge A_2 < 0 \bigwedge A_3 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge 0 < B_4 \bigwedge 0 < A_1 \bigwedge A_2 < 0 \bigwedge 0 < A_3 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge 0 < A_1 \bigwedge A_2 < 0 \bigwedge A_3 < 0 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge B_3 < 0 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge A_2 < 0 \bigwedge 0 < A_3 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge B_2 < 0 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge 0 < A_3 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge A_3 < 0 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge 0 < A_3 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge 0 < B_4 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge A_3 < 0 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge 0 < B_3 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge A_3 < 0 \bigwedge 0 < A_4$,

- $0 < B_1 \bigwedge 0 < B_2 \bigwedge 0 < B_3 \bigwedge 0 < B_4 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge A_3 < 0 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge 0 < B_3 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge 0 < A_3 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge B_3 < 0 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge A_3 < 0 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge 0 < B_3 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge A_3 < 0 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_3 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge 0 < A_3 \bigwedge 0 < A_4$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_3 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge B_3 < 0 \bigwedge 0 < B_4 \bigwedge 0 < A_1 \bigwedge A_3 < 0 \bigwedge A_4 < 0$,
- $0 < B_1 \bigwedge 0 < B_2 \bigwedge 0 < B_3 \bigwedge 0 < B_4 \bigwedge 0 < A_2 \bigwedge 0 < A_3 \bigwedge 0 < A_4$,
- $B_2 < 0 \bigwedge B_3 < 0 \bigwedge B_4 < 0 \bigwedge 0 < A_1 \bigwedge 0 < A_2 \bigwedge A_3 < 0 \bigwedge A_4 < 0$.

The above steps are formalised in Algorithm 2. The termination and correctness of the algorithm are obvious.

Remark 6.1. From our experiments, the runtime can be the bottleneck for Algorithm 2 rather than memory requirements. It should be mentioned that Yang *et al.* [40] proposed a more direct method that may also be used to compute the necessary and sufficient conditions above. Their method avoids the quasi-linearisation process and may be more efficient when the polynomials involved are of high degree.

Both Yang's approach and ours are quite time-consuming for large problems, as the complexities of existing algorithms for cylindrical algebraic decomposition prove to be double-exponential. For the third-order system given in Section 2, Algorithm 2 takes 139 seconds to terminate in Maple 17 running on AMD A8-6500 CPU 3.50 GHz with 20G RAM under Windows 7 OS. Moreover, we found that systems with higher order could not be resolved within 5 hours. However, efficiency is less important as the computation of this stage is performed off-line, and powerful parallel computers may be used to accelerate the process.

Furthermore, Safey El Din [27, 28] and others have proposed a new approach to find sample points of semi-algebraic sets, based on the computation of critical points. The complexity of their method is $\mathcal{O}(d^7 D^{4n})$, where d and D are the parameter number and the degree of the involved system, respectively. Thus it is reasonable to hope for a significant performance boost if the cylindrical algebraic decomposition procedure in our methods is replaced with the critical point computation. This is an interesting issue for further investigation.

Finally, we note that points in the parameter space $\mathbb{R}^3$ are covered in the sense of Lebesgue measure, except for those points that annihilate N. For these exceptional points, we may add the equation $N = 0$ to (6.1), with a and b might viewed as parameters while c, t_1, t_2 and t_3 are the variables. Repeating this process, we finally cover all points in the parameter space and so obtain the complete necessary and sufficient conditions on the parameters a, b and c such that system (2.3) has non-negative solutions. (Since somewhat tedious, we have not listed the complete necessary and sufficient conditions.)

Algorithm 2: $\Omega, N := \mathsf{BooleanF}(\Phi)$

Input: Φ — an m-order linear system.
Output: Ω — the necessary and sufficient conditions that the present value $v(0)$ should be $(-1)^m$ for the time-optimal control of driving Φ from $\boldsymbol{u}$ to the origin $\mathbf{0}$, provided that $N \neq 0$.

$\mathscr{F}$:= the polynomial set corresponding to Φ with $\boldsymbol{u}$ as its parameters and $t_1, \cdots, t_m$ as its variables (see Section 2 for details);
Let $\boldsymbol{u} < t_1 < \cdots < t_m$ and decompose $\mathscr{F}$ into regular systems $[\mathscr{T}_i, \mathscr{S}_i]$, $i = 1, \cdots, l$;
Δ := the set of indices i's such that $[\mathscr{T}_i, \mathscr{S}_i]$ is the main branch;
$\Gamma := \emptyset$; $\mathscr{B} := \emptyset$; $N = 1$;
for $j \in \Delta$ **do**
  Suppose that $\mathscr{T}_j = [T_1(t_1), \cdots, T_m(t_1, \cdots, t_m)]$ is quasi-linear, for otherwise we apply the quasi-linearization technique in Section 3.1;
  $\mathbb{S} := \mathsf{Simplify}(\{T_1 = 0, \cdots, T_m = 0, t_1 \geq 0, \cdots, t_m \geq 0\})$;
  $\Gamma := \Gamma \cup \{\mathbb{S}\}$;
  $\mathscr{P}$:= the set of all the factors of the border polynomial of $\mathbb{S}$;
  $\mathscr{B} := \mathscr{B} \cup \mathscr{P}$;
  $N := N \cdot \prod_{S \in \mathscr{S}_i} S$;
end
Θ := the set of sample points at all the regions of parameter space divided by zeros of polynomials in $\mathscr{B}$.
Θ^* := the subset of Θ, where at least one $\mathbb{S} \in \Gamma$ has non-negative solutions.
Suppose that the signs of all polynomials in $\mathscr{B}$ can exactly characterise the regions corresponding to points in Θ^*, for otherwise we add to $\mathscr{B}$ certain elements in the generalised discriminant list of $\mathbb{S} \in \Gamma$.
Suppose that $\mathscr{B} = \{B_1, \cdots, B_k\}$;
$\Omega := \emptyset$;
for $\bar{\boldsymbol{u}} \in \Theta^*$ **do**
  $\omega := B_1 \lessgtr 0 \bigwedge \cdots \bigwedge B_k \lessgtr 0$, where $\lessgtr$ could be either $>$ or $<$ depending on $B_i(\bar{\boldsymbol{u}}) > 0$ or $B_i(\bar{\boldsymbol{u}}) < 0$;
  $\Omega := \Omega \cup \{\omega\}$;
end
$N := N \cdot \prod_{B \in \mathscr{B}} B$;
return;

6.2. The on-line phase

In the on-line stage, we sample the state $\bar{\boldsymbol{u}}$ of the considered m-order system Φ and verify whether or not the complete necessary and sufficient conditions obtained by the algorithm BooleanF are satisfied when $\boldsymbol{u} = \bar{\boldsymbol{u}}$ — thus

- if the answer is positive, then the present value $v(0)$ is assumed to be $(-1)^m$;

- otherwise, set $v(0) = -(-1)^m$.

(The system will then be driven to the origin in minimum time.) The computation mainly identifies the signs of certain polynomials, so it can be completed for real-time control.

7. Conclusion

Optimal control is widely used in modern system science. In the context of switching surfaces in optimal control, many problems can be reduced to solving certain semi-algebraic systems. In this article, new methods for time-optimal control were presented and illustrated by a simple example. Complex and real versions for optimal control were considered. Based on triangular decomposition and relevant symbolic computation, our methods are more general and more systematic than those given in Ref. [30] — and since the computation is exact, they are quite different from existing numerical approaches. Their high complexity can be addressed by moving the computational burden off-line, such that the modified version is feasible for real-time control. The future development of faster computer algebra systems and more efficient algorithms for basic operations (including triangular decomposition and cylindrical algebraic decomposition) will also significantly improve the performance of our methods.

Acknowledgments

The authors wish to thank the anonymous referees for their helpful comments. This work has been supported by the National Natural Science Foundation of China (Tianyuan Special Foundation Nos. 11326210 and 11326211), the open fund of Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis (No. HCIC201302) and the Fundamental Research Funds for the Central Universities (No.YWF-14-RSC-103).

References

[1] M. Äthans and P.L. Falb, *Optimal Control: An Introduction to the Theory and Its Applications*, McGraw-Hill, New York (1966).

[2] F. Chai, X.-S. Gao and C. Yuan, *A characteristic set method for solving Boolean equations and applications in cryptanalysis of stream ciphers*, J. Systems Science and Complexity **21**, 191–208 (2008).

[3] C. Chen and M. Moreno Maza, *Algorithms for computing triangular decompositions of polynomial systems*, Proc. 36th International Symposium on Symbolic and Algebraic Computation (A. Leykin, ed.), pp. 83–90, ACM, New York (2011).

[4] S.-C. Chou and X.-S. Gao, *Ritt-Wu's decomposition algorithm and geometry theorem proving*, Proc. 10th International Conference on Automated Deduction (M.E. Stickel, ed.), pp. 207–220, Springer, Berlin (1990).

[5] G.E. Collins and H. Hong, *Partial cylindrical algebraic decomposition for quantifier elimination*, J. Symbolic Comp. **12**, 299–328 (1991).

[6] G.E Collins and R. Loos, *Real zeros of polynomial*, in *Computer Algebra: Symbolic and Algebraic Computation* (B. Buchberger, G.E. Collins and R. Loos, eds.), pp. 83–94, Springer, New York (1983).
[7] P. Deuflhard, *Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive Algorithms*, Springer, Berlin (2011).
[8] J. Fiene and G. Niemeyer, *Toward switching motor control*, IEEE/ASME Trans. Mechatronics **11**, 27–34 (2006).
[9] G. Gallo and B. Mishra, *Efficient algorithms and bounds for Wu-Ritt characteristic sets*, in *Effective Methods in Algebraic Geometry* (T. Mora and C. Traverso, eds.), pp. 119–142. Birkhauser, Berlin (1991).
[10] G. Gallo and B. Mishra, *Wu-Ritt characteristic sets and their complexity*, in *Discrete and Computational Geometry: Papers from the DIMACS Special Year* (J.E. Goodman, R.D. Pollack and W. Steiger, eds), pp. 111–136, American Mathematical Society, Providence(1991).
[11] X.-S. Gao and S.-C. Chou. On the dimension of an arbitrary ascending chain. *Chinese Science Bulletin*, 38(5):396–399, 1993.
[12] X.-S. Gao and Z. Huang, *Characteristic set algorithms for equation solving in finite fields*, J. Symbolic Comp. **47**, 655–679 (2012).
[13] T. Ge and J.S. Chang, *Bang-Bang control class-d amplifiers: power-supply noise*, IEEE Trans. Circuits and Systems II: Express Briefs **55**, 723–727 (2008).
[14] M. Jin, X. Li and D. Wang, *A new algorithmic scheme for computing characteristic sets,* J. Symbolic Comp. **51**, 433–449 (2012).
[15] M. Kalkbrener, *A generalized Euclidean algorithm for computing triangular representations of algebraic varieties*, J. Symbolic Comp. **15**, 143–167 (1993).
[16] B.S. Kumar and A. Ng, *A Bang-Bang control approach to manoeuver spacecraft in a formation with differential drag*, Proc. AIAA Guidance, Navigation and Control Conference and Exhibit, pp. 18–21, American Institute of Aeronautics and Astronautics, Reston (2008).
[17] D. Lazard, *A new method for solving algebraic systems of positive dimension*, Discrete Applied Math. **33** 147–160 (1991).
[18] E.B. Lee and L. Markus, *Foundations of Optimal Control Theory*, Krieger, Malabar (1967).
[19] X. Li, C. Mou and D. Wang, *Decomposing polynomial sets into simple sets over finite fields: the zero-dimensional case*, Computers and Mathematics with Applications **60**, 2983–2997 (2010).
[20] X. Li and D. Wang, *Computing equilibria of semi-algebraic economies using triangular decomposition and real solution classification*, ArXiv e-prints arXiv:1308.5029 (2013).
[21] Y. Li, *Some properties of triangular sets and improvement upon algorithm CharSer*, in *Artificial Intelligence and Symbolic Computation* (J. Calmet, T. Ida, and D. Wang, eds), pp. 82–93, Springer, New York (2006).
[22] B. Mishra, *Algorithmic Algebra*, Springer, New York (1993).
[23] M. Moreno Maza, *On triangular decompositions of algebraic varieties*, Technical Report 4/99, NAG, UK. (Presented at the MEGA-2000 Conference, Bath.)
[24] L.Y. Pao and G.F. Franklin, *Proximate time-optimal control of third-order servomechanisms*, IEEE Trans. Automatic Control **38**, 560–580 (1993).
[25] J.F. Ritt, *Differential Algebra*, American Mathematical Society, Providence (1950).
[26] M. Rubagotti, M.L Della Vedova, and A. Ferrara, *Time-optimal sliding-mode control of a mobile robot in a dynamic environment*, IET Control Theory and Applications **5**, 1916–1924 (2011).
[27] M.S. El Din, *Testing sign conditions on a multivariate polynomial and applications*, Mathematics in Comp. Sc. **1**, 177–207 (2007).
[28] M.S. El Din and E. Schost, *Polar varieties and computation of one point in each connected component of a smooth real algebraic set*, Proc. 2003 International Symposium on Symbolic and

Algebraic Computation, pp. 224–231. ACM, New York (2003).

[29] A.J. Sommese and C.W. Wampler, *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science*, World Scientific, Singapore (2005).

[30] U. Walther, T.T. Georgiou and A. Tannenbaum, *On the computation of switching surfaces in optimal control: a Gröbner basis approach*, IEEE Trans. Automatic Control **46**, 534–540 (2001).

[31] D. Wang, *An elimination method for polynomial systems*, J. Symbolic Comp. **16**, 83–114 (1993).

[32] D. Wang, *Decomposing polynomial systems into simple systems*, J. Symbolic Comp. **25**, 295–314 (1998).

[33] D. Wang, *Computing triangular systems and regular systems*, J. Symbolic Comp. **30**, 221–236 (2000).

[34] D. Wang, *A generalized algorithm for computing characteristic sets*, in Computer Mathematics: Proc. Fifth Asian Symposium (K. Shirayanagi and K. Yokoyama, eds.), pp. 165–174, Springer, Berlin (2001).

[35] D. Wang, *A strategy for speeding-up the computation of characteristic sets.*, Proc. 17th International Symposium on Mathematical Foundations of Computer Science (I.M. Havel and V. Koubek, eds.), pp. 504–510, Springer, Berlin (1992).

[36] D. Wang, *Elimination Methods*, Springer, New York (2001).

[37] W.-T. Wu, *Basic principles of mechanical theorem proving in elementary geometries*, J. Automated Reasoning **2**, 221–252 (1986).

[38] W.-T. Wu, *On zeros of algebraic equations: an application of Ritt principle*, Kexue Tongbao **31**, 1–5 (1986).

[39] L. Yang and J.-Z. Zhang, *Searching dependency between algebraic equations: an algorithm applied to automated reasoning*, in *Artificial Intelligence in Mathematics* (J. Johnson, S. McKee and A. Vella, eds.), pp. 147–156, Oxford University Press, Oxford (1994).

[40] L. Yang, X. Hou and B. Xia, *A complete algorithm for automated discovering of a class of inequality-type theorems*, Science in China Series F **44**, 33–49 (2001).

[41] Y. Zhao and P. Tsiotras, *Time-optimal path following for fixed-wing aircraft*, J. Guidance, Control and Dynamics **36**, 83–95 (2012).