A Robust Control System Design by a Special Quantifier Elimination

Hirokazu Anai * and Shinji Hara **

* Department of Mathematics and Computer Science
University of Passau, Germany
e-mail : anai@fmi.uni-passau.de

** Department of Computational Intelligence and System Science
Tokyo Institute of Technology, Japan
e-mail : hara@cs.dis.titech.ac.jp

MIP-0004
March 01, 2000

Abstract

For multi-objective design and robust control synthesis problems, the methods based on Quantifier Elimination (QE) have been proposed. QE based methods are really suitable for such problems but, in general, they have a drawback on computational complexity. In this paper, we will propose an efficient symbolic method for a parameter space approach based on Sign Definite Condition (SDC) by a special quantifier elimination using *Sturm-Habicht sequence.* Then we examine its feasibility, in particular, for multi-objective control using low degree fixed-structure controller by showing the several experimental results.

1 Introduction

Multi-objective design and robust control synthesis are of great practical interest and main concerns in the control system design. A parameter space design method is known to be one of the useful tools to deal with such problems. In this paper we will propose the symbolic method for a parameter space approach by a special quantifier elimination (QE[§]) method. Then we will investigate the feasibility of our method, in particular, for multi-objective control using *low degree* fixed-structure controller.

Recently, for such problems, the methods based on QE were proposed by several researchers [8][15][18][2]. For example, in [8] it is shown that how certain robust multi-objective design problems can be reduced to QE problems and actually solved by using QEPCAD[††]. In [15] it is shown that, in feedback design of linear time-invariant systems, robustness and several performance specifications (H_∞ norm constraint, gain and phase margins) on the close-loop system can also be solved as QE problems by using QEPCAD. QE based approach is really effective for such problems. However, unfortunately, the size of the problems which can be solved by QE based approach is very limited. Because the computational complexity of the general QE algorithm based on CAD algorithm is doubly exponential in the number of variables.

Many important design specifications such as H_∞ norm constraint, stability margins *etc*, which are frequently used as indices of the robustness, to sign definite condition. See [13][16][17]. The sign definite condition is a very simple (first-order) formula and suited for a QE procedure in view of computational efficiency. Moreover, In [13] it is also proposed that SDC is checked by using Routh-Hurwitz like criterion proposed by D.Šiljak for positive realness [21]. A parameter space approach based on SDC using D.Šiljak's criterion is essentially equivalent to performing QE for the particular inputs

$$\forall x > 0, \ f(x) > 0 \tag{1}$$

where $f(x)$ is a polynomial with real coefficients. So this method is regarded as a special QE algorithm for the particular input first-order formula (1) and more efficient than the general QE algorithm based on CAD algorithm. However in the method using D.Šiljak's criterion, there remains some issues related to singular cases (see [10]) and specialization of parameters.

Hence, in this paper, we propose a robust control system design and synthesis method based on a special QE method for SDC using Sturm-Habicht

§As for quantifier elimination, see Appendix A.

††QEPCAD is a package for QE based on the Cylindrical Algebraic Decomposition (CAD) algorithm presented by G.E.Collins[5].

sequence and examine its computational complexity to show the applicability not to toy problems but to practical control problems. Our approach consists in the combination of SDC and the special QE using Sturm-Habicht sequence and aim at practically applicability. A combinatorial algorithm to solve the particular QE problem $\forall x,\ f(x) > 0$ based on Sturm-Habicht sequence is proposed by L.González-Vega *et.al.*[11]. We utilize their algorithm with modifying for our situation (1). The method proposed here is more efficient than the method using Routh-Hurwitz like criterion by D.Šiljak and has a good specialization property.

Moreover, our approach has several advantages compared with a matrix inequality (LMI) approach based on numerical optimization, if we restrict our attention to low degree fixed-structure control synthesis problems.

- To extract the feasible region of design parameters for some specification, many recursive computations for distinct parameter values are not needed and obtained result does not have any parameter "holes".

- When we consider fixed-order or fixed-structure controller synthesis problems, many cases are reduced to BMI problems. Our approach is also applicable to such nonlinear cases without any changes.

- In multi-objective design, our approach yields non-conservative results and increase the computational complexity linearly in the number of the objectives.

The organization of the rest of the paper is as follows: The idea of robust control synthesis based on SDC is explained in Section 2. Section 3 is devoted to our proposed QE algorithm using Strum-Habicht sequence for checking the SDCs. We examine the computational complexity by numerical experiments including PID controller synthesis and summarize the tractability and the applicability to practical control problems in Section 4. Section 5 provides a practical design example, where we show the effectiveness of the proposed method for integrated design as well as fixed-structure robust controller synthesis. Section 6 addresses the concluding remarks.

2 Control synthesis based on SDC

2.1 Sign definite condition (SDC)

In this paper we use **R** and **Q** for the fields of real numbers and rational numbers, respectively.

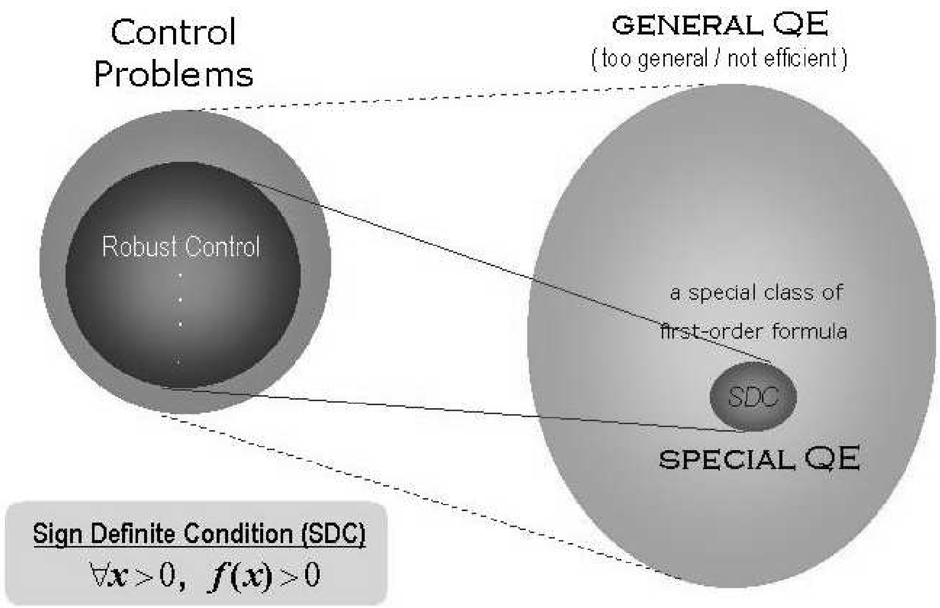

Figure 1: Relevance of our approach

Definition 1 *Let $f(x)$ be a polynomial in x over $\mathbf{R}$, i.e., $f(x) \in \mathbf{R}[x]$. $f(x)$ is sign definite in the interval $x \in [a, b]$ such that $a < b$ $(\in \mathbf{R})$, denoted by $f(x) \in \mathbf{N}_0[a, b]$, if $f(x)$ preserves its sign in $[a, b]$, or does not cross zero in $[a, b]$. If $f(x)$ is sign definite in $[a, b]$ and $f(x) > 0$ then $f(x)$ is positive sign definite in $[a, b]$.*

Note that in actual computation we consider the polynomial $f(x)$ over $\mathbf{Q}$. This restriction is needed since we utilize a computer algebra system. In this paper we, in particular, consider the parametric case, *i.e.,* the coefficients of $f(x)$ contain some real parameters, say, $p_1, \cdots, p_s$. Strictly speaking, this means $f(x)$ is a polynomial over the rational function fields $\mathbf{R}(p_1, \cdots, p_s)$, *i.e.,* $f(x) \in \mathbf{R}(p_1, \cdots, p_s)[x]$.

The sign definition condition have emerged as the important problem in a parameter space approach for robust control design. The specifications such as

- H_∞ norm constraint,
- frequency restricted norm constraint,
- gain and phase margin constraint, and
- D-stability

that are frequently used as indices of robustness of feedback control systems, are reduced to sign definite conditions (see [13][16][17][21]). This fact makes it appealing to look into the SDC. Hereafter, without loss of generality, it is enough to consider the problem

$$f(x) \in \mathbf{N}_0[\ 0, +\infty\ ], \tag{2}$$

because the condition $f(x) \in \mathbf{N}_0[a, b]$ can be translated to the condition $f(z) \in \mathbf{N}_0[0, +\infty]$ by a bilinear transformation $z = -\frac{x-a}{x-b}$.

2.2 H_∞ norm constraint

Among the specifications that can be reduced to SDC, here we show how H_∞ norm constraint is transformed to SDC (see [13]). First we have the following lemma:

Lemma 2 [3] A stable transfer function $G(s) = C(sI - A)^{-1}B + D$ with degree n satisfies

$$||G(s)||_\infty < \gamma$$

if and only if the following conditions hold;
 i) $D^T D < \gamma^2 I$,
 ii) Hamilton matrix

$$H = \begin{bmatrix} A & 0 \\ C^T C & -A^T \end{bmatrix} - \begin{bmatrix} B \\ C^T D \end{bmatrix} \times (\gamma^2 I - D^T D)^{-1} [-D^T C \ B^T]$$

 has no eigenvalues on imaginary axis.

Since the characteristic polynomials h of Hamilton matrices are even polynomials, *i.e.*,

$$h(s^2) = |sI - H| = \sum_{i=0}^{n} h_i x^{2i},$$

this condition is equivalent that h has no root in pure imaginary number and on the origin. Let $s^2 = x$ then the condition is that $h(x) = \sum_{i=0}^{n} h_i x^i$ has no negative real roots and no root on the origin. Finally we have the sign definite condition:

$$f(x) = (-1)^n\ h(x) > 0, \quad \forall x \geq 0.$$

Moreover, frequency restricted norm, a generalization of H_∞ norm, defined by

$$||G||_{[\omega_1, \omega_2]} = \sup_{\omega_1 \leq \omega \leq \omega_2} \bar{\sigma}(G(j\omega))$$

can be also reduced to SDC:

$$f(x) \in N_0[-\omega_2^2, -\omega_1^2]$$

where $\bar{\sigma}(G(j\omega))$ is the maximal singular value of G.

Example 3 We consider a PI control system shown in Fig.3. The compensator is fixed as $C(s) = k + \frac{m}{s}$.

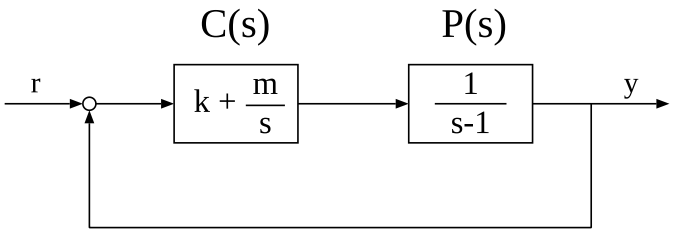

Figure 2: PI control system

The complementary sensitivity function is given by

$$T(s) = \frac{P(s)C(s)}{1 + P(s)C(s)} = \frac{ks + m}{s^2 + (k-1)s + m}. \tag{3}$$

Now we consider the specifications

$$||T(s)||_{[\omega_t, \infty]} < \gamma_t. \tag{4}$$

From the characteristic polynomial of the Hamilton matrix concerning with complementary sensitivity, the specification (4) is reduced to SDC:

$$f_t(x) = b_2 x^2 + b_1 x + b_0 \ \in \mathbf{N}_0[0, +\infty] \tag{5}$$

where

$$\begin{aligned} b_2 &= 1, \\ b_1 &= -2\omega_t^2 + 2m - (1-k)^2 + \tfrac{k^2}{\gamma_t^2}, \\ a_0 &= \omega_t^4 - (2m - (1-k)^2 + \tfrac{k^2}{\gamma_t^2})\omega_t^2 + m^2(1 - \tfrac{1}{\gamma_t^2}). \end{aligned}$$

3 Algorithm

3.1 Šiljak's criterion

In [13][16][17], it is shown that SDC can be readily checked by the following lemma based on the Routh-Hurwitz like criterion proposed by D.Šiljak [21]:

Lemma 4 Let $f(x) = \sum_{i=0}^{n} a_i x^i \in \mathbf{R}[x]$. $f(x)$ is sign definite in $x \in [0, +\infty]$ if and only if $V[f(x)] = n$ holds, where $V[f]$ is the number of sign changes of the most left column of the Modified Routh Array of f defined by

$$\begin{array}{ccccc} (-1)^n a_n & (-1)^{n-1} a_{n-1} & \cdots & -a_1 & a_0 \\ (-1)^n n a_n & (-1)^{n-1}(n-1) a_{n-1} & \cdots & -a_1 & \\ \vdots & & & & \\ a_0 & & & & \end{array}$$

The first two rows of Routh array above are formed by the coefficients of the polynomial $f(-x)$ and $f'(-x)$, and following rows in Routh array are formed by the coefficients of the polynomial remainder sequence generated by Euclidean divisions. Here we enumerate the issues when we use the Routh type criterion.

- In computing the remainder sequence by Euclidean divisions, we suffer from computational complexity due to exponential coefficient growth.
- In the parametric coefficient cases, there remains the problem concerning specialization of parameters; Since rational functions may appear in the sequence due to Euclidean division procedure, "division by 0" may occur by specialization.
- Moreover, we cannot proceed the whole procedure to compute modified Routh array in singular cases which occur when (i) an element of the first column becomes zero, or (ii) all the elements in a row of the array vanish simultaneously. (See Remark 4 in [21], or [10] for details.)

Hence it is desirable to develop the more efficient method which resolves the above issues. Usage of Sturm-Habicht sequence enable us to clear them by virtue of subresultants instead of Euclidean divisions.

3.2 Sturm-Habicht sequence

Let $f(x) \in \mathbf{R}[x]$ with degree n. Sturm-Habicht sequence of a polynomial $f(x)$ is defined as the *subresultant* sequence starting from $f(x)$ and $f'(x)$ modulo some specified sign changes (see Appendix B for the definition), and it has the following properties:

- desirable worst-case computational complexity :

  Subresultant sequence computation is a polynomial time. (A remainder sequence computation by a naive Euclidean division is not.)

- ∘ desirable specialization properties: Since Sturm-Habicht sequence is defined through determinants, rational functions does not appear in it.

Moreover, Sturm-Habicht sequence makes us free from the care of singular cases and can be used for real root counting as is Sturm sequence as follows:

Theorem 5 ([12]) *Let the Sturm-Habicht sequence of $P(x) \in \mathbf{R}[x]$ be*

$$\{SH_j(P(x))\} = \{g_0(x), \cdots, g_n(x)\}$$

and $\alpha, \beta \in \mathbf{R} \cup \{-\infty, +\infty\}$ s.t. $\alpha < \beta$. We define $W_{SH}(P;\alpha)$ as the number of sign variations in the list $\{g_0(\alpha), \cdots, g_n(\alpha)\}$. Then $W_{SH}(P;\alpha,\beta) \equiv W_{SH}(P;\alpha) - W_{SH}(P;\beta)$ gives a number of real roots of P in $[\alpha, \beta]$.

Example 6 Sturm-Habicht sequence $\{SH_j(F_2)\}$ $(j = 2, 1, 0)$ of $F_2 = c_2x^2 + c_1x + c_0$ is as follows:

$$\begin{array}{rcl} SH_2(F_2) & = & c_2x^2 + c_1x + c_0, \\ SH_1(F_2) & = & 2c_2x + c_1, \\ SH_0(F_2) & = & c_2c_1^2 - 4c_0c_2^2. \end{array}$$

Note that no rational polynomials appears in Sturm-Habicht sequence. On the other hand, modified Routh array of $F_2(x)$ is given as

$$\begin{array}{ccc} c_2 & -c_1 & c_0 \\ 2c_1 & -c_0 & \\ \frac{-c_0c_2+2c_1^2}{2c_1} & c_0 & \\ \frac{c_2c_0^2-6c_1^2c_0}{-c_2c_0+2c_1^2} & & \\ c_0 & & \end{array}$$

In the most left column, rational functions appear due to Euclidean division. This leads to bad specialization property, *i.e.,* "division by 0" by specialization.

3.3 Checking SDC by Sturm-Habicht sequence

For every j, the principal j-th Sturm-Habicht coefficient is defined as the coefficient of degree j of $SH_j(f)$. We denote the principal j-th Sturm-Habicht coefficient by $st_j(f)$ and the constant term of $SH_j(f)$ by $ct_j(f)$.

The sign definiteness of f in the interval $[0, +\infty]$ is equivalent to that f has no real roots in $[0, +\infty]$. Hence, an equivalent condition to the sign definition condition in $[0, +\infty]$ is obtained according to the following proposition (*cf.* Theorem 5):

Lemma 7 A polynomial $f(x)$ is sign definite in $[0, +\infty]$ if and only if

$$W_{SH}(f; 0, +\infty) = 0.$$

By definitions we have

$$\begin{aligned} W_{SH}(f; 0, +\infty) &= W_{SH}(f; 0) - W_{SH}(f; +\infty) \\ &= V(\{ct_n(f), \cdots, ct_0(f)\}) - V(\{st_n(f), \cdots, st_0(f)\}) \qquad (*) \end{aligned}$$

where V stands for the number of sign changes over the sequences. The last formula $(*)$ gives us how we count the number $W_{SH}(f; 0, +\infty)$ concretely. Since $ct_0(f) = st_0(f)$ and $st_n(f) = st_{n-1}(f)$ and we need $2(n+1) - 2 = 2n$ sign evaluations; $\{-, 0, +\}^{2n}$.

If we have Sturm-Habicht sequence for f, we construct the (quantifier-free) equivalent condition for SDC of f by the following procedure:

1. Consider all the 3^{2n} (at most) possible sign combinations over the polynomials $ct_i(f)$'s and $st_i\ (f)$'s,

2. Choose all sign conditions that satisfy $W_{SH}(f; 0, +\infty) = 0$ according to $(*)$,

3. Construct semi-algebraic sets generated by $ct_i(f)$'s and $st_i(f)$'s for each selected sign conditions and combine them as a union.

The obtained conditions are of the form of the union of semi-algebraic sets. In the case of multi-objective design, what we do is to compute the possible regions for all given specifications and superpose them in a parameter space.

3.4 Simplification

The result through above procedure obviously tends to be large and complicated, and hence we should reduce the result as simple as possible. Some possible simplifications are as follows:

- Manual simplifications by deleting some sign conditions trivially false (*i.e.* empty) or decreasing the number of unions by using the well-known rules; $< \cup > \to \neq$, $< \cup = \to \leq$, $> \cup = \to \geq$, are indicated in [11].

- We, fortunately, have some sophisticated softwares for automatic formula simplification (see [7]), which are implemented on a QE package

"REDLOG" ‡ and another QE package on a computer algebra system 'Risa/Asir" ¶.

4 Computational Complexity

Our approach consists of two parts: reduction to SDC and special QE computation. The dominant part of our approach is QE part. In particular, the construction of Sturm-Habicht sequence occupies the total computation time. Here we show some experimental results to demonstrate the tractability of our proposed method for practical control problems. All the computations were done by using a computer algebra system *Risa/Asir* and were executed on a PC with Pentium 200MHz CPU.

4.1 Generic polynomials

By using QEPCAD‡‡ , we can immediately solve the SDC for generic polynomials $F_n = \sum_{i=0}^{n} c_i x^i$, *i.e.*, $\forall x \; (\; x > 0 \rightarrow F_n > 0 \;)$ up to $n = 3$. However we could not solve the QE problems by QEPCAD for $n \geq 4$ due to the lack of memory.

On the other hand, we can solve it for generic polynomials up to $n = 8$ in our method as shown in Table 1. Table 1 shows the timing data to compute Sturm-Habicht sequence for generic polynomials $F_n(x)$. Once we compute Sturm-Habicht sequence of $F_n(x)$, the result can be used for another polynomials with degree n by substituting the coefficients c_i by those of an input polynomial. The results for the generic cases should be stored in a database to be called upon, whenever needed. This greatly improves the total efficiency. In the case of polynomials with many parameters it seems to be better that we compute Sturm-Habicht sequence in this way.

n	time (sec)	n	time (sec)
2	0.002	6	1.533
3	0.006	7	34.120
4	0.028	8	> 3600
5	0.121	9	—

Table 1: Sturm-Habicht sequence computation for generic polynomials

‡REDLOG is developed at University of Passau (Germany) on a computer algebra system REDUCE, see [6].

¶*Risa/Asir* is developed at Fujitsu labs, see [19], anonymous -ftp via: `endeavor.fujitsu.co.jp:/pub/isis/asir`.

‡‡These computation by QEPCAD are executed on Sun Ultra Sparc 1 Model 140.

4.2 PID-controller synthesis

Table 2 shows the timing data to compute Sturm-Habicht sequence of the polynomials $f_t(z)$, for which we check the SDC in analyzing sensitivity of PI control systems with compensators $C(s) = k + \frac{m}{s}$ and PID control systems with compensators $C(s) = k + \frac{m}{s} + \frac{d \cdot s}{1+0.1s}$. PI and PID control systems have same structure as Fig.1 and the compensator has 2 and 3 design parameters, respectively. As a target specification, here we consider the frequency restricted norm constraint for complementary sensitivity function : $||T(s)||_{[20,+\infty]} < -10$. This is equivalent to a SDC $f_t(z) > 0, \forall z > 0$. The numerators of the plants $p(s)$ are fixed as 1 and the denominators for each degree are given randomly. Noted that the computation of $f_t(z)$ is achieved immediately.

degree of $p(s)$ w.r.t. s	PI (sec)	PID (sec)
2	0.001	0.3709
3	0.029	1.931
4	0.111	9.807
5	0.459	35.840
6	1.528	145.700
7	4.718	443.200
8	13.090	1346.000
9	35.630	3644.000
10	82.700	7689.000
11	266.600	—
12	443.200	—
13	1176.000	—
14	1838.000	—
15	4333.000	—

Table 2: Sturm-Habicht sequence computation for PI and PID control systems

As a practical example, we quote the flexible beam example in [9]. The plant transfer function is given by

$$P(s) = \frac{-6.4750s^2 + 4.0302s + 175.7700}{s(5s^3 + 3.5682s^2 + 139.5021s + 0.0929)}.$$

We consider the PID control system for this plant with a same controller as above and the same frequency restricted norm constraint for complementary sensitivity $||T(s)||_{[20,+\infty]} < -10$. Then $f_t(z)$ is obtained in 0.55 sec and Sturm-Habicht for $f_t(z)$ is computed in 115.50 sec.

4.3 Combinatorial part

There are several possibilities to improve the combinatorial part. We can prune the impossible sign combinations before counting the number of sign changes owing to the followings;

$\langle a\rangle$ In the case of positive sign definite condition, we have to add one more condition such that "(head coefficient of f) > 0". This implies that $st_n > 0$ and $st_{n-1} > 0$. Hence all possible sign conditions are reduced to $3^{2(n-1)}$.

$\langle b\rangle$ From the algebraic properties of Sturm-Habicht sequence, it it impossible that more than two consecutive zeros appear in the sequence.

$\langle c\rangle$ When we determine the design parameters, we usually do not choose the parameter values on the boundaries of possible ranges of parameters. This implies that for actual design we do not have to check the sign combinations including 0 (except identically 0 case). Hence we should consider 2^{2n} sign combinations.

For example, in the case of generic polynomials with degree 4 there are totally $3^8 = 6561$ sign combinations to verify the number of sign changes. After pruning impossible sign combinations by $\langle a\rangle$, $\langle b\rangle$, and checking the number of sign changes, we have 561 feasible sign combinations. Furthermore, this formula can be simplified by deleting trivially empty semialgebraic sets manually. Finally we have 477 feasible sign combinations. For practical control problems, the number of possible sign combinations can become rather small as in §5. For $g(\omega)$ in §5.2, whose degree is 4, finally we have only one sign combination.

4.4 Summary

Here we summarize the computational complexity of our approach based on the computational results above.

Tractability : Our approach is practically applicable to the systems up to order 15 for the case of the number of design parameters in fixed-structure controller is 2 (e.g. PI control systems), and to the systems up to order 10 for the case of the number of design parameters the is 3 (e.g. PID control systems). In the case that controller has more than 3 parameters, our approach is practically applicable to the systems up to order 7 by using stored general forms.

Applicability : Our approach outputs a disjoint union $\mathcal{R}$ of semi-algebraic sets R_i which describes the possible range of design parameters Θ; $\mathcal{R} = \bigcup_{i=1}^{n} R_i$. And the obtained results are applicable to

- visualization of possible region of design parameters by a projection to 2 or 3 dimensional space,

- pre-processing (reduction to sub-problems) for numerical optimization such that

$$\min_{\Theta \in \mathcal{R}} F(\Theta) = \min_i \left\{ \min_{\Theta \in R_i} F(\Theta) \right\}$$

  where $F(\Theta)$ is an objective function in Θ,

- reduction of the VC-dimension for randomized algorithm

5 Integrated design examples

Here we show some computational results applying our method to practical integrated design examples, which demonstrate the tractability of our approach, We note that the first example can not be reduced to a convex optimization problem, and hence it is difficult to obtain the exact solution by numerical optimization.

We consider a swing-arm positioning mechanism for small disc storage devices shown in Fig.2 taken from [14]. It works basically as follows: when we apply a

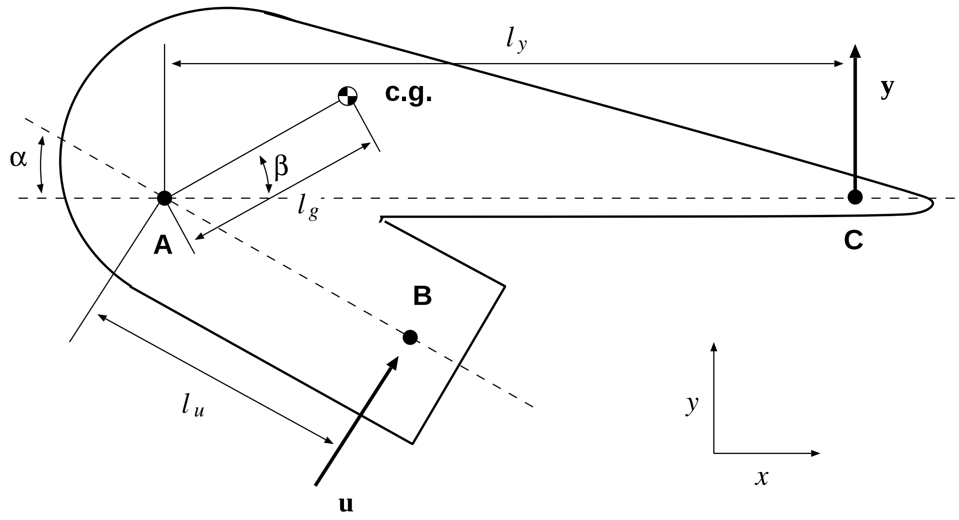

Figure 3: Geometry of the swing-arm

force input u to the point B, the swing-arm rotates around the pivot A with i the $x-y$ plane, and the sensing point C moves to a desired position. We design the shape of the swing-arm such that the resulting transfer function from u to $\dot{y}$ is positive real (PR). We employ the equation of motion, linearized around the equilibrium state, given in [14] by

$$M\ddot{q} + D\dot{q} + Kq = bu, \quad y = cq$$

where

$$q = \begin{bmatrix} x_g \\ y_g \\ \gamma \end{bmatrix}, \quad M = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & J \end{bmatrix},$$
$$D = dS, \ K = kS$$
$$S = \begin{bmatrix} 1 & 0 & l_g sin(\beta) \\ 0 & 1 & -l_g cos(\beta) \\ l_g sin(\beta) & -l_g cos(\beta) & l_g^2 \end{bmatrix},$$
$$b = \begin{bmatrix} sin(\alpha) \\ cos(\alpha) \\ l_u - l_g cos(\alpha + \beta) \end{bmatrix},$$
$$c = \begin{bmatrix} 0 & 1 & l_y - l_g cos(\beta) \end{bmatrix}.$$

and (x_g, y_g) $[m]$ is the displacements of the center of gravity and γ $[rad]$ is the angle between the x-axis and the line AC, measured counter clockwise.

In this equation, the flexibility of the pivot is modeled by two linear springs in the x and the y directions with small damping and assumed that the stiffness and the damping coefficients are the same for both directions. The values of the other swing-arm parameters are taken from [20] and shown in Table 3.

mass of swing-arm	m	0.033	kg
moment of inertia	J	1.7×10^{-5}	kg m^2
actuator point (angle)	α	—	*deg*
actuator point (length)	l_u	—	m
sensor point (length)	l_y	—	m
c.g. location (angle)	β	10	*deg*
c.g. location (length)	l_g	0.02	m
stiffness of pivot	k	1.5×10^6	N/m
damping of pivot	d	4.4	Ns/m

Table 3: Swing-arm parameters

Checking PR condition : For a scalar transfer function, positive real function is defined as follows:

Definition 8 *A real function $G(s) = q(s)/p(s)$ with relatively prime polynomials $p(s)$ and $q(s)$ is called (strictly)* positive real *if and only if*

(i) the polynomial $f(s) \equiv p(s) + q(s)$ is Hurwitz, and

(ii) $Re[G(i\omega)] > 0$ for all real ω.

Positive-realness of a given function is established as follows: The condition (i) is verified by using well-known Liénard-Chipart criterion. The condition (ii) is equivalent to a SDC

$$g(\omega) \equiv p_r(\omega)q_r(\omega) + q_i(\omega)p_i(\omega) > 0, \ \forall\omega > 0$$

where $p(i\omega) = p_r(\omega) + ip_i(\omega)$, $q(i\omega) = q_r(\omega) + iq_i(\omega)$ and hence verified by our method.

Simultaneous design of an actuator point B and a sensing point C (nonlinear case): The goal is to obtain simultaneously the region of actuator point B and a sensing point C yielding PR transfer functions. Thus, α, l_u and l_y are the design parameters. It is noted that the problem can not be reduced to a convex problem, and hence it is very difficult to find the exact region by numerical optimization. Instead, our approach can provide the exact region as will shown below. We define the new design parameter vector

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} sin(\alpha)/l_u \\ cos(\alpha)/l_u \end{bmatrix}$$

and the new control input $v = l_u u$. Then we have

$$bu = (b_1 + b_2\theta)v$$

where

$$b_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ l_g sin(\beta) & = l_g cos(\beta) \end{bmatrix}.$$

Note that the transfer function $T_{\dot{y}u}(s)$ from u to $\dot{y}$ is PR if and only if the transfer function $T_{\dot{y}v}(s)$ from v to $\dot{y}$ is so. Hence we try to compute the region of θ for which $T_{\dot{y}v}(s)$ is PR, and then to find the corresponding region in the original parameters.

A state space realization for $T_{\dot{y}v}(s)$ is given by

$$\left[\begin{array}{c|c} A & B(\theta) \\ \hline C & 0 \end{array}\right] = \left[\begin{array}{cc|c} 0 & I & 0 \\ -M^{-1}K & -M^{-1}D & M^{-1}(b_1 + b_2\theta) \\ \hline 0 & c & 0 \end{array}\right].$$

Then we have $T_{\dot{y}v}(s) = Q(s)/P(s)$ where

$$\begin{aligned} Q(s) = {} & (((3781008l_y - 1163605212/15625)\theta_1 + (-21448944l_y + 15366537516/15625)\theta_2 + 1089000000l_y - 2859859200)s^3 + \\ & (171864000000000l_y\theta_1 + (-974952000000000l_y + 45300000000000)\theta_2 + 990193600000000000l_y - 974952000000000)s^2 + 1320000000000000000000l_y s + \\ & 2250000000000000000000000l_y), \end{aligned}$$

$$\begin{aligned} P(s) = {} & 18513s^5 + 6853440s^4 + 2336984672000s^3 + \\ & 398640000000000s^2 + 6795000000000000000s. \end{aligned}$$

Then $f(s) = P(s) + Q(s)$ is Hurwitz if and only if

$$[D_4 > 0 \ \wedge \ D_2 > 0 \ \wedge \ A_4 > 0 \ \wedge \ A_2 > 0 \ \wedge A_0 > 0] \tag{6}$$

(see Appendix C for $f(s), D_4, D_2, A_4, A_2$ and A_0).

Next we compute Sturm-Habicht sequence of $g(\omega)$ and we have

$$\{ct_i\}_{i=4,3,2,1,0} = \{S_1, 0, -S_1, 0, S_1\}$$

$$\{st_i\}_{i=4,3,2,1,0} = \{S_3, S_3, S_2, S_4, S_1\}$$

(see Appendix C for $g(\omega)$ and S_i's). Additionally, we need the condition that the head coefficient of $g(\omega)$ is positive *i.e.* $S_3 > 0$ to ensure the positivity of $g(\omega)$. Finally, we have that the condition (ii) holds if and only if

$$[S_1 > 0 \ \wedge \ S_2 < 0 \ \wedge \ S_3 > 0 \ \wedge \ S_4 > 0] \cup [S_1 > 0 \ \wedge \ S_2 < 0 \ \wedge \ S_3 > 0 \ \wedge \ S_4 < 0] \cup [S_1 > 0 \ \wedge \ S_2 > 0 \ \wedge \ S_3 > 0 \ \wedge \ S_4 < 0] \tag{7}$$

Consequently, by superposing two possible regions (6) and (7) of design parameters θ_1, θ_2 and l_y in the parameter space, we have the feasible region of the design parameters for positive-realness.

Here we fix the sensing point and let $l_y = 0.06[m]$. Now we obtain the region of actuator point B yielding PR transfer functions. Thus, α and l_u are the design parameters. In this case, $f(s)$ is Hurwitz if and only if

$$[D_4' > 0 \ \wedge \ D_2' > 0 \ \wedge \ A_4' > 0 \ \wedge \ A_2' > 0] \tag{8}$$

(see Appendix C for D_4', D_2', A_4', A_2', and A_0').

As for the condition (ii), after removing the sign conditions which is obviously empty, we have necessary and sufficient conditions

$$[S_1' > 0 \ \wedge \ S_2' > 0 \ \wedge \ S_3' > 0 \ \wedge \ S_4' < 0] \tag{9}$$

(see Appendix C for S_i''s). Consequently, by superposing two possible regions (9) and (11) of design parameters θ_1 and θ_2 in the parameter space, we have the feasible region of θ_1 and θ_2 for positive-realness

$$[D_4' > 0 \ \wedge \ D_2' > 0 \ \wedge \ A_4' > 0 \ \wedge \ A_2' > 0] \cup \ [S_1' > 0 \ \wedge \ S_2' > 0 \ \wedge \ S_3' > 0 \ \wedge \ S_4' < 0] \tag{10}$$

which is shown in Fig. 3 as a shaded cell. This region is transformed to the region in α and l_u and described in the $x - y$ plane as shown in Fig. 4. Integer lattice points in (10) are described in Fig. 4 as dots. And the region (10) corresponds to that below the dotted line. (All the computations needed here has been done in about one minute.)

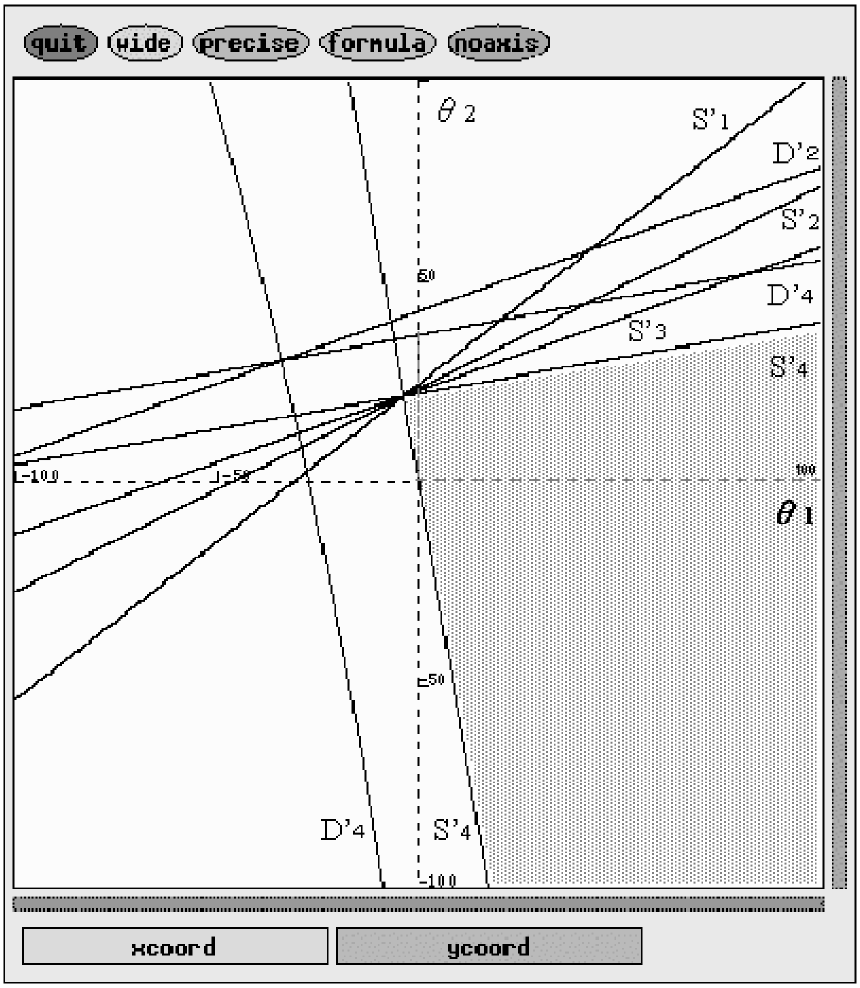

Figure 4: Admissible parameter space for PR

6 Conclusion

In this paper, we have proposed a method of robust control design based on SDC by a special QE method using Sturm-Habicht sequence, in particular, for multi-objective control design using low degree fixed-structure controller. Our approach is more efficient than the method using Routh-Hurwitz like criterion and has a good specialization property. Moreover, compared with the matrix inequality approach based on numerical optimizations, our approach based on a special QE has several advantages such as applicability to parametric and nonlinear cases, possibility to obtain non-conservative results and less complexity for multi-objective design. We have demonstrated our method by applying it to some examples and showed by computational experiments on a computer algebra system that our approach is

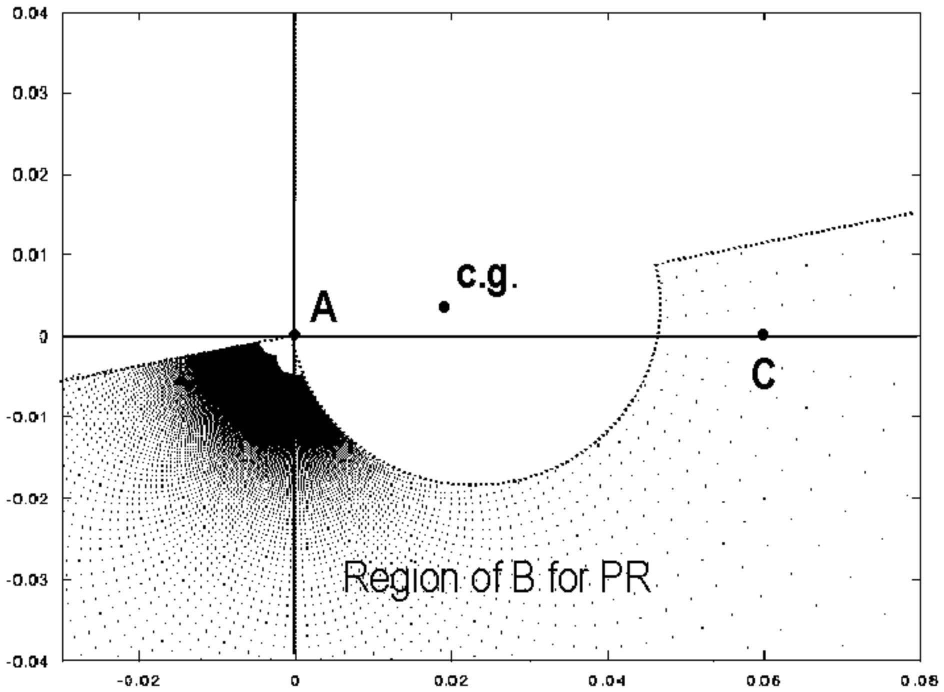

Figure 5: Region of actuation points for positive-realness

practically appricable one.

Acknowledgments The authors would like to thank Dr. J.Kaneko, Dr. K.Yokoyama and Prof. T.Iwasaki for their invaluable comments and advice. The research is supported in part by The Grant-in-Aid for COE Research Project of Super Mechano-Systems by The Ministry of Education, Science, Sport and Culture in Japan.

References

[1] J. Ackermann. *Robust Control — Systems with Uncertain Physical Parameters.* Springer-Verlag, 1993.

[2] H. Anai. On solving semidefinite programming by quantifier elimination. *In Proc. of American Control Conference, Philadelphia*, pages 2814–2818, 1998.

[3] S. Boyd *et.al.* A bisection method for computing the h_∞ norm of a transfer matrix and related problems. *Math. Contr. Sig. Syst.*, 2:207–216, 1989.

[4] B. Caviness and J. Johnson, editors. *Quantifier Elimination and Cylindrical Algebraic Decomposition.* Texts and monographs in symbolic computation. Springer-Verlag, 1998.

[5] G. E. Collins. *Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition, LNCS 32.* Springer Verlag, 1975.

[6] A. Dolzmann and T. Sturm. REDLOG computer algebra meets computer logic. Technical Report MIP-9603, Universitä Passau, Fakultät Für Mathematik und Informatik, Feb. 1996.

[7] A. Dolzmann and T. Sturm. Simplification of quantifier-free formulae over ordered fields. *Journal of Symbolic Computation*, 24(2):209–231, Aug. 1997. Applications of quantifier elimination (Albuquerque, NM, 1995).

[8] P. Dorato, W.Yang, and C.Abdallah. Robust multi-objective feedback design by quantifier elimination. *J. Symb. Comp.* **24**, pages 153–159, 1997.

[9] J. Doyle, B.A.Francis, and A.R.Tannenbaum. *Feedback control theory.* Macmillan Publishing Company, 1992.

[10] F. R. Gantmacher. *The Theory of Matrices*, volume II. Chelsea pub. co., 1959.

[11] L. González-Vega. A combinatorial algorithm solving some quantifier elimination problems. In B. Caviness and J. Johnson, editors, *Quantifier Elimination and Cylindrical Algebraic Decomposition*, Texts and monographs in symbolic computation, pages 365–375. Springer-Verlag, 1998.

[12] L. González-Vega, H. Lombardi, T. Recio, and M.-F. Roy:. Sturm-habicht sequence. In *Proceedings of ISSAC"89*, pages 136–146, Portland, 1989. ACM Press.

[13] S. Hara, T. Kimura, and R. Kondo. H $_\infty$ control system design by a parameter space approach. In *Proceedings of MTNS-91*, pages 287–292, Kobe, Japan, 1991.

[14] T. Iwasaki. Critical open-loop properties for closed-loop performance. In *Proc. COE/Super Mechano-Systems Workshop'98*, pages 170–185, 1998.

[15] M. Jirstrand. *Algebraic Methods for Modeling and Design in Control.* PhD thesis, Linköping University, 1996.

[16] T. Kimura and S. Hara. A robust control system design by a parameter space approach based on sign definition condition. In *Proceedings of KACC-91*, pages 1533–1538, Soul, Korea, 1991.

[17] R. Kondo, S. Hara, and T. Kaneko. Parameter space design for H_∞ control (in japanese). *Trans. of SICE*, 27(6):714–716, 1991.

[18] A. Neubacher. *Parametric Robust Stability by Quantifier Elimination.* PhD thesis, Risc-Linz, Oct, 1997.

[19] M. Noro and T.Takeshima. Risa/Asir – a computer algebra system. In *ISSAC '92: proceedings of the International Symposium on Symbolic and Algebraic Computation*, pages 387–396. ACM Press, 1992.

[20] K. Ono and T. Teramoto. Design methodology to stabilize the natural modes of vibration of a swing-arm positioning mechanism. *ASME Adv. Info. Strage Syst.*, 4:343–359, 1992.

[21] D. D. Šiljak. New algebraic criterion for positive realness. *Journal of the Franklin Institute*, 291:109–120, 1971.

Appendix

A. Quantifier elimination : QE deals with the *first-order formulas*, which consists of polynomial equations, inequalities, quantifiers ($\forall, \exists$) and Boolean operators ($\wedge, \vee, \neg, \rightarrow, etc$). QE procedure is an algorithm to compute equivalent quantifier-free formula for a given first-order formula over the real closed fie ld. For example, for the input

$$\forall x(x^2 + bx + c > 0),$$

QE outputs the equivalent quantifier-free formula; $b^2 - 4c < 0$. See [4] for the details about QE.

B. Sturm-Habicht sequence : We briefly show the definition of Sturm-Habicht sequence and the relation between Sturm-Habicht sequence and the number of real roots (see [12] for derails).

Definition 9 *[12] Let P, Q be polynomials in $\mathbf{R}[x]$*

$$P = \sum_{k=0}^{n} a_k x^k \quad Q = \sum_{k=0}^{m} b_k x^k$$

where n and m be non-negative integers and let $\ell = min(n,m)$. For $i = 0, 1, \cdots, \ell$ we define the subresultant associated to P, n, Q and m of index i as follows:

$$Sres_i(P, n, Q, m) = \sum_{j=0}^{i} M_j^i(P, Q) x^j$$

where $M_j^i(P, Q)$ is the determinant of the matrix composed by the columns $1, 2, \cdots,$ $n + m - 2i - 1$ and $n + m - i - j$ in the matrix $s_i(P, n, Q, m)$:

$$s_i = \begin{array}{c} \overbrace{\hspace{5cm}}^{n+m-i} \\ \left(\begin{array}{ccccc} a_n & \cdots & a_0 & & \\ & \ddots & & \ddots & \\ & & a_n & \cdots & a_0 \\ b_n & \cdots & b_0 & & \\ & \ddots & & \ddots & \\ & & b_n & \cdots & b_0 \end{array}\right) \end{array} \begin{array}{l} \\ \left.\begin{array}{c} \\ \\ \\ \end{array}\right\} m-i \\ \left.\begin{array}{c} \\ \\ \\ \end{array}\right\} n-i \end{array}$$

Definition 10 *[12] Let P, Q be polynomials in $\mathbf{R}[x]$ with degrees n, m, respectively. Here n, m be non-negative integers. Let $v = n + m - 1$ and $\delta_k = (-1)^{\frac{k(k+1)}{2}}$ for every integer k. The Sturm-Habicht sequence associated to P and Q is defined as the list of the polynomials $\{SH_j(P, Q)\}_{j=0,\cdots,v+1}$ given by*

- $SH_{v+1}(P, Q) = P$,
- $SH_v(P, Q) = P'Q$, *and*
- $SH_j(P, Q) = \delta_{v-j} Sres_j(P, p, P'Q, v)$

for every $j = 0, 1, \cdots, v-1$ *where* $P' = \frac{dP}{dx}$. *When* $Q = 1$, $\{SH_j(P, 1)\}_{j=0,\cdots,v+1}$ *is called the Sturm-Habicht sequence of* P.

C. Results in Swing-arm example :

$$\begin{aligned} f(s) = {} & 289265625s^5 + ((59078250000l_y - 1163605212)\theta_1 + (-335139750000l_y + 15366537516)\theta_2 + 17015625000000l_y - \\ & 228054750000)s^4 + (7877100000000l_y\theta_1 + (-44685300000000l_y + 2076250000000)\theta_2 + 4537500000000000l_y + \\ & 36470700200000000)s^3 + (2685375000000000000l_y\theta_1 + (-15233625000000000000l_y + 707812500000000000)\theta_2 + \\ & 1547177500000000000000l_y - 9004875000000000000)s^2 + (206250000000000000000000l_y + 1061718750000000000000000)s + \\ & 35156250000000000000000000000l_y, \end{aligned}$$

$$\begin{aligned} g(\omega) = {} & ((36798364968750l_y - 1132769673882)\theta_1 + (-208750171781250l_y + 9499435599951)\theta_2 + 4632503906250000l_y - 208750171781250)w^4 + \\ & ((-3344651713095000000000l_y + 65889145129500000000)\theta_1 + (18973577229585000000000l_y - 8699577542810000000000)\theta_2 - \\ & 421054363125000000000000l_y + 18973577229585000000000)w^2 + 76029679687500000000000000000l_y\theta_1 + \\ & (-431302007812500000000000000000l_y + 20039941406250000000000000000)\theta_2 + 9571289062500000000000000000000l_y - \\ & 431302007812500000000000000000. \end{aligned}$$

$\boxed{(6)}$

$$D_4 = (1211811197127300000l_y^4 + 1245227926435835556992l_y^3 - 245730193236239619 6l_y^2)\theta_1^3 + ((-2062312788475170000000\theta_2 + 3492969766056540000000)l_y^4 + (-211099662756875455551168\theta_2 + 158670036896061433881 60)l_y^3 + (125974240237066212684\theta_2 + 136387633019153114491 2)l_y^2 + (-12953937710749959 00\theta_2 + 314843791779038756480 4)l_y - 143516675897398795761)\theta_1^2 + ((1169911079545131000000\theta_2^2 - 3962991504161844000000 0\theta_2 + 2358994522500000000000)l_y^4 + (119288462595425073498 24\theta_2^2 - 176571579738966781205760\theta_2 + 807738951024205200000 00)l_y^3 + (-119023267003504748161 2\theta_2^2 + 932162622819095221276 8\theta_2 + 322623496656956820937920)l_y^2 + (331066391421197124 00\theta_2^2 - 1765575654351601152614 4\theta_2 + 47401282886558376993940 8)l_y - 17072027251110562 5\theta_2^2 + 79391209947275136279 6\theta_2 -1810433744829187291598 4)\theta_1 + (-2212228170384111000000\theta_2^3 + 11240651017565046000000 \theta_2^2 - 1338213021750000000000 0\theta_2 + 45295593750000000000000)l_y^4 + (-224688842333765652997 44\theta_2^3 + 491044570979331145539840\theta_2^2 + 547594772990364000000 0\theta_2 + 40664705112075000000000 0)l_y^3 + (314660909283863381197 2\theta_2^3 - 962892015476614766113 12\theta_2^2 + 245455125399045704725440\theta_2 + 113920036923280132600000 00)l_y^2 + (-145996004193369089100\theta_2^3 + 238081229568297444159 6\theta_2^2 - 437488292085411557640144\theta_2 + 101572351042374870240000 00)l_y + 22545270897975731 25\theta_2^3 + 429249890174932568 61\theta_2^2 + 1990668642424279663111 2\theta_2 - 37721631423833878950000 0$$

$$D_2 = (6409990125000l_y^2 - 126251165502l_y)\theta_1^2 + ((-727253257500000\theta_2 + 5538585937500000)l_y^2 + (40730167659 72\theta_2 + 18881022006000000)l_y - 332773460025\theta_2 - 584538523939524)\theta_1 + (2062785161250000\theta_2^2 - 31419351562500000\theta_2 + 106347656250000000)l_y^2 + (-190425929035 98\theta_2^2 - 1060739978017500 00\theta_2 + 2369031585937500000)l_y + 4394596903 25\theta_2^2 + 48926867629213332\theta_2 - 78684788175125000$$

$$A_0 = l_y$$

$$A_2 = 42966l_y\theta_1 + (-243738\theta_2 + 24754840)l_y + 113325\theta_2 - 144078$$

$$A_4 = (447562500l_y - 8815191)\theta_1 + (-25389375 00\theta_2 + 128906250000)l_y + 1164131 63\theta_2 - 1727687500$$

$\boxed{(7)}$

$$S_1 = 2162622l_y\theta_1 + (-12268146l_y + 570025)\theta_2 + 272250000l_y - 12268146,$$

$$S_2 = (608118493290l_y - 11979844569)\theta_1 + (-3449741314470l_y + 158174137142)\theta_2 + 76555338750000l_y - 3449741314470,$$

$$S_3 = (33790968750l_y - 1040192538)\theta_1 + (-191689781250l_y + 8723081359)\theta_2 + 4253906250000l_y - 191689781250,$$

$$
\begin{aligned}
S_4 = {} & (-131525904787846569000l_y^2 - 289274297231090375820l_y + 13046970536127163251)\theta_1^2 +((149224321468976153400l_y^2 + 158301764100117432102 0l_y - 716423557963053762 36)\theta_2 \\
& -331152902157577500000 0l_y^2 - 362671847510200463466 00l_y + 164099843267959563426 0)\theta_1 +(-42326069061822498810 0l_y^2 + 328914076295560537320l_y - 13962545539331773576)\theta_2^2 \\
& +(187856784588008250000 00l_y^2 - 814565665982429497620 0l_y + 328914076295560537320)\theta_2 -208442292764062500000 000l_y^2 + 187856784588008250000 00l_y - 423260690618224988100.
\end{aligned}
$$

$\boxed{(8)}$

$$
\begin{aligned}
D_4' &= 316105676834325001470\theta_1^3 + (-143861788784333667663\theta_2 + 9335433271499477426938 5)\theta_1^2 +(212905040871515052984\theta_2^2 - 469667541040413368974860\theta_2 + 19991935079026618520967000)\theta_1 \\
& \quad -103107463685775488204\theta_2^3 - 926128178987806163988 85\theta_2^2 - 947563068948488177785 6000\theta_2 +475891029943857811937 500000,\\
D_2' &= 3875223629970\theta_1^2 + (-1267687819167\theta_2 + 1420654264488690 0)\theta_1 + 987669353978\theta_2^2 -396215692702167 00\theta_2 + 1682140565153125000,\\
A_2' &= 21483\theta_1 - 27494\theta_2 + 11176770,\\
A_4' &= 180385559\theta_1 - 359230087\theta_2 + 6006687500.
\end{aligned}
$$

$\boxed{(9)}$

$$
\begin{aligned}
S_1' &= 32439933\theta_1 - 41515940\theta_2 + 101671350,\\
S_2' &= 1225363251420\theta_1 - 2440517086310\theta_2 + 5717895052650,\\
S_3' &= 9872655870\theta_1 - 27783055160\theta_2 + 63544593750,\\
S_4' &= -108920915586547101576\theta_1^2 + (596897755526334929436\theta_2 -136738539214819984134 00)\theta_1 + 106214013804406217651\theta_2^2 \\
& \quad -230492202105535478130 0t2 - 116280592602001220250 0.
\end{aligned}
$$