A symbolic-numeric method for the parametric H_∞ loop-shaping design problem

Guillaume Rance[1], Yacine Bouzidi[2], Alban Quadrat[3] and Arnaud Quadrat[4]

***Abstract*— In this paper, we present a symbolic-numeric method for solving the H_∞ loop-shaping design problem for low order single-input single-output systems with parameters. Due to the system parameters, no purely numerical algorithm can indeed solve the problem. Using Gröbner basis techniques and the Rational Univariate Representation of zero-dimensional algebraic varieties, we first give a parametrization of all the solutions of the two Algebraic Riccati Equations associated with the H_∞ control problem. Then, following the works of [1], [9] on the spectral factorization problem, a certified symbolic-numeric algorithm is obtained for the computation of the positive definite solutions of these two Algebraic Riccati Equations. Finally, we present a certified symbolic-numeric algorithm which solves the H_∞ loop-shaping design problem for the above class of systems. This algorithm is illustrated with a standard example.**

I. Introduction

Nowadays, automatic control usually consists in the solving of complex problems. As a consequence, this is usually done numerically for a particular system and hardware performances are required to handle the underlying numerical computations: the more a system or a problem is refined, i.e., takes into consideration finer dynamics of the physical system or constraints, the more time-computation is required.

An alternative to this approach is to solve these problems symbolically, that is to say to solve the problems corresponding to a certain class of systems. The goal is to compute the solution of a control problem for a set of systems depicted by some parameters. Its solution (e.g., a stabilizing controller) then depends on the system parameters. In what follows, this approach will be called a "*symbolic method*". One of the greatest benefits of this approach is to simply obtain the solution for a particular system belonging to the class of systems by numerically evaluating the system parameters in the closed-form solution. Since only evaluation operations are required, this task can be achieved at a cheap computational cost.

The purpose of this paper is to develop a symbolic method for *the H_∞ loop-shaping design problem* [8], [14], [16] for linear systems containing parameters. This synthesis problem yields the computation of a stabilizing controller which takes into account robustness objectives in the frequency domain (e.g., stability margins) and the modeling of different transfer functions via perturbations. Practically, H_∞ control provides a natural *compromise* between the performance and the robustness of the control-loop system. The desired performance properties and robustness conditions can be obtained by *"weighting"* the plant transfer function, which results in a "*weighted plant*".

On a practical point of view, such an approach is interesting in the design stage of a project, when a designer wants to know if the global design of its system has a chance to reach the specifications, or if a global rework is needed. As the global design is likely to change at some stage, the engineer has to quickly provide a range of performances and robustness that the closed-loop system could achieve in order to *choose the appropriate global design*. This approach is also interesting in *adaptive control* where the controller depends on the measurement of variables or parameters (e.g., a mechanical mode) and has to be re-computed in real time (i.e., quickly relative to the dynamics of the system). As the technologies of embedded processors are not powerful enough to compute H_∞ optimization problems in real time, the symbolic approach developed in this paper is justified.

Solving the H_∞ control problem for a system with parameters requires solving an *Algebraic Riccati Equation* (ARE), whose resolution was studied in [7] using *Gröbner basis techniques*. A method to solve an H_∞ control problem was proposed in [9] based on properties of the spectral factorization of the Hamiltonian matrix associated with the ARE [10]. In this article, a different method is proposed by means of the study of the polynomial system defined by an ARE (Section III). A *Rational Univariate Representation* of the solutions of this polynomial system is given (Section IV). In Section V, the equivalence between our method and the one developed in [9] is shown. In Section VI, an algorithm is given to compute the optimal solution to the H_∞ control problem by means of certified root isolation methods. Finally, in Section VII, a standard example illustrates our method.

II. The standard H_∞-control problem

In this paper, we will consider a single-input single-output (SISO) finite-dimensional linear system (Figure 1) defined by $y_1 = G\, e_1$, where the transfer function G is given by

$$G := \frac{c_n\, s^n + c_{n-1}\, s^{n-1} + \ldots + c_1\, s + c_0}{a_n\, s^n + a_{n-1}\, s^{n-1} + \ldots + a_1\, s + a_0}\,, \qquad (1)$$

$a_i,\, c_i \in \mathbb{R}$ for $i = 0, \ldots, n$ and n is the order of G (i.e., $a_n \neq 0$). In what follows, to get a strictly proper transfer

[1]Guillaume Rance is with Safran Electronics & Defense, 100 Avenue de Paris, Massy, France. guillaume.rance@safrangroup.com

[2]Yacine Bouzidi is with Inria Lille - Nord Europe, Non-A project, 40 Avenue Halley, Bat A - Park Plaza, 59650 Vileneuve d'Ascq, France. yacine.bouzidi@inria.fr

[3]Alban Quadrat is with Inria Lille - Nord Europe, Non-A project, 40 Avenue Halley, Bat A - Park Plaza, 59650 Vileneuve d'Ascq, France. alban.quadrat@inria.fr

[4]Arnaud Quadrat is with Safran Electronics & Defense, 100 Avenue de Paris, Massy, France. arnaud.quadrat@safrangroup.com

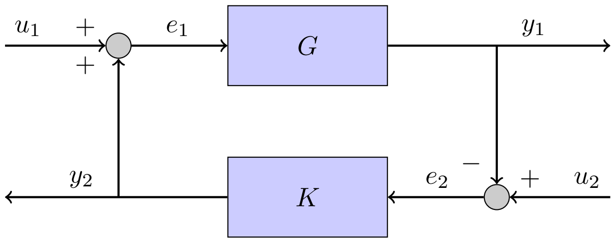

Fig. 1. Control scheme

function G, we set:

$$a_n := 1, \quad c_n := 0. \tag{2}$$

Let us consider its controller canonical form defined by the following state-space representation:

$$\dot{x} = A\,x + B\,e_1, \quad y_1 = C\,x, \tag{3}$$

$$\begin{array}{c} A := \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 0 & 1 \\ -a_0 & -a_1 & \dots & -a_{n-2} & -a_{n-1} \end{pmatrix} \in \mathbb{R}^{n\times n}, \\ B := (0 \ \dots \ 0 \ \ 1)^T \in \mathbb{R}^{n\times 1}, \\ C := (c_0 \quad \dots \quad c_{n-1}) \in \mathbb{R}^{1\times n}. \end{array} \tag{4}$$

Let $\mathbb{R}(s)$ be the field of rational functions with real coefficients. Let $K \in \mathbb{R}(s)$ be a controller and let us consider the closed-loop system defined in Figure 1. If we denote the sensitivity transfer function by $S := (1 + G\,K)^{-1}$, then:

$$\begin{pmatrix} e_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} S & K\,S \\ G\,S & G\,K\,S \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}.$$

Let us consider the following standard problem.

Robust Control Problem (RCP): *Given $\gamma > 0$, find a controller K which stabilizes G (i.e., such that the rational transfer functions S, $K\,S$ and $G\,S$ are proper and stable) and is such that:*

$$\left\| \begin{pmatrix} S & K\,S \\ G\,S & G\,K\,S \end{pmatrix} \right\|_\infty < \gamma. \tag{5}$$

For more details, see [8], [16], [6], [14] and the references therein. The RCP yields a compromise between the performance of the closed-loop system and the robustness with respect to the perturbations u_1 and u_2.

We briefly state a standard result of H_∞-*control theory*.

Theorem 1: [8, Cor. 5.1], [16, Ch. 18], [6, Th. 3.2] Let (A, B, C) be a controllable and observable state-space representation (3) of the transfer function G defined by (1). Then, the optimal value of γ of (5) is given by

$$\gamma_{\mathrm{opt}} = \sqrt{1 + \lambda_{\max}(Y X)},$$

where X and Y are respectively the unique real positive definite solutions of the following *Algebraic Riccati Equations* (ARE)

$$\mathscr{R} \ := \ X\,A + A^T\,X - X\,B\,B^T\,X + C^T\,C = 0, \tag{6}$$

$$\widetilde{\mathscr{R}} \ := \ Y\,A^T + A\,Y - Y\,C^T\,C\,Y + B\,B^T = 0, \tag{7}$$

and $\lambda_{\max}$ is the greatest eigenvalue of $Y\,X$ (all the eigenvalues of $Y\,X$ are real positive).

For $\gamma > \gamma_{\mathrm{opt}}$, a controller K_γ satisfying the RCP is defined by

$$\dot{z} = A_\gamma\,z + B_\gamma\,e_2, \quad y_2 = C_\gamma\,z, \tag{8}$$

with the following notations:

$$\begin{cases} Z_\gamma := \left(I_n + Y\,X - \gamma^2\,I_n\right)^{-1}, \\ A_\gamma := A - B\,B^T\,X + \gamma^2\,Z_\gamma\,Y\,C^T\,C, \\ B_\gamma := -\gamma^2\,Z_\gamma\,Y\,C^T, \\ C_\gamma := B^T\,X. \end{cases}$$

We note that $\mathscr{R}^T = \mathscr{R}$ and $\widetilde{\mathscr{R}}^T = \widetilde{\mathscr{R}}$. For SISO systems, the next lemma shows that (7) is a consequence of (6).

Lemma 1: We have $Y := QXQ$, where $Q = Q^T$ is defined by

$$Q^{-1} := \mathscr{P} = \begin{pmatrix} \mathscr{P}_1 & \cdots & \mathscr{P}_n \end{pmatrix}, \tag{9}$$

where:

$$\mathscr{P}_i^{\,T} := C \sum_{j=0}^{n-i} a_{n-j}\,A^{n-i-j}, \quad i = 1, \dots, n.$$

Corollary 1: Let $\rho_r(M)$ be the set of all the real eigenvalues of $M \in \mathbb{R}^{n\times n}$ and $\lambda_\star(M) := \max_{\lambda\in\rho_r(M)} |\lambda|$. We have:

$$\gamma_{\mathrm{opt}} = \sqrt{1 + \lambda_{\max}(Y X)} = \sqrt{1 + \lambda_\star^2(Q\,X)}. \tag{10}$$

In this paper, we focus on the computation of X in the case where the *a_i's and c_j's are unknown parameters and not fixed numerical values*. In particular, *numerical algorithms* for the computation of the positive definite solutions of ARE cannot be used. We have to consider *symbolic methods* to take into account the parameters a_i's and c_j's.

III. Polynomial system description of an ARE

We first study the equation $\mathscr{R} = 0$. For $1 \le i \le j \le n$, $x_{i,j}$ denotes the entries of X and we set $x_{i,0} = x_{0,j} = 0$. From direct computations, we get the following proposition.

Proposition 1: The equation $\mathscr{R} = 0$, defined by (6), is equivalent to the following system of $\frac{n(n+1)}{2}$ polynomial equations in the $\frac{n(n+1)}{2}$ unknowns $\{x_{i,j}\}_{1\le i\le j\le n}$:

$$\begin{array}{r} -\left(x_{i,n} + a_{i-1}\right)\left(x_{j,n} + a_{j-1}\right) \\ +x_{i,j-1} + x_{i-1,j} + a_{i-1}\,a_{j-1} + c_{i-1}\,c_{j-1} = 0, \\ 1 \le i \le j \le n. \end{array} \tag{11}$$

Let us introduce the following notations:

$$\begin{array}{ll} k < 0, & b_k := 0, \\ 0 \le k \le n-1, & b_k := x_{k+1,n} + a_k, \\ k = n, & b_k := 1, \\ k > n, & b_k := 0. \end{array} \tag{12}$$

Theorem 2: For $k = 1 \ldots n$, we set $x_{k,0} = x_{0,k} := 0$, and for $(i, j) \in \mathbb{N}^2$, we define:

$$\begin{cases} N(i,j) := i-1, & 2 \leq i+j \leq n+1, \\ N(i,j) := n-j+1, & n+1 < i+j \leq 2\,n+1. \end{cases}$$

The elements of X solution of (6) are determined only by the b_k's as follows

$$\begin{aligned} x_{k,n} &= b_{k-1} - a_{k-1}, \\ x_{i,j-1} &= \sum_{k=0}^{N(i,j)} (-1)^k\, b_{i-1-k}\, b_{j-1+k} - \theta_{N(i,j)}, \end{aligned} \tag{13}$$

where $1 \leq k \leq n$, $1 \leq i < j \leq n$, and θ_m is defined by:

$$\theta_m := \sum_{k=0}^{m} (-1)^k \left(a_{i-1-k}\, a_{j-1+k} + c_{i-1-k}\, c_{j-1+k}\right). \tag{14}$$

Proof: The diagonal elements of the matrix $\mathscr{R}$ yield:

$$1 \leq i \leq n,\; 2\,x_{i-1,i} = 2\,x_{i,i-1} = b_{i-1}^2 - \left(a_{i-1}^2 + c_{i-1}^2\right). \tag{15}$$

Using (11) and (12), the i^{th} row of X can be deduced from the $(i-1)^{\text{th}}$ row, which allows us to write $x_{i,j-1}$ in terms of $x_{i-1,j}$ which itself can be expressed in terms of $x_{i-2,j+1}, \ldots$ After m substitutions, for $1 \leq i < j \leq n-m$, we obtain

$$\begin{aligned} x_{i,j-1} = \;& (-1)^{m+1}\, x_{i-1-m,j+m} \\ & + \sum_{k=0}^{m} (-1)^k\, b_{i-1-k}\, b_{j-1+k} - \theta_m, \end{aligned}$$

where θ_m is defined by (14).

We distinguish two halting conditions in the induction:

1) We reach the first row of X, i.e. an element of the form $x_{i-1-m,j+m} = x_{0,j+m} = 0$ for $m = i-1$. In this case, note that we have $1 \leq j+m \leq n$, i.e. $2 \leq i+j \leq n+1$. By reaching this limit, one obtains the following expression:

$$\begin{aligned} &\forall (i,j) \in \mathbb{N}^2, \quad i < j, \quad 2 \leq i+j \leq n+1, \\ &x_{i,j-1} = \sum_{k=0}^{i-1} (-1)^k\, b_{i-1-k}\, b_{j-1+k} - \theta_{i-1}. \end{aligned} \tag{16}$$

2) We reach the last column of X, i.e. an element of the form $x_{i-1-m,j+m} = x_{i-1-m,n}$ for $m = n-j$. In this case, note that we have $1 \leq i-1-m \leq n$, i.e. $n+1 < i+j \leq 2\,n+1$. By reaching this limit, using $x_{i+j-n-1,n} = b_{i+j-n-2}\, b_n - a_{i+j-n-2}\, a_n$ (see (2) and (12)), we obtain the following expression:

$$\begin{aligned} &\forall (i,j) \in \mathbb{N}^2, \quad i < j, \quad n+1 < i+j \leq 2\,n+1, \\ &x_{i,j-1} = \sum_{k=0}^{n-j+1} (-1)^k\, b_{i-1-k}\, b_{j-1+k} - \theta_{n-j+1}. \end{aligned} \tag{17}$$

Hence, the entries $x_{i,j}$ of X can be explicitly expressed in terms of the b_k's defined in (12), i.e., up to constant a_{k-1}, in terms of the elements of the last column $x_{k,n}$ of X. ■

We note that for $i = 1$, using $x_{1,0} = 0$, (15) then yields:

$$\mathcal{B}_0 := b_0^2 - (a_0^2 + c_0^2) = 0.$$

For $i \neq 1$, $x_{i-1,i}$ appears twice in (15) and (16) or (17). Thus, combining (15) and (13) for $i = j = k+1$, we obtain a polynomial system of $n-1$ equations in $n-1$ unknowns $\{b_i\}_{1 \leq i \leq n-1}$. If M is defined by:

$$\begin{cases} M(k) := k, & 1 \leq k \leq \dfrac{n-1}{2}, \\ M(k) := n-k, & \dfrac{n-1}{2} < k \leq n-1, \end{cases} \tag{18}$$

then this system, denoted by $\mathcal{B}$, is defined by

$$\mathcal{B} := \left\{\mathcal{B}_k = 0\right\}_{0 \leq k \leq n-1}, \tag{19}$$

where the polynomials $\mathcal{B}_k$ are given by

$$\begin{cases} \mathcal{B}_0 := b_0^2 - d_0, \\ \mathcal{B}_k := b_k^2 + 2 \displaystyle\sum_{m=1}^{M(k)} (-1)^m b_{k-m}\, b_{k+m} - d_{2k}, \\ \qquad\qquad\qquad\qquad 1 \leq k \leq n-1, \end{cases} \tag{20}$$

and the constants d_{2k} are defined by

$$\begin{cases} d_0 & := & a_0^2 + c_0^2, \\ d_{2k} & := & 2 \displaystyle\sum_{m=1}^{M(k)} (-1)^m \,\left(a_{k-m}\, a_{k+m} + c_{k-m}\, c_{k+m}\right) \\ & & \quad + a_k^2 + c_k^2, \qquad 1 \leq k \leq n-1, \\ d_{2n} & := & 1. \end{cases} \tag{21}$$

Remark 1: We point out that the parameters a_i's and c_j's appear only in the constant terms of the $\mathcal{B}_k$'s (compare with (11)), which highly simplifies the study of (19).

If the d_{2k}'s are fixed, then the polynomial system (21) defines all the transfer functions G (see (1)) which give the same solution of (6). The study of this algebraic variety will be studied in details in a future publication.

Let $\mathbb{K} := \mathbb{Q}(d_0, .., d_{2n})$ be the field for rational functions in $d_0, \ldots, d_{2n}$, $\overline{\mathbb{K}}$ its *algebraic closure* and $\mathbb{K}' := \mathbb{Q}(a_0, \ldots, a_{n-1}, c_0, \ldots, c_{n-1})$. Using *Gröbner basis techniques* (see, e.g., [4] and the references therein), we can now state some results on the polynomial system (19).

Theorem 3: The polynomial system $\mathcal{B}$ given by (19) and defined over $R := \mathbb{K}[b_0, .., b_{n-1}]$ is *zero-dimensional* (i.e., $\mathcal{A} := R/\langle \mathcal{B} \rangle$, where $\langle \mathcal{B} \rangle := \{\sum_{i=1}^{n} r_k\, \mathcal{B}_k \mid r_k \in R\}$ is the ideal of R generated by $\mathcal{B}$, is a finite-dimensional $\mathbb{K}$-vector space). The dimension of the $\mathbb{K}$-vector space $\mathcal{A}$ is 2^n. Moreover, $\mathcal{B}$ is a *reduced Gröbner basis* of $\langle \mathcal{B} \rangle$ with respect to *the degree reverse lexicographic (DRL) order* $b_{n-1} \succ \ldots \succ b_0$.

Proof: According to the *Buchberger criterion* (see, e.g., [4] and the references therein), $\mathcal{B}$ is a *Gröbner basis* with respect to the DRL order $b_{n-1} \succ \ldots \succ b_0$. Moreover, this Gröbner basis is reduced according to [4, ch. 2, §7, Definition 4]. Since *the leading monomial* of each polynomial in the Gröbner basis $\mathcal{B}$ is of the form b_k^2, by [4, ch. 5, §6, Theorem 4], $\mathcal{A}$ is a finite-dimensional $\mathbb{K}$-vector space. Using the notation $b^\alpha := \prod_{i=0}^{n-1} b_i^{\alpha_i}$ where $\alpha := (\alpha_0, \ldots, \alpha_{n-1}) \in \mathbb{N}^n$, a basis of $\mathcal{A}$ is defined by $\{b^\alpha\}_{\alpha \in [\![0,1]\!]^n}$. Finally, the dimension of $\mathcal{A}$ is equal to 2^n. ■

Remark 2: Combining Theorems 2 and 3, we get that the polynomial system $\mathscr{R} = 0$ associated with an ARE of a SISO system (1) is a finite-dimensional $\mathbb{K}$-vector space of dimension 2^n with basis $\{b^\alpha\}_{\alpha \in [\![0,1]\!]^n}$ (see (12)).

IV. Parametrization of the solutions

In Section III, we show that we can reduce the problem of computing the solutions of the ARE equation $\mathscr{R} = 0$ to those of a zero-dimensional system $\mathcal{B}$. This system is given by its Gröbner basis $\mathcal{B}$ with respect to the DRL order. From the end-user point of view, a convenient way to express the solutions of such a system is to use a *Rational Univariate Representation* (RUR) [11], [12], i.e., a one-to-one mapping

$$\begin{array}{ccc} V(\langle \mathcal{B}\rangle) & \longrightarrow & V(\langle p\rangle) \\ b = (b_0, \cdots, b_{n-1}) & \longmapsto & \xi \\ \left(\dfrac{p_0(\xi)}{q(\xi)}, \cdots, \dfrac{p_{n-1}(\xi)}{q(\xi)}\right) & \longleftarrow & \xi, \end{array} \tag{22}$$

between the solutions of the system $\mathcal{B}$ defined by

$$V(\langle \mathcal{B}\rangle) := \left\{ b \in \overline{\mathbb{K}}^n \mid r(b) = 0, \forall\, r \in \langle \mathcal{B}\rangle \right\},$$

and the roots of a univariate polynomial p. In order to achieve the one-to-one condition, the above representation is computed with respect to a *separating element* t, that is a linear combination $t := \sum_{i=0}^{n-1} u_i\, b_i$ of the b_i's, that takes different values when evaluated at different points of $V(\langle \mathcal{B}\rangle)$.

Generically, i.e. for almost any value of the parameters, the system $\mathcal{B}$ admits $t := b_{n-1}$ as a separating element. In that case, the above representation has the following form

$$\begin{cases} p(b_{n-1}) = 0, \\ b_{n-1} = b_{n-1}, \\ b_{n-2} = \dfrac{p_{n-2}(b_{n-1})}{q(b_{n-1})}, \\ \qquad \cdots \\ b_0 = \dfrac{p_0(b_{n-1})}{q(b_{n-1})}, \end{cases} \tag{23}$$

where $p, p_0, \ldots, p_{n-2}, q$ are in $\mathbb{K}[b_{n-1}]$ and p and q are coprime polynomials, i.e., $\gcd(p, q) = 1$.

For instance, given explicit values for the parameters, using (23), we can compute certified numerical approximations of the real solutions by first isolating the real roots of p by means of intervals [13], and then substituting these intervals into the expressions $\frac{p_i(b_{n-1})}{q(b_{n-1})}$ for $i = 0, \ldots, n-2$ to get the corresponding isolating intervals for the b_i's.

From the computation point of view, a RUR of $\mathcal{B}$ can be obtained using the algorithm given in [12]. This algorithm requires the knowledge of a basis of $\mathcal{A} := R/\langle \mathcal{B}\rangle$ (e.g., $\{b^\alpha\}_{\alpha \in [\![0,1]\!]^n}$; see the proof of Theorem 3) and a reduction algorithm which computes normal forms modulo the ideal $\langle \mathcal{B}\rangle$. In order to explicitly characterize the polynomials appearing in the RUR, we first introduce the multiplication $\mathbb{K}$-endomorphism of $\mathcal{A}$ associated with $r \in R$ defined by

$$\begin{array}{rccc} m_r: & \mathcal{A} & \longrightarrow & \mathcal{A} \\ & \overline{a} & \longmapsto & \overline{r\, a}, \end{array}$$

where $\overline{a}$ denotes the residue class of $a \in R$ in $\mathcal{A}$ (i.e., modulo $\langle \mathcal{B}\rangle$). A representative of $\overline{a}$ is the normal form of a with respect to the Gröbner basis $\mathcal{B}$. Using the basis $\{b^\alpha\}_{\alpha \in [\![0,1]\!]^n}$, the $\mathbb{K}$-endomorphism m_r can be defined by means of a matrix $M_r \in \mathbb{K}^{2^n \times 2^n}$. Then, p is the characteristic polynomial of $M_{b_{n-1}}$. Moreover, if

$$\overline{p}(b_{n-1}) := \frac{p}{\gcd\left(p, \frac{dp}{db_{n-1}}\right)} = \sum_{i=0}^{d} v_i\, b_{n-1}^{d-i} \in \mathbb{K}[b_{n-1}]$$

denotes the *square-free part* of p and if we note

$$H_j(b_{n-1}) := \sum_{i=0}^{j} v_i\, b_{n-1}^{j-i} \in \mathbb{K}[b_{n-1}], \quad j = 0, \ldots, d-1,$$

then, for $k = 0, \ldots, n-2$, we can define:

$$\begin{cases} q(b_{n-1}) := \displaystyle\sum_{i=0}^{d-1} \mathrm{Trace}(M_{b_{n-1}}^i)\, H_{d-i-1}(b_{n-1}), \\ p_k(b_{n-1}) := \displaystyle\sum_{i=0}^{d-1} \mathrm{Trace}(M_{b_k}\, M_{b_{n-1}}^i)\, H_{d-i-1}(b_{n-1}). \end{cases}$$

For more details, see [11], [12]. Note that the polynomials p, $p_0, \ldots, p_{n-2}$, q defining a RUR (22) are not unique. The above formulas give us a way to compute the RUR. But the polynomials appearing are usually not the shortest ones. For small orders (i.e., $n < 5$), a RUR can easily be obtained by direct computations yielding simple expressions.

Finally, if b_{n-1} is a separating element and the ideal $\langle \mathcal{B}\rangle$ is *radical* (i.e., $\sqrt{\langle \mathcal{B}\rangle} := \{r \in R \mid \exists\, l \in \mathbb{N} :\ r^l \in \langle \mathcal{B}\rangle\} = \langle \mathcal{B}\rangle$), a similar representation can be computed by performing a change of order for the Gröbner basis (passing from the DRL order to the *lexicographic order* $b_0 \succ \ldots \succ b_{n-1}$) [12]. However, the size of the outputs is then much more larger than the one given by a RUR [12], a fact increasing the computational cost. For instance, on a regular machine (Intel Core i7 2.60 GHz processor and 16 GB of RAM), we can compute the RUR of $\mathcal{B}$ for $n = 6$ with all the parameters a_i's and c_j's in 20 minutes `Maple 18` CPU time. By comparison, a similar computation based on a change of order for the Gröbner basis $\mathcal{B}$ did not finish after 3 days (due to a lack of memory).

Remark 3: If p is square-free (i.e., the discriminant $\mathrm{Disc}(p) \in \mathbb{K}$ of p is not zero), then the ideal $\langle \mathcal{B}\rangle$ is radical and b_{n-1} is a separating element. In a future work, the conditions of non degeneracy will be further studied.

Finally, combining (23) and (13), all the entries $x_{i,j}$ of X can be expressed in terms of $b_{n-1} = x_{n,n} - a_{n-1}$ (see (12)).

Remark 4: Note that the above effective method yields the explicit resolution of an ARE for a general SISO system with parameters. It can be used to study many standard problems (optimal control and estimation problems such as H_∞ or H_2 optimal control, LQ control, model order reduction, Kalman filtering, …) for SISO system with parameters.

Once an explicit expression of the solutions of $\mathscr{R} = 0$ is known (see (23)), we then have to characterize the positive definite one, i.e. X of Theorem 1, among the 2^n solutions.

V. From ARE to spectral factorization

Another way to obtain a solution of $\mathscr{R}$ is by means of the invariant subspaces of the Hamiltonian matrix defined by:

$$\mathscr{H} := \begin{pmatrix} A & -B\,B^T \\ -C^T\,C & -A^T \end{pmatrix} \in \mathbb{K}'^{2n\times 2n}.$$

Using (4), the computation of the characteristic polynomial of $\mathscr{H}$, denoted by $f(\lambda) := \det(\mathscr{H} - \lambda\, I_n)$, gives:

$$f(\lambda) = (-1)^n \sum_{i=0}^{n}\sum_{j=0}^{n} \delta_{i,j} \lambda^{i+j},\ \delta_{i,j} := (-1)^i\,(a_i\,a_j + c_i\,c_j)\,. \tag{24}$$

We have:

$$f(\lambda) := \lambda^{2n} + (-1)^n \sum_{k=0}^{n-1} \left(\sum_{\substack{0\leq i,j\leq n \\ i+j=2\,k}} \delta_{i,j}\,\lambda^{2k} + \sum_{\substack{0\leq i,j\leq n \\ i+j=2\,k+1}} \delta_{i,j}\,\lambda^{2k+1} \right). \tag{25}$$

Note that the coefficient of f of degree $2\,k+1$ in λ vanishes for $k = 0, \ldots n-1$. Moreover, some coefficients of λ^{2k} appear twice and after tedious computation, we obtain

$$f(\lambda) := (-1)^n \sum_{k=0}^{n} (-1)^k\, d_{2k}\,\lambda^{2k}, \tag{26}$$

where d_{2k} is defined in (21). As a consequence, f is an even polynomial, i.e. $f(-\lambda) = f(\lambda)$.

The even property of f leads to the *spectral factorization problem* which consists in finding a polynomial

$$g(\lambda) := \sum_{k=0}^{n} \beta_k\,\lambda^k$$

such that

$$f(\lambda) = (-1)^n\, g(\lambda)\, g(-\lambda), \tag{27}$$

where all the roots of g have negative real parts. We first concentrate on the condition (27). We have:

$$f(\lambda) = (-1)^n \sum_{i=0}^{n}\sum_{j=0}^{n} (-1)^i\,\beta_i\,\beta_j\,\lambda^{i+j}. \tag{28}$$

After similar tedious computation as the ones done to get (26) , using the change of variable $i := k-m$ and $j := k+m$, we obtain

$$f(\lambda) = \sum_{k=0}^{n} (-1)^{n+k} \left(\beta_k^2 + 2 \sum_{m=1}^{M(k)} (-1)^m \beta_{k-m}\,\beta_{k+m} \right) \lambda^{2k}, \tag{29}$$

where $M(k)$ is defined by (18). Combining (26) and (29), we finally obtain

$$\begin{cases} \beta_0^2 = d_0, \\ {\beta_k}^2 + 2 \sum_{m=1}^{M(k)} (-1)^m\,\beta_{k-m}\,\beta_{k+m} = d_{2k}, \end{cases}$$

for $1 \leq k \leq n-1$, which coincides with (20) with $\beta_k = b_k$ for $k = 0, \ldots, n-1$.

Let us now characterize $b_{n-1} = \beta_{n-1}$ so that all the roots of g have negative real parts, or equivalently characterize $b_{n-1} = \beta_{n-1}$ which defines the unique positive definite solution X of (6). In the sequel, *the parameters are fixed to explicit values*. Since f is even, if λ is a solution of f, then so is $-\lambda$. Let $\{\alpha_i\}_{i=1,\ldots,n}$ be the roots of f with negative real parts (see (26)). We want to determine $b_{n-1} = \beta_{n-1}$, and thus all the $\beta_i = b_i$'s by (23) such that all the α_i's are roots of g. Given g satisfying (27), then α_i is either a root of $g(\lambda) = 0$ or of $g(-\lambda) = 0$. Since $\beta_n = 1$, $-\beta_{n-1}$ is a sum of the roots of g, we get $\beta_{n-1} = -\sum_{i=1}^{n} \epsilon_i\,\alpha_i$ where $\epsilon_i \in \{-1,\,1\}$. If $\{\nu_k\}_{k=1,\ldots,2^n}$ is the set of complex solutions of $\mathcal{B}$, where $\nu_k := (\nu_{k,0}, \ldots, \nu_{k,n-1})$, and $\nu_{k,n-1}$ is the n^{th} coordinate of ν_k, using $\beta_{n-1} = b_{n-1}$, then we get $\nu_{k,n-1} = -\sum_{i=1}^{n} \epsilon_{k,i}\,\alpha_i$ for $= 1, \ldots, 2^n$, where $\epsilon_{k,i} \in \{-1,\,1\}$. Moreover, we have

$$\sigma := -\sum_{i=1}^{n} \alpha_i = \max_{k\in\{1,\ldots,2^n\}} \nu_{k,n-1}, \tag{30}$$

For more details, see [10]. We obtain the following theorem.

Theorem 4 ([10]): The positive definite solution X of $\mathscr{R} = 0$ corresponds to the real solution ν_k of $\mathcal{B}$ such that $b_{n-1}(\nu_k)$ is maximal.

Proposition 2: We have $b_0 = \sqrt{a_0^2 + c_0^2}$.

Proof: Using (19) and (21), we have $b_0^2 = a_0^2 + c_0^2$. We also note that we have

$$b_0 = \beta_0 = (-1)^n \prod_{i=1}^{n} \lambda_i,$$

where λ_i's are roots of g with negative real parts. Let us distinguish the following two cases:

- If n is even then $b_0 = (-1)^n \prod_{i=1}^{n/2} \lambda_i\,\overline{\lambda_i} = \prod_{i=1}^{n/2} |\lambda_i|^2 > 0$.
- If n is odd then we have
$$b_0 = (-1)^n\,\lambda_0 \prod_{i=1}^{(n-1)/2} \lambda_i\,\overline{\lambda_i} = (-1)\,\lambda_0 \prod_{i=1}^{(n-1)/2} |\lambda_i|^2,$$
where $\lambda_0 < 0$ so that $b_0 > 0$ and concludes the proof.

■

VI. Certified computation of γ_{opt}

As stated in Theorem 1, an optimal solution of the RCP is given by (10). Using the RUR (23) of $\mathcal{B}$ and (13), we can express the entries $x_{i,j}$ (resp., $y_{i,j}$) of X (resp., Y) in terms of $b_{n-1} = x_{n,n} - a_{n-1}$, and then we compute the characteristic polynomial of QX:

$$h(\lambda, b_{n-1}) := \det(QX - \lambda\, I_n).$$

The numerator of h, denoted by $h_\star$, belongs to $\mathbb{K}'[\lambda, b_{n-1}]$. Hence, the problem of computing γ_{opt} is equivalent to that of the maximal real solution of $h_\star(\lambda, \sigma) = 0$, where σ is defined by (30) (i.e., the maximal real root of p). We note again that σ defines the solutions X and Y.

When explicit values of the parameters are given, this problem can be tackled using purely numerical methods.

For instance, one approach consists in first isolating the root σ of p by means of an interval, substituting this interval into $h_\star(\lambda, b_{n-1})$ and isolating again in order to find $\lambda_\star$ (see Corollary 1). But such a numerical method fails to provide a *certified output* (see [5], [13]), especially when $h_\star(\lambda, \sigma)$ is not square-free in λ. In order to achieve a certification, we can instead consider the following *triangular system*

$$\begin{cases} p(b_{n-1}) = 0, \\ h_\star(\lambda, b_{n-1}) = 0, \end{cases} \tag{31}$$

and compute a certified numerical approximation of the solution $(\sigma, \lambda_\star)$ [5], [13].

There exist several exact methods for solving (31) and thus for deducing the desired solution $(\sigma, \lambda_\star)$. We can mention the algorithm of [3] which uses some *separation bounds* in order to isolate the roots of the polynomial $h_\star(\lambda, b_{n-1})$ over a root of p, or the algorithm given in [2] which requires the computation of a decomposition of (31) into a set of square-free triangular systems so that for any root α of p the polynomial $h_\star(\lambda, \alpha)$ is square-free.

We now use again the idea developed in Section IV and compute a RUR of the solutions of (31). For a well chosen separating element $\tau := u_1\, b_{n-1} + u_2\, \lambda$, these solutions can be expressed by:

$$\begin{cases} v(\tau) = 0, \\ b_{n-1} = \dfrac{v_1(\tau)}{w(\tau)}, \\ \lambda = \dfrac{v_2(\tau)}{w(\tau)}. \end{cases} \tag{32}$$

Finally, in order to obtain the solution $(\sigma, \lambda_\star)$, it is sufficient to isolate the real roots of v, evaluate the resulting intervals in the expressions $\frac{v_1(\tau)}{w(\tau)}$ and $\frac{v_2(\tau)}{w(\tau)}$, which yields boxes isolating the solutions, and finally choose the one that satisfies the requirement (i.e., the maximal λ obtained from the maximal b_{n-1}).

Getting $\lambda_\star$, we deduce $\gamma_{\text{opt}} := \sqrt{1+\lambda_\star^2}$, which allows us to compute an optimal H_∞ controller by Theorem 1.

Finally, we recapitulate the above approach for solving the RCP by outlining an algorithm.

Algorithm 1:

1) Using (20) and (21), compute $\mathcal{B}$.
2) Compute a RUR (23) of the ideal $\langle \mathcal{B} \rangle$.
3) Using Theorem 2, compute X in terms of b_{n-1}.
4) Compute Q defined by (9).
5) Compute $Q\,X$ and the numerator of its characteristic polynomial $h_\star$.
6) Compute a RUR of (31) to get (32).
7) Isolate the solutions of (32) by means of 2D-boxes and select the desired box corresponding to $(\sigma, \lambda_\star)$.
8) Compute $\gamma_{\text{opt}} := \sqrt{1+\lambda_\star^2}$.
9) Compute $Y := Q\,X\,Q$.
10) For $\gamma > \gamma_{\text{opt}}$, compute the controller K_γ, defined by (8), which satisfies the RCP.

VII. EXAMPLE: A TWO-MASS-SPRING SYSTEM

We illustrate the above approach with the two-mass-spring problem (Figure 2) considered in [15], [14], which mathematical model is similar to the one of a 2-dimensional gyro-stabilized sight [1].

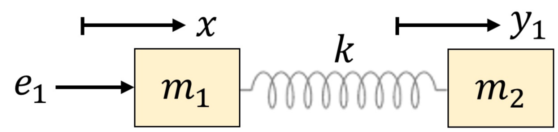

Fig. 2. Two-mass-spring system

Two masses m_1 and m_2 are linked by a spring of stiffness k. With the notations of Figure 1, we study the displacement of m_2, denoted by y_1, while m_1 is excited by a force e_1. We consider the transfer function of the *physical plant*

$$P := \frac{c}{s^2\,(s^2 + a_2)}, \;\; c := \frac{k}{m_1\, m_2}, \;\; a_2 := \frac{m_1 + m_2}{m_1\, m_2}\, k,$$

between the input e_1 and the output y_1. As in [14, §2.6, §4], we consider a weight $w > 0$ and define the *fictive plant* by

$$G := w\, P = \frac{c_0}{s^2\,(s^2 + a_2)}, \quad c_0 := w\, c > 0, \quad a_2 > 0,$$

which will be used in Algorithm 1. As seen above, we can design the loop-shaping controller K_γ stabilizing G. From this controller K_γ, we then get the robust stabilizing controller $C_\gamma := w\, K_\gamma$ of the physical plant P.

For the system G, the polynomials $\mathcal{B}_k$'s defined in Theorem 3, are given by

$$\begin{cases} \mathcal{B}_0 := b_0^2 - d_0 = 0, \\ \mathcal{B}_1 := b_1^2 - 2\, b_0\, b_2 - d_2 = 0, \\ \mathcal{B}_2 := b_2^2 - 2\, b_1\, b_3 + 2\, b_0 - d_4 = 0, \\ \mathcal{B}_3 := b_3^2 - 2\, b_2 - d_6 = 0, \end{cases} \tag{33}$$

where the constants d_{2k}'s are defined by:

$$d_0 := c_0^2, \quad d_2 := 0, \quad d_4 := a_2^2, \quad d_6 := -2\, a_2.$$

Using Proposition 2, we get $b_0 = c_0$. Moreover, if $b_3 = 0$, then the last equation of (33) yields $b_2 = a_2$, and thus the last but one gives $b_0 = c_0 = 0$, i.e. $G = 0$. Since in the sequel we suppose that $G \neq 0$, and thus, $b_3 \neq 0$, a RUR of the system $\{\mathcal{B}_1 = 0, \mathcal{B}_2 = 0, \mathcal{B}_3 = 0\}$ for the separating element b_3 can easily be obtained by direct computations

$$\begin{cases} b_1 = \dfrac{b_3^4 + 4\, a_2\, b_3^2 + 8\, c_0}{8\, b_3}, \\ b_2 = \frac{1}{2}\, b_3^2 + a_2, \\ b_3 = b_3, \end{cases} \tag{34}$$

where b_3 satisfies the following polynomial equation:

$$p(b_3) := b_3^8 + 8\, a_2\, b_3^6 + 16\,(a_2^2 - 3\, c_0)\, b_3^4 - 64\, a_2\, c_0\, b_3^2 + 64\, c_0^2 = 0. \tag{35}$$

[1] A gyro-stabilized sight is a system of cameras controlled in rotation by motors and gyroscopes.

Noting $\tau = b_3^2/\sqrt{8\,c_0}$, we obtain:

$$c_0\,\tau^4 + \sqrt{8\,c_0}\,a_2\,\tau^3 + 2\,(a_2^2 - 3\,c_0)\,\tau^2 - \sqrt{8\,c_0}\,a_2\,\tau + c_0 = 0. \quad (36)$$

The latter polynomial is anti-palindromic, so we can apply the change of variable $\varsigma := \tau - \tau^{-1}$ in order to get:

$$p_s(\varsigma) := c_0\,\varsigma^2 + 2\sqrt{2\,c_0}\,a_2\,\varsigma + 2\,a_2^2 - 4\,c_0 = 0. \quad (37)$$

The discriminant of (37) is $16\,c_0^2 \neq 0$. Thus, p_s has two distinct real roots, which are defined by:

$$\varsigma_+ := -\sqrt{\frac{2}{c_0}}\,a_2 + 2, \quad \varsigma_- := -\sqrt{\frac{2}{c_0}}\,a_2 - 2.$$

Since $a_2 > 0$, we have $\varsigma_- < 0$, $\varsigma_+ > \varsigma_-$, $|\varsigma_-| > |\varsigma_+|$. Then, τ verifies $\tau^2 - \tau\,\varsigma - 1 = 0$ which discriminant is $\varsigma^2 + 4 > 0$. Its solutions are given by:

$$\begin{cases} \tau_+(\varsigma_\pm) := \dfrac{1}{2}\left(\varsigma_\pm + \sqrt{\varsigma_\pm^2 + 4}\right), \\ \tau_-(\varsigma_\pm) := \dfrac{1}{2}\left(\varsigma_\pm - \sqrt{\varsigma_\pm^2 + 4}\right), \end{cases} \quad \varsigma_\pm \in \{\varsigma_+, \varsigma_-\}.$$

Then all the roots of (36) are real. It is clear that $\tau_+(\varsigma_+)$ is the maximal real root of (36), which leads to the following expression of σ, maximal real root of p:

$$\sigma := \sqrt{2}\sqrt{\left(\sqrt{2\,c_0} - a_2\right) + \sqrt{\left(\sqrt{2\,c_0} - a_2\right)^2 + 2\,c_0}}. \quad (38)$$

Using Theorem 2 (see (13)), the matrix X is defined by

$$X := \begin{pmatrix} c_0\,b_1 & c_0\,b_2 & c_0\,b_3 & c_0 \\ c_0\,b_2 & b_1\,b_2 - c_0\,b_3 & b_1\,b_3 - c_0 & b_1 \\ c_0\,b_3 & b_1\,b_3 - c_0 & b_2\,b_3 - b_1 & b_2 - a_2 \\ c_0 & b_1 & b_2 - a_2 & b_3 \end{pmatrix},$$

where b_1 and b_2 are defined by (34). By (9), we have:

$$\mathscr{Q} := \begin{pmatrix} 0 & 0 & 0 & c_0^{-1} \\ 0 & 0 & c_0^{-1} & 0 \\ 0 & c_0^{-1} & 0 & -a_2\,c_0^{-1} \\ c_0^{-1} & 0 & -a_2\,c_0^{-1} & 0 \end{pmatrix}.$$

The numerator $h_\star$ of the characteristic polynomial of $\mathscr{Q}\mathscr{X}$ divided by c_0^2, i.e., $h_s := h_\star\,c_0^{-2}$, is defined by:

$$h_s(\lambda, b_3) = 4\,c_0\,\lambda^4 - \alpha(b_3)\,\lambda^3 + \beta(b_3)\,\lambda^2 + \alpha(b_3)\,\lambda + 4\,c_0, \quad (39)$$

$$\alpha := b_3^4 + 8\,c_0, \quad \beta := -2\,(b_3^4 + 4\,c_0).$$

Since h_s is *anti-palindromic*, using $t := \lambda - \lambda^{-1}$ ($\lambda = 0$ is not a root of h_s since its constant term $4\,c_0$ is non-zero), h_s can then be rewritten as:

$$h_s(t, b_3) = 4\,c_0\,t^2 - \alpha(b_3)\,t - 2\,b_3^4. \quad (40)$$

The real roots of (40) are then given by:

$$\begin{cases} t_+(b_3) := \dfrac{1}{8\,c_0}\left(\alpha(b_3) + \sqrt{\alpha(b_3)^2 + 32\,c_0\,b_3^4}\right), \\ t_-(b_3) := \dfrac{1}{8\,c_0}\left(\alpha(b_3) - \sqrt{\alpha(b_3)^2 + 32\,c_0\,b_3^4}\right). \end{cases}$$

The roots of $h_\star$ are then of the form:

$$\begin{cases} \lambda_+(t_\pm) := \dfrac{1}{2}\left(t_\pm + \sqrt{t_\pm^2 + 4}\right), \\ \lambda_-(t_\pm) := \dfrac{1}{2}\left(t_\pm - \sqrt{t_\pm^2 + 4}\right), \end{cases} \quad t_\pm \in \{t_+, t_-\}.$$

Let us determine the root of $h_\star$ of greatest absolute value. Since $t_+\,t_- = -\frac{2\,b_3^4}{4\,c_0} < 0$ and $t_+ > t_-$, t_- is necessarily negative while t_+ is positive. Furthermore, since $t_+ + t_- = \frac{\alpha(b_3)}{4\,c_0} > 0$, then we get $t_+ = |t_+| > -t_-$, i.e., $|t_+| > |t_-|$. The root of $h_\star$ of greatest absolute value is given by:

$$\lambda_\star := \lambda_+(t_+) = \frac{1}{2}\left(t_+ + \sqrt{t_+^2 + 4}\right). \quad (41)$$

Hence, using (38), the minimal γ is given by:

$$\gamma_{\mathrm{opt}}(\sigma) := \sqrt{1 + \left(\frac{t_+(\sigma)}{2} + \sqrt{1 + \left(\frac{t_+(\sigma)}{2}\right)^2}\right)^2}. \quad (42)$$

Remark 5: (42) is also valid for $a_2 < 0$.

Using the explicit formula (8) given in Theorem 1, we can compute a robust stabilizing controller K_γ of G and, since $K_\gamma\,G = C_\gamma\,P$, we deduce a stabilizing controller C_γ of P satisfying $\|S\|_\infty < \gamma$ and $\|G\,K_\gamma\,S\|_\infty < \gamma$ [14, Cor. 5.1]. Figure 3 shows the plot of the function γ_{opt} in terms of the values of the parameters a_2 and c_0. We note that we have:

$$\gamma_{\inf} = \inf_{a_2 > 0,\, c_0 > 0} \gamma_{\mathrm{opt}} = \sqrt{4 + 2\sqrt{2}} \approx 2.6131.$$

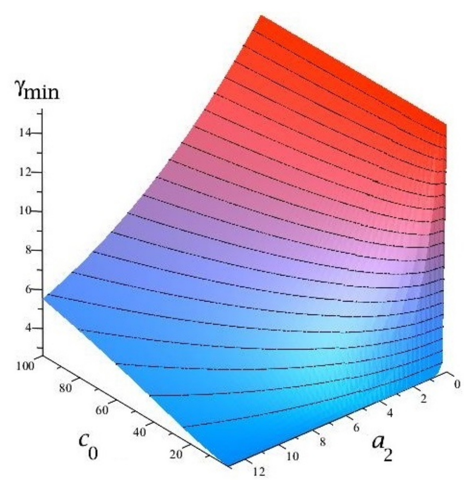

Fig. 3. $\gamma_{\mathrm{opt}} = \Gamma(a_2, c_0)$

The weight w can be seen as a scalar tuning parameter for γ_{opt}. As a consequence, for $\gamma > \gamma_{\mathrm{opt}}$, we can obtain an auto-tuned controller $C_\gamma(c, a_2) = K_\gamma(c, a_2)/w$. By on-line identification of the modal frequency a_2 and of the gain c, the controller $C_\gamma(c, a_2)$ can be computed in real-time by a simple embedded calculator (which does not have to contain optimization routines).

According to [14, Th.2.10], we have the following gain and phase margins:

$$\left\{\begin{array}{l}\Delta_{\mathrm{G}}(P,C_\gamma)=\Delta_{\mathrm{G}}(G,K_\gamma)\geq \dfrac{1+\gamma^{-1}}{1-\gamma^{-1}},\\ \Delta_{\Phi}(P,C_\gamma)=\Delta_{\Phi}(G,K_\gamma)\geq 2\arcsin\left(\gamma^{-1}\right).\end{array}\right.$$

For $\gamma_{\mathrm{opt}}=3$, with $\gamma_{\mathrm{opt}}>\gamma_{\inf}$, we have

$$\left\{\begin{array}{l}\Delta_{\mathrm{G}}(P,C_{\gamma_{\mathrm{opt}}})=\Delta_{\mathrm{G}}(G,K_{\gamma_{\mathrm{opt}}})\geq 6\,\mathrm{dB},\\ \Delta_{\Phi}(P,C_{\gamma_{\mathrm{opt}}})=\Delta_{\Phi}(G,K_{\gamma_{\mathrm{opt}}})\geq 39^\circ,\end{array}\right.$$

and to achieve $\gamma_{\mathrm{opt}}=3$, w can be chosen as follows:

$$w=952\,\frac{6561\sqrt{2}-8\sqrt{7}\sqrt{223074\sqrt{2}-129472}}{72048449-49968576\sqrt{2}}\,\frac{a_2^2}{c_0}.$$

Remark 6: Since we usually cannot solve by radicals univariate polynomials of degree larger than or equal to 4, we have to use the symbolic-numeric method described in Algorithm 1. Let us show how to compute the second RUR at Step 6 of Algorithm 1 for the above example. We can first compute a RUR of the polynomial system defined by p and h_s given by (35) and (40) with $T=b_3^2$, i.e.:

$$\left\{\begin{array}{l}T^4+8\,a_2\,T^3+16\,(a_2^2-3\,c_0)\,T^2-64\,a_2\,c_0\,T+64\,c_0^2=0,\\ 4\,c_0\,t^2-\left(T^2+8\,c_0\right)t-2\,T^2=0.\end{array}\right. \tag{43}$$

After some eliminations, we obtain $T=\delta(t)\,\theta^{-1}(t)$, where

$$\left\{\begin{array}{rcl}\delta(t)&=&-c_0\,t^4+\left(-4\,a_2^2+16\,c_0\right)t^3-8\,c_0\,t^2\\ &&+16\,\left(a_2^2-4\,c_0\right)\,t-16\,c_0,\\ \theta(t)&=&2\,a_2\left(t^3-2\,t^2-12\,t-8\right),\end{array}\right. \tag{44}$$

and t satisfies the following polynomial equation:

$$\begin{array}{l}c_0^2\,t^8-8\,c_0\,(a_2^2+4\,c_0)\,t^7+16\,(a_2^4+17\,c_0^2)\,t^6\\ -32\,c_0\,(a_2^2+4\,c_0)\,t^5-32\,(4\,a_2^4+61\,c_0^2)\,t^4\\ +128\,c_0\,(a_2^2+4\,c_0)\,t^3+256\,(a_2^4+17\,c_0^2)\,t^2\\ +512\,c_0\,(a_2^2+4\,c_0)\,t+256\,c_0^2=0.\end{array} \tag{45}$$

Substituting $t=\lambda-\lambda^{-1}$ into (44) and (45), we obtain a RUR for the polynomial system defined by (35) and (39). For explicit values of a_2 and c_0, root isolation techniques can then be used to compute the H_∞ controller.

For instance, for $a_2=10$ and $c_0=1$, the above computations can be directly obtained by applying `Maple` command `RootFinding[Isolate]` to (43) with options `method = "RS"`, `output=interval`. In our example, we obtain:

$$\begin{array}{l}\sigma=[0.481024117223, 0.481024117224],\\ \lambda_\star=[2.436943741649, 2.436943741650].\end{array}$$

VIII. Conclusion

In this paper, we have initiated a new symbolic-numeric method for solving the H_∞ loop-shaping design problem for low order SISO systems with parameters. To do that, we first showed that solving an ARE of size $n\times n$ is equivalent to solving a certain zero-dimensional polynomial system of n equations in n unknowns, and that a RUR can be used to parametrize all its real solutions. Then, using a result of spectral factorization [10], we showed how to solve an H_2 control problem. The computation of the controller satisfying the robust 4-block problem requires a second RUR computation and symbolic/numeric root isolation techniques.

When the univariate polynomials of the two RUR can be reduced to polynomials of degrees less than or equal to 4, then an explicit formula of the H_∞ controller can be obtained. Then, the dependence of this controller to the system parameters can then be analytically studied, which is useful, e.g., in the applications to gyro-stabilized sights. For higher order systems, a certified numerical root isolation technique has to be used to compute the H_∞ controller.

Further works include the search for tractable RURs in order to deal with highest order systems. Furthermore, our approach will be applied to small order models of gyro-stabilized sights to develop an adaptive control scheme.

References

[1] H. Anai, S. Hara, M. Kanno, K. Yokoyama. Parametric polynomial factorization using the sum of roots and its application to a control design problem. *J. Symb. Comp.*, 44 (2009), 703–725.

[2] F. Boulier, C. Chen, Changbo, F. Lemaire, M. Moreno Maza. Real root isolation of regular chains. *Asian Symposium on Computer Mathematics*, R. Feng et al. (eds.), Springer, 2009, 33–48.

[3] Jin-San Cheng, Xiao-Shan Gao, Chee-Keng Yap. Complete numerical isolation of real roots in zero-dimensional triangular systems. *J. Symb. Comp.*, 44 (2009), 768–785.

[4] D. Cox, J. Little, D. O'Shea. Varieties, and Algorithms *Springer*, 2015.

[5] G. Collins, A. Akritas. Polynomial real roots isolation using Descartes' rule of signs. *SYMSAC*, 1976, 272–275.

[6] G. Duc, S. Font. Commande H_∞ et μ-analyse, des outils pour la robustesse. *Hermes*, 1999.

[7] K. Forsman, J. Eriksson. Solving the ARE symbolically. 1993.

[8] K. Glover, D. C. McFarlane. Robust stabilization of normalized coprime factor plant descriptions with H_∞-bounded uncertainty. *IEEE Trans. Automat. Contr.*, 34 (1989), 821–830.

[9] M. Kanno, S. Hara. Symbolic-numeric hybrid optimization for plant/controller integrated design in H_∞ loop-shaping design. *Journal of Math-for-Industry*, 4 (2012), 135–140.

[10] M. Kanno, K. Yokoyama, H. Hanai, S. Hara. Solution of algebraic Riccati equations using the sum of roots. *ISSAC 2009*, 2009, 215–222.

[11] F. Rouillier. Solving zero-dimensional systems through the Rational Univariate Representation. *Applicable Algebra in Engineering Communication and Computing*, 9 (1999), 433–461.

[12] F. Rouillier. Algorithmes pour l'étude des solutions réelles des systèmes polynomiaux. *Habilitation thesis*, 2007.

[13] F. Rouillier, P. Zimmerman. Efficient isolation of polynomial's real roots. *J. Comput. Applied Mathematics*, 162 (2004), 33–50.

[14] G. Vinnicombe. Uncertainty and Feedback $-$ H_∞ Loop-shaping and the ν-gap metric. *Imperial College Press*, 2001.

[15] B. Wie, D. S. Bernstein. A benchmark problem for robust control design. *J. Guidance, Control, and Dynamics*, 15, (1992), 1057–1059.

[16] K. Zhou, J.C. Doyle, K. Glover. Robust and Optimal Control. *Prentice-Hall*, 1996.