



## A new method for the computation of all stabilizing controllers of a given order

K. Saadaoui & A. B. Özgüler

To cite this article: K. Saadaoui & A. B. Özgüler (2005) A new method for the computation of all stabilizing controllers of a given order, *International Journal of Control*, 78:1, 14-28, DOI: [10.1080/00207170412331332506](https://doi.org/10.1080/00207170412331332506)

To link to this article: <http://dx.doi.org/10.1080/00207170412331332506>



Published online: 19 Feb 2007.



[Submit your article to this journal](#)



Article views: 89



[View related articles](#)



Citing articles: 18 [View citing articles](#)

# A new method for the computation of all stabilizing controllers of a given order

K. SAADAoui and A. B. ÖZGÜLER\*

Department of Electrical and Electronics Engineering, Bilkent University, Bilkent, Ankara, TR-06533, Turkey

(Received 12 September 2003; revised 28 October 2004; accepted 1 November 2004)

A new method is given for computing the set of all stabilizing controllers of a given order for linear, time invariant, scalar plants. The method is based on a generalized Hermite–Biehler theorem and the successive application of a modified constant gain stabilizing algorithm to subsidiary plants. It is applicable to both continuous and discrete time systems.

## 1. Introduction

An analytic method of determining the set of all stabilizing constant gains for linear, time invariant, scalar plants was derived in Özgüler and Koçan (1994) for continuous-time systems. The solution was based on a generalization of Hermite–Biehler theorem to the case of signature computation. The main advantage of the method in comparison with other analytic methods such as D-decomposition or Routh–Hurwitz criterion based methods is that it replaces the (finite number of) checks for stability required in such methods with a certain check of sign sequences.

In Datta *et al.* (2000), a computational characterization of all stabilizing proportional-integral (PI) and proportional-integral-derivative (PID) controllers was derived. This method is also based on the results reported in Özgüler and Koçan (1994), (see Brualdi 2000). The computational method of Datta *et al.* (2000) has been extended to compute all stabilizing PID gains for discrete-time systems in Xu *et al.* (2001). Alternatively, in Munro and Söylemez (2000) and Söylemez *et al.* (2003) the limiting values of proportional, derivative and integral action terms of the set of stabilizing PID controllers are calculated using a Nyquist plot based approach. Because of the structural differences between PID and first-order controllers, direct application of these methods to first-order controllers is not possible although in both types

of controllers only three parameters are involved. The quest for an analytic design method for first-order controllers (phase-lead, phase-lag) has been around for decades. Many classical control textbooks such as Phillips and Harbor (2000) and Dorf and Bishop (2001), contain attempts to deductively obtain a first-order stabilizing controller. In Phillips and Harbor (2000), for example, an analytic method for designing a first-order controller is suggested although the authors emphasize that the design is not guaranteed to succeed and it may lead to an unstable system. In this paper, we solve the problem of determining the set of all stabilizing controllers of a given degree for an arbitrary plant. We will solve the problem for first-order and second-order controllers and show how to extend the algorithm to higher-order controllers. The method developed is based on the application of a modified proportional controller algorithm to a number of auxiliary plants.

There are several classical solutions to the problem of finding the set of all stabilizing proportional controllers, i.e. given coprime polynomials  $q(s)$  and  $p(s)$  with real coefficients, determine the set of all  $\alpha$  such that  $\phi(s, \alpha) = q(s) + \alpha p(s)$  has degree in  $s$  equal to the degree of  $q$  and is Hurwitz stable. However, extensions of these methods to higher order controllers is not obvious.

(i) *Root-locus method*: This is the most widely used graphical solution to the problem of finding the set of all stabilizing proportional controllers. However, as the order of the controller increases the number

\*Corresponding author. Email: ozguler@ee.bilkent.edu.tr

of parameters increases accordingly. Hence, it is difficult to use this method to solve the problem at hand.

- (ii) *Routh–Hurwitz criterion*: With a first-order controller, an example can show that solving the problem with this method is very difficult because we have to solve a highly non-linear set of inequalities.
- (iii) *Neimark D-decomposition*: First let us briefly describe this method (Neimark 1999). Let

$$q(j\omega) = \tilde{h}(\omega) + j\omega\tilde{g}(\omega), \quad p(j\omega) = \tilde{f}(\omega) + j\omega\tilde{e}(\omega)$$

where  $\tilde{h}, \tilde{g}, \tilde{f}$ , and  $\tilde{e}$  are real and even polynomials of  $\omega$ . Then,  $\phi(j\omega, \alpha) = \tilde{h}(\omega) + \alpha\tilde{f}(\omega) + j\omega[\tilde{g}(\omega) + \alpha\tilde{e}(\omega)]$ . If  $\phi(s, \alpha)$  has a  $j\omega$ -axis zero, then as  $\alpha$  is real,  $\tilde{h}(\omega) + \alpha\tilde{f}(\omega) = 0$  and  $\tilde{g}(\omega) + \alpha\tilde{e}(\omega) = 0$ . Eliminating  $\alpha$  from these two equalities, we have

$$\omega[\tilde{g}(\omega)\tilde{f}(\omega) - \tilde{h}(\omega)\tilde{e}(\omega)] = 0. \quad (1)$$

Consequently, if  $\phi(s, \alpha)$  has a  $j\omega$ -axis zero, then (1) holds for some  $\omega \in [0, \infty)$ . Let the roots in  $\omega \in [0, \infty)$  of equation (1) be  $\omega_i, i = 1, \dots, \tilde{k}$  and define

$$\alpha_i = \begin{cases} -\tilde{h}(\omega_i)/\tilde{g}(\omega_i) & \text{if } \tilde{f}(\omega_i) \neq 0 \\ -\tilde{g}(\omega_i)/\tilde{e}(\omega_i) & \text{if } \omega_i\tilde{e}(\omega_i) \neq 0. \end{cases} \quad (2)$$

If  $\tilde{f}(\omega_i) = 0$  and  $\omega_i\tilde{e}(\omega_i) = 0$ , then let  $\alpha_i = \infty$ . The values  $\alpha_i$  so defined satisfy  $\phi(j\omega_i, \alpha_i) = 0$  for  $i = 1, \dots, \tilde{k}$ . We have so far shown that  $\phi(s, \alpha)$  has a  $j\omega$ -axis zero for some  $\alpha$  if and only if  $\alpha \in \{\alpha_i, i = 1, \dots, \tilde{k}\}$ . By the continuity of the roots of  $\phi(s, \alpha)$  with respect to  $\alpha$ , the following description of the solution set is immediate: Let  $\{\omega_i\}$  be the roots in  $[0, \infty)$  of equation (1) and let  $\{\alpha_i\}$  be as defined in equation (2). Let the distinct values of  $\alpha_i, i = 1, \dots, \tilde{k}$  be ordered as

$$\infty > \alpha_{i_1} > \dots > \alpha_{i_{\tilde{k}}} > -\infty$$

and let  $\alpha_{i_0} := \infty$  and  $\alpha_{i_{\tilde{k}+1}} := -\infty$  for convenience. Then for  $l = 1, \dots, \tilde{k}$  the interval  $(\alpha_{i_l}, \alpha_{i_{l+1}})$  is in the solution set if and only if at one point  $\alpha$  in  $(\alpha_{i_l}, \alpha_{i_{l+1}})$  the polynomial  $\phi(s, \alpha)$  is Hurwitz stable. Since the union of all candidate intervals cover  $\mathbf{R}$ , this is a complete description of the solution set. Thus the method requires the determination of roots of equation (1),  $\alpha_i$  and at most  $k + 1$  applications of some stability criterion such as Routh or Hurwitz at the interior point of each interval.

Since the number of parameters increases for a higher-order controller, a direct application of this method to determine higher order controllers is not obvious.

The paper is organized as follows. In §2, some preliminary results are presented. In §3, an improved proportional controller algorithm is given. The algorithm is comparable with the one given in Munro *et al.* (1999) and offers several advantages over the ones given in Özgüler and Koçan (1994) and Datta *et al.* (2000). An algorithm for determining stabilizing first-order controllers is presented in §4. It is then applied to plants with interval type uncertainties in §5. In §6, we give an algorithm for the computation of second-order controllers and show how to extend this algorithm to higher-order controllers. Finally, §7 contains some concluding remarks.

## 2. Preliminaries

Given a set of polynomials  $\psi_1, \dots, \psi_k \in \mathbf{R}[s]$  not all zero and  $k > 1$ , their *greatest common divisor* (with highest coefficient 1) is unique and it is denoted by  $\gcd, \{\psi_1, \dots, \psi_k\}$ . If  $\gcd, \{\psi_1, \dots, \psi_k\} = 1$ , then we say  $(\psi_1, \dots, \psi_k)$  is *coprime*. The derivative of  $\psi$  is denoted by  $\psi'$ . Let  $\mathbf{C}$  denote the set of complex numbers and let  $\mathbf{C}_-, \mathbf{C}_0$  and  $\mathbf{C}_+$  denote the points in the open left half,  $j\omega$ -axis and the open right half of the complex plane, respectively. Then, the set  $\mathcal{H}$  of Hurwitz stable polynomials are  $\mathcal{H} = \{\psi(s) \in \mathbf{R}[s]: \psi(s) = 0 \Rightarrow s \in \mathbf{C}_-\}$ . The *signature*  $\sigma(\psi)$  of a polynomial  $\psi \in \mathbf{R}[s]$  is the difference between the number of its  $\mathbf{C}_-$  roots and  $\mathbf{C}_+$  roots. Given  $\psi \in \mathbf{R}[s]$ , the *even–odd components*  $(a, b)$  of  $\psi(s)$  are the unique polynomials  $a, b \in \mathbf{R}[u]$  such that  $\psi(s) = a(s^2) + sb(s^2)$ . It is possible to state a necessary and sufficient condition for the Hurwitz stability of  $\psi$  in terms of its even–odd components  $(a, b)$ . This result is known as the Hermite–Biehler theorem stated in Proposition 1 below in a slightly modified form. Let us define the *signum function*  $\mathcal{S}: \mathbf{R} \rightarrow \{-1, 0, 1\}$  by

$$\mathcal{S}r = \begin{cases} -1 & \text{if } r < 0 \\ 0 & \text{if } r = 0 \\ 1 & \text{if } r > 0. \end{cases}$$

**Proposition 1** (Gantmacher 1959, §XV, 14): *A non-zero polynomial  $\psi \in \mathbf{R}[s]$  is Hurwitz stable if and only if its even–odd components  $(a, b)$  are such that  $b \neq 0$  and at the distinct real negative roots  $v_1 > v_2 > \dots > v_k$  of  $b$*

the following holds

$$\deg \psi = \begin{cases} Sb(0)[Sa(0) - 2Sa(v_1) \\ \quad 7 + 2Sa(v_2) + \cdots + (-1)^k 2Sa(v_k)], \deg \psi \text{ odd} \\ Sb(0)[Sa(0) - 2Sa(v_1) + 2Sa(v_2) + \cdots + \\ \quad (-1)^{k+1} Sa(-\infty)], \deg \psi \text{ even.} \end{cases} \quad (3)$$

The following is a generalization of Proposition 1 to not necessarily Hurwitz stable polynomials.

**Lemma 1** (Özgüler and Koçan 1994): *Let a non-zero polynomial  $\psi \in \mathbf{R}[s]$  have the even-odd components  $(a, b)$ . Suppose  $b \neq 0$  and  $(a, b)$  is coprime. Then,  $\sigma(\psi) = r$  if and only if at the real negative roots of odd multiplicities  $v_1 > v_2 > \cdots > v_k$  of  $b$  the following holds*

$$r = \begin{cases} Sb(0_-)[Sa(0) - 2Sa(v_1) \\ \quad + 2Sa(v_2) + \cdots + (-1)^k 2Sa(v_k)], \deg \psi \text{ odd} \\ Sb(0_-)[Sa(0) - 2Sa(v_1) \\ \quad + 2Sa(v_2) + \cdots + (-1)^{k+1} Sa(-\infty)], \deg \psi \text{ even} \end{cases} \quad (4)$$

where  $b(0_-) := (-1)^{m_0} b^{(m_0)}(0)$ ,  $m_0$  is the multiplicity of  $u=0$  as a root of  $b(u)$ , and  $b^{(m_0)}(0)$  denotes the value at  $u=0$  of the  $m_0$ th derivative of  $b(u)$ .

The following result, which is used in Algorithm 3, determines the number of real negative roots of a real polynomial.

**Lemma 2:** *A non-zero polynomial  $\psi \in \mathbf{R}[u]$ , such that  $\psi(0) \neq 0$ , has  $r$  real negative roots without counting the multiplicities if and only if the signature of the polynomial  $\psi(s^2) + s\psi'(s^2)$  is  $2r$ . All roots of  $\psi$  negative, and distinct if and only if  $\psi(s^2) + s\psi'(s^2) \in \mathcal{H}$ .*

**Proof:** We first assume that  $(\psi, \psi')$  is coprime. Suppose that  $\psi(u)$  has  $r$  real negative distinct roots  $u_1 > u_2 > \cdots > u_r$ . Since  $\psi'(u)$  is the derivative of  $\psi(u)$ , it follows that between any two consecutive real negative roots  $u_i$  and  $u_{i+1}$  of  $\psi(u)$  there is an odd number of real negative roots of  $\psi'(u)$ :  $v_{i1} > v_{i2} > \cdots > v_{ij}$ , where  $j$  is an odd integer. Since

$$\mathcal{S}\psi(v_{i1}) = \mathcal{S}\psi(v_{i2}) = \cdots = \mathcal{S}\psi(v_{ij}),$$

it follows that

$$2\mathcal{S}\psi(v_{i1}) - 2\mathcal{S}\psi(v_{i2}) + \cdots + (-1)^j 2\mathcal{S}\psi(v_{ij}) = 2\mathcal{S}\psi(v_{i1}).$$

In the interval  $(-\infty, u_r)$ ,  $\psi'(u)$  must have an even number or real roots otherwise  $\psi(u)$  have a real root

in this interval contradicting the fact that  $\psi(u)$  has  $r$  real negative roots. Assume that  $\psi(0) > 0$ . If  $\psi'(u)$  has an even number,  $k$ , of real roots  $v_{01}, v_{02}, \dots, v_{0k}$ , between 0 and  $u_1$ , then  $\psi'(0_-) > 0$  and

$$2\mathcal{S}\psi(v_{01}) - 2\mathcal{S}\psi(v_{02}) + \cdots + (-1)^k 2\mathcal{S}\psi(v_{0k}) = 0.$$

Finally,  $\mathcal{S}\psi(0) = 1$ ,  $\mathcal{S}\psi(v_{11}) = -1$ ,  $\mathcal{S}\psi(v_{21}) = 1, \dots$ ,  $\mathcal{S}\psi(-\infty) = (-1)^r$ . Using these facts in equation (4) of Lemma 1, we get

$$\begin{aligned} \mathcal{S}\psi'(0_-)[\mathcal{S}\psi(0) - 2\mathcal{S}\psi(v_{01}) + \cdots - 2\mathcal{S}\psi(v_{11}) + \cdots \\ + (-1)^r \mathcal{S}\psi(-\infty)] \\ = \mathcal{S}\psi(0) - 2\mathcal{S}\psi(v_{11}) + 2\mathcal{S}\psi(v_{21}) \\ - 2\mathcal{S}\psi(v_{31}) + \cdots + (-1)^r \mathcal{S}\psi(-\infty) = 2r. \end{aligned}$$

If  $\psi'(u)$  has an odd number of roots between 0 and  $u_1$ , then  $\psi'(0_-) < 0$ . In this case, we obtain again the same result

$$\begin{aligned} \mathcal{S}\psi'(0_-)[\mathcal{S}\psi(0) - 2\mathcal{S}\psi(v_{01}) + \cdots + 2\mathcal{S}\psi(v_{11}) - \cdots \\ + (-1)^r \mathcal{S}\psi(-\infty)] \\ = -[\mathcal{S}\psi(0) - 2\mathcal{S}\psi(v_{01}) + 2\mathcal{S}\psi(v_{11}) \\ - 2\mathcal{S}\psi(v_{21}) + \cdots + (-1)^r + 1\mathcal{S}\psi(-\infty)] = 2r. \end{aligned}$$

Similar arguments apply in the case  $\psi(0) < 0$  to give the same result; namely

$$\begin{aligned} \mathcal{S}\psi'(0_-)[\mathcal{S}\psi(0) - 2\mathcal{S}\psi(v_{01}) + \cdots + 2\mathcal{S}\psi(v_{11}) - \cdots \\ + (-1)^{r+1} \mathcal{S}\psi(-\infty)] = 2r. \end{aligned}$$

Therefore, by Lemma 1, signature of  $\psi(s^2) + s\psi'(s^2)$  is  $2r$ . Conversely, suppose that the signature of  $\psi(s^2) + s\psi'(s^2)$  is  $2r$ . Using the second equation of (4) in Lemma 1, it follows that  $\psi(u)$  changes sign exactly  $r$  times for  $u < 0$ . Hence,  $\psi(u)$  has  $r$  real negative roots.

Now, let us examine the case of non-coprime pair  $(\psi, \psi')$ . Since complex roots of  $\psi(u)$  and  $\psi'(u)$  do not affect the signature of  $\psi(s^2) + s\psi'(s^2)$ , we consider only the case of common real negative roots. Assume that  $\psi(u)$  and  $\psi'(u)$  have a common real negative root  $u_1$ , then  $\psi(u) = (u - u_1)\psi_1(u)$  and  $\psi'(u) = \psi_1(u) + (u - u_1)\psi_1'(u)$ . Since  $u_1$  is also a root of  $\psi'(u)$ , it follows that  $u_1$  is a root of  $\psi_1(u)$ . This shows that whenever  $(\psi, \psi')$  are not coprime,  $\psi(u)$  has a root of multiplicity greater than 1. Let  $\psi(u)$  have a real negative root  $u_1$  with multiplicity greater than 1. Repeating the same analysis as above, using the fact that  $u_1$  is also a root of  $\psi'(u)$ , and that  $\mathcal{S}\psi(u_1) = 0$ , it follows that  $\psi(u)$  has  $r$  real negative roots without counting

the multiplicities if and only if the signature of  $\psi(s^2) + s\psi'(s^2)$  is  $2r$ .

If  $\psi(u)$  has all its roots real, negative, and distinct, then  $r = \deg \psi$ . By the part we have just proved, signature of  $\psi(s^2) + s\psi'(s^2)$  is  $2r$  which is the degree of  $\psi(s^2) + s\psi'(s^2)$ . Hence,  $\psi(s^2) + s\psi'(s^2) \in \mathcal{H}$ . The converse follows by Hermite–Biehler theorem.  $\square$

### 3. Proportional controllers

We now describe a slight extension of the constant stabilizing gain algorithm of Özgüler and Koçan (1994). Given a plant  $g(s) = p(s)/q(s)$ , where  $p, q \in \mathbf{R}[s]$  are coprime with  $m = \deg p$  less than or equal to  $n = \deg q$ , the set  $A_r(p, q) := \{\alpha \in \mathbf{R} : \sigma g[\phi(s, \alpha)] = \sigma[q(s) + \alpha p(s)] = r\}$  is the set of all real  $\alpha$  such that  $\phi(s, \alpha)$  has signature equal to  $r$ .

Let  $(h, g)$  and  $(f, e)$  be the even–odd components of  $q$  and  $p$ , respectively, so that  $q(s) = h(s^2) + sg(s^2)$ ,  $p(s) = f(s^2) + se(s^2)$ . Let  $d := \gcd\{f, e\}$  so that  $f = d\bar{f}$ ,  $e = d\bar{e}$ , for coprime polynomials  $f, \bar{e} \in \mathbf{R}[u]$ . Then, the polynomial  $\bar{p}(s) := f(s^2) + s\bar{e}(s^2) = p(s)/d(s^2)$  is free of  $\mathbf{C}_0$  roots except possibly a simple root at  $s=0$ . Let  $(H, G)$  be the even–odd components of  $q(s)\bar{p}(-s)$ . Also let  $F(s^2) := p(s)\bar{p}(-s)$ . By a simple computation, it follows that

$$\left. \begin{aligned} H(u) &= h(u)\bar{f}(u) - ug(u)\bar{e}(u) \\ G(u) &= g(u)\bar{f}(u) - h(u)\bar{e}(u) \\ F(u) &= f(u)\bar{f}(u) - ue(u)\bar{e}(u). \end{aligned} \right\} \quad (5)$$

By an appropriate choice of  $d(u)$ , it can be ensured that  $G(0_-) > 0$ , where  $G(0_-) := (-1)^{m_0} G^{(m_0)}(0)$  with  $m_0$  being the multiplicity of  $u=0$  as a root of  $G(u)$ . If  $G \not\equiv 0$  and if they exist, let the *real negative zeros with odd multiplicities* of  $G(u)$  be  $\{v_1, \dots, v_k\}$  with the ordering  $v_1 > v_2 > \dots > v_k$ , with  $v_0 := 0$  and  $v_{k+1} := -\infty$  for notational convenience, and let the *real negative zeros with even multiplicities* of  $G(u)$  be  $\{u_1, \dots, u_l\}$ .

The following algorithm determines whether  $A_r(p, q)$  is empty or not and outputs its elements when it is not empty:

#### Algorithm 1:

*Step 1.* Consider all the sequences of signums

$$\mathcal{I} = \begin{cases} \{i_0, i_1, \dots, i_k\} & \text{for odd } r - m \\ \{i_0, i_1, \dots, i_{k+1}\} & \text{for even } r - m \end{cases}$$

where  $i_0 \in \{-1, 0, 1\}$  and  $i_j \in \{-1, 1\}$  for  $j = 1, \dots, k + 1$ .

*Step 2.* Choose all sequences that satisfy

$$r - \sigma(p) = \begin{cases} i_0 - 2i_1 + \dots + 2(-1)^k i_k & \text{for odd } r - m \\ i_0 - 2i_1 + \dots + 2(-1)^k i_k & \\ \quad + (-1)^{k+1} i_{k+1} & \text{for even } r - m. \end{cases}$$

*Step 3.* For each sequence of signums  $\mathcal{I} = \{i_j\}$  that satisfy Step 2, let

$$\alpha_{\max} = \max \left\{ -\frac{H}{F}(v_j) \right\} \text{ for all } v_j$$

for which  $i_j S F(v_j) = 1$

and

$$\alpha_{\min} = \min \left\{ -\frac{H}{F}(v_j) \right\} \text{ for all } v_j$$

for which  $i_j S F(v_j) = -1$ .

The set  $A_r(p, q)$  is non-empty if and only if for at least one signum sequence  $\mathcal{I}$  satisfying Step 2,  $\alpha_{\max} < \alpha_{\min}$  holds.

*Step 4.*  $A_r(p, q)$  is equal to the union of intervals  $(\alpha_{\max}, \alpha_{\min})$  for each sequence of signums  $\mathcal{I}$  that satisfy Step 3. The set of points  $\hat{A} := \{-(H/F)(u_j), j = 1, \dots, l : F(u_j) \neq 0\}$  must be excluded from  $A_r(p, q)$  as they correspond to values of  $\alpha$  for which  $q(s) + \alpha p(s)$  has zeros on the  $j\omega$ -axis.

From a computational point of view, application of Algorithm 1 is expensive. The main disadvantage comes from checking condition 2. In order to find the suitable signum sequences, we have to check condition 2 for  $2^{k+2}$  different candidate signum sequences in case  $p(s)$  has no roots in  $\mathbf{C}_0$  and  $n - m$  is even. In case  $p(s)$  has no roots in  $\mathbf{C}_0$  and  $n - m$  is odd, the number of sequences is  $2^{k+1}$ . Therefore, the number of sequences explodes exponentially as  $k$  increases. Since some sequences that satisfy condition 2 fail to satisfy condition 3, it is possible to improve Algorithm 1. In order to reduce the number of arithmetic operations needed in Algorithm 1, we have to first identify the signum sequences for which condition 3 holds then proceed to check condition 2. We now show that two different signum sequences  $\mathcal{I}_1, \mathcal{I}_2$  cannot correspond to the same interval. Let us define the



following sets:

$$\begin{aligned}\mathcal{J}_1^+ &= \{j : i_j \in \mathcal{I}_1, i_j \mathcal{S}F(v_j) = 1\} \\ \mathcal{J}_1^- &= \{j : i_j \in \mathcal{I}_1, i_j \mathcal{S}F(v_j) = -1\} \\ \mathcal{J}_2^+ &= \{j : i_j \in \mathcal{I}_2, i_j \mathcal{S}F(v_j) = 1\} \\ \mathcal{J}_2^- &= \{j : i_j \in \mathcal{I}_2, i_j \mathcal{S}F(v_j) = -1\}.\end{aligned}$$

Since  $\mathcal{I}_1 \neq \mathcal{I}_2$ , it follows that  $\mathcal{J}_1^+ \neq \mathcal{J}_2^+$  and  $\mathcal{J}_1^- \neq \mathcal{J}_2^-$ . Using condition 3 in Algorithm 1

$$\max_{j \in \mathcal{J}_1^+} \frac{H}{F}(v_j) \neq \max_{j \in \mathcal{J}_2^+} \frac{H}{F}(v_j)$$

and/or

$$\min_{j \in \mathcal{J}_1^-} \frac{H}{F}(v_j) \neq \min_{j \in \mathcal{J}_2^-} \frac{H}{F}(v_j).$$

In both cases  $\mathcal{I}_1$  and  $\mathcal{I}_2$  correspond to two different intervals as the endpoints of the intervals are different.

**Algorithm 2:**

Step 1. Calculate

$$\alpha_j = \begin{cases} -\frac{H}{F}(v_j), j = 0, \dots, k & \text{for odd } r - m \\ -\frac{H}{F}(v_j), j = 0, \dots, k + 1 & \text{for even } r - m \end{cases}$$

and sort the distinct  $\alpha_j$ 's in ascending order

$$\bar{\alpha}_0 < \bar{\alpha}_1 < \dots < \bar{\alpha}_{k+2} < \bar{\alpha}_{k+3}$$

where  $\bar{\alpha}_0 = -\infty$  and  $\bar{\alpha}_{k+3} = \infty$ .

Step 2. Identify all the sequences of signums

$$\mathcal{I} = \begin{cases} \{i_0, i_1, \dots, i_k\} & \text{for odd } r - m \\ \{i_0, i_1, \dots, i_{k+1}\} & \text{for even } r - m \end{cases}$$

where  $i_0 \in \{-1, 0, 1\}$  and  $i_j \in \{-1, 1\}$  for  $j = 1, \dots, k + 1$ , that correspond to the intervals  $(\bar{\alpha}_j, \bar{\alpha}_{j+1})$  for  $j = 0, \dots, k + 2$ .

Step 3. For each signum sequence  $\mathcal{I}_j$  from Step 2, if

$$r - \sigma(p) = \begin{cases} i_0 - 2i_1 + 2i_2 - 2i_3 + \dots + 2(-1)^k i_k & \text{for odd } r - m \\ i_0 - 2i_1 + 2i_2 - 2i_3 + \dots + (-1)^{k+1} i_{k+1} & \text{for even } r - m \end{cases}$$

holds, then  $(\bar{\alpha}_j, \bar{\alpha}_{j+1}) \in A_r(p, q)$ .

In Step 2 above it is easy to identify the signum sequences that lead to different intervals. Since  $\alpha_j$ 's are ordered in ascending order and  $\mathcal{S}F(v_j)$ ,  $j = 1, \dots, k + 1$  are known, we can determine  $\mathcal{J}^-$  and  $\mathcal{J}^+$  for a particular interval  $(\bar{\alpha}_i, \bar{\alpha}_{i+1})$ . This is equivalent to determining whether  $i_j = 1$  or  $i_j = -1$  for  $j = 0, 1, \dots, k + 1$  and therefore identifying  $\mathcal{I}$  for that particular interval. Algorithm 2 is similar to Neimark D-decomposition described in the introduction with the advantage that the application of some stability criterion at one interior point of each interval is replaced by Step 3. Using Neimark D-decomposition the problem can be solved with  $\mathbf{O}(n^3)$  arithmetic operations whereas Algorithm 2 requires only  $\mathbf{O}(n^2)$  arithmetic operations.

The algorithm above is easily specialized to determine all stabilizing proportional controllers  $c(s) = \alpha$  for the plant  $g(s)$ . This is achieved by replacing  $r$  in Step 3 of the algorithm by  $n$ , the degree of  $\phi(s, \alpha)$ . In case of plants with no unstable zeros and having a relative degree less than or equal to 2, and only in case of such plants (see Remark 3.2 in Saadaoui (2003)),  $A_r(p, q)$  may contain an infinite interval on the real axis. The algorithm above identifies such cases by outputting  $(H/F)(v_k) = \infty$  or  $(H/F)(v_{k+1}) = \infty$ , depending on whether the relative degree is odd or even.

**Remark 1:** By Step 3 of Algorithm 2, a necessary condition for the existence of an  $\alpha \in A_r(p, q)$  is that the odd part of  $[q(s) + \alpha p(s)]\bar{p}(-s)$  has at least

$$\bar{r} = \max \left\{ 0, \left\lfloor \frac{|r - \sigma(p)| - 1}{2} \right\rfloor \right\}$$

real negative roots with odd multiplicities. When solving a constant stabilization problem, this lower bound is

$$\bar{r} = \max \left\{ 0, \left\lfloor \frac{n - \sigma(p) - 1}{2} \right\rfloor \right\}.$$

□

**Remark 2:** The above algorithm can be modified (Datta *et al.* 2000) to give a linear program for determining the values of two parameters instead of only one. This is possible whenever we can modify the characteristic equation such that these parameters appear only in the even part. □

#### 4. First-order controllers

A first-order controller

$$c(s) = \frac{\alpha_2 s + \alpha_3}{s + \alpha_1}$$

applied to a plant transfer function  $g(s) = p(s)/q(s)$  gives the closed loop characteristic polynomial

$$\begin{aligned}\phi_0(s, \alpha_1, \alpha_2, \alpha_3) &= (s + \alpha_1)q(s) + (\alpha_2s + \alpha_3)p(s) \\ &= q_0(s) + \alpha_3p_0(s)\end{aligned}$$

where

$$\left. \begin{aligned}q_0(s) &= (s + \alpha_1)q(s) + \alpha_2sp(s) \\ p_0(s) &= p(s).\end{aligned}\right\} \quad (6)$$

Multiplying  $\phi_0(s, \alpha_1, \alpha_2, \alpha_3)$  by  $\bar{p}_0(-s)$  (recall that  $\bar{p}_0(s)$  denotes  $p_0(s)$  after division by the greatest common factor of its even-odd parts), we obtain

$$\begin{aligned}\psi_1(s, \alpha_1, \alpha_2, \alpha_3) &= \phi_0(s, \alpha_1, \alpha_2, \alpha_3)\bar{p}_0(-s) \\ &= s^2G(s^2) + \alpha_1H(s^2) + \alpha_3F(s^2) \\ &\quad + s[H(s^2) + \alpha_1G(s^2) + \alpha_2F(s^2)].\end{aligned} \quad (7)$$

Note that  $\alpha_1, \alpha_2$  appear in the odd part and  $\alpha_1, \alpha_3$  appear in the even part. (As pointed out in Xu *et al.* (2001), it is no longer possible to exploit the results given in Datta *et al.* (2000) and proceed.)

The reasoning behind the algorithm which determines the set of parameters  $\alpha_1, \alpha_2, \alpha_3$  of a stabilizing first-order controller can be explained as follows. Suppose  $\phi_0(s)$  is Hurwitz stable for some  $\alpha_1, \alpha_2, \alpha_3 \in \mathbf{R}$ . By Remark 1, it follows that the odd part  $H(u) + \alpha_1G(u) + \alpha_2F(u)$  of  $\psi_1(s)$  has at least  $r_1 = \lfloor (n - \sigma(p_0))/2 \rfloor$  real negative roots with odd multiplicities. Suppose  $H(u) + \alpha_1G(u) + \alpha_2F(u)$  has  $r_1$  real negative roots with odd multiplicities. By Lemma 2,  $\sigma[\phi_1(s)] = 2r_1$ , where

$$\begin{aligned}\phi_1(s) &= H_1(s) + \alpha_1G_1(s) + \alpha_2F_1(s) \\ &= q_1(s) + \alpha_2p_1(s)\end{aligned} \quad (8)$$

and

$$\begin{aligned}H_1(s) &= H(s^2) + sH'(s^2) \\ G_1(s) &= G(s^2) + sG'(s^2) \\ F_1(s) &= F(s^2) + sF'(s^2) \\ q_1(s) &= H_1(s) + \alpha_1G_1(s) \\ p_1(s) &= F_1(s).\end{aligned}$$

In order to find the suitable ranges of  $\alpha_1$  and  $\alpha_2$ , we modify  $\phi_1(s)$  as follows. Let  $B := \gcd\{F, F'\}$  so that  $F = B\bar{F}, F' = B\bar{F}'$  (the prime notation is still kept in  $F'$  although strictly speaking,  $F'$  is not the derivative

of a polynomial) for coprime polynomials  $\bar{F}, \bar{F}' \in \mathbf{R}[u]$ . Also let  $\bar{p}_1(s) := \bar{F}(s^2) + s\bar{F}'(s^2)$ . By a simple computation, it follows that

$$\begin{aligned}\psi_2(s) &= \phi_1(s)\bar{p}_1(-s) = H_{2e}(s^2) + \alpha_1G_{2e}(s^2) + \alpha_2F_{2e}(s^2) \\ &\quad + s[H_{2o}(s^2) + \alpha_1G_{2o}(s^2)],\end{aligned}$$

where

$$\left. \begin{aligned}H_{2e}(u) &= H(u)\bar{F}(u) - uH'(u)\bar{F}'(u) \\ G_{2e}(u) &= G(u)\bar{F}(u) - uG'(u)\bar{F}'(u) \\ F_{2e}(u) &= F(u)\bar{F}(u) - uF'(u)\bar{F}'(u) \\ H_{2o}(u) &= H'(u)\bar{F}(u) - H(u)\bar{F}'(u) \\ G_{2o}(u) &= G'(u)\bar{F}(u) - G(u)\bar{F}'(u).\end{aligned}\right\} \quad (9)$$

Once more by Remark 1, since  $\sigma[\phi_1(s)p_1(-s)] = 2r_1 - \sigma[p_1(s)]$  the odd part of  $\phi_1(s)\bar{p}_1(-s)$  should have at least  $r_2 = \lfloor (|2r_1 - \sigma(p_1)| - 1)/2 \rfloor$  real negative roots with odd multiplicities. Now the set of  $\alpha_1 \in \mathbf{R}$  which achieves  $r_2$  real negative roots with odd multiplicities in  $H_{2o}(u) + \alpha_1G_{2o}(u)$  can be determined by applying Algorithm 2 to

$$\begin{aligned}q_2(s) &= H_2(s) = H_{2o}(s^2) + sH'_{2o}(s^2) \\ p_2(s) &= G_2(s) = G_{2o}(s^2) + sG'_{2o}(s^2).\end{aligned}$$

The algorithm below traces the above steps backwards by repetition of the steps (i)–(iii) below:

(i) Pick a value of  $\alpha_1$  such that the number of real negative roots with odd multiplicities of  $H_{2o}(u) + \alpha_1G_{2o}(u)$  is  $r_2$  or greater.

(ii) Determine using Algorithm 2 all  $\alpha_2 \in \mathbf{R}$  such that  $\sigma[\phi_1(s)] = 2r_1$ . By Lemma 2 and Remark 3, this is equivalent to determining values of  $\alpha_2$  such that  $H(u) + \alpha_1G(u) + \alpha_2F(u)$  has  $r_1$  real negative roots with odd multiplicities.

(iii) For every  $\alpha_2$  determined, find using Algorithm 2 again, all  $\alpha_3$  such that  $\phi_1(s)$  is Hurwitz stable.

### Algorithm 3:

*Step 1.* Partition the real axis into intervals (or union of intervals) such that the number of real negative roots with odd multiplicities of  $H_{2o}(u) + \alpha_1G_{2o}(u)$  is constant in each interval.

*Step 2.* Fix  $r_1 = \lfloor (n - \sigma(p_0))/2 \rfloor$ .

(a) Find admissible range of  $\alpha_1$  from the intervals found in the first step.

(i) Fix an  $\alpha_1$  in the admissible range.

- (ii) Apply Algorithm 2 to  $q_1(s)$  and  $p_1(s)$ . (This calculates admissible values of  $\alpha_2$  such that  $H(u) + \alpha_1 G(u) + \alpha_2 F(u)$  has  $r_1$  real negative roots with odd multiplicities.)
- A. Fix an  $\alpha_2$  from the range determined in 2.(a.ii).
- B. Apply Algorithm 2 to  $q_0(s)$  and  $p_0(s)$ . (This calculates all admissible values of  $\alpha_3$  such that  $\phi_0$  is in  $\mathcal{H}$ .)
- C. Increment  $\alpha_2$  and go to Step 2(a.ii.B).
- (iii) Increment  $\alpha_1$  and go to Step 2(a.ii).
- (b) If  $r_1 < \deg(H)$ , then increment  $r_1$  by one and go to Step 2(a).

Algorithm 2 is repeatedly used on three auxiliary plants:

$$g_0(s) = \frac{p_0(s)}{q_0(s)} = \frac{p(s)}{(s + \alpha_1)q(s) + \alpha_2 sp(s)}$$

$$g_1(s) = \frac{p_1(s)}{q_1(s)} = \frac{F_1(s)}{H_1(s) + \alpha_1 G_1(s)}$$

$$g_2(s) = \frac{p_2(s)}{q_2(s)} = \frac{G_2(s)}{H_2(s)}$$

to give the admissible values of  $(\alpha_1, \alpha_2, \alpha_3)$ .

**Remark 3:** In Step (ii) above, only values of  $\alpha_2$  leading to  $H(u) + \alpha_1 G(u) + \alpha_2 F(u)$  having  $r_1$  real negative roots with odd multiplicities are calculated. If  $H(u) + \alpha_1 G(u) + \alpha_2 F(u)$  has a real negative root  $u_0$  of even multiplicity, then  $u_0$  is also a root of  $H'(u) + \alpha_1 G'(u) + \alpha_2 F'(u)$  with odd multiplicity. This corresponds to a conjugate pair of roots (with odd multiplicity) of  $\phi_2(s)$  on the  $j\omega$ -axis. Values of  $\alpha_2$  leading to this situation are excluded from the solution set by Algorithm 2. If  $H(u) + \alpha_1 G(u) + \alpha_2 F(u)$  has a real negative root  $u_1$  with odd multiplicity (not a simple root), then  $\phi_2(s)$  has a conjugate pair of roots (with even multiplicity) on the  $j\omega$ -axis. We can easily modify Step 3 in Algorithm 2 such that values of  $\alpha_2$  leading to the latter situation are included in the solution set.  $\square$

**Example 1:** Consider determining proper first-order controllers to stabilize the plant  $g(s) = p(s)/q(s)$ , where

$$q(s) = s^5 + 3s^4 + 29s^3 + 15s^2 - 3s + 60,$$

$$p(s) = s^3 - 6s^2 + 2s + 1.$$

The roots of  $q_0(s)$  are  $\{-1.2576 \pm j5.1476, -1.5574, 0.5363 \pm j1.0414\}$  and those of  $p_0(s)$  are

$\{-0.2705, 0.6587, 5.6119\}$  so that this is an unstable and non-minimum phase plant. Using

$$H(u) = -u^4 - 49u^3 - 142u^2 - 339u + 60$$

$$G(u) = -9u^3 - 194u^2 - 43u - 123$$

$$F(u) = -u^3 + 32u^2 - 16u + 1.$$

A necessary condition for the existence of a stabilizing first-order controller is that  $H(u) + \alpha_1 G(u) + \alpha_2 F(u)$  has at least  $r_1 = \lfloor (n - \sigma(p_0))/2 \rfloor = 3$  real negative roots with odd multiplicities. As  $\gcd(F, F') = 1$ , we multiply  $\phi_1(s)$  by  $p_1(-s)$ . For  $r_1 = 3$ ,  $\sigma(\phi_1) - \sigma(p_1) = 6$  and the odd part of  $\phi_1(s)p_1(-s)$  must have at least  $r_2 = \lfloor (|2r_1 - \sigma(p_2)| - 1)/2 \rfloor = 2$  real negative roots with odd multiplicities. Using Algorithm 1,  $\alpha_1 \in (-2.2917, 0.3088)$ . Similarly, for  $r_1 = 4$ , we find  $r_2 = 3$  and  $\alpha_1 \in (0.3088, 3.6000)$ . Now let us follow the steps of Algorithm 3 for a fixed value of  $\alpha_1$  from the above intervals. For  $\alpha_1 = 1$ , we have

$$q_1(s) = -s^8 - 4s^7 - 58s^6 - 174s^5 - 336s^4 - 672s^3 - 382s^2 - 382s - 63$$

$$p_1(s) = -s^6 - 3s^5 + 32s^4 + 64s^3 - 16s^2 - 16s + 1.$$

Using Step 2(a.ii) in Algorithm 3, the range of admissible values of  $\alpha_2$  for which  $H(u) + \alpha_1 G(u) + \alpha_2 F(u)$  has four negative distinct roots is  $\alpha_2 \in (-3.1602, 1.3297)$ . With  $\alpha_2 = 1$ , we obtain

$$q_0(s) = s^6 + 4s^5 + 33s^4 + 38s^3 + 14s^2 + 58s + 60$$

$$p_0(s) = s^4 - 6s^3 + 2s + 1.$$

Step 2(a.ii.B) in Algorithm 3 gives the solution  $\alpha_3 \in (-17.0988, -11.5621)$  for  $\alpha_1 = \alpha_2 = 1$ . Application of Algorithm 3, with a 0.05 increment of  $\alpha_2$  in Step 2(a.ii.C) and a 0.1 increment of  $\alpha_1$  in Step 2(a.iii), results in the set of stabilizing  $(\alpha_1, \alpha_2, \alpha_3)$  values shown in figure 1.

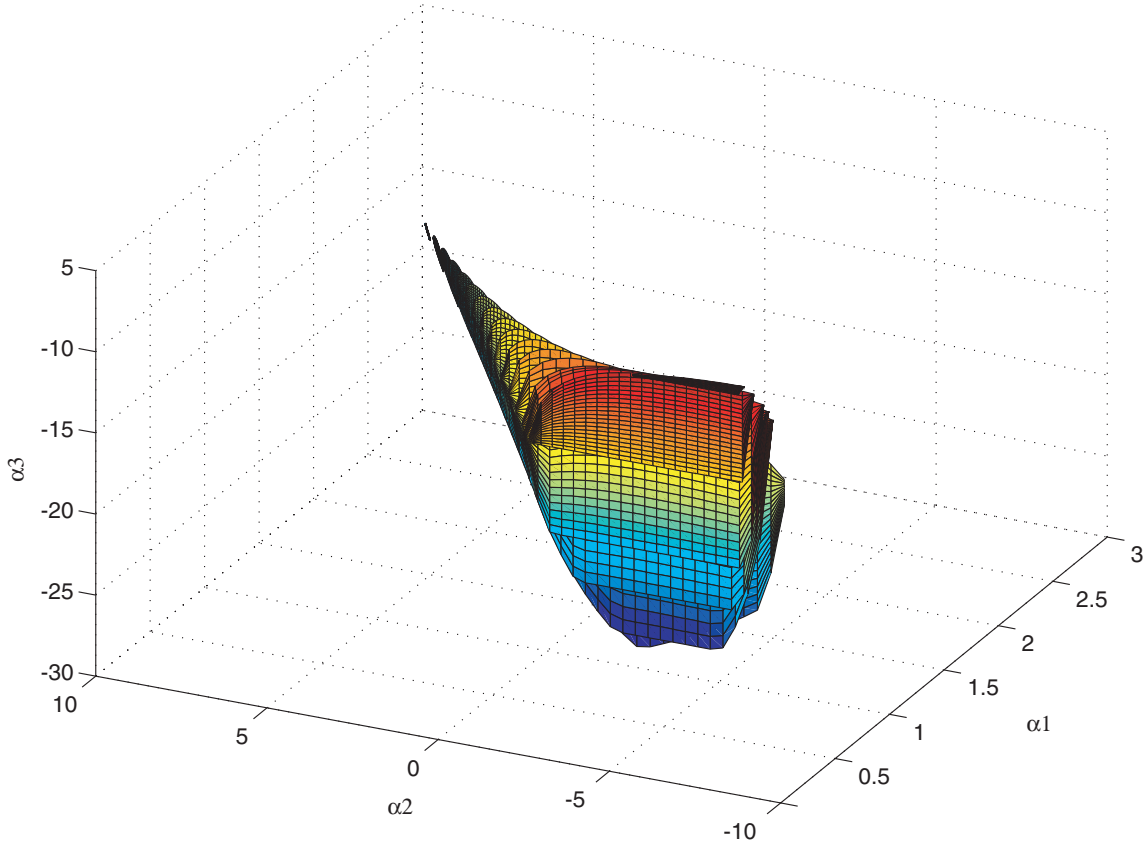
## 5. Uncertain systems

The method described in the previous sections can be applied to plants with interval type uncertainty. Let  $g(s)$  be the transfer function of an uncertain system

$$g(s) = \frac{p(s)}{q(s)} = \frac{\sum_{i=0}^m x_i s^i}{\sum_{j=0}^n y_j s^j} \quad (10)$$

where  $n > m$ ,  $x_m \neq 0$ ,  $y_n \neq 0$  and  $x_i \in [x_{i-}, x_{i+}]$ ,  $i = 1, \dots, m$  and  $y_j \in [y_{j-}, y_{j+}]$ ,  $j = 1, \dots, n$ . Let  $p_k(s)$  and  $q_l(s)$ ,  $k, l = 1, 2, 3, 4$  be the four Kharitonov polynomials corresponding to  $p(s)$  and  $q(s)$ , respectively.




 Figure 1. Stabilizing set of  $(\alpha_1, \alpha_2, \alpha_3)$  values for example 1.

Let  $p_k(s)$ ,  $k = 1, 2, 3, 4$  be the four Kharitonov segments of  $p(s)$ , i.e.

$$p_1^\lambda(s) = (1 - \lambda)p_1(s) + \lambda p_2(s)$$

$$p_2^\lambda(s) = (1 - \lambda)p_1(s) + \lambda p_3(s)$$

$$p_3^\lambda(s) = (1 - \lambda)p_2(s) + \lambda p_4(s)$$

$$p_4^\lambda(s) = (1 - \lambda)p_3(s) + \lambda p_4(s)$$

$$q_1^\lambda(s) = (1 - \lambda)q_1(s) + \lambda p_2(s)$$

$$q_2^\lambda(s) = (1 - \lambda)q_1(s) + \lambda p_3(s)$$

$$q_3^\lambda(s) = (1 - \lambda)q_2(s) + \lambda p_4(s)$$

$$q_4^\lambda(s) = (1 - \lambda)q_3(s) + \lambda p_4(s)$$

where  $\lambda \in [0, 1]$ . The four Kharitonov segments  $q_l(s)$ ,  $l = 1, 2, 3, 4$  of  $q(s)$  can be defined similarly. Let  $g_{\text{seg}}(s)$  denote the family of 32 segment plants

$$g_{\text{seg}}(s) = \left\{ g_{kl}(s, \lambda) \mid g_{kl}(s, \lambda) = \frac{p_k^\lambda(s)}{q_l(s)} \right.$$

$$\left. \text{or } g_{kl}(s, \lambda) = \frac{p_k(s)}{q_l^\lambda(s)}, \quad k, l = 1, 2, 3, 4, \text{ and } \lambda \in [0, 1] \right\}.$$

It is well known (Barmish 1994) that the family  $g(s)$  is stabilized by a particular controller, if and only if the 32 segment plants  $g_{\text{seg}}$  are stabilized by the same controller. Let  $\tilde{g}_{\text{seg}}(s)$  denote the family of 16 segment plants

$$\tilde{g}_{\text{seg}}(s) = \left\{ g_{kl}(s, \lambda) \mid g_{kl}(s, \lambda) = \frac{p_k^\lambda(s)}{q_l(s)}, \right. \\ \left. k, l = 1, 2, 3, 4, \text{ and } \lambda \in [0, 1] \right\}.$$

It is shown in Ho *et al.* (1998) (Munro and Söylemez 2000) that ‘the entire family  $g(s)$  is stabilized by a particular PID controller, if and only if each segment plant  $g_{kl}(s) \in \tilde{g}_{\text{seg}}(s)$  is stabilized by that same PID controller’. In reaching this result the structure of the PID controller was used to reduce the 32 segment plants to only 16. Since we are considering first-order controllers, the numerator and denominator of the controller are convex directions (Barmish 1994). It is shown in Barmish (1994) that stabilizing an interval plant  $g(s)$  by a first-order controller is equivalent to stabilizing 16 vertex plants; namely,

$$g_v(s) = \left\{ g_{kl}(s) \mid g_{kl}(s) = \frac{p_k(s)}{q_l(s)}, \quad k, l = 1, 2, 3, 4 \right\}.$$

The stabilizing controller, if any, can be determined by first calculating  $\alpha_1$  which is the intersection of  $\alpha_1 s$  found for the 16 plants mentioned above. We can then apply the algorithm of the previous section for the 16 vertex plants to find  $\alpha_2$  and  $\alpha_3$ . The following example is from Saadaoui and Özgüler (2003).

**Example 2:** Consider a proper first-order controller to stabilize the interval plant  $g(s) = p(s)/q(s)$  where

$$q(s) = s^5 + y_4 s^4 + y_3 s^3 + y_2 s^2 + y_1 s + y_0$$

$$p(s) = s^3 + x_2 s^2 + x_1 s + x_0$$

and

$$x_0 \in [-1, -2] \quad x_1 \in [2, 2], \quad x_2 \in [-6, -5]$$

$$y_0 \in [60, 65], \quad y_1 \in [-5, -3], \quad y_2 \in [14, 15]$$

$$y_3 \in [29, 29], \quad y_4 \in [3, 4].$$

We get the following Kharitonov polynomials

$$q_1(s) = s^5 + 3s^4 + 29s^3 + 15s - 5s + 60$$

$$q_2(s) = s^5 + 3s^4 + 29s^3 + 15s - 3s + 60$$

$$q_3(s) = s^5 + 4s^4 + 29s^3 + 14s - 3s + 65$$

$$q_4(s) = s^5 + 4s^4 + 29s^3 + 14s - 5s + 65$$

$$p_1(s) = p_3(s) = s^3 - 6s^2 + 2s - 1$$

$$p_2(s) = p_4(s) = s^3 - 5s^2 + 2s - 2$$

a suitable range of  $\alpha_1$  was determined to be  $\alpha_1 \in (-1.54, 0.97)$ . This is the intersection of suitable ranges of  $\alpha_1$  for the 16 vertex plants. Using Algorithm 2 for the 16 vertex plants, the set of stabilizing  $(\alpha_1, \alpha_2, \alpha_3)$  values are shown in figure 2.  $\square$

## 6. Second-order controllers

In this section, we will show that Algorithm 3 can be extended to compute all stabilizing parameters of a higher-order controller. We will give a detailed

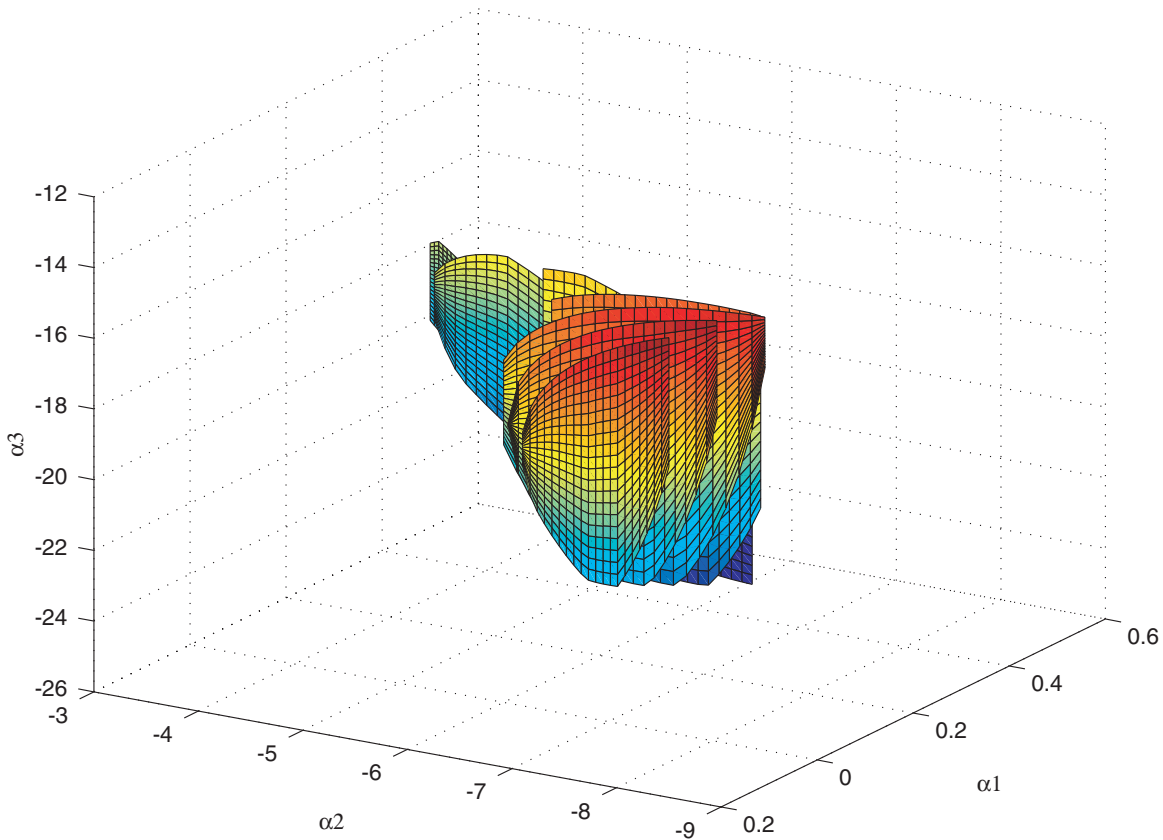


Figure 2. Stabilizing set of  $(\alpha_1, \alpha_2, \alpha_3)$  values.

derivation of the second-order controller case and show how to find the  $j$ th parameter in a  $l$ th order controller. Now, we describe an algorithm that determines the set of all stabilizing second-order controllers for a given plant. A second-order controller

$$c(s) = \frac{\alpha_3 s^2 + \alpha_4 s + \alpha_5}{s^2 + \alpha_1 s + \alpha_2}$$

applied to  $g(s)$  gives the closed loop characteristic polynomial

$$\begin{aligned} \phi_0(s, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) &= (s^2 + \alpha_1 s + \alpha_2)q(s) \\ &\quad + (\alpha_3 s^2 + \alpha_4 s + \alpha_5)p(s) \\ &= q_0(s) + (\alpha_3 s^2 + \alpha_5)p_0(s) \end{aligned} \quad (11)$$

where

$$\left. \begin{aligned} q_0(s) &= (s^2 + \alpha_1 s + \alpha_2)q(s) + \alpha_4 s p(s) \\ p_0(s) &= p(s). \end{aligned} \right\} \quad (12)$$

Multiplying  $\phi_0(s, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5)$  by  $\bar{p}_0(-s)$  we obtain

$$\begin{aligned} \psi_1(s, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) &= \phi_0(s, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5)\bar{p}_0(-s) \\ &= s^2 H(s^2) + \alpha_1 s^2 G(s^2) + \alpha_2 H(s^2) \\ &\quad + \alpha_3 s^2 F(s^2) + \alpha_5 F(s^2) \\ &\quad + s[s^2 G(s^2) + \alpha_1 H(s^2) \\ &\quad + \alpha_2 G(s^2) + \alpha_4 F(s^2)]. \end{aligned} \quad (13)$$

The reasoning behind the algorithm which determines the set of parameters  $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ , and  $\alpha_5$  of a stabilizing second-order controller can be explained as follows. Suppose  $\phi_0(s)$  is Hurwitz stable for some  $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5 \in \mathbf{R}$ . By Remark 1, it follows that the odd part  $uG(u) + \alpha_1 H(u) + \alpha_2 G(u) + \alpha_4 F(u)$  of  $\psi_1(s)$  has at least  $r_1 = \lfloor (n+1 - \sigma(p_0))/2 \rfloor$  real negative roots with odd multiplicities. Suppose  $uG(u) + \alpha_1 H(u) + \alpha_2 G(u) + \alpha_3 F(u)$  has  $r_1$  real negative roots with odd multiplicities. By Lemma 2,  $\sigma[\phi_1(s)] = 2r_1$ , where

$$\begin{aligned} \phi_1(s) &= G_1^u(s) + \alpha_1 H_1(s) + \alpha_2 G_1(s) + \alpha_4 F_1(s) \\ &= q_1(s) + \alpha_4 p_1(s) \end{aligned}$$

and

$$\left. \begin{aligned} H_1(s) &= H(s^2) + sH'(s^2) \\ G_1(s) &= G(s^2) + sG'(s^2) \\ F_1(s) &= F(s^2) + sF'(s^2) \\ G_1^u(s) &= s^2 G(s^2) + s[G(s^2) + s^2 G'(s^2)] \\ q_1(s) &= G_1^u(s) + \alpha_1 H_1(s) + \alpha_2 G_1(s) \\ p_1(s) &= F_1(s). \end{aligned} \right\} \quad (14)$$

In order to find the suitable ranges of  $\alpha_1, \alpha_2$  and  $\alpha_4$ , we modify  $\phi_1(s)$  as follows. Let  $B := \gcd\{F, F'\}$  so that  $F = B\bar{F}, F' = B\bar{F}'$  for coprime polynomials  $\bar{F}, \bar{F}' \in \mathbf{R}[u]$ . Let  $\bar{p}_1(s) := \bar{F}(s^2) + s\bar{F}'(s^2)$ . By a simple computation, it follows that

$$\begin{aligned} \psi_2(s, \alpha_1, \alpha_2, \alpha_4) &= \phi_1(s)\bar{p}_1(-s) \\ &= G_{2e}^u(s^2) + \alpha_1 H_{2e}(s^2) + \alpha_2 G_{2e}(s^2) \\ &\quad + \alpha_4 F_{2e}(s^2) \\ &\quad + s[G_{2o}^u(s^2) + \alpha_1 H_{2o}(s^2) + \alpha_2 G_{2o}(s^2)] \end{aligned}$$

where

$$\left. \begin{aligned} G_{2e}^u(u) &= uG(u)\bar{F}(u) - u[G(u) + uG'(u)]\bar{F}'(u) \\ G_{2o}^u(u) &= [G(u) + uG'(u)]\bar{F}(u) - uG(u)\bar{F}'(u) \\ H_{2e}(u) &= H(u)\bar{F}(u) - uH'(u)\bar{F}'(u) \\ H_{2o}(u) &= H'(u)\bar{F}(u) - H(u)\bar{F}'(u) \\ G_{2e}(u) &= G(u)\bar{F}(u) - uG'(u)\bar{F}'(u) \\ G_{2o}(u) &= G'(u)\bar{F}(u) - G(u)\bar{F}'(u) \\ F_{2e}(u) &= F(u)\bar{F}(u) - uF'(u)\bar{F}'(u). \end{aligned} \right\} \quad (15)$$

Again by Remark 1, it follows that the odd part  $G_{2o}^u(s^2) + \alpha_1 H_{2o}(s^2) + \alpha_2 G_{2o}(s^2)$  has at least  $r_2 = \lfloor (2r_1 - \sigma(p_1) - 1)/2 \rfloor$  real negative roots with odd multiplicities. Repeating the same procedure once more, suppose that  $G_{2o}^u(s^2) + \alpha_1 H_{2o}(s^2) + \alpha_2 G_{2o}(s^2)$  has  $r_2$  real negative roots with odd multiplicities. By Lemma 2,  $\sigma[\phi_2(s)] = 2r_2$ , where

$$\begin{aligned} \phi_2(s) &= G_2^u(s) + \alpha_1 H_2(s) + \alpha_2 G_2(s) \\ &= q_2(s) + \alpha_2 p_2(s) \end{aligned}$$

and

$$\left. \begin{aligned} G_2^u(s) &= G_{2o}^u(s^2) + sG_{2o}^u(s^2) \\ H_2(s) &= H_{2o}(s^2) + sH_{2o}'(s^2) \\ G_2(s) &= G_{2o}(s^2) + sG_{2o}'(s^2) \\ q_2(s) &= G_2^u(s) + \alpha_1 H_2(s) \\ p_2(s) &= G_2(s). \end{aligned} \right\} \quad (16)$$

The same steps above are repeated for  $\phi_2(s)$ . Let  $C := \gcd\{G_{2o}, G'_{2o}\}$  so that  $G_{2o} = C\bar{G}'_{2o}$ ,  $G'_{2o} = C\bar{G}_{2o}$  for coprime polynomials  $\bar{G}_{2o}, \bar{G}'_{2o} \in \mathbf{R}[u]$ . Let  $\bar{p}_2(s) := \bar{G}_{2o}(s^2) + s\bar{G}'_{2o}(s^2)$ . Multiplying  $\phi_2(s)$  by  $p_2(-s)$ , we get

$$\begin{aligned}\psi_3(s, \alpha_1, \alpha_2) &= \phi_2(s)\bar{p}_2(-s) \\ &= G_{3e}^u(s^2) + \alpha_1 H_{3e}(s^2) + \alpha_2 G_{3e}(s^2) \\ &\quad + s[G_{3o}^u(s^2) + \alpha_1 H_{3o}(s^2)]\end{aligned}$$

where

$$\left. \begin{aligned}G_{3e}^u(u) &= G_{2o}^u(u)\bar{G}_{2o}(u) - uG_{2o}^u(u)'_{2o}(u) \\ G_{3o}^u(u) &= G_{2o}^u(u)\bar{G}'_{2o}(u) - G_{2o}^u(u)\bar{G}_{2o}(u) \\ H_{3e}(u) &= H_{2o}(u)\bar{G}_{2o}(u) - uH_{2o}'(u)\bar{G}'_{2o}(u) \\ H_{3o}(u) &= H_{2o}'(u)\bar{G}_{2o}(u) - H_{2o}(u)\bar{G}'_{2o}(u) \\ G_{3e}(u) &= G_{2o}(u)\bar{G}_{2o}(u) - uG_{2o}'(u)\bar{G}'_{2o}(u).\end{aligned}\right\} \quad (17)$$

Once more by Remark 1, the odd part of  $\psi_3(s)$  has at least  $r_3 = \lfloor (|2r_2 - \sigma(p_2)| - 1)/2 \rfloor$  real negative roots with odd multiplicities. Now the set of  $\alpha_1 \in \mathbf{R}$  which achieves  $r_3$  real negative roots with odd multiplicities in  $G_{3o}^u(u) + \alpha_1 H_{3o}(u)$  can be determined by applying Algorithm 2 to

$$\begin{aligned}q_3(s) &= G_{3o}^u(s) = G_{3o}^u(s^2) + sG_{3o}^u(s^2) \\ p_3(s) &= H_3(s) = H_{3o}(s^2) + sH_{3o}'(s^2).\end{aligned}$$

The algorithm below traces the above steps backwards by repetition of the steps (i)–(iv) below:

- (i) Pick a value of  $\alpha_1$  such that the number of real negative roots with odd multiplicities of  $G_{3o}^u(u) + \alpha_1 H_{3o}(u)$  is  $r_3$  or greater.
- (ii) Determine using Algorithm 2 all  $\alpha_2 \in \mathbf{R}$  such that  $\sigma[\phi_2(s)] = 2r_2$ . By Lemma 2 and Remark 3, this is equivalent to determining values of  $\alpha_2$  such that  $G_{2o}^u(u) + \alpha_1 H_{2o}(u) + \alpha_2 G_{2o}(u)$  has  $r_2$  real negative roots with odd multiplicities.
- (iii) For every  $\alpha_2$  found, determine using Algorithm 2 all  $\alpha_4 \in \mathbf{R}$  such that  $\sigma[\phi_1(s)] = 2r_1$ . By Lemma 2 and Remark 3, this is equivalent to determining values of  $\alpha_4$  such that  $uG(u) + \alpha_1 H(u) + \alpha_2 G(u) + \alpha_4 F(u)$  has  $r_1$  real negative roots with odd multiplicities.
- (iv) For every  $\alpha_4$  determined, find using extension of Algorithm 2, all  $\alpha_3, \alpha_5$  such that  $\phi_0(s)$  is Hurwitz stable.

The following algorithm determines all  $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ , and  $\alpha_5$  such that  $\phi(s, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) \in \mathcal{H}$ .

**Algorithm 4:**

- Partition the real axis into intervals (or union of intervals) such that the number of real negative roots with odd multiplicities of  $G_{3o}^u(u) + \alpha_1 H_{3o}(u)$  is constant in each interval.
- Fix  $r_1 = \lfloor (n + 1 - \sigma(p_0))/2 \rfloor$ .
  - (1) Fix  $r_2 = \lfloor \frac{2r_1 - \sigma(p_1)}{2} \rfloor$ .
  - (2) Find admissible range of  $\alpha_1$  from the intervals found in the first step.
    - (a) Fix an  $\alpha_1$  in the admissible range.
    - (b) Apply Algorithm 2 to  $q_2(s)$  and  $p_2(s)$  given by (16). (This calculates admissible values of  $\alpha_2$  such that  $G_{2o}^u(u) + \alpha_1 H_{2o}(u) + \alpha_2 G_{2o}(u)$  has  $r_2$  real negative roots with odd multiplicities.)
      - (i) Fix an  $\alpha_2$  from the range determined in (2.b).
      - (ii) Apply Algorithm 2 to  $q_1(s)$  and  $p_1(s)$  given by equation (14). (This calculates all admissible values of  $\alpha_4$  such that  $uG(u) + \alpha_1 H(u) + \alpha_2 G(u) + \alpha_4 F(u)$  has  $r_1$  real negative roots with odd multiplicities.)
        - (A) Fix an  $\alpha_4$  from the range determined in (2.b.ii).
        - (B) Apply modified Algorithm 2 to  $q_0(s)$  and  $p_0(s)$  given by equation (12). (This calculates all admissible values of  $\alpha_3$  and  $\alpha_5$  such that  $\phi_0$  of equation (1) is in  $\mathcal{H}$ .)
        - (C) Increment  $\alpha_4$  and go to Step 2(b.ii.A).
      - (iii) Increment  $\alpha_2$  and go to Step 2(b.i).
    - (c) Increment  $\alpha_1$  and go to Step 2(a).
  - (3) If  $r_2 < \deg(G_{2o}^u)$ , then increment  $r_2$  by one and go to Step 2.
- If  $r_1 < \deg(uG)$  then increment  $r_1$  by one and go to Step 1.

Algorithm 2 is repeatedly used on four auxiliary plants

$$\begin{aligned}g_0(s) &= \frac{p_0(s)}{q_0(s)} = \frac{p(s)}{(s^2 + \alpha_1 s + \alpha_2)q(s) + \alpha_4 s p(s)} \\ g_1(s) &= \frac{p_1(s)}{q_1(s)} = \frac{F_1(s)}{G_1^u(s) + \alpha_1 H_1(s) + \alpha_2 G_1(s)} \\ g_2(s) &= \frac{p_2(s)}{q_2(s)} = \frac{G_2(s)}{G_2^u(s) + \alpha_1 H_2(s)} \\ g_3(s) &= \frac{p_3(s)}{q_3(s)} = \frac{H_3(s)}{G_3^u(s)}\end{aligned}$$

to give the admissible values of  $(\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5)$ .

**Remark 4:** The method can also be applied to discrete-time plants using a bilinear transformation of the complex plane. Let the controller transfer function be

$$c(z) = \frac{\alpha_3 z^2 + \alpha_4 z + \alpha_5}{\alpha_1 z^2 + \alpha_2 z + 1}.$$

By the bilinear transformation  $z = (w + 1)/(w - 1)$ , we get

$$c(w) = \frac{(\alpha_3 + \alpha_4 + \alpha_5)w^2 + (2\alpha_2 - 2\alpha_5)w + \alpha_3 - \alpha_4 + \alpha_5}{(\alpha_1 + \alpha_2 + 1)w^2 + (2\alpha_1 - 2)w + \alpha_1 - \alpha_2 + 1}.$$

For a  $c(w)$  in this form,  $\alpha_1, \alpha_2, \alpha_3$ , and  $\alpha_5$  appear both in the even and odd parts of  $\psi(w, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) = \phi(w, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5)\bar{p}(-w)$ . Let  $\bar{\alpha}_3 = \alpha_3 + \alpha_4 + \alpha_5$ ,  $\bar{\alpha}_4 = \alpha_3 - \alpha_5$  and  $\bar{\alpha}_5 = \alpha_3 - \alpha_4 + \alpha_5$ . Then, by a simple computation it follows that

$$\begin{aligned} \psi(w) = & w^2 H(w^2) + H(w^2) - 2w^2 G(w^2) + \alpha_1 [w^2 H(w^2) \\ & + H(w^2) + 2w^2 G(w^2)] \\ & + \alpha_2 [w^2 H(w^2) - H(w^2)] + \bar{\alpha}_3 w^2 F(w^2) + \bar{\alpha}_5 F(w^2) \\ & + w [w^2 G(w^2) - 2H(w^2) \\ & + G(w^2) + \alpha_1 (w^2 G(w^2) + 2H(w^2) + G(w^2)) \\ & + \alpha_2 (w^2 G(w^2) - G(w^2) + \bar{\alpha}_4 F(w^2)]. \end{aligned}$$

Stabilizing controller parameters  $\alpha_1, \alpha_2, \bar{\alpha}_3, \bar{\alpha}_4$  and  $\bar{\alpha}_5$  can be calculated using Algorithm 4. Since

$$\begin{bmatrix} \bar{\alpha}_3 \\ \bar{\alpha}_4 \\ \bar{\alpha}_5 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & -1 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix}$$

and the linear transformation is invertible, we can calculate the values of  $\alpha_3, \alpha_4$  and  $\alpha_5$  as

$$\begin{bmatrix} \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 0 & -\frac{1}{2} \\ \frac{1}{4} & -\frac{1}{2} & \frac{1}{4} \end{bmatrix} \begin{bmatrix} \bar{\alpha}_3 \\ \bar{\alpha}_4 \\ \bar{\alpha}_5 \end{bmatrix}.$$

The method hence applies to discrete-time plants of arbitrary order.

**Example 3:** Consider determining a strictly proper second-order controllers

$$c(s) = \frac{\alpha_3 s + \alpha_4}{s^2 + \alpha_1 s + \alpha_2}$$

to stabilize the plant  $g(s) = p(s)/q(s)$ , where

$$\begin{aligned} q(s) &= s^5 + 4s^4 + 29s^3 + 15s^2 - 3s + 60 \\ p(s) &= s^3 - 6s^2 + 2s + 1. \end{aligned}$$

The roots of  $q_0(s)$  are  $\{-1.2576 \pm j5.1476, -1.5574, 0.5363 \pm j1.0414\}$  and those of  $p_0(s)$  are  $\{-0.2705, 0.6587, 5.6119\}$  so that this is an unstable and non-minimum phase plant. Using equation (1), we have

$$\begin{aligned} H(u) &= -u^4 - 49u^3 - 142u^2 - 339u + 60 \\ G(u) &= -9u^3 - 194u^2 - 43u - 123 \\ F(u) &= -u^3 + 32u^2 - 16u + 1. \end{aligned}$$

A necessary condition for the existence of a stabilizing second-order controller is that  $uG(u) + \alpha_1 H(u) + \alpha_2 G(u) + \alpha_3 F(u)$  has at least  $r_1 = \lfloor (n + 1 - \sigma(p_0))/2 \rfloor = 3$  real negative roots with odd multiplicities. As  $\gcd(F, F') = 1$ , we multiply  $\phi_1(s)$  by  $p_1(-s) = F(s^2) - sF'(s^2)$ . For  $r_1 = 3$ ,  $\sigma(\phi_1) - \sigma(p_1) = 6$  and the odd part of  $\phi_1(s) p_1(-s)$  must have at least  $r_2 = \lfloor (|2r_1 - \sigma(p_1)| - 1)/2 \rfloor = 2$  real negative roots with odd multiplicities. In a similar way we can determine  $r_3 = \lfloor (|2r_2 - \sigma(p_2)| - 1)/2 \rfloor = 1$ . For  $r_1 = 4$  we obtain  $r_2 = 3$  and  $r_3 = 2$ . Now let us follow the steps of Algorithm 4 for a fixed value of  $\alpha_1$ . For  $\alpha_1 = 1$ , using Step 2(b) in Algorithm 4, the range of admissible values of  $\alpha_2$  for which  $G_{20}''(u) + \alpha_1 H_{20}(u) + \alpha_2 G_{20}(u)$  has at least two negative real roots is  $(-14.3402, 1.5032)$ . With  $\alpha_2 = 0.5$ , we obtain

$$\begin{aligned} q_1(s) &= -10s^8 - 40s^7 - 247.5s^6 - 742.5s^5 - 282s^4 \\ &\quad - 564s^3 - 483.5s^2 - 483.58s - 1.5 \\ p_1(s) &= -s^6 - 3s^5 + 32s^4 + 64s^3 - 16s^2 - 16s + 1. \end{aligned}$$

Step 2(b.ii) in Algorithm 4 gives the following solution  $\alpha_3 \in (-15.8926, -8.5154)$  for  $\alpha_1 = 1$  and  $\alpha_2 = 0.5$ . With  $\alpha_3 = -10$ , we obtain

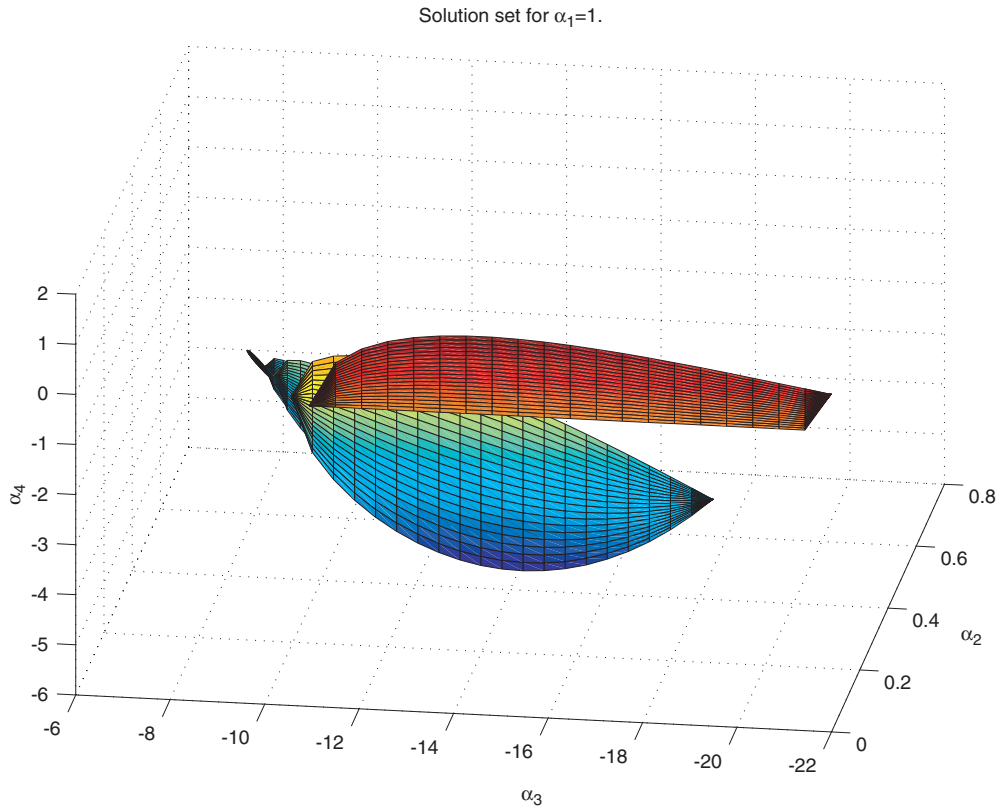
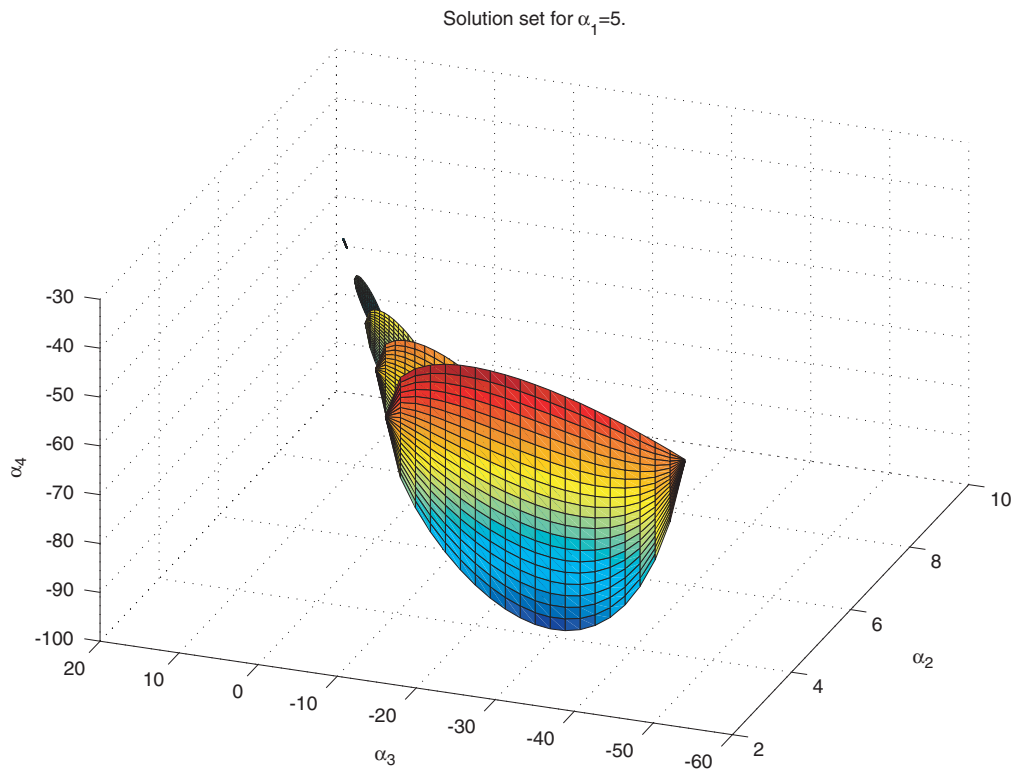
$$\begin{aligned} q_0(s) &= s^7 + 4s^6 + 32.5s^5 + 35.5s^4 + 86.5s^3 + 44.5s^2 \\ &\quad + 48.5s + 30 \\ p_0(s) &= s^3 - 6s^2 + 2s + 1. \end{aligned}$$

Step 2(b.ii.A) in Algorithm 4 gives the following solution  $\alpha_4 \in (-4.0566, -2.8786)$  for  $\alpha_1 = 1, \alpha_2 = 0.5$  and  $\alpha_3 = -10$ . The solution set for  $\alpha_1 = 1$  is shown in figure 3.

Figures 4 and 5 shows the results for  $\alpha_1 = 5$  and  $\alpha_1 = 15$ , respectively.

**Remark 5:** In this section, we gave a complete derivation of an algorithm that determines all stabilizing second-order controllers for a given plant. Algorithm 2 is repeatedly applied to a number of auxiliary plants



Figure 3. Stabilizing set of  $(\alpha_2, \alpha_3, \alpha_4)$  values for  $\alpha_1 = 1$ .Figure 4. Stabilizing set of  $(\alpha_2, \alpha_3, \alpha_4)$  values for  $\alpha_1 = 5$ .

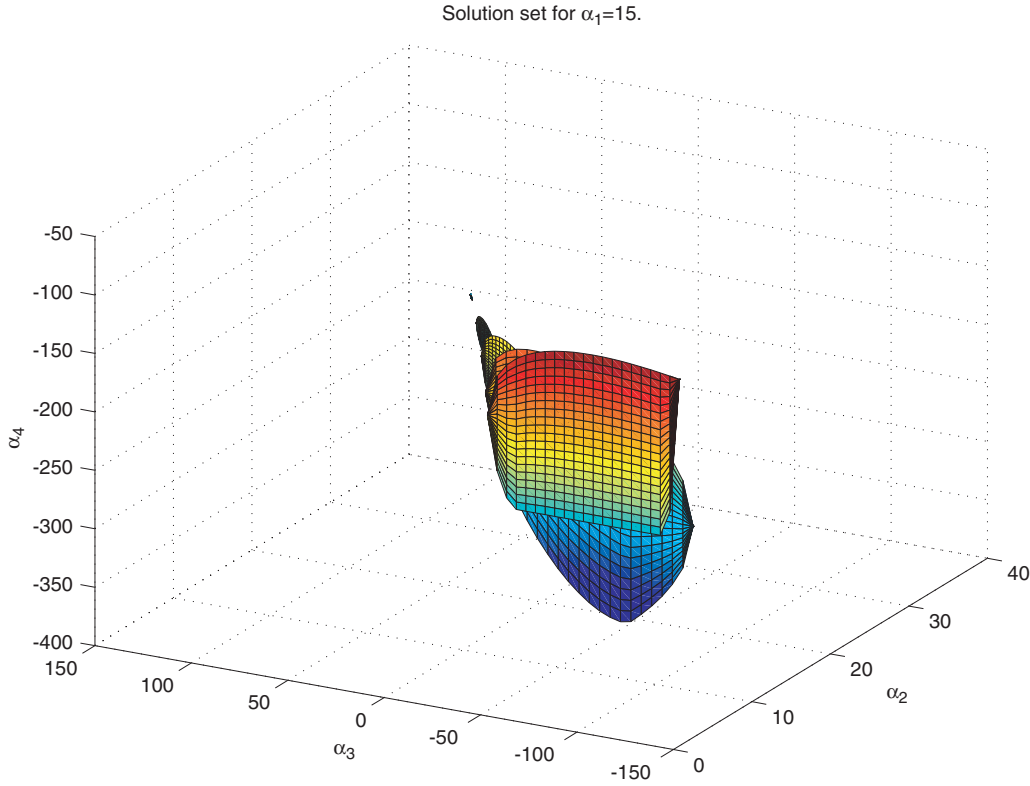


Figure 5. Stabilizing set of  $(\alpha_2, \alpha_3, \alpha_4)$  values for  $\alpha_1 = 15$ .

$(g_0(s) = p_0(s)/q_0(s), g_1(s) = p_1(s)/q_1(s), g_2(s) = p_2(s)/q_2(s),$  and  $g_3(s) = p_3(s)/q_3(s))$ . The above algorithm can be extended to high-order controllers. As the number of parameters of the controller increases, the number of auxiliary plants increases accordingly. For an  $l$ th order controller (we assume here that  $l$  is even and let  $k = 3l/2$ )

$$c(s) = \frac{1}{s^l + \alpha_1 s^{l-1} + \alpha_2 s^{l-2} + \dots + \alpha_l} \times \left( s[\alpha_{l+1} s^{l-2} + \alpha_{l+2} s^{l-4} + \dots + \alpha_k] + \alpha_{k+1} s^l + \alpha_{k+2} s^{l-2} + \dots + \alpha_{2l+1} \right)$$

we can determine recursively  $\phi_i s$  and  $\psi_i s$  as

$$\begin{aligned} \phi_0(s) &= (s^l + \alpha_1 s^{l-1} + \alpha_2 s^{l-2} + \dots + \alpha_l)q(s) \\ &\quad + s[\alpha_{l+1} s^{l-2} + \alpha_{l+2} s^{l-4} + \dots + \alpha_k]p(s) \\ &\quad + [\alpha_k + 1s^l + \alpha_k + 2s^{l-2} + \dots + \alpha_{2l+1}]p(s) \\ &= q_0(s) + [\alpha_k + 1s^l + \alpha_k + 2s^{l-2} + \dots + \alpha_{2l+1}] \\ &\quad \times p_0(s) \\ \psi_1(s) &= \phi_0(s)\bar{p}_0(s) \\ &= \psi_{1e}(s^2) + s\psi_{1o}(s^2) \end{aligned}$$

$$\begin{aligned} \phi_1(s) &= \psi_{1o}(s^2) + s\psi'_{1o}(s^2) \\ &= q_1(s) + \alpha_1 p_1(s) \\ &\vdots \\ \psi_j(s) &= \phi_{j-1}(s)\bar{p}_{j-1}(s) \\ &= \psi_{je}(s^2) + s\psi_{jo}(s^2) \\ \phi_j(s) &= \psi_{jo}(s^2) + s\psi'_{jo}(s^2) \\ &= q_j(s) + \alpha_j p_j(s) \\ &\vdots \\ \phi_k(s) &= q_k(s) + \alpha_k p_k(s). \end{aligned}$$

Hence, at each step we can determine  $p_i$  and  $q_i$  for  $i = 0, 1, \dots, k$ . It is also possible to determine  $r_i s$  recursively, i.e.  $r_0 = \lfloor (n + l - \sigma(p_0))/2 \rfloor$  and  $r_i = \lfloor (|2r_{i-1} - \sigma(p_{i-1})| - 1)/2 \rfloor$  for  $i = 1, 2, 3, \dots, k$ . At the  $j$ th step of the algorithm as  $q_j(s)$ ,  $p_j(s)$  and  $r_j$  are all known, we can determine  $\alpha_j$  using Algorithm 2.

## 7. Conclusions

We have presented a computational method to determine the set of all stabilizing controllers with

an arbitrary but fixed order for a given plant. The method consists essentially of a learned search in a subset of the controller parameter space. This subset is a substantially narrowed down version of the controller parameter space and is obtained by using our results on a semi-analytic method of determining all stabilizing constant feedback gains, applied to a number of subsidiary plants. Stabilization being the most basic requirement in most controller design problems, an inventory of all stabilizing controllers of a given order is most convenient for searching, among such controllers, those that satisfy further performance criteria, such as those imposed on unit-step response, closed-loop system frequency response, or  $H_\infty$ -norm of certain transfer functions. If one is able to translate a design requirement into a constraint on the controller parameters, then our method easily accommodates the incorporation of that requirement into the design. Otherwise, a further search in the admissible subset of the parameter space, i.e. the subset that corresponds to the stabilizing controllers, needs be performed.

The application of our result, given in § 5, to stabilization of uncertain systems is just one example of how further requirements can be incorporated into the choice of controllers. Other examples given in Saadaoui (2003) illustrates applications to finding controllers that give a desired degree of damping in unit-step response or that lead to the smallest  $H_\infty$ -norm for disturbance-to-output transfer function, and the like. The future direction in this research is then, incorporation of yet other design specifications into our algorithm that computes stabilizing fixed order controllers.

The main motivation for considering fixed-order controllers of course comes from the desire to reduce controller complexity and to determine as low order a controller as possible for a given high-order plant. There are mainly three approaches to the problem of reducing controller complexity: (i) Design a high-order controller first and then approximate it with a low-order one (see, e.g. Anderson and Liu 1989). (ii) Reduce the order of the plant model so that a low-order controller is easier to find (see, e.g. a survey in Antoulas *et al.* 2001). (iii) Fix the order of the controller

and search parameters that achieve a specified performance, as we have done in this paper. In view of the fact that methods in the category of (i) or (ii) are still at the stage of development, the tool we have presented in this paper will be of great help in designing low-order controllers.

## References

- B.D.O. Anderson and Y. Liu, "Controller reduction: concepts and approaches", *IEEE Transactions on Automatic Control*, 1989, 34, 802–812.
- A.C. Antoulas, D.C. Sorenson and S. Gugercin, "A survey of model reduction methods for large-scale systems", *Contemporary Mathematics*, 2001, 280, 193–219.
- B.R. Barmish, *New Tools for Robustness of Linear Systems*, New York: Macmillan Publishing Company, 1994.
- R.A. Brualdi, "From the editor-in-chief", *Linear Algebra and its Applications*, 2000, 320, 214–215.
- A. Datta, M.T. Ho and S.P. Bhattacharyya, *Structure and Synthesis of PID Controllers*, New York: Springer-Verlag, 2000.
- R.C. Dorf and R.H. Bishop, *Modern Control Systems*, 9th edn, New Jersey: Prentice Hall, 2001.
- F.R. Gantmacher, *The Theory of Matrices, Vol. II*, New York: Chelsea Publishing Company, 1959.
- M.T.A. Ho, A. Datta and S.P. Bhattacharyya, "Design of P, PI and PID controllers for interval plants", *Proceedings of American Control Conference*, 1998.
- N. Munro and M.T. Söylemez, "Fast calculation of stabilizing PID controllers of uncertain parameter systems", *Proceedings of ROCOND'2000*, Prague, Czechoslovakia, 2000.
- N. Munro, M.T. Söylemez and H. Baki, "Computation of D-stabilizing low-order compensators", Report, Control Systems Centre, UMIST, 1999.
- Y.I. Neimark, *Stability of Linearized Systems*, Leningrad Aeronautical Engineering Academy, Leningrad, Russia, 1949.
- A.B. Özgüler and A.A. Koçan, "An analytic determination of stabilizing feedback gains", Report, Institut für Dynamische Systeme, Report no. 321, Universität Bremen, 1994.
- C.L. Phillips and R.D. Harbor, *Feedback Control Systems*, 4th edn, New Jersey: Prentice Hall, 2000.
- K. Saadaoui, *Fixed order controller design via parametric methods*. PhD Thesis, Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800 Turkey, 2003.
- K. Saadaoui and A.B. Özgüler, "Computation of stabilizing first-order controllers for interval plants", *Proceedings of IEEE International Conference on Signals, Systems, Decision and Information Technology*, Tunisia, 2003.
- M.T. Söylemez, N. Munro and H. Baki, "Fast calculation of stabilizing PID controllers", *Automatica*, 2003, 45, 56–65.
- H. Xu, A. Datta and S.P. Bhattacharyya, "Computation of all stabilizing PID gains for digital control systems", *IEEE Transactions Automatic Control*, 2001, 46, 647–652.