ARTICLE IN PRESS

Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Systems & Control Letters ■■■ (■■■■) ■■■–■■■

SYSTEMS & CONTROL LETTERS

www.elsevier.com/locate/sysconle

On the complete instability of interval polynomials☆

F. Dabbene[a], B.T. Polyak[b], R. Tempo[a,*]

[a] *IEIIT-CNR, Politecnico di Torino, C.so Duca degli Abruzzi 24, 10129 Torino, Italy*
[b] *Institute for Control Science RAS 65 Profsoyuznaya Str., 117997 Moscow, Russia*

Received 18 May 2006; received in revised form 9 November 2006; accepted 16 November 2006

Abstract

In this paper, we study "complete instability" of interval polynomials, which is the counterpart of classical robust stability. That is, the objective is to check if *all* polynomials in the family are unstable. If not, a subsequent goal is to find a stable polynomial. To this end, we first propose a randomized algorithm which is based on a (recursive) necessary condition for Hurwitz stability. The second contribution of this paper is to provide a probability-one estimate of the volume of stable polynomials. These results are based on a combination of deterministic and randomized methods. Finally, we present two numerical examples and simulations showing the efficiency of the proposed methodology for small and medium-size problems.

Keywords: Robust stability; Randomized algorithms; Interval polynomials

1. Introduction

In this paper, we consider an interval polynomial of the form

$$p(s,k) = k_0 + k_1 s + k_2 s^2 + \cdots + k_n s^n, \tag{1}$$

where $k_i \in [k_i^-, k_i^+]$, $k_i^- > 0$, $i = 0, 1, \ldots, n$. The celebrated theorem of Kharitonov, see [8], states that $p(s,k)$ is Hurwitz if and only if four specific vertex polynomials are Hurwitz.

We now turn our attention to the design counterpart of this result, which has been called the *complete instability* problem, see [12]. The jargon "one-in-a-box" problem has been also used within the parametric stability community to denote it, see, for instance, [2].

To be more precise, we pose the following question: Is the interval polynomial completely unstable or there exists at least one Hurwitz polynomial in the family $p(s,k)$? If the answer to this question is affirmative, then the objective is to find such a Hurwitz polynomial. The one-in-a-box problem is a special case of fixed-order stabilization and static output feedback, and is deemed to be NP-hard even though a definitive assessment is presently not known. Negative results regarding the number of high-dimensional faces of the hyperrectangle

$$\mathcal{K} \doteq \{k : k_i \in [k_i^-, k_i^+],\ \ i = 0, 1, \ldots, n\}$$

required to be checked have been shown in [12]. In [3], a useful parameterization of stable polynomials is given. This parameterization is based upon a specific construction that leads to fixed-order proper controllers for unity feedback systems.

To evidence the difficulty of this problem, in [11] it is shown that, for a polynomial of order $n \geqslant 3$ with coefficients restricted in the interval [0, 1], the volume of stable polynomial $\mathcal{V}_{\text{stab}}$ is bounded by

$$\mathcal{V}_{\text{stab}} \leqslant \frac{1}{((n+1)/2)!}.$$

Following previous research on mixed methods for fixed-order controller design, see e.g. [6], and in the spirit of the randomized algorithms literature, see, for instance, [14] and references therein, we propose a mixed deterministic/

☆ The results of this paper have been obtained within a joint bilateral project between Consiglio Nazionale delle Ricerche, CNR of Italy, and Institute for Control Science, Russian Academy of Sciences of Russia. The financial support is gratefully acknowledged.

* Corresponding author. Tel.: +39 011 564 5408; fax: +39 011 564 5429.
E-mail addresses: fabrizio.dabbene@polito.it (F. Dabbene), boris@ipu.rssi.ru (B.T. Polyak), tempo@polito.it (R. Tempo).

0167-6911/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.sysconle.2006.11.002

randomized approach to solve the one-in-a-box problem. We assume that the odd coefficients of $p(s,k)$ are random variables distributed in the intervals $k_{2i+1} \in [k_{2i+1}^-, k_{2i+1}^+]$, $i = 0, 1, \dots, n_o$, where $n_o = (n-1)/2$ if n is odd or $n_o = n/2 - 1$ if n is even. That is, letting

$$k_o \doteq [k_1, k_3, \dots, k_{2n_o+1}]^T,$$

we define the *odd* hyperrectangle

$$\mathscr{K}_o \doteq \{k_o : k_{2i+1} \in [k_{2i+1}^-, k_{2i+1}^+],\ i = 0, 1, \dots, n_o\}.$$

We also assume that the even coefficients of $p(s,k)$ vary in the intervals $k_{2i} \in [k_{2i}^-, k_{2i}^+]$, $i = 0, 1, \dots, n_e$, where $n_e = n/2$ if n is even or $n_e = (n-1)/2$ if n is odd. Formally, letting

$$k_e \doteq [k_0, k_2, \dots, k_{2n_e}]^T,$$

we define the *even* hyperrectangle

$$\mathscr{K}_e \doteq \{k_e : k_{2i} \in [k_{2i}^-, k_{2i}^+],\ i = 0, 1, \dots, n_e\}.$$

In this case, no probabilistic assumption is made or used for the coefficients k_e.

For the odd coefficients k_o, a new powerful randomized technique is proposed. This technique is based on a necessary condition for stability in terms of a special recursion which involves only the coefficients k_{2i-3}, k_{2i-1}, and k_{2i+1}, $i = 2, 3, \dots, n_o$. The condition leads to a very efficient recursive algorithm which outperforms standard Monte Carlo methods based on randomization of independent identically distributed (iid) points within a hyperrectangle.

On the other hand, for the even coefficients k_e we use deterministic methods which require to check if the intersection of the hyperrectangle $\mathscr{K}_e$ and a polyhedral cone is non-empty. In such a case, specific points within the intersection can be immediately computed solving one single linear program.

The results of the paper allow to find a Hurwitz polynomial $p(s,k) = p(s, k_e, k_o)$ within the interval polynomial family, if one exists. In addition, we also estimate the volume of the set of Hurwitz interval polynomials. This estimate is obtained using a new technical result, and requires standard methods for computing the volume of polytopes.

The same approach can be followed when randomized methods are utilized for the even coefficients and deterministic methods are used for the odd coefficients, thus leaving some flexibility in the design procedure. The paper finally includes some simulations showing that the method works very well for polynomials of order up to 16. Rejection rates and numerical comparisons with other techniques are also given. These comparisons show the superiority of the approach proposed here.

2. Randomized methods for odd coefficients

We write the polynomial $p(s,k)$ in terms of even and odd polynomials with coefficient vectors k_e and k_o. That is, for $s = \jmath\omega$, we write

$$p(\jmath\omega, k) = p_e(\omega^2, k_e) + \jmath\omega p_o(\omega^2, k_o),$$

where

$$p_e(\omega^2, k_e) \doteq u_e^T(\omega^2) k_e,$$
$$p_o(\omega^2, k_o) \doteq u_o^T(\omega^2) k_o,$$

and

$$u_e^T(\omega^2) = [1, -\omega^2, \omega^4, \dots, (-1)^{n_e}\omega^{2n_e}],$$
$$u_o^T(\omega^2) = [1, -\omega^2, \omega^4, \dots, (-1)^{n_o}\omega^{2n_o}].$$

As discussed in the Introduction, we assume that the odd coefficients of $p(s,k)$ are random variables distributed in the intervals $k_{2i+1} \in [k_{2i+1}^-, k_{2i+1}^+]$, $i = 0, 1, \dots, n_o$.

We now state a necessary condition for stability of the interval polynomial $p(s,k)$ which is used in the randomized algorithm presented subsequently.

Theorem 1 (*Necessary condition for stability*). *Consider the odd polynomial $p_o(\omega^2, k_o)$, with fixed $k_o \in \mathscr{K}_o$. Then, a necessary condition for Hurwitz stability of the interval polynomial $p(s,k)$, $k_e \in \mathscr{K}_e$, and k_o fixed, is that*

$$k_{2i+1} \leqslant C(i, n_o) \frac{k_{2i-1}^2}{k_{2i-3}}, \quad i = 2, 3, \dots, n_o, \tag{2}$$

where

$$C(i, n_o) \doteq \frac{i-1}{i} \frac{n_o - i + 1}{n_o - i + 2}. \tag{3}$$

Proof. Due to the Hermite–Biehler Theorem, see e.g. [2], a necessary condition for a polynomial to be stable is that its odd part $p_o(\omega^2, k_o)$ has only real positive distinct roots

$$\omega_1^2 < \omega_2^2 < \dots < \omega_{n_o}^2.$$

The proof is completed observing that (2) is a necessary condition for a polynomial to have positive real roots, see Lemma 3 in the Appendix. □

Remark 1. As stated in the proof of Theorem 1, the Hermite–Biehler Theorem provides a necessary and sufficient condition for the stability of the whole polynomial $p(s,k)$ based on the positivity of the roots of the odd polynomial $p_o(s,k)$ only. Additional details regarding necessary conditions for a polynomial to have all real distinct roots can be found in [11]. Notice that, for $n_o \leqslant 2$, condition (2) is also sufficient for the polynomial $p_o(\omega^2, k_o)$ to have real positive distinct roots.

Other necessary conditions for stability, based on the coefficients of the whole polynomial $p(s,k)$, such as those proposed in [10], can in principle be used within our framework. However, Theorem 1 allows us to treat in a (more-efficient) deterministic way the remaining even coefficients.

The idea exploited in the randomized algorithm below is that generating samples in the entire odd hyperrectangle $\mathscr{K}_o$ is superfluous, and we only need to generate samples within a subset of the odd hyperrectangle defined by Theorem 1. Furthermore, the recursion given in this result is used in order to generate

Please cite this article as: F. Dabbene, et al., On the complete instability of interval polynomials, Systems Control Lett. (2006), doi: 10.1016/j.sysconle.2006.11.002

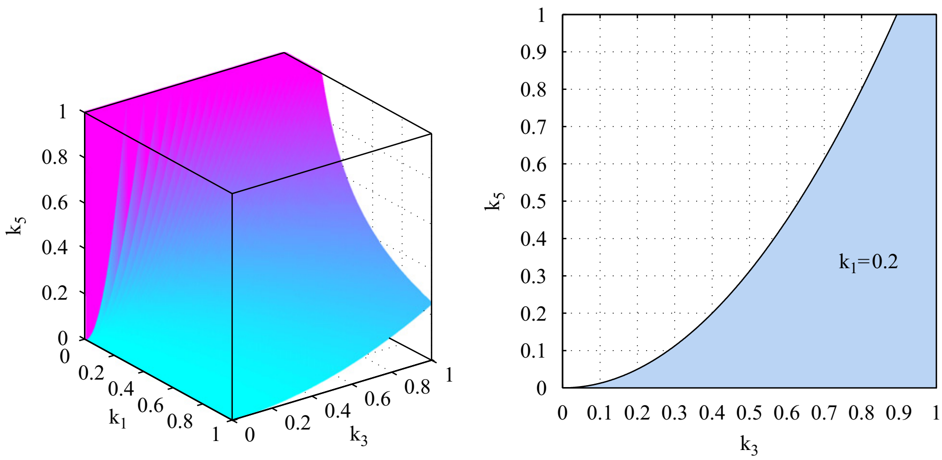

Fig. 1. Plot of the sets $\mathscr{K}_o^{nec}$ and $\mathscr{K}_o$ for $n_o = 2$.

Table 1
Estimated ratio between the volumes of the sets $\mathscr{K}_o^{nec}$ and $\mathscr{K}_o$ based on 100, 000 samples

Degree of odd polynomial n_o	Estimated ratio $\hat{\eta}(n_o)$
2	0.02794
3	0.00113
4	2e − 5
$\geqslant 5$	0.00000

samples sequentially. More precisely, we define the set

$$\mathscr{K}_o^{nec} \doteq \left\{ k_o \in \mathscr{K}_o : k_{2i+1} \leqslant C(i, n_o) \frac{k_{2i-1}^2}{k_{2i-3}}, i = 2, 3, \ldots, n_o \right\}.$$

Fig. 1 shows a plot of the set $\mathscr{K}_o^{nec}$ for $n_o = 2$, and its cross-section for $k_1 = 0.2$. In Table 1, we report the (estimated) ratio $\hat{\eta}(n_o)$ between the volumes of the sets $\mathscr{K}_o^{nec}$ and $\mathscr{K}_o$, for $\mathscr{K}_o = [0.001,\ 1]^{n_o}$. This ratio is obtained by randomization generating uniform random samples in the set $\mathscr{K}_o$. It can be seen that this ratio approaches zero very rapidly, and is negligible for $n_o \geqslant 4$. The conclusion is therefore immediate: a randomized algorithm based on random samples within the set $\mathscr{K}_o^{nec}$ outperforms classical randomized algorithms based on uniform independent sample generation within the odd hyperrectangle.

An algorithm for generating samples within $\mathscr{K}_o^{nec}$ is presented next. We remark that the samples drawn by Algorithm 1 are not uniformly distributed in $\mathscr{K}_o^{nec}$. However, the algorithm is based on an *importance sampling* technique which provides a variance reduction. This is discussed in detail in Section 4.

Algorithm 1. *Generates a vector of coefficients* $k_o \in \mathscr{K}_o^{nec}$.

1. `generate` k_1, k_3 `uniformly in` $[k_1^-, k_1^+], [k_3^-, k_3^+]$,
2. `for` $i = 2, 3, \ldots, n_o$
   (a) `construct the interval`

$$I_{2i+1} \doteq \left[k_{2i+1}^-, \min \left\{ k_{2i+1}^+, C(i, n_o) \frac{k_{2i-1}^2}{k_{2i-3}} \right\} \right]$$

   (b) `if` I_{2i+1} `is empty go to 1`
   (b) `else generate` k_{2i+1} `uniformly in the interval` I_{2i+1}.

3. Deterministic methods for even coefficients

Once the odd coefficients k_o of the polynomial $p(s, k)$ are obtained using the method proposed in the previous section, the even coefficients k_e can be easily determined solving one linear program. To show this fact, we state the following theorem. For a closely related result, see Ref. [15].

Theorem 2. *Suppose that* $k_o \in \mathscr{K}_o$ *is generated according to Algorithm* 1. *Then, the set* $\mathscr{H}_e$ *of all* $k_e \in \mathscr{K}_e$ *providing a Hurwitz interval polynomial* $p(s, k)$ *is either empty or is given by*

$$\mathscr{H}_e = \mathscr{K}_e \cap \mathscr{C}_e, \tag{4}$$

where $\mathscr{C}_e$ *is the interior of a polyhedral cone and is defined as*

$$\mathscr{C}_e \doteq \{k_e : V_e k_e < \mathbf{0}\}$$

$\mathbf{0}$ *being the zero vector and*

$$V_e \doteq \begin{bmatrix} -u_e^T(0) \\ u_e^T(\omega_1^2) \\ -u_e^T(\omega_2^2) \\ \vdots \\ (-1)^{n_o} u_e^T(\omega_{n_o}^2) \end{bmatrix}.$$

Proof. The Hermite–Biehler Theorem, see e.g. [2], can be stated as follows: a polynomial $p(s, k)$ is Hurwitz if and only if the roots of the polynomials $p_o(\omega^2, k_o)$ and $p_e(\omega^2, k_e)$ are all real, distinct, interlacing, and have the same sign. We observe that for $\omega = 0$, $p_o(0, k_o) = k_0 > 0$. Hence, the condition can be written as: $p_o(\omega^2, k_o)$ has all positive, distinct, real roots $0 < \omega_1^2 < \omega_2^2 < \cdots < \omega_{n_o}^2$ and $p_e(\omega^2, k_e)$

at these roots has alternating signs given by $p_e(0, k_e) > 0$, $p_e(\omega_1^2, k_e) < 0$, $p_e(\omega_2^2, k_e) > 0, \ldots$. The proof is completed observing that this is exactly the condition defining $\mathscr{C}_e$. □

Clearly, checking if the polytope $\mathscr{H}_e$ is empty can be accomplished solving one linear feasibility program. If $\mathscr{H}_e$ is non-empty, then a specific coefficient vector $k_e \in \mathscr{H}_e$ can be easily computed. For example, we could use the analytic center of the inequalities defining (4), see e.g. [1] for definition. The conclusion is that the polynomial $p(s, k_e, k_o)$ having odd coefficients k_o given by Algorithm 1 and even coefficients k_e provided by Theorem 2 is Hurwitz.

Next, we state the proposed algorithm for generating a stable polynomial in the interval family (1).

Algorithm 2. *Generates a vector of coefficients $k \in \mathscr{K}$ such that $p(s, k)$ is Hurwitz.*

1. generate k_o according to Algorithm 1
2. if the polynomial $p_o(\omega^2, k_o)$ has all real positive distinct roots
   (a) using the roots $\omega_1^2 < \omega_2^2 < \cdots < \omega_{n_o}^2$ construct V_e and the set

   $$\mathscr{C}_e = \{k_e : V_e k_e < \mathbf{0}\}$$

   (b) if the set $\mathscr{H}_e = \mathscr{K}_e \cap \mathscr{C}_e$ is non-empty, let k_e be the analytic center of $\mathscr{H}_e$ and return (k_o, k_e)
   (b) else go to 1.

4. On the volume of stable polynomials

The objective of this section is to estimate the volume $\mathscr{V}_{\text{stab}}$ of stable polynomials with coefficients in $\mathscr{K}$. Formally, this volume is defined as

$$\mathscr{V}_{\text{stab}} = \int_{\mathscr{H}} \mathrm{d}k,$$

where

$$\mathscr{H} = \{k \in \mathscr{K} : p(s, k) \text{ is Hurwitz}\}.$$

To this end, we propose a randomized algorithm which requires samples of k_o provided by Algorithm 1 and relies on a deterministic computation of the volume of the polytope $\mathscr{H}_e$.

Remark 2 (*Volume of a polytope*). It is well known that the exact computation of the volume of a polytope is a computationally difficult problem, see e.g. [4]. Nevertheless, algorithms which work "reasonably well" for medium-size problems have been developed since the early eighties, see [9]. A different approach followed in this literature is to develop efficient polynomial-time randomized algorithms for computing the volume of convex bodies, see [5]. In this case, clearly, we obtain only probabilistic estimates of the volume.

In order to compute the volume of $\mathscr{H}$, we propose an algorithm which is based on a multidimensional extension of a particular Monte Carlo technique for volume estimation. For the sake of simplicity, in the next section we discuss this technique for the two-dimensional case.

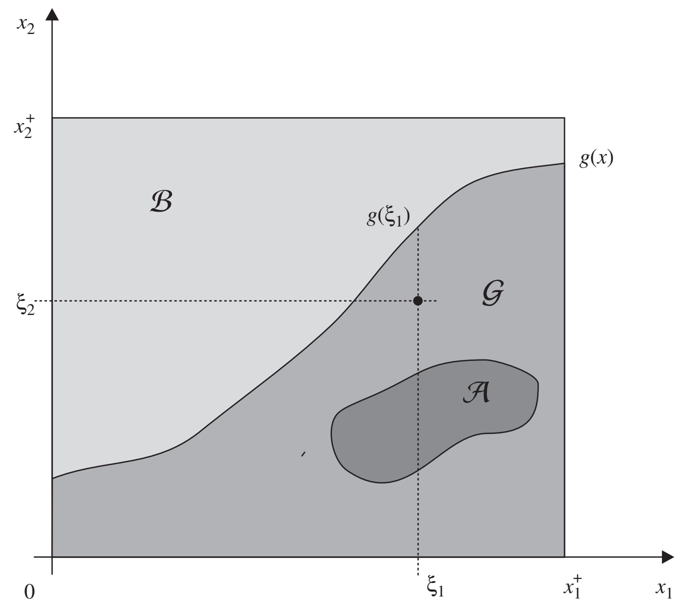

Fig. 2. Two-dimensional plot of the proposed algorithm for computing a weighted volume. A random sample ξ_1 is drawn uniformly in $[0, x_1^+]$, and the second component ξ_2 is drawn uniformly in the interval $[0, g(\xi_1)]$.

4.1. Randomized volume estimation

In this section, we discuss the problem of estimating efficiently the (weighted) volume $\mathscr{V}_{\mathscr{A}}$ of a two-dimensional bounded set $\mathscr{A} \subseteq \mathbb{R}^2$. The results presented are instrumental to the construction of a probabilistic estimate of the volume of stable polynomials $\mathscr{V}_{\text{stab}}$, discussed in Section 4.2. Let the volume of $\mathscr{A}$ be defined as

$$\mathscr{V}_{\mathscr{A}} \doteq \int_{\mathscr{A}} w(\xi)\,\mathrm{d}\xi, \qquad (5)$$

where $w(\cdot)$ is a given weighting function. Assume that $\mathscr{A} \subseteq \mathscr{G} \subseteq \mathscr{B}$, where

$$\mathscr{B} \doteq \{x \in \mathbb{R}^2 : x_1 \in [0, x_1^+],\ x_2 \in [0, x_2^+]\}$$

and

$$\mathscr{G} \doteq \{x \in \mathbb{R}^2 : x_1 \in [0, x_1^+],\ x_2 \in [0, g(x_1)]\},$$

where $g(\cdot) : \mathbb{R}^2 \to \mathbb{R}$ is a measurable function which takes positive values in the interval $[0, x_1^+]$. For simplicity, we also assume $g(x_1) \leqslant x_2^+$. This situation is depicted in Fig. 2.

Following the same philosophy proposed in Algorithm 1, we generate a random sample ξ_1^ℓ uniformly in $[0, x_1^+]$, and ξ_2^ℓ uniformly in $[0, g(x_1)]$. Then, we define the (weighted) indicator function of the set $\mathscr{A}$ as

$$\mathscr{X}_{\mathscr{A}}(x) \doteq \begin{cases} w(x) & \text{if } x \in \mathscr{A}, \\ 0 & \text{otherwise.} \end{cases}$$

We have the following lemma.

Lemma 1. *With the notation above, with probability one the volume of* $\mathscr{A}$ *is given by*

$$\mathscr{V}_{\mathscr{A}} = \lim_{N\to\infty} \frac{1}{N} \sum_{\ell=1}^{N} \mathscr{X}_{\mathscr{A}}(\xi^\ell)\, x_1^+ g(\xi_1^\ell). \tag{6}$$

Proof. First, notice that we can write the density function of a random variable ξ generated according to the proposed method as $f_\xi(\xi)=f_1(\xi_1)f_2(\xi_2|\xi_1)$ where $f_1(\xi_1)$ is the marginal density of ξ_1 given by

$$f_1(\xi_1) \doteq \begin{cases} \dfrac{1}{x_1^+} & \text{if } \xi_1 \in [0, x_1^+], \\ 0 & \text{otherwise} \end{cases}$$

and $f_2(\xi_2|\xi_1)$ is the conditional density

$$f_2(\xi_2|\xi_1) \doteq \begin{cases} \dfrac{1}{g(\xi_1)} & \text{if } \xi_2 \in [0, g(\xi_1)], \\ 0 & \text{otherwise.} \end{cases}$$

By definition of expected value, we have

$$\begin{aligned} \mathbb{E}\{x_1^+ g(\xi_1)\mathscr{X}_{\mathscr{A}}(\xi)\} &= \int \mathscr{X}_{\mathscr{A}}(\xi)\, x_1^+ g(\xi_1)\, f_\xi(\xi)\,\mathrm{d}\xi \\ &= \int_0^{x_1^+}\int_0^{g(\xi_1)} \mathscr{X}_{\mathscr{A}}(\xi)\,\mathrm{d}\xi = \mathscr{V}_{\mathscr{A}}, \end{aligned}$$

where the last equality follows from the fact that $\mathscr{A} \in \mathscr{G}$. The statement of the lemma is validated using the strong law of large numbers, see, for instance, [14]. □

4.2. Computation of $\mathscr{V}_{\text{stab}}$

We now return to the problem of estimating the volume $\mathscr{V}_{\text{stab}}$ of the set $\mathscr{H}$. To this end, we first introduce some useful notation. Let k_{o}^ℓ be the ℓth vector sample k_{o} generated by Algorithm 1. We denote by

$$I_{2i+1}^\ell, \quad i = 0, 1, \ldots, n_{\text{o}}$$

the intervals I_{2i+1} corresponding to k_{o}^ℓ. Moreover, with a slight abuse of notation, we let

$$I_1^\ell \doteq [k_1^-, k_1^+], \quad I_3^\ell \doteq [k_3^-, k_3^+].$$

We denote by $|I_{2i+1}^\ell|$ the lengths of the intervals, if they are non-empty, otherwise we let $|I_{2i+1}^\ell| = 0$. Similarly, we define the set $\mathscr{H}_{\text{e}}(k_{\text{o}}^\ell)$ as the set $\mathscr{H}_{\text{e}}$ given in (4) corresponding to k_{o}^ℓ, and we denote by $\mathscr{V}_{\mathscr{H}_{\text{e}}}(k_{\text{o}}^\ell)$ its volume. Finally, let $\mathscr{H}_{\text{o}}$ be the set of polynomial coefficients $k_{\text{o}} \in \mathscr{K}_{\text{o}}$ such that $p(s, k_{\text{o}})$ has all distinct positive roots.

We are now in the position of defining the following weighted indicator function:

$$\mathscr{X}_{\mathscr{H}_{\text{o}}}(k_{\text{o}}) \doteq \begin{cases} \mathscr{V}_{\mathscr{H}_{\text{e}}}(k_{\text{o}}) & \text{if } x \in \mathscr{H}_{\text{o}}, \\ 0 & \text{otherwise.} \end{cases}$$

Theorem 3. *Suppose that the vectors* k_{o}^ℓ, $\ell = 1, \ldots, N$, *are generated according to Algorithm* 1. *Then, with probability one the volume of stable polynomials* $\mathscr{V}_{\text{stab}}$ *is given by*

$$\mathscr{V}_{\text{stab}} = \lim_{N\to\infty} \frac{1}{N} \sum_{\ell=1}^{N} \mathscr{X}_{\mathscr{H}_{\text{o}}}(k_{\text{o}}^\ell) \prod_{i=0}^{n_{\text{o}}} |I_{2i+1}^\ell|.$$

The proof, not reported here, is a multidimensional extension of Lemma 1 and easily follows from induction arguments. In particular, the quantity $\prod_{i=0}^{n_{\text{o}}}|I_{2i+1}^\ell|$ represents the multidimensional extension of $x_1^+ g(\xi_1^\ell)$ in (6), while $\mathscr{X}_{\mathscr{H}_{\text{o}}}(\cdot)$ plays the role of $\mathscr{X}_{\mathscr{A}}(\cdot)$.

4.3. Connections with importance sampling

In this section, we highlight the connections of the proposed algorithm for computing the volume of stable polynomials with a classical variance reduction technique known as *importance sampling*, see, for instance, [13].

For simplicity, we return to the two-dimensional case discussed in Section 4.1, and consider unit weighting $w(x) \equiv 1$ in (5). We first remark that, in principle, the volume $\mathscr{V}_{\mathscr{A}}$ of the set $\mathscr{A} \subseteq \mathscr{B}$ could be estimated using a direct Monte Carlo method:

$$\mathscr{V}_{\mathscr{A}} \simeq \frac{1}{N}\mathscr{V}_{\mathscr{B}} \sum_{\ell=1}^{N} \mathscr{X}_{\mathscr{A}}(\xi^\ell),$$

where the samples ξ^ℓ are drawn *uniformly*[1] in $\mathscr{B}$ and $\mathscr{V}_{\mathscr{B}} \doteq \int_{\mathscr{B}} \mathrm{d}\xi$. In other words, we may use the estimate

$$\eta_{\mathscr{U}} \doteq \mathscr{V}_{\mathscr{B}}\mathscr{X}_{\mathscr{A}}(\xi), \quad \xi \sim \mathscr{U}_{\mathscr{B}}(\xi) \tag{7}$$

$\mathscr{U}_{\mathscr{B}}(\xi)$ being the uniform density on $\mathscr{B}$ defined as

$$\mathscr{U}_{\mathscr{B}}(\xi) \doteq \begin{cases} \dfrac{1}{\mathscr{V}_{\mathscr{B}}} & \text{if } \xi \in \mathscr{B}, \\ 0 & \text{otherwise.} \end{cases}$$

In fact, the expected value of $\eta_{\mathscr{U}}$ is equal to the volume of $\mathscr{A}$, i.e.

$$\mathbb{E}\{\eta_{\mathscr{U}}\} = \mathscr{V}_{\mathscr{A}}.$$

On the other hand, in Lemma 1 we introduce a *weighting pdf* $f_\xi(\xi)$, writing

$$\mathscr{V}_{\mathscr{A}} = \int \mathscr{X}_{\mathscr{A}}(\xi)\,\mathrm{d}\xi = \int \frac{\mathscr{X}_{\mathscr{A}}(\xi)}{f_\xi(\xi)} f_\xi(\xi)\,\mathrm{d}\xi,$$

where $f_\xi(\xi)$ is the density of the samples drawn according to Algorithm 2 (see proof of Lemma 1 for details), and use the importance sampling estimate

$$\eta_{\text{is}} \doteq v(\xi)\mathscr{X}_{\mathscr{A}}(\xi), \quad \xi \sim f_\xi(\xi) \tag{8}$$

with $v(\xi) = 1/f_\xi(\xi)$. Notice in fact that $v(\xi) = x_1^+ g(\xi_1)$ for $\xi \in \mathscr{A}$.

This is exactly the variance reduction techniques proposed in importance sampling, see [13]. In particular, in the following

[1] In the "one-in-a-box" setting, this corresponds to generate uniform samples k_{o}^ℓ in the set $\mathscr{K}_{\text{o}}$.

Please cite this article as: F. Dabbene, et al., On the complete instability of interval polynomials, Systems Control Lett. (2006), doi: 10.1016/j.sysconle.2006.11.002

lemma, we show that the variance of estimate (8) improves upon the variance of classical Monte Carlo estimate (7).

Lemma 2. *With the notation above, the variance of estimate* (8) *is smaller than the variance of the Monte Carlo estimate* (7), *that is*

$$\mathrm{Var}(\eta_{\mathrm{is}}) \leqslant \mathrm{Var}(\eta_{\mathscr{U}}).$$

Proof. We first notice that $\mathbb{E}\{\eta_{\mathscr{U}}\} = \mathbb{E}\{\eta_{\mathrm{is}}\} = \mathscr{V}_{\mathscr{A}}$, therefore it is sufficient to show that

$$\mathbb{E}\{\eta_{\mathrm{is}}^2\} \leqslant \mathbb{E}\{\eta_{\mathscr{U}}^2\}.$$

This is immediately seen, since

$$\begin{aligned}\mathbb{E}\{\eta_{\mathrm{is}}^2\} &\doteq \int \left(\frac{\mathscr{X}_{\mathscr{A}}(\xi)}{f_\xi(\xi)}\right)^2 f_\xi(\xi)\,\mathrm{d}\xi \\ &= \int \frac{\mathscr{X}_{\mathscr{A}}^2(\xi)}{f_\xi(\xi)}\,\mathrm{d}\xi = \int x_1^+ g(\xi_1)\mathscr{X}_{\mathscr{A}}^2(\xi)\,\mathrm{d}\xi \\ &\leqslant \int \mathscr{V}_{\mathscr{B}}\mathscr{X}_{\mathscr{A}}^2(\xi)\,\mathrm{d}\xi \\ &= \int \mathscr{V}_{\mathscr{B}}^2\mathscr{X}_{\mathscr{A}}^2(\xi)\frac{1}{\mathscr{V}_{\mathscr{B}}}\,\mathrm{d}\xi = \int \eta_{\mathscr{U}}^2 \mathscr{U}_{\mathscr{B}}(\xi)\,\mathrm{d}\xi \\ &= \mathbb{E}\{\eta_{\mathscr{U}}^2\}. \qquad \square\end{aligned}$$

5. Numerical examples

5.1. Example 1

In the first example, we studied the problem of finding a stable polynomial in a box and of computing a probability-one estimate of the volume of stability. To this end, we considered the following interval polynomial of degree five:

$$p(s,k) = [1,5] + [1,5]s + [4,8]s^2 + [6,10]s^3 + [4,8]s^4 + [6,10]s^5.$$

First, we constructed the four Kharitonov polynomials, see [8] for details:

$$\begin{aligned} p_1(s) &\doteq k_0^- + k_1^- s + k_2^+ s^2 + k_3^+ s^3 + k_4^- s^4 + k_5^- s^5 \\ &= 1 + s + 8s^2 + 10s^3 + 4s^4 + 6s^5, \\ p_2(s) &\doteq k_0^+ + k_1^+ s + k_2^- s^2 + k_3^- s^3 + k_4^+ s^4 + k_5^+ s^5 \\ &= 5 + 5s + 4s^2 + 6s^3 + 8s^4 + 10s^5, \\ p_3(s) &\doteq k_0^+ + k_1^- s + k_2^- s^2 + k_3^+ s^3 + k_4^+ s^4 + k_5^- s^5 \\ &= 5 + s + 4s^2 + 10s^3 + 8s^4 + 6s^5, \\ p_4(s) &\doteq k_0^- + k_1^+ s + k_2^+ s^2 + k_3^- s^3 + k_4^- s^4 + k_5^+ s^5 \\ &= 1 + 5s + 8s^2 + 6s^3 + 4s^4 + 10s^5. \end{aligned}$$

The four polynomials are all unstable. Hence, we pose the question: Is the family robustly unstable or there exists a stable polynomial in the box $\mathscr{K}$? To answer this question, we resort to the procedure proposed in the paper. First notice that, in this case, $n_o = 2$, and therefore condition (2) is necessary and sufficient for the odd polynomial $p_o(\omega^2, k_o)$ to possess distinct positive real roots. Algorithm 1 converged in one iteration[2] to the vector of coefficients

$$k_o = [3.1951 \;\; 9.7263 \;\; 6.4700]^{\mathrm{T}}. \tag{9}$$

The resulting polynomial $p_o(\omega^2, k_o)$ has two real positive roots $\omega_1^2 = 1.0184$, $\omega_2^2 = 0.4849$. Therefore, the set $\mathscr{C}_e$ of Theorem 2 is defined by the three inequalities

$$\begin{aligned} &-k_0 < 0, \\ &k_0 - 0.4849k_2 + 0.2352k_4 < 0, \\ &-k_0 + 1.0184k_2 - 1.0371k_4 < 0. \end{aligned}$$

The set $\mathscr{H}_e = \mathscr{K}_e \cap \mathscr{C}_e$ is represented in Fig. 3. Any point in the set $\mathscr{H}_e$ corresponds to a stable polynomial. In particular, we chose the point

$$k_e = [1.4282 \;\; 6.6994 \;\; 6.3374]^{\mathrm{T}} \tag{10}$$

which is the analytic center of the set. Putting together (9) and (10), we obtained the polynomial

$$p(s,k) = 6.4700s^5 + 6.3374s^4 + 9.7263s^3 + 6.6994s^2 + 3.1951s + 1.4282$$

having stable roots

$$\begin{aligned} \lambda_1 &= -0.6297, \\ \lambda_{2,3} &= -0.0772 \pm 0.9083\jmath, \\ \lambda_{4,5} &= -0.0977 \pm 0.6421\jmath. \end{aligned}$$

Then, for statistical purposes, we ran Algorithm 2 for 10, 000 trials. In all the runs, the algorithm was able to find a stable polynomial in the box in at most four iterations, and in the 99% of the cases it converged at the first one.

Finally, using Theorem 3, we computed a probability-one estimate of the volume of stability with $N = 5000$ samples obtaining

$$\mathscr{V}_{\mathrm{stab}} = 18.7154.$$

Notice that, assuming a uniform measure in the box $\mathscr{K}$, this volume corresponds to a very small probability of stability of 0.0046.

To compare with standard Monte Carlo approximation methods, we considered the following running average version of the volume estimation proposed in Theorem 3:

$$\widehat{\mathscr{V}}_{\mathrm{stab}}^{\ell+1} = \frac{\ell}{\ell+1}\widehat{\mathscr{V}}_{\mathrm{stab}}^{\ell} + \frac{1}{\ell+1}\mathscr{X}_{\mathscr{H}_o}(k_o^\ell)\prod_{i=0}^{n_o}|I_{2i+1}^\ell|.$$

The convergence of this estimate was compared to standard volume estimate based on rejection from the box. The result, depicted in Fig. 4, shows that the proposed methodology has faster convergence.

[2] One iteration corresponds to the generation of one vector of coefficients k_o and the solution of one linear feasibility program for checking that $\mathscr{C}_e$ is non-empty.

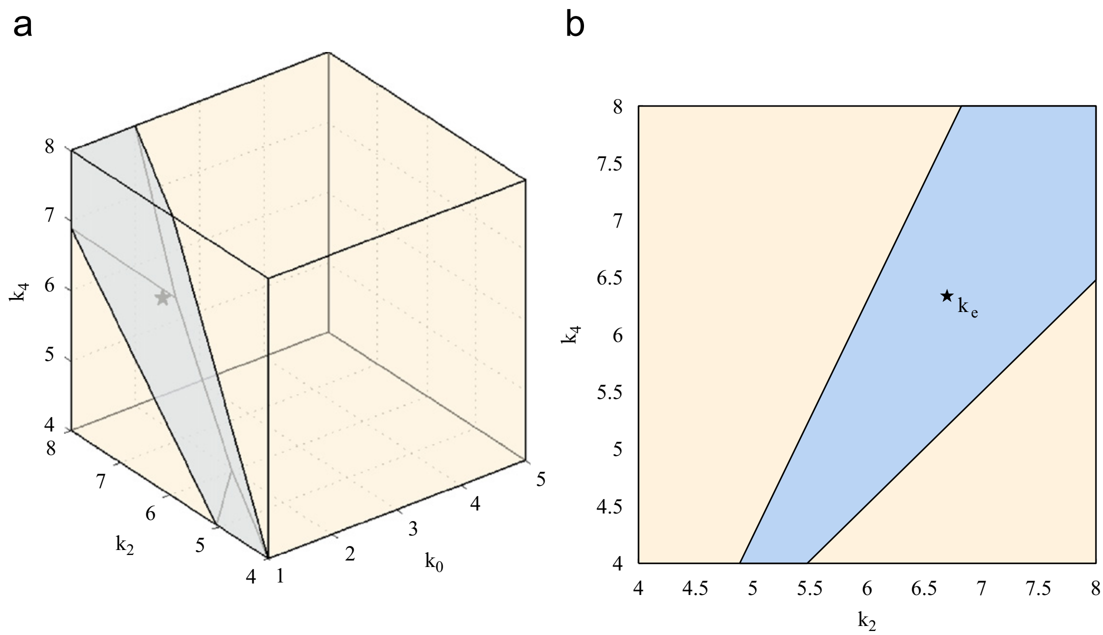

Fig. 3. (a) Set $\mathscr{H}_e = \mathscr{K}_e \cap \mathscr{C}_e$ for Example 1. The star represents the analytic center of the set. (b) Section of $\mathscr{H}_e$ at the analytic center ($k_0 = 1.4282$).

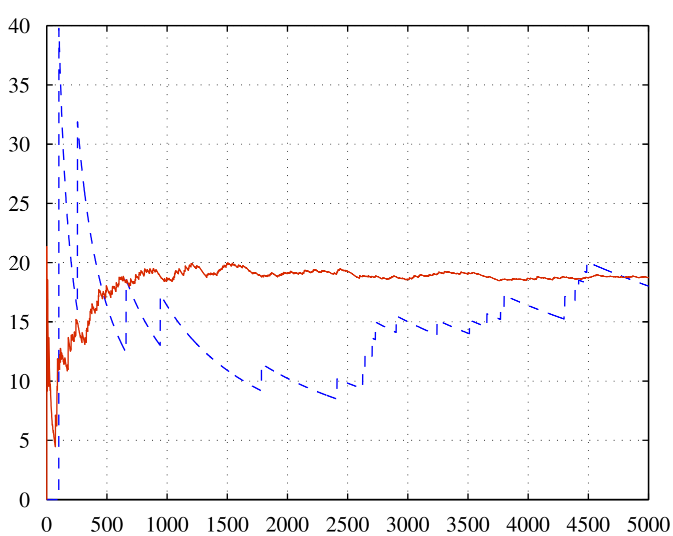

Fig. 4. Estimate of the volume of stability $\mathscr{V}_{stab}$. Solid line: running average version of Theorem 3. Dashed line: standard Monte Carlo estimation based on rejection from the box.

Table 2
Number of sample generations necessary for finding a stable polynomial of degree n

Degree of polynomial	Expected number of iterations	
	Algorithm 2	Standard MC
3	1	2
4	1	62
5	1	50
6	1	1108
7	2	28,334
8	2	∞
9	3	∞
10	5	∞
11	7	∞
12	34	∞
13	80	∞
14	626	∞
15	4099	∞
16	6461	∞
17	76,968	∞
18	90,093	∞

5.2. Example 2

To test the efficiency of the proposed approach for finding stable polynomials in an interval family, we constructed a benchmark interval polynomial of degree n with coefficients ranging in the box

$$\mathscr{K} = \{k : k_i \in [\alpha, 1],\ i = 0, \ldots, n\}$$

with $\alpha = 10^{-10}$. For increasing values of n, we ran Algorithm 2 and computed the number of iterations (number of sample generations of Algorithm 1) necessary for finding a stable polynomial.

In Table 2 we report the expected number of iterations estimated over 1000 different runs of the algorithm, compared with the expected number of iterations required using a fully randomized approach based on rejection from the box $\mathscr{K}$. The symbol ∞ means that no stable polynomial was found after 10^5 sample generations. The table shows that the method is very efficient for polynomials of degree up to 16, and it works reasonably well for polynomials of degree 17 and 18.

6. Conclusion

In this paper, we studied a mixed deterministic/randomized approach for the complete instability of interval polynomials. An interesting feature of this method is the exploitation of very efficient randomized algorithms suitably tuned on the specific stability problem.

Future research will be devoted to studying extensions in various directions including, in particular, stability of discrete time system. In this case, odd and even polynomials can be replaced by the symmetric and antisymmetric parts of the polynomial:

$$p(z,k) = p_{\rm sym}(z,k) + p_{\rm asym}(z,k)$$

with $p_{\rm sym}(z,k) \doteq \frac{1}{2}(p(z,k) + z^n p(z^{-1},k))$ and $p_{\rm asym}(z,k) \doteq \frac{1}{2}(p(z,k) - z^n p(z^{-1},k))$. However, a critical step that remains is the development of recursive necessary conditions similar to those presented in Theorem 1 of this paper.

Appendix

We state a technical lemma due to Newton, see [7, Theorem 51].

Lemma 3 (*Necessary condition for real roots*). *If a polynomial*

$$p(s) = a_0 + a_1 s + \cdots + a_n s^n$$

with non-vanishing coefficients $a_0, a_n \neq 0$, *has all real roots, then*

$$a_{i-1}a_{i+1} \leqslant C(i,n)a_i^2 \tag{11}$$

where $C(i,n)$ *is defined in* (3).

Proof. If $p(s)$ has all real roots, then its derivative $p'(s)$ also has all real roots. In fact, between two subsequent real roots of p there is a real root of p'. On the other hand, the polynomial

$$q(s) \doteq s^n p(1/s) = a_n + a_{n-1}s + \cdots + a_0 s^n$$

has also n real roots, which are the reciprocal of the roots of p. Notice that the differentiation of the polynomial $q(s)$ is equivalent to a "backward" differentiation of the original polynomial $p(s)$. After $(n-i-1)$ "forward" differentiations and $(i-1)$ "backward" differentiations of the polynomial $p(s)$, we obtain a second degree polynomial $d_0 + d_1 s + d_2 s^2$, whose coefficients depend on the original coefficients of $p(s)$. By construction, this polynomial should have all real roots. This is equivalent to require $d_1^2 \geqslant 4d_0 d_2$. Rewriting this inequality in terms of the original coefficients $a_0, \ldots, a_n$, by lengthy but straightforward computations involving binomial coefficients, we obtain (11). □

References

[1] E.W. Bai, Y. Ye, R. Tempo, Bounded error parameter estimation: a sequential analytic center approach, IEEE Trans. Automat. Control 44 (1999) 1107–1117.
[2] B.R. Barmish, New Tools for Robustness of Linear Systems, Macmillan, New York, 1994.
[3] T.E. Djaferis, D.L. Pepyne, C.M. Cushing, A new parameterization of stable polynomials, IEEE Trans. Automat. Control 47 (2002) 1546–1550.
[4] M.E. Dyer, A.M. Frieze, The complexity of computing the volume of a polyhedron, SIAM J. Comput. 17 (1988) 967–974.
[5] M.E. Dyer, A.M. Frieze, R. Kannan, A random polynomial-time algorithm for approximating the volume of convex bodies, J. ACM 38 (1991) 1–17.
[6] Y. Fujisaki, Y. Oishi, R. Tempo, Randomized and deterministic algorithms for stabilization with fixed order controllers, in: Proceedings of the American Control Conference, Portland, 2005.
[7] G.H. Hardy, J.E. Littlewood, G. Polya, Inequalities, Cambridge University Press, Cambridge, 2001 (first printing—1934).
[8] V.L. Kharitonov, Asymptotic stability of an equilibrium position of a family of systems of linear differential equations, Differentsial'nye Uraveniya 14 (1978) 1483–1485.
[9] J.B. Lasserre, An analytical expression and an algorithm for the volume of a convex polyhedron in $\mathbf{R}^n$, J. Optim. Theory and Appl. 39 (1983) 363–377.
[10] A.V. Lipatov, N.I. Sokolov, Some sufficient conditions for stability and instability of continuous linear stationary systems, Automat. Remote Control 39 (1979) 1258–1291.
[11] A. Nemirovski, B. Polyak, Necessary conditions for stability of polynomials and their use, Automat. Remote Control 55 (1994) 1644–1649.
[12] P. Padmanabhan, C.V. Hollot, Complete instability of a box of polynomials, IEEE Trans. Automat. Control 37 (1992) 1230–1233.
[13] R.Y. Rubinstein, Monte-Carlo Optimization Simulation and Sensitivity of Queueing Networks, Wiley, New York, 1986.
[14] R. Tempo, G. Calafiore, F. Dabbene, Randomized Algorithms for Analysis and Control of Uncertain Systems, Communications and Control Engineering Series, Springer, London, 2005.
[15] H. Lin, C.V. Hollot, Results on positive pairs of polynomials and their application to the construction of stability domains, Int. J. Control 46 (1987) 153–159.

Please cite this article as: F. Dabbene, et al., On the complete instability of interval polynomials, Systems Control Lett. (2006), doi: 10.1016/j.sysconle.2006.11.002