

POLYNOMIAL TOOLBOX 2.0 ROBUST CONTROL OF SYSTEMS WITH PARAMETRIC UNCERTAINTIES

Michael Šebek¹,

PolyX, s.r.o.²

info@polyx.cz - www.polyx.cz - www.polyx.com

ABSTRACT

Polynomial Toolbox is a new MATLAB toolbox for systems, signals and control analysis and design using polynomial methods. See www.polyx.cz for more details. This paper demonstrates on several examples how the Polynomial Toolbox solves various robust control problems for plants with parametric uncertainties.

INTRODUCTION

Modern control theory addresses various problems involving uncertainty. In a large class of practical design problems, the uncertainty can be traced down to particular coefficients of the plant transfer matrix. It usually originates from various physical parameters whose values are specified only within given bounds. An ideal solution of these problems is to find a robust controller – a simple, fixed and off-line designed controller that nevertheless guarantees desired behavior and/or stability for all expected values of the uncertain parameters.

The Polynomial Toolbox offers several simple tools that are useful for robust control analysis and design for the systems with parametric uncertainties. Some of them are briefly introduced in this paper. The underlying methods as well as other solutions that can also be built from the Polynomial Toolbox macros are described in Barmish (1996), Bhattacharyya, Chapellat and Keel (1995) and other textbooks.

SINGLE PARAMETER UNCERTAINTY

Many systems of practical interest depend just on a single uncertain parameter. At the time of design, the parameter is only known to fall within a given interval. Quite often even more complex problems (with more complex uncertainty structure) can be reduced to the single parameter case. Even though the uncertain parameter is single, it may well appear in several coefficients of the transfer matrix at the same time.

To analyze an uncertain polynomial $p(s, q) = 3 + (10 + q)s + 12s^2 + (6 + q)s^3 + s^4$ with a single uncertain parameter $q \in \mathbb{R}$, first check whether $p(s, q)$ is stable for $q = 0$. Then find its left-sided and right-sided stability margins, that is, the smallest negative q_{\min} and the largest positive q_{\max} such that $p(s, q)$ remains stable for any

¹ The author is also with UTIA Prague and CVUT Prague.

² PolyX, s.r.o., Jarní 4, 16000 Praha 6, tel. 0603844561, fax. 02-6884554

$q \in (q_{\min}, q_{\max})$. With the Polynomial Toolbox, this is an easy task: First express the given polynomial as $p(s, q) = p_0(s) + qp_1(s)$ and enter the data

```
p0 = 3 + 10*s + 12*s^2 + 6*s^3 + s^4, p1 = s + s^3
p0 =
    3 + 10s + 12s^2 + 6s^3 + s^4
p1 =
    s + s^3
```

Then type

```
isstable(p0)
ans =
    1
```

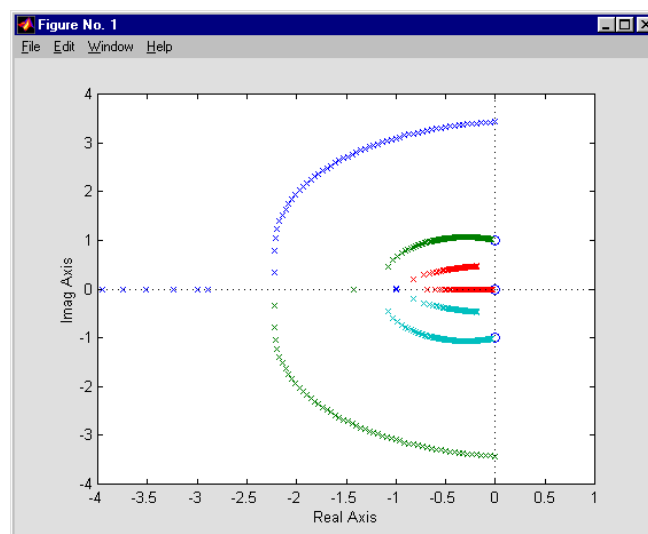
to verify nominal stability (that is, the case of $q = 0$) and, finally, call

```
[qmin,qmax]=stabint(p0,p1)
qmin =
   -5.6277
qmax =
    Inf
```

to get the desired stability margins. This result discloses that $p(s, q)$ is not merely stable for $q = 0$, but also for all $q \in (-5.6277, \infty)$. When $q = -5.6277$, the stability is lost.

Combining the Polynomial Toolbox and Control System Toolbox, the result can be visualized by plotting root-locus for a fictitious plant $p_1(s)/p_0(s)$ under a fictitious feedback gain q ranging $(-5.6277, \infty)$. Type

```
rlocus(ss(p1,p0),qmin:.1:100)
```



to demonstrate that, indeed, all the roots of $p(s, q) = p_0(s) + qp_1(s)$ stay inside of stability region for all $q \in \mathcal{A} = [0.6277, 100]$.

INTERVAL POLYNOMIALS

Another important class of uncertain systems is those described by interval polynomials where uncertainties in different coefficients are considered to be independent. An interval polynomial looks like

$$p(s, q) = \sum_{i=0}^n [q_i^-, q_i^+] s^i$$

with $[q_i^-, q_i^+]$ denoting the bounding interval for the i -th coefficient. In the

Polynomial Toolbox, the interval polynomial is described by the 'lower' and 'upper' members

$$p^-(s) = \sum_{i=0}^n q_i^- s^i \quad \text{and} \quad p^+(s) = \sum_{i=0}^n q_i^+ s^i .$$

In many applications the interval polynomials arise when an original uncertainty structure is known but too complex (e.g., highly nonlinear) to be tractable yet it can be overbounded by simple interval once an independent uncertainty structure is imposed.

Checking robust stability is easy for interval polynomials of continuous-time nature: such an interval polynomial is robustly if and only if just certain four 'extreme' polynomials (called Kharitonov) are stable. For example, consider the *continuous-time* interval polynomial (Barmish, 1996)

$$p(s, q) = [0.45, 0.55] + [1.95, 2.05]s + [2.95, 3.05]s^2 + [5.95, 6.05]s^3 + [3.95, 4.05]s^4 + [3.95, 4.05]s^5 + s^6$$

To test it using the Polynomial Toolbox, we first enter it via two 'lumped' polynomials

```
pminus = 0.45+1.95*s+2.95*s^2+5.95*s^3+3.95*s^4+3.95*s^5+s^6;
ppplus = 0.55+2.05*s+3.05*s^2+6.05*s^3+4.05*s^4+4.05*s^5+s^6;
```

Then the robust stability is checked simply by typing

```
[stability]=kharit(pminus,ppplus)
stability =
    1
```

In fact, the macro verifies that all the four polynomials are stable and we conclude that the interval polynomial is robustly stable.

On the other hand, no Kharitonov-like results are available for *discrete-time* polynomials (of degree 4 and higher). Yet the following graphics method can be applied. Consider the interval polynomial

$$p(z, q) = [10, 20] + [20, 30]z + [128, 138]z^2 + [260, 270]z^3 + 168z^4 .$$

To test its robust stability, we write

$$p(z, q) = p_0(z) + q_1 p_1(z) + q_2 p_2(z) + q_3 p_3(z) + q_4 p_4(z) ,$$

where $p_0(z) = 10 + 20z + 128z^2 + 260z^3 + 168z^4$, $p_1(z) = 1$, $p_2(z) = z$, $p_3(z) = z^2$ and $p_4(z) = z^3$. Such an expression is called *polytopic*. It enables us to describe each interval coefficient by a separate uncertain parameter q_i ranging $[0,10]$. To analyze the interval polynomial, we enter the data

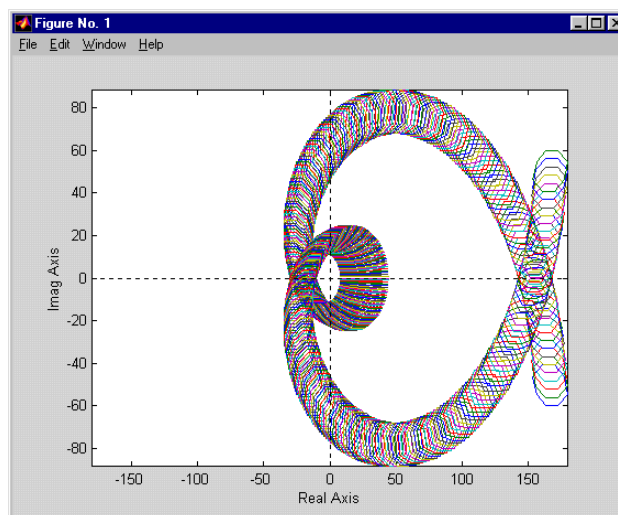
```
p0 = 10 + 20*z + 128*z^2 + 260*z^3 + 168*z^4;
p1=1,p2=z,p3=z^2,p4=z^3, Qbounds=[1 10;1 10;1 10;1 10];
```

and check that p_0 is stable

```
isstable(p0)
ans =
    1
```

Then we plot the sets (called value sets) $p(c, q)$ for $q \in Q_{\text{bounds}}$ and the generalized frequency c sweeping around unit circle. To better recognize what is happening in the neighborhood of the critical point 0, we only show here a zoomed figure for generalized frequencies $e^{j\omega}$ in the critical range $\omega \in [0.6\pi, 1.4\pi]$:

```
ptopplot(p0,p1,p2,p3,p4,Qbounds,exp(j*(0.3:0.001:0.7)*2*pi))
```



We can see that the point zero is excluded from all the octagons ($0 \notin p(c, q)$ for all c on the unit circle) and we can conclude that the discrete-time interval polynomial is robustly stable.

POLYTOPES OF POLYNOMIALS

More general class of systems is described by uncertain polynomials that linearly depend on several parameters, but each parameter may occur simultaneously in several coefficients. Such an uncertain polynomial may look like

$$p(s, q) = \sum_{i=1}^n a_i(q) s^i$$

where each coefficient $a_i(q)$ is an affine linear function of q . Uncertain polynomials with this uncertainty structure form polytopes in the space of polynomials. Similarly to the single parameter case, they can always be expressed as

$$p(s, q) = p_0(s) + q_1 p_1(s) + q_2 p_2(s) + \dots + q_n p_n(s) .$$

In the Polynomial Toolbox a polytope of polynomials having n parameters is always described by the $[n+1]$ polynomials $p_0(s), p_1(s), \dots, p_n(s)$ along with n parameter bounding intervals $[q_1^-, q_1^+], \dots, [q_n^-, q_n^+]$. The affine linear uncertainty structure is quite important as it is preserved under feedback interconnection. Unfortunately, there is not enough space here to present a full example. However, a special case of polytopic uncertainty was treated in the last example, where an interval polynomial was expressed and analyzed in this setting.

REFERENCES

- Barmish B. R. (1996). *New Tools for Robust stability of Linear Systems*. Macmillan, New York.
- Bhattacharyya S. P., Chapellat H. and Keel L.H. (1995). *Robust Control: The Parametric Approach*. Prentice Hall, Upper Saddle River, N.J.
- *Polynomial Toolbox 2: Manual* (1999). PolyX, Ltd., Prague, CZ. See www.polyx.cz or www.polyx.com .
- *Polynomial Toolbox 2: Commands* (1999). PolyX, Ltd., Prague, CZ. See www.polyx.cz or www.polyx.com

