Global Optimization of H_∞ problem: Application to robust control synthesis under structural constraint

Dominique Monnet[1], Jordan Ninin[1], and Benoit Clement[1]

ENSTA-Bretagne, LabSTIC, IHSEV team, 2 rue Francois Verny, 29806 Brest, France,
dominique.monnet@ensta-bretagne.fr, jordan.ninin@ensta-bretagne.fr,
benoit.clement@ensta-bretagne.fr

Abstract. In this paper, a new technique to synthesis structured Robust Control Law is developed. This technique is based on global optimization methods using a Branch-and-Bound algorithm. The original problem is reformulated as a min/max problem with non-convex constraint. Our approach uses techniques based on interval arithmetic to compute bounds and accelerate the convergence.

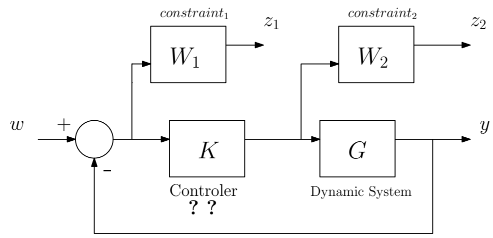
1 Context

Controlling an autonomous vehicle or a robot requires the synthesis of control laws for steering and guiding. To generate efficient control laws, a lot of specifications, constraints and requirements have been translated into norm constraints and then into an constraint feasibility problem. This problem has been solved, sometimes with relaxations, using numerical methods based on LMI (Linear Matrix Inequalities) or SDP (Semi Definite Program) [2, 3]. The main limitation of these approaches are the complexity of the controller for implementation in an embedded system. But, if a physical structure is imposed to the law control in order to make easier the implementation, the synthesis of this robust control law is much more complex. And this complexity has been identified as a key issue for several years. A efficient first approach was given by Apkarian and Noll based on local non-smooth optimization [1].

In this talk, we will present a new approach based on **global optimization** in order to generate **robust control laws**.

2 H_∞ control synthesis under structural constraints

We illustrate our approach on an example on the control of a periodic second order system G with a PID controller K subjected to two frequency constraints on the error e and on the command u of the closed-loop system. The objective is to find $\boldsymbol{k} = (k_p, k_i, k_d)$ to stabilize the closed-loop system and minimizing the H_∞ norm of the controlled system to ensure the robustness.



The H_∞ norm of a dynamic system P is defined as follow:

$$||P||_\infty = \sup_{\omega}(\sigma_{\max}(P(j\omega))),$$

with $\sigma_{\max}$ the greatest singular value of the transfert function P and j the imaginary unit.

In our particular case, the closed-loop system can be interpreted as two SISO systems (Single In Single Out). The H_∞ norm of a SISO system is the maximum of the absolute value of the transfer function. Indeed, to minimize the H_∞ norm of our example, we need to solve the following min/max problem:

$$\begin{cases} \min\limits_{\boldsymbol{k}} \max \left(\sup\limits_{\omega} \left| \dfrac{W_1(j\omega)}{1+G(j\omega)K(j\omega)} \right|, \sup\limits_{\omega} \left| \dfrac{W_2(j\omega)K(j\omega)}{1+G(j\omega)K(j\omega)} \right| \right), \\ \\ s.t. \qquad\qquad \text{The closed-loop system must be stable.} \end{cases} \tag{1}$$

The stability constraint of a closed-loop is well-known: the roots of denominator part of the transfer function $\frac{1}{1+G(s)K(s)}$ must have a non-positive real part [4]. Using Routh-Hurwitz stability criterion [5], this constraint can be reformulated as a set of non-convex constraints.

Proposition 1. *Let us consider a polynomial* $Q(s) = a_n s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0$. *The real part of its root are negative if the first column of the following table are positive:*

$v_{1,1} = a_n$	$v_{1,2} = a_{n-2}$	$v_{1,3} = a_{n-4}$	$v_{1,4} = a_{n-6}$	$\dots$
$v_{2,1} = a_{n-1}$	$v_{2,2} = a_{n-3}$	$v_{2,3} = a_{n-5}$	$v_{2,4} = a_{n-7}$	$\dots$
$v_{3,1} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,2} \\ v_{2,1} & v_{2,2} \end{vmatrix}$	$v_{3,2} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,3} \\ v_{2,1} & v_{2,3} \end{vmatrix}$	$v_{3,3} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,4} \\ v_{2,1} & v_{2,4} \end{vmatrix}$	$\dots$	$\dots$
$v_{4,1} = \frac{-1}{v_{3,1}} \begin{vmatrix} v_{2,1} & v_{2,2} \\ v_{3,1} & v_{3,2} \end{vmatrix}$	$v_{4,2} = \frac{-1}{v_{3,1}} \begin{vmatrix} v_{2,1} & v_{2,3} \\ v_{3,1} & v_{3,3} \end{vmatrix}$	$\dots$	$\dots$	$\dots$
$v_{5,1} = \frac{-1}{v_{4,1}} \begin{vmatrix} v_{3,1} & v_{3,2} \\ v_{4,1} & v_{4,2} \end{vmatrix}$	$\dots$	$\dots$	$\dots$	$\dots$
$\vdots$	$\ddots$	$\ddots$	$\ddots$	$\ddots$

Indeed, the H_∞ control synthesis under structural constraint is reformulated as a min/max problem with non-convex constraints.

3 Global optimization of min/max problems

In order to solve Problem (1), our approach is based on an Brand-and-Bound technique [7]. At each iteration, the domain under study is bisected to improve the computation of bounds. Boxes are eliminated if and only if it is certified that no point in the box can produce a better solution than the current best one, or that at least one constraint cannot be satisfied by any point in such a box.

The non-convex contraint can be handled with constraint programming technique. In our approach, we use the ACID algorithm [8] which reduces the width of the boxes and so accelerate the convergence of the branch-and-bound.

But, the key point concerns the computation of the bounds of the objective function. In our example, the objective function can be reformulate as the following expression:

$$f(x) = \sup_{\omega \in [\omega_{\min}, \omega_{\max}]} g(x, \omega). \tag{2}$$

At each iteration, Algorithm 1 is used to compute a lower bound of this function over a box $[\boldsymbol{x}]$. This algorithm is also a branch-and-bound algorithm based on Interval Arithmetic. But, for not wasting time, we limit the maximum number of iterations for computing faster lower bounds. Each element $([\boldsymbol{\omega}], ub_{\boldsymbol{\omega}})$ stored in $\mathcal{L}$ are composed with: (i) $[\boldsymbol{\omega}]$ a sub-interval of $[\boldsymbol{\omega}_{\min}, \boldsymbol{\omega}_{\max}]$ and (ii) $ub_{\boldsymbol{\omega}}$ an upper bound of g over $[\boldsymbol{x}] \times [\boldsymbol{\omega}]$.

Algorithm 1 Computation of bounds of f over a box $[\boldsymbol{x}]$

Require: g: the function under study (see Equation 2); $\boldsymbol{x}$: a initial box; $\mathcal{L}$: the list of boxes; *nbIter*: the maximal number of iterations.

Initialization: $(lb_{out}, ub_{out}) = (-\infty, \infty)$.
for nb := 1 **to** nbIter **do**
  Extract an element $(\boldsymbol{\omega}, ub_{\boldsymbol{\omega}})$ from $\mathcal{L}$.
  Bisect $\boldsymbol{\omega}$ into two sub-boxes $\boldsymbol{\omega}_1$ and $\boldsymbol{\omega}_2$.
  for i:=1 **to** 2 **do**
    Compute $lb_{\boldsymbol{\omega}_i}$ and $ub_{\boldsymbol{\omega}_i}$ a lower and an upper bound of $g(x, \boldsymbol{\omega})$ over $[\boldsymbol{x}] \times [\boldsymbol{\omega}_i]$ using Interval Arithmetic techniques [6].
    if $lb_{\boldsymbol{\omega}_i} > lb_{out}$ **then**
      $lb_{out} := lb_{\boldsymbol{\omega}_i}$, {Update the best lower bound}
      Remove from $\mathcal{L}$ all the element k such as $ub_{\boldsymbol{\omega}_k} < lb_{out}$,
    end if
    if $ub_{\boldsymbol{\omega}_i} > ub_{out}$ **then**
      $ub_{out} := ub_{\boldsymbol{\omega}_i}$, {Update the worst upper bound}
    end if
    if $ub_{\boldsymbol{\omega}_i} > lb_{out}$ **then**
      Add $(\boldsymbol{\omega}, ub_{\boldsymbol{\omega}_i})$ in $\mathcal{L}$,
    end if
  end for
end for
return (lb_{out}, ub_{out}): a lower and an upper bound of f over $\boldsymbol{x}$.

Thanks to Interval Analysis, at the end of Algorithm 1, we can ensure that the value of the maximum of f over $[\boldsymbol{x}]$ is include in $[lb_{out}, ub_{out}]$.

4 Application

In our example, we consider a second-order system and weighting functions W_1 and W_2 penalizing the error signal and control signal respectively:

$$G(s) = \frac{1}{s^2 + 1.4s + 1}, \ K(s) = k_p + \frac{k_i}{s} + \frac{k_d s}{1+s}.$$

$$W_1(s) = \frac{s+100}{100s+1}, \qquad W_2(s) = \frac{10s+1}{s+10}.$$

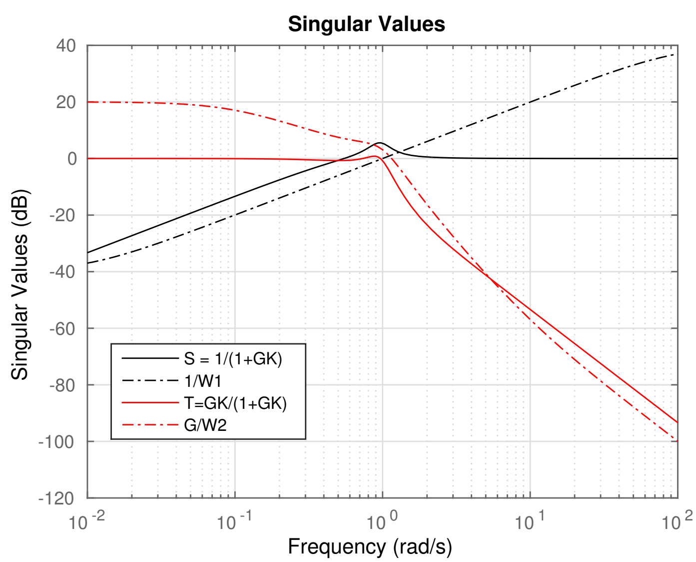
We want to find k_p, k_i and k_d the coefficients of the structured controller K such that the closed-loop system respects the constraints W_1^{-1} and W_2^{-1}. The control is bounded in $[-2, 2]$, and we limit the interval of ω to $[10^{-2}, 10^2]$.

Our algorithm gives the following result:

$$\max\left(\sup_\omega \left|\frac{W_1(j\omega)}{1+G(j\omega)K(j\omega)}\right|, \sup_\omega \left|\frac{W_2(j\omega)K(j\omega)}{1+G(j\omega)K(j\omega)}\right|\right) = 2.1414$$

$$\text{with } k_p = -0.0425,\ k_i = 0.4619,\ k_d = 0.2566$$

Unfortunately, the value of the solution of the min/max problem is greater than 1. So, the constraints W_1^{-1} and W_2^{-1} are not respected.



In this example, the main advantage of our global optimization approach is that unlike classical method based on non-smooth optimization, we can certify that no robust solution of our problem exists.

References

1. P. Apkarian and D. Noll. Nonsmooth h_∞ synthesis. *Automatic Control, IEEE Transactions on*, 51(1):71–86, Jan 2006.
2. D. Arzelier, B. Clement, and D. Peaucelle. Multi-objective h2/hinfinity/impulse-to-peak control of a space launch vehicle. *European Journal of Control*, 12(1), 2006.
3. S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, June 1994.
4. I. R. Petersen et R. Tempo. Robust control of uncertain systems: Classical results and recent developments. *Automatica*, 50(5), 2014.
5. A. Hurwitz. Ueber die Bedingungen, unter welchen eine Gleichung nur Wurzeln mit negativen reellen Theilen besitzt. *Mathematische Annalen*, 46(2):273–284, June 1895.
6. L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer-Verlag, London, 2001.
7. J. Ninin. *Optimisation Globale basée sur l'Analyse d'Intervalles : Relaxation Affine et Limitation de la Mémoire*. PhD thesis, Institut National Polytechnique de Toulouse, Toulouse, 2010.
8. G. Trombettoni and G. Chabert. Constructive interval disjunction. In *Principles and Practice of Constraint Programming–CP 2007*, pages 635–650. Springer, 2007.