Available online at www.sciencedirect.com
ScienceDirect
IFAC PapersOnline CONFERENCE PAPER ARCHIVE
IFAC-PapersOnLine 49-5 (2016) 345–352

Evolutionary Fixed-Structure Controller Tuning Against Multiple Requirements

Philippe Feyel*

**Sagem Défense Sécurité, Optronics & Defense division, France*
philippe.feyel@sagem.com

Abstract: This work deals with a new *Matlab* tool for computing fixed-structure controller against multiple requirements. Based on evolutionary computation, it allows solving control problems based on complex specifications, i.e. criterion whose gradients or sub gradients can't be easily formulated: thus it appears to be well adapted to the industrial framework. The tool is described and few examples show its efficiency in computing fixed-structure controller against complex specifications.

Keywords: Evolutionary Optimization, Robust Control, Controller Tuning, H-infinity, Fixed-Structure.

1. INTRODUCTION

In the industrial framework, the control engineer must design a unique control law that valid on a single prototype, with a sufficient degree of robustness to satisfy a complex high level close-loop specification on many systems. As shown in Fig. 1, the specification is very often in the form of requirements related to *m* any quantities (here noted Z_j).

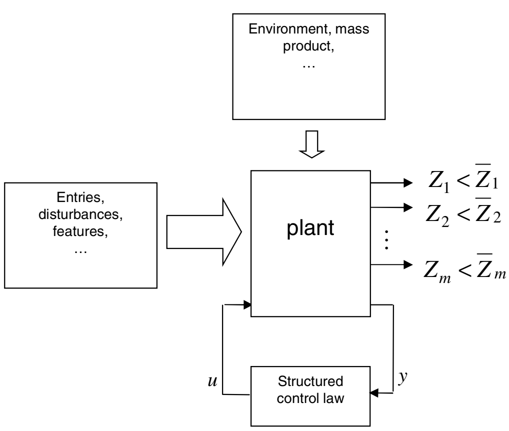

Fig. 1. Close-loop with high level specification

To assist in his task, H_∞ synthesis (Doyle *et al.*, 1989) has emerged as an efficient tool for robust control in the industrial community. However, this approach is well known to have two major drawbacks.

First, the choice of the "best" weighting filters, used to shape close-loop transfers, is a hard task, as they are supposed to express as close as possible numerous and complex specifications. For that purpose, a very time-consuming iterative procedure has to be used in the design process to get a controller satisfying all specifications.

Secondly, it is also well known that the order of H_∞ controllers is high: it is equal to that of the synthesis model and the weights. Thus the classical approach is to a posteriori simplify this controller but with the loss of guarantees on the overall optimality and the specification fulfilment.

In this paper, we try to make the methodology for computing controllers be more efficient and more direct with a less costly development time by calculating a structured controller directly optimized on the high level specification. For that purpose, we introduce *HinfStoch_ControllerTuning*, a new numerical *Matlab* tool for computing such controllers provided that high level specifications are in the form of *m* evaluable requirements $Z_1 < \overline{Z}_1, Z_1 < \overline{Z}_2, \ldots, Z_m < \overline{Z}_m$.

Here the optimization of complex criterion is made possible by the use of efficient evolutionary optimization dedicated to structured control problems (Feyel *et al.*, 2014). Indeed, such techniques do not need to formulate gradients, the only constraint being the capability of evaluating criterion. Due to stochastic optimization, the main drawback deals with the time consuming calculation, of course quite reasonable in comparison with the *try and error* iterative procedures used to tune weighting functions.

This tool can be viewed as a complementary tool to other powerful commercial tools such as *Looptune* (Gahinet *et al.*, 2011) and more recently *Systune* (Apkarian, 2013) both based on non-smooth optimization. Indeed, S*ystune* allows tuning fixed-structure controller towards predefined criterion (whose gradients or sub gradients are known) such as step time responses, gain of close-loop transfers, stability in multiple plant case, etc… Examples have shown its efficiency (Mathworks, 2014).

HinfStoch_ControllerTuning has key features which make its use very interesting in the industrial framework:

- Any kind of requirements may be optimized provided that they can be evaluated ; thus complex criterion such as in (Anderson, 2010) can be optimized,
- Experimental data (such as disturbances temporal records) can be directly used for the controller computation,

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.137

- Parallel computation (Luszczek, 2006) is well adapted to evolutionary computation and thus makes the controller computation time quite reasonable,
- Because all requirements are specified in the tool using any *Matlab* instructions, all features of *Matlab* can be used. For instance nothing prevents the user from coupling the tool with *Simulink* in order to evaluate requirements with servo-loops simulated directly in a non-linear framework.

We emphasize that for requirements already predefined in *Systune*, using *Systune* is preferable because much faster (although *HinfStoch_ControllerTuning* can yield to comparable results but in a longer time). In all other cases, using the tool introduced in this paper is interesting.

The paper is organized as follows. In section II, we briefly recall some theorical elements about the computation of fixed-structure controllers with evolutionary techniques. Then *HinfStoch_ControllerTuning* is introduced in section III, and examples of uses, showing the flexibility and the capability of the tool, are described in section IV.

2. THEORICAL BACKGROUND

2.1 Optimizing controllers against multiple requirements

Given a plant $P(s)$ depicted in Fig. 2 on which $e \in \Re^{n_e}$ is the external input vector including disturbances, sensor noise, references, $z \in \Re^{n_z}$ are the controlled outputs, $u \in \Re^{n_u}$ is the control input vector and $y \in \Re^{n_y}$ are the measured variables. The H$_\infty$ standard control problem consists at first in designing some weighting functions $W_o(s)$ and $W_i(s)$ to shape some closed-loop transfers to satisfy robustness and/or performance specifications ; then a stabilizing controller can be computed which minimizes $\|W_o(s)F_l(P,K)W_i(s)\|_\infty < \gamma$.

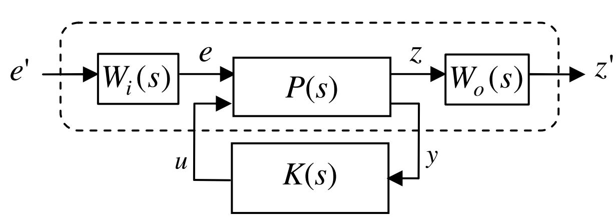

Fig. 2. Generalized control form with weights.

The main difficulty in this design relies obviously on the choice of well-suited pre- and post-compensators to handle the high-level specification requirements with good stability margins. Thus, in the full-order case, the tuning of weighing functions is a time consuming task which consists in solving the following problem:

$$\begin{array}{c}\min_{W_0(s),W_i(s)} \gamma \\ \text{subject to } Z_1 < \overline{Z}_1,\ldots,Z_m < \overline{Z}_m\end{array}, \quad (1)$$

where each Z_i is a high-level requirement.

The H$_\infty$ loop-shaping approach (McFarlane *et al.*, 1992) is a particular case of the previous H$_\infty$ control problem where weighting functions used to shape an open-loop target, are externalized as in (2) ($H(s)$ is the system to be controlled).

$$\left\| \begin{pmatrix} W_o(s) \\ W_i(s)^{-1}K(s) \end{pmatrix} S(s)\begin{pmatrix} W_o(s)^{-1} & H(s)W_i(s) \end{pmatrix} \right\|_\infty < \gamma . \quad (2)$$
$$S(s) = \left(I + H(s)K(s)\right)^{-1}$$

An important property of the H$_\infty$ loop-shaping approach is the direct link between γ and the generalized gain and phase margins (respectively noted ΔG and $\Delta\Phi$) (Vinnicombe, 2000). Indeed:

$$\Delta G \geq \frac{1+\gamma^{-1}}{1-\gamma^{-1}},\ \Delta\Phi \geq 2\operatorname{asin}\left(\gamma^{-1}\right). \quad (3)$$

Thus $\gamma = 3$ ensures that $\Delta G \geq 6$ dB and $\Delta\Phi \geq 40°$. Note that $\gamma > 1$ in this approach. Furthermore good values for ΔG and $\Delta\Phi$ don't imply a small γ value.

A drawback of the H$_\infty$ loop-shaping approach is the high order of computed controllers. Thus the question of simultaneously tuning the frequency weights and the final controller appears as a natural extension of the problem (1), that is:

$$\begin{array}{c}\min_{W_0(s),W_i(s),K(s)} \gamma \\ \text{subject to} \\ K(s) \text{ stabilizing and } Z_1 < \overline{Z}_1,\ldots,Z_m < \overline{Z}_m\end{array} . \quad (4)$$

In fact, optimizing the weights may be redundant with the optimization of the fixed-order final controller because at the end of the optimization process, the controller contains all the dynamical information needed to satisfy the specifications. However, the obtained value for γ is already considered as a robustness indicator relative to the weights with the relations (3) in the H$_\infty$ loop-shaping framework.

Thus, without loss of generality, we can simplify the problem (2) by using directly static weights D_i and D_o (like "scalings") in the place of the frequency weights $W_i(s)$ and $W_o(s)$, which leads to the scaled stabilization problem (5).

$$\left\| \begin{pmatrix} D_o \\ D_i^{-1}K(s) \end{pmatrix} S(s)\begin{pmatrix} D_o^{-1} & H(s)D_i \end{pmatrix} \right\|_\infty < \gamma . \quad (5)$$
$$S(s) = \left(I + H(s)K(s)\right)^{-1}$$

The optimization problem (4) simplified to (6).

$$\begin{array}{c}\min_{D_0,D_i,K(s)} \gamma \\ \text{subject to} \\ K(s) \text{ stabilizing and } Z_1 < \overline{Z}_1,\ldots,Z_m < \overline{Z}_m\end{array} . \quad (6)$$

Note that relations (3) remain satisfied.

A robust controller $K(s)$ aims to satisfy some complex specifications, which can be expressed by constraints such as