Improvement of the LMI change of variables for multi-objective control design problems

Emile Simon^{*1} and $\mathrm{Vincent} \ \mathrm{Wertz}^1$

¹Dpt. of Mathematical Engineering, ICTEAM Institute, Université Catholique de Louvain, 4 avenue Georges Lemaître, 1348 Louvain-la-Neuve, Belgium

Summary

The aim of this paper is to propose an improvement of a classical change of variables, useful to solve multi-objective control design problems that can be formulated with Linear Matrix Inequalities (LMIs). For multi-objective problems, the only approach guaranteed to converge to global optima is to use an iterative approach where the order of the design parameter grows (until no improvement is obtained to some relative decrease). The issue is that the order of the design parameter, and the size of associated LMIs, may easily grow to unacceptably large values (computationally and/or for implementation in practice). The change of variables considered here was proposed to mitigate that issue, i.e. to reduce the inflation of number of variables of the corresponding optimization problems. Here we propose a method exploiting this change of variables further, in order to obtain a sequence of increasing order design parameters converging faster towards the global optimum.

Keywords: Bilinear and Linear matrix inequalities (BMIs and LMIs), Youla parameterization, Multi-objective control

1 Introduction

The systems considered in this work are Linear finite-dimensional Time-Invariant (LTI) plants. The objectives addressed are control design objectives that admit a Linear Matrix Inequalities (LMIs) representation. For a direct comparison of the results of [1], discrete-time systems are considered. Note however that the technique can be straightforwardly translated to the continuous-time case.

For single objective problems, there exists a well-known change of variables requiring full-order controllers [2], leading to global optima. Here we consider multi-objective problems, for which that change of variables cannot be used (except with an additional constraint, which imply that the solutions thus found are suboptimal).

The current work proposes an improvement of the technique developed in [1]. It was already known at the time of that work how to solve LMI-representable multi-objective control problems toward globally optimal solutions: by optimizing Youla parameters confined to finite-dimensional subspaces of increasing dimensions, belonging to the infinite-dimensional space RH_{∞} of all proper and stable transfer matrices. By increasing the order of the Youla parameter and optimizing it, the sequence of objective values decreases monotonously towards the global optimum (which result was first proven in the SISO case in [3] and implemented for the MIMO case in [4]). The problem resided already then in the inflation of size of the optimization problem. The work of [1] introduced a change of variables, which allowed to considerably reduce the number of required variables compared to the previous solutions of [3, 4]. The aim of the current work is to push this contribution further.

^{*}Correspondence to Emile Simon, Tel. +32 10478032, Fax +32 10472180, Email: Emile.Simon@uclouvain.be

The drawback of the change of variables proposed in [1] is that the state-space matrices A_q and B_q of the Youla parameter Q must be fixed before each optimization. Because of this restriction, obtaining an objective value close to the global optimum will typically lead to a Youla parameter with a very large order. Such parameter is unpractical for implementation and requires to solve large LMIs which will imply optimization problems (computing time, limit of memory, numerical issues).

Starting from the technique proposed in [1], we develop a method lifting this restriction and (partially) removing its associated conservatism. This method is also solely based on solving LMIs and is summarized as follows: at each optimization of the Youla parameter of a given order, instead of solving only the LMI problem where the state-space matrices A_q and B_q are fixed, all four state-space matrices are further improved with an iterative LMI algorithm. This approach can only lead to better (or at least the same) objective values for each order of the Youla parameter than without this improvement, which implies that the sequence of increasing order Youla parameters converges faster with respect to the order to the global optimum (or at least as fast).

The structure of the paper is as follows. First the notations are given in Section 2, along with a reminder of the results of [1]. Then the contributed method is presented in Section 3. An example of obtained results compared to those of [1] is given in Section 4. The convergence of the algorithm is discussed in Section 5. Finally the conclusions are drawn.

2 Notations and previous results

2.1 Notations

We consider the LTI generalized plant P with the following state-space representation:

$$\begin{pmatrix} x^+ \\ \overline{z_j} \\ y \end{pmatrix} = \begin{pmatrix} A & B_i & B \\ \overline{C_j} & D_{ij} & E_j \\ C & F_i & 0 \end{pmatrix} \begin{pmatrix} x \\ \overline{w_i} \\ u \end{pmatrix}$$
(1)

where x is the state vector, w and z are respectively the input(s) and output(s) of the objective(s) channel(s), u and y are respectively the control input(s) and the measured output(s). The $^+$ denotes the value at the next sampling time. The indices i allow to designate each component of w and the indices j each component of z. P is of minimal realization with McMillan degree n. The controller is a LTI system represented by:

$$\left(\frac{x_c^+}{u}\right) = \left(\frac{A_c}{C_c} \frac{B_c}{D_c}\right) \left(\frac{x_c}{y}\right) \tag{2}$$

which must be designed to ensure good closed-loop properties. The closed-loop state-space representation is written as $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}$ for which the performance channel $T_{ij} : w_i \to z_j$ has the realization:

$$\begin{pmatrix} A & B_i \\ \hline C_j & D_{ij} \end{pmatrix} = \begin{pmatrix} A + BD_cC & BC_c \\ B_cC & A_c \\ \hline C_j + E_jD_cC & E_jC_c \\ \hline D_{ij} + E_jD_cF_i \end{pmatrix}$$
(3)

Note that it is assumed that (A, B) is stabilizable and (A, C) is detectable, which ensures the existence of stabilizing controllers. Now the famous $\mathcal{H}_2/\mathcal{H}_\infty$ multi-objective problem is recalled, which is the main example considered in [1] and the most typical multi-objective control problem. This objective is to impose two specifications on the closed-loop system which are to keep an upper bound γ_1 on the \mathcal{H}_2 norm of the first performance channel and an upper bound γ_2 on the \mathcal{H}_∞ norm of the second performance channel : $||T_{11}||_2 < \gamma_1$ and $||T_{22}||_\infty < \gamma_2$. The trade-off between these two objectives can be drawn by minimizing the functional $\gamma_1 + \lambda \gamma_2$ for several fixed values of the positive scalar λ .

The techniques developed in [1] and here rely on LMI optimization and thus require LMI - representable objectives. We then recall the matrix inequalities formulation of the discrete-time $\mathcal{H}_2/\mathcal{H}_{\infty}$ multi-objective. For the \mathcal{H}_2 norm: $||T_{11}||_2 < \gamma_1$ iff $\exists \mathcal{X}_1, Z$ s.t.: $tr(Z) < \gamma_1$,

$$\begin{pmatrix} \mathcal{X}_1 & * & * \\ 0 & \gamma_1 I & * \\ \mathcal{X}_1 \mathcal{A} & \mathcal{X}_1 \mathcal{B}_1 & \mathcal{X}_1 \end{pmatrix} \succ 0, \quad \begin{pmatrix} \mathcal{X}_1 & * & * \\ 0 & \gamma_1 I & * \\ \mathcal{C}_1 & \mathcal{D}_{11} & Z \end{pmatrix} \succ 0 \tag{4}$$

For the \mathcal{H}_{∞} norm: $||T_{22}||_{\infty} < \gamma_2$ iff $\exists \mathcal{X}_2$ s.t.:

$$\begin{pmatrix} \mathcal{X}_2 & * & * & * \\ 0 & \gamma_2 I & * & * \\ \hline \mathcal{X}_2 \mathcal{A} & \mathcal{X}_2 \mathcal{B}_2 & \mathcal{X}_2 & * \\ \mathcal{C}_2 & \mathcal{D}_{22} & 0 & \gamma_2 I \end{pmatrix} \succ 0$$
(5)

The optimization variables are the following:

- Lyapunov matrices $\mathcal{X}_1 = \mathcal{X}'_1 \succ 0, \mathcal{X}_2 = \mathcal{X}'_2 \succ 0$
- Controller state-space matrices A_c, B_c, C_c, D_c
- Objective scalars $\gamma_1, \gamma_2 > 0$ and matrices $Z_1 = Z'_1 \succ 0, Z_2 = Z'_2 \succ 0$

The notations tr denotes the trace, ' the transposed matrix, the * indicates a symmetrical term and $M \succ 0$ denotes that the matrix M is strictly positive definite (i.e. all its eigenvalues are strictly positive). Note that since only two performance channels are considered, the indices i, j and ij were replaced by either 1 or 2 as in [1]. These inequalities also ensure the Hurwitz stability of \mathcal{A} , thus the asymptotic stability of the closed-loop system. Remark that without changing the variables these matrix inequalities are Bilinear Matrix Inequalities (BMIs), because of the terms with products of the Lyapunov matrices and the design parameter state-space matrices. It is well known that problems involving BMIs are NP-hard in general [5], whereas solving LMIs can be done exactly and efficiently with interior point methods [6]. Hence particular techniques are used to turn these (non-convex) BMI problems into LMI reformulations or sub-problems. These transformations imply however in most cases the introduction of some conservatism, i.e. the loss of optimality of the solution.

For example, the $\mathcal{H}_2/\mathcal{H}_\infty$ problem already admitted a conservative full-order solution proposed in [7], using the extra technical constraint of a unique Lyapunov matrix $\mathcal{X} = \mathcal{X}_1 = \mathcal{X}_2$ and the change of variables introduced by [2] (requiring full-order control). This solution (the so-called *mixed controller*) is often used as an initial solution, which is the case in [1]. Note that the change of variables of [2] provides the global solution for the single-objective \mathcal{H}_2 or \mathcal{H}_∞ problems with a full-order controller.

We now recall some key elements presented in [1] that will be useful for the development of the contribution. The notations used in the current paper are consistent with those of that previous work.

2.2 Key previous results

The change of variables proposed in [1] requires to have the generalized plant P respecting a certain triangular structure. The adequate tool to enforce this condition is the Youla parameterization. This requires to choose an initial controller defined by two matrices K, L such that the matrices A + BK and A + LC are stable (i.e. Hurwitz in continuous time or Schur in discrete time), and then the original P can be extended into the new representation (so-called 'observer structure'):

$$\begin{pmatrix} z_j \\ \hat{y} \end{pmatrix} = \begin{pmatrix} A + BK & -BK & B_i & B \\ 0 & A + LC & B_i + LF_i & 0 \\ \hline C_j + E_j K & -E_j K & D_{ij} & E_j \\ 0 & C & F_i & 0 \end{pmatrix} \begin{pmatrix} w_i \\ \hat{u} \end{pmatrix}$$
(6)

Then a Youla parameter $Q \in \operatorname{RH}_{\infty}$, allowing to explore the set of all stabilizing controllers, is introduced as $\hat{u} = Q\hat{y}$ (more details are provided in [1, 8]). This extended P has the structure necessary for the development in [1]. For convenience, the following notations are used for the extended P representation:

$$\begin{pmatrix} z_j \\ \hat{y} \end{pmatrix} = \begin{pmatrix} A_1 & \hat{A} & B_{i1} & \hat{B} \\ 0 & A_2 & B_{i2} & 0 \\ \hline C_{j1} & C_{j2} & D_{ij} & E_j \\ 0 & \hat{C} & F_i & 0 \end{pmatrix} \begin{pmatrix} w_i \\ \hat{u} \end{pmatrix}$$
(7)

Note that [1] proposes a systematic (but not straightforward) choice to build K, L from the conservative mixed controller mentioned in the previous section. Such choice can also be used here, simpler alternatives being to use pole placement or linear quadratic design (e.g. with functions place and dlqr in Matlab C). This choice is actually that of the initial solution and is not a point of interest here. Indeed, our contribution is a technique to converge faster than [1] to the global optimum for increasing orders of the Youla parameter.

Now we can replace the state-space matrices of the original P by those of the extended P and the A_c, B_c, C_c, D_c matrices of the controller C by the A_q, B_q, C_q, D_q matrices of the Youla parameter Q in the closed-loop representation. This gives the extended closed-loop representation which has the structure required to apply the change of variables of [1].

With this change of variables, the matrix inequalities are then rewritten by replacing the terms \mathcal{X}_{ij} , $\mathcal{X}_{ij}\mathcal{A}$, $\mathcal{X}_{ij}\mathcal{B}_i$, \mathcal{C}_j respectively by $\mathbf{X}_{ij}(v)$, $\mathbf{A}_{ij}(v)$, $\mathbf{B}_{ij}(v)$, $\mathbf{C}_{ij}(v)$ defined hereunder. $\mathbf{D}_{ij} = D_{ij} + E_j D_q F_i$ does not change. The new variables are denoted by v, where the Lyapunov matrices \mathcal{X}_{ij} have been restructured and changed into Q_{ij} , R_{ij} , S_{ij} (see [1]).

$$\begin{split} \mathbf{A}(v) &= \begin{pmatrix} A_1 Q_{ij} & t_1 & t_2 \\ 0 & R_{ij11}A_2 + R_{ij12}B_q \hat{C} & R_{ij12}A_q \\ 0 & R'_{ij12}A_2 + R_{ij22}B_q \hat{C} & R_{ij22}A_q \end{pmatrix}, \\ t_1 &= A_1 S_{ij1} - S_{ij1}A_2 - S_{ij2}B_q \hat{C} + \hat{A} + \hat{B}D_q \hat{C}, \\ t_2 &= A_1 S_{ij2} - S_{ij2}A_q + \hat{B}C_q, \\ \mathbf{B}_{ij}(v) &= \begin{pmatrix} \hat{B}D_q F_i + B_{i1} - S_{ij1}B_{i2} - S_{ij2}B_q F_i \\ R_{ij11}B_{i2} + R_{ij12}B_q F_i \\ R'_{ij12}B_{i2} + R_{ij22}B_q F_i \end{pmatrix}, \\ \mathbf{C}_{ij}(v) &= \begin{pmatrix} Q_{ij} & C_{j1}S_{ij1} + C_{j2} - E_jD_q \hat{C} & C_{j1}S_{ij2} - E_jC_q \\ * & R_{ij} \end{pmatrix}, R_{ij} &= \begin{pmatrix} R_{ij11} & R_{ij12} \\ * & R_{ij22} \end{pmatrix}. \end{split}$$

These expressions are not written explicitly under that form in [1] but are easily derived from the other expressions (as was made in [9] with other notations). Note that the notations are consistent with those of [1], except for the additional detail of the indices ij (sometimes omitted or regrouped as j in [1]) and the splitting of the terms R_{ij} and $S_{ij} = (S_{ij1} \ S_{ij2})$. The reason for this is to have expressions where all four matrices of Q appear explicitly. Also, note that Q_{ij} are new variables and otherwise Q denotes the Youla parameter (like in [1]).

Then, the technique of [1] considers finite-dimensional subspaces of RH_{∞} for the Youla parameter Q. The reason is that the expressions above are not yet affine in the optimization variables because some products between variables still remain. Thus, Finite Impulse Response (FIR) expansions are

used to define the realization of $Q(z) = Q_0 + Q_1 z^{-1} + \dots + Q_p z^{-n_q}$ as:

$$\left(\begin{array}{c|cccc} A_q & B_q \\ \hline C_q & D_q \end{array}\right) = \begin{pmatrix} 0 & I & 0 & \cdots & 0 & 0 \\ 0 & 0 & I & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & I & 0 \\ \hline 0 & 0 & \cdots & 0 & 0 & I \\ \hline Q_{n_q} & Q_{n_q-1} & \cdots & Q_2 & Q_1 & Q_0 \end{pmatrix}$$
(8)

This fixes the A_q, B_q matrices and hence renders the expressions affine: the BMIs are turned into LMIs. Note that the order (i.e. number of states) of the Youla parameter is actually pn_q , where n_q is the size of the FIR expansion and p is the number of inputs of the parameter (i.e. the number of measurements in y = the number of columns in B_q). Also, we remark that an alternative formulation can be used where the fixed matrices are A_q and C_q instead of B_q (in which case the order of the Youla parameter becomes mn_q , with m the number of inputs in u). The restriction of fixed A_q, B_q matrices implies the conservatism of the approach in [1] for fixed-order parameters. Nevertheless, increasing the order of the Youla parameter Q improves the objective value monotonously towards the global optimum [3, 4]. The problem is that the order required for a given relative decrease might lead to too big LMIs and Youla parameters. Therefore a technique will be proposed in the next section that will lead to a faster convergence meaning smaller LMIs and Youla parameters.

3 Contribution

The contribution developed here stems from the key observation that we can identify three subsets of variables: $v_{\alpha} = \{S_{ij2}, R_{ij12}, R_{ij22}\}, v_{\beta} = \{A_q, B_q\}$ and $v_{\gamma} = \{C_q, D_q, Q_{ij}, S_{ij1}, R_{ij11}, \gamma_{ij}, Z_{ij}\}$. The technique developed in [1] only considers LMIs with the variables $v_1 = \{v_{\alpha}, v_{\gamma}\}$ and defines v_{β} such that Q is a FIR filter. Here we propose to also use LMIs with the variables $v_2 = \{v_{\gamma}, v_{\beta}\}$. Indeed, observe that in the expressions obtained with the change of variables, only the v_{α} variables are involved in product terms with v_{β} (and the v_{γ} variables always appear affinely). Thus, instead of fixing the v_{β} variables, the v_{α} variables can be fixed to turn the BMIs into LMIs.

Note that since each of the variables S_{ij2} , R_{ij12} , R_{ij22} of v_{α} appear in product terms with each of the variables A_q , B_q of v_{β} , there exists no meaningful alternative choice to split the variables. On a related note, we remark that this choice of structuring the variables is actually very similar to what was proposed by [10] (applicable for model reduction problems).

The main loop of the method proposed here is the same as the algorithm in [1]. At each iteration of the loop, the order of the Youla parameter is increased. This leads to a sequence of Youla parameters which corresponding costs converge to the globally optimal value of the multi-objective (MO) problem.

Main loop (= Outer loop)

- 1. Choose an initial controller to extend P into the observer structure (6).
- 2. Run the **fixed-order Youla parameters optimizations**, for increasing orders until no improvement is obtained to a given relative decrease.

The improvement contributed here lies in the point 2., i.e. for each

Fixed-order Youla parameters optimizations

In [1], one fixed-order parameter optimization is performed as follows:

- A) Fix the matrices A_q, B_q (= v_β), according e.g. to FIR filters (8).
- B) Optimize the variables $v_1 (\ni C_q, D_q)$ by minimizing the MO formulated under the LMI constraints affine in v_1 .

Here, this optimization is performed further with the following algorithm (or **inner loop**):

- A) Choose initial matrices $A_q, B_q (= v_\beta)$.
- B) Fix the variables v_{β} at the values obtained at the previous step and optimize the variables v_1 $(\ni C_q, D_q)$ by minimizing the MO problem formulated under the LMI constraints affine in v_1 .
- C) Fix the variables v_{α} at the values obtained at the previous step and optimize the variables v_2 ($\ni A_q, B_q, C_q, D_q$) by minimizing the MO problem formulated under the LMI constraints affine in v_2 .
- D) Iterate successively B) and C), until the objective decreases less than a chosen relative decrease.

This type of algorithm can only lead to a monotonous decrease of the objective, and at worst remains at the solution obtained after the first optimization. More comments will be made in Section 5 about the convergence of this inner loop algorithm. But at least, since the first two steps of this inner loop are exactly the end of the method in [1], this algorithm can only lead to the same or a better MO value than that obtained in [1] for each fixed order-parameter optimization. This means that for the sequence of increasing order Youla parameters, the corresponding sequence of decreasing costs can only converge faster with respect to the order (or at least as fast) to the global optimum of the MO than the sequence obtained in [1].

This is the main point of the contribution, which deepens the results in [1]: i.e. it expands the main contribution of [1], which implied already a considerable reduction of the growth of size of the optimization problems compared to the previous solution from [3, 4]. Indeed, thanks to the highlighted new insight on the change of variables [1] and resulting inner loop algorithm, the contribution of [1] is pushed further by exploiting more thoroughly the benefits of that change of variables (that is, that this change of variables leads not to one but two useful LMI formulations).

On top of the improvement of the inner loop algorithm, we also present the following additional improvement. Considering that now the A_q , B_q matrices are also optimized with the inner loop algorithm, it would be a waste of computational effort to start each optimization of fixed order from the FIR filter matrices A_q , B_q from (8). Therefore, an additional improvement is to extend the solution Q_{end,n_q} , obtained after the optimization of the Youla parameter of order pn_q , into a larger order initial solution Q_{ini,n_q+1} for the optimization of the Youla parameter of order $p(n_q+1)$ at the initial step A) of the inner loop algorithm. We propose to do this as follows:

$$Q_{ini,n_q+1} = \begin{pmatrix} A_{q_{n_q}} & 0 & B_{q_{n_q}} \\ 0 & 0 & I \\ \hline C_{q_{n_q}} & 0 & D_{q_{n_q}} \end{pmatrix} \equiv \begin{pmatrix} A_{q_{n_q}} & B_{q_{n_q}} \\ \hline C_{q_{n_q}} & D_{q_{n_q}} \end{pmatrix} = Q_{end,n_q}$$
(9)

where the zero and identity matrices have appropriate dimensions. Since the entry(ies) of the additional state(s) in the C_q matrix is(are) zero, we have that $Q_{ini,n_q+1} \equiv Q_{end,n_q}$ (i.e. these parameters are equivalent in the input-output sense). Thus the closed-loop and its properties remain the same, in particular the MO cost is not modified by this 'extension' of Q. This ensures that the sequence of MO costs when increasing the order of the Youla parameter (or 'Growing Youla') decreases monotonously.

4 Results on example

The results obtained with the $\mathcal{H}_2/\mathcal{H}_\infty$ example used in [1] are now presented¹. The discrete-time plant *P* considered there is defined as follows:

$$\begin{pmatrix} \frac{x^{+}}{z_{1}} \\ z_{2} \\ y \end{pmatrix} = \begin{pmatrix} 0.5 & 1 & 1.5 & 1 & 1 & 0 & 0 \\ -1 & 3 & 2.1 & 2 & 0 & 0 & 0 \\ 1 & -1 & -0.6 & 1 & 0 & 1 & 0 \\ -2 & 2 & -1 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{x}{w_{1}} \\ w_{2} \\ u \end{pmatrix}$$
(10)

The multi-objective used is chosen with $\lambda = 1$, i.e. to minimize $||T_{11}||_2 + ||T_{22}||_{\infty}$.

The choice of the initial controller K, L to extend P into the observer structure (6) (i.e. the step 1. of the main loop) is not made like in [1], which is not straightforward to reproduce, but is obtained by linear quadratic design for discrete-time systems (see the hyperlinked files for details). Anyhow this choice does not affect the global optimum of the MO problem, to which the increasing order solutions converge, as pointed out in [1].

4.1 Main results

The MO values obtained for the sequences of increasing order Youla parameters are given in Fig. 1.



Figure 1: Comparison of the two sequences of MO values

The multi-objective values 'FIR filters (Scherer 2000)' are those obtained with the separated Youla parameters designs of [1], where the matrices A_q, B_q are fixed for each order n_q according to the FIR definition (8)².

The second set of results 'Growing Youla' is obtained with the contributed method, i.e. with the inner loop algorithm improving the multi-objective value for each order n_q , and where the solution obtained at a previous order n_q is extended to be used as initial solution for the order $n_q + 1$ optimization.

¹Matlab files to reproduce the results available on http://www.mathworks.com/matlabcentral/fileexchange/36706.

²Note that in Fig. 5 of [1] the objectives values converged to another value than in Fig. 1 here. The technique of [1] was however applied here with extreme care. This has been discussed with the author of [1]. The results in [1] should correspond to a different P than that in (10).

The objective of accelerating the convergence to the global optimum is fulfilled, for example the growing controller converges to 102.9801 at $n_q = 9$ (the solution at $n_q = 10$ is smaller by a relative decrease under 10^{-14}), whereas FIR filters will require much larger orders to get such relative decrease (actually with the desktop computer used³, such value could not be reached with FIR filters, e.g. the relative decrease obtained between the solutions at $n_q = 14$ and 15 is only of 10^{-4}).

4.2 Computing time VS relative accuracies

In Fig. 1 above we observed that indeed the objective value of the Growing Youla was smaller for all orders n_q than the values reached by the FIR filters of [1]. This however does not give any information regarding the computing time.

In the figure hereunder we draw a comparison of the computing times required against the relative distances $\Gamma_{n_q}/\Gamma_{opt} - 1$ reached, Γ_{n_q} being the objective obtained at the size n_q design and Γ_{opt} being the best solution found (underestimated at 102.9799 \cong 102.97996/(1+10⁻⁷)), for each order n_q of the Growing Youla or the FIR parameters.

The FIR designs are stopped at $n_q = 25$ (after which are encountered memory problems) and the growing Youla is stopped at $n_q = 10$ (after which the relative decrease falls under the machine precision). Note that for the growing Youla, the computing time of the size n_q design is the *total time* required to optimize Q_{n_q} and all the smaller order designs. Whereas for the FIR designs, it is only the time needed for each Q_{n_q} independently (therefore this comparison favors the FIR designs, for which in practice several optimizations of different increasing orders must be tried until no significant improvement is observed). The results are drawn in Fig. 2.



Figure 2: Relative distances reached VS computing times required

It appears clearly that for [1] the computing times and orders of the FIR expansion required explode to obtain a small relative distance. This is not surprising since the number of variables in the LMIs varies with the square of n_q (with this example: $n_q^2 + 18n_q + 76$ for FIR designs).

With the Growing Youla parameter optimizations, the computing time is larger for smaller orders of the parameter, but then the corresponding sequence of multi-objective value converges much faster towards the globally optimal value. This is due to the fact that most of the computing effort is used for smaller orders optimizations, that bring the multi-objective value already close to the globally optimal value for small orders, i.e. the sequence of solutions already reaches at moderate orders a small relative distance to the best value found. This translates as requiring at first larger computing times for smaller order optimizations, but is worth it since the Growing Youla is able to reach a small

³An HP Compaq dc7800©, Intel Q9300©, 2.5GHz, 3.48Go RAM, software Matlab 2007b©, OS Windows XP.

relative distance to the global optimum for moderate orders which require much less computing time than the computing time that would be required to reach such relative distance with FIR parameters.

Indeed, we see in Fig. 2 that obtaining the Growing Youla of order $n_q = 5$ or 6 required about the same computing time than obtaining the FIR parameter of order 12, and that they have close multi-objective values. Because of the quadratic increase of the number of variables with n_q , such threshold value for n_q should exist for all systems. On a related note we remark that the Growing Youla reaches the best value found 102.97996 at $n_q = 9$, whereas such value could not be reached with FIR filters (the FIR filter of order 25 required around five times more computing time to reach only the objective value 102.9809).

The objective here is to propose a method which more quickly converges than [1] towards the globally optimal value of the multi-objective problem, and not so much to propose a method for actual implementation (such as PID optimization). The motivation is that once the globally optimal value is known, the conservatism of other designs can be evaluated. To optimize controllers with specific structure (thus with non-increasing order, unlike here), we recommend to use specialized non-smooth non-convex methods, in particular the Matlab ©implementations HIFOO [11] and hinfstruct [12] (which programs currently only handle continuous-time systems).

Similar results as above were observed with other examples. The presented example suffices to illustrate that the proposed method converges at least as fast to the global optimum for increasing order Youla parameters. This is guaranteed in theory, therefore the method should be systematically considered to improve [1], and in practice we have observed here with one example that it was actually significantly faster. Since fixed-order designs are shown NP-hard [13], a comparative complexity analysis to observe the gain of complexity is difficult to develop, so the gain of performance cannot be known a priori and is only observed a posteriori.

Still, it is of interest to discuss the convergence of the inner loop algorithm. This is done in the following section.

5 Discussion

The question studied here is whether the inner loop algorithm may converge towards locally optimal solutions for a fixed order design. Note that this property would be a nice bonus, but is not required. Indeed, the FIR solutions of [1] are not at all locally optimal for the fixed-order problems, but the main loop (i.e. the increasing order solutions) does converge to the global optimum of the multi-objective problem.

The discussion developed in this section is first made by drawing several relevant observations from the literature, concerning the convergence of iterative LMI algorithms. Then in a second subsection we point out reasons behind the efficiency of the proposed inner loop algorithm⁴.

5.1 Convergence of iterative LMI algorithms

No formal proof can be given that the proposed inner loop algorithm is guaranteed to converge to a locally optimal solution by starting from any feasible solution.

This is a common feature of many iterative LMI algorithms available in the literature, from which we draw the following examples: the V-K iteration of [15], the cone complementary linearization of [16] and the dual iteration of [17]. These methods consider the feasibility problem of stabilization and the last two are very successful in practice for this problem. Nevertheless, they do not have a formal proof of convergence to locally optimal solutions for minimizations problems (as also outlined in [18, 19, Intro.], [17, Sec. II]). This is also the case with the algorithm proposed in [10], which is the most similar to the algorithm proposed here.

⁴This discussion summarizes elements presented at [14] along with alternative insights.

The local optimality is not guaranteed because the sequences of solutions of coordinate-descent methods, though guaranteed monotonically nonincreasing, may lead to 'partial optimal' solutions or 'stagnation'/'dead' points (depending on the authors). This was already outlined in [20], again more specifically in [21] with a counter-example, and also in [22] providing evidence that such a scheme should almost always fail to reach local optimality. The reason behind this can be simply stated as in [19]: optimality in two fixed directions does not imply optimality in all possible directions.

Of course the convergence to global optima is also not guaranteed for the fixed-order problems. This is clear here since these problems can be formulated under an LMI formulation *plus* a non-convex rank constraint, shown to be NP-hard [13]. Remember that this NP-hardness applies to the fixed-order problems and not to the increasing order sequence, for which the cost function converges to the global optimum because the rank constraint disappears.

5.2 Reasons for efficiency in practice

On a more optimistic note, iterative LMI algorithms are heuristics that may be efficient to deal with BMI algorithms. The question then is why.

An important reason why iterative LMI algorithms of coordinate-descent type (i.e. where different subsets of variables are optimized alternatively) may be efficient in practice lies essentially in the choice of the optimization variables, or coordinates. This choice is important for two reasons pointed out at several places in the literature. The interested reader is advised to consult [14] and the references therein presenting these reasons in details, only summarized here.

The first reason is that the choice of coordinates must lead to subsets (i.e. the feasible sets of the conservative LMI formulations of the BMI problem) with better solutions at each iteration. This is necessary to get an improving sequence of solutions and not to remain stuck at the same solution. This last situation typically happens with the V-K iteration [15], for which the objective is not improved at the steps where the controller is fixed.

With the inner loop algorithm proposed here, the controller is partly free (steps B) or entirely free (steps C) and the successively explored convex subsets can generate an improving sequence of solutions. Such improvement is guaranteed for the steps B) of the algorithm, under the very mild assumption that the C_q, D_q matrices of the parameter are not already optimal for its A_q, B_q matrices towards the desired objective. At the steps 3), only a part of the Lyapunov-related matrices is fixed and A_q, B_q, C_q, D_q are free. Thanks to the fact that all four state-space matrices of Q are free, this allows to benefit from the degree of freedom that LTI systems admit an infinity of equivalent state-space representations. Therefore, this mitigates the impact of having to fix part of the Lyapunov-related matrices at these steps. Such a similar hypothesis is likewise formulated in [10].

The second reason of the importance of the the choice of variables is its impact on the structure of the resulting BMI problem. For instance, the change of variables of [1] implies the great benefit of a set of variables v_{γ} already appearing affinely in the BMI formulation. That is, the structure of that problem can be written in a summarized fashion as $v_{\alpha}v_{\beta} + v_{\gamma}$. Thanks to this structure, the number of variables involved in product terms (i.e. the so-called 'complicating variables', here v_{α} and v_{β}) has been reduced compared to the previous BMI formulation where almost all variables were complicating.

Next to the choice of variables, we also point out a technical improvement for the inner loop algorithm, proposed in [14] and also used with the example results presented above. It consists of not optimizing the objective at each step but rather solving the feasibility problem whether there exist a better solution than the previous one. This means that at each step of the inner loop algorithm, we do not completely minimize the objective but rather solve the problem of finding a strictly smaller objective⁵. This modification proved useful in practice to avoid early convergence to dead points and

⁵In Matlab, at each step k of the inner loop algorithm, instead of using mincx to minimize the linear objective $\Gamma_k(=\text{'cx'})$ s.t. LMIs_k $\prec 0$, use feasp to solve the feasibility problem LMIs_k $\prec 0 \cap \Gamma_k < \Gamma_{k-1}$. See the hyperlinked files in http://www.mathworks.com/matlabcentral/fileexchange/36706 for more details.

led to better solutions, as illustrated with a fixed-order design example in [14, Fig. 2]. This technique has similar features with the method of centers described in [23, Sec. 2.4] to solve LMIs. However the proof of convergence given there does not apply for iterative LMI algorithms dealing with non-convex BMIs.

6 Conclusion

The topic treated in this paper is that of multi-objective control design problems that can be formulated with LMIs. The exact formulation of such problems involves BMIs constraints because of products between optimization variables: the Lyapunov matrices and the state-space matrices of the design parameter. The only approach with guaranteed convergence towards the global optima of these multiobjective problems relies on using finite dimensional subspaces for the design parameter. By increasing the order of the design parameter, and re-optimizing it after each increase, the objective value is improved monotonously towards its global optimum. This approach was used in [1], where the main contribution is a change of variables reducing the inflation of size of the optimization problems.

The aim of the current paper is to improve the method proposed in [1]. The issue with the way the change of variables proposed in [1] was used in that paper is that the state-space matrices A_q and B_q of the design parameter had to be fixed prior to optimization and could not be improved. Here the observation is made that this change of variables actually allows to formulate, not just the affine terms proposed in [1], but also another set of affine terms where the state-space matrices A_q and B_q appear as free variables and other variables are fixed.

Thanks to this key observation, the fixed-order design parameters obtained in [1] can be directly improved with the contributed iterative LMI algorithm: once an optimization is made with one set of variables, a following optimization can be made with the other set of variables, and this is repeated iteratively until no improvement is reached to a given relative decrease. Although this kind of scheme is not guaranteed to converge to locally optimal solutions, monotonous decrease is guaranteed. This means that the proposed algorithm can only improve the solutions obtained in [1], or at the very least remain at the same solutions, which is not surprising since this scheme comes as a second layer on top of the method of [1].

Since a fixed-order design can only be improved or at least remain at the same solution, the sequence of increasing order design parameters obtained here can only converge faster or at least as fast as in [1] to the global optimum of the multi-objective problem.

This is illustrated with the multi-objective $\mathcal{H}_2/\mathcal{H}_\infty$ example from [1]. The objective values obtained are indeed better than those from [1], which translates as a sequence of increasing order design parameters converging faster (i.e. for smaller orders) to the global optimum. Thanks to this, a significantly smaller order design parameter is obtained for a given relative decrease than that obtained in [1]. Moreover, the LMIs involved are smaller and therefore involve significantly less variables than in [1], which in practice translates as a much shorter computing time required to reach a desired relative decrease.

Then, the convergence of the proposed algorithm has been discussed. This discussion is based on some main elements drawn from the literature regarding iterative LMI algorithms, as well as some more practical observations. From these points, reasons are identified why the algorithm will in practice improve the fixed-order design parameters obtained in [1], at least towards better solutions even if not necessarily locally optimal ones. In the end, the method can only speed up the convergence of the increasing order design parameters towards the global optimum and should therefore be used as a systematic refinement of the method in [1].

7 Acknowledgments

The authors would like to thank François Glineur for pointing out the parallelism with the analytic center of an LMI, and Carsten Scherer for useful feedback including the design example. Anonymous reviewers are also acknowledged for their pertinent comments and outlining some typos. This research is supported by the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization) funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office.

References

- C.W. Scherer. An efficient solution to multi-objective control problem with LMI objectives. <u>Systems & Control Letters</u>, volume 40:43-57, 2000. http://dx.doi.org/10.1016/S0167-6911(99)00122-X
- [2] I. Masubuchi, A. Ohara and N. Suda. LMI-based controller synthesis: An unified formulation and solution. In Proceedings of American Control Conf., 1995.
- [3] C. Scherer. Multiobjective $\mathcal{H}_2/\mathcal{H}_{\infty}$ control <u>IEEE Transactions on Automatic Control</u>, volume 40:1054–1062, 1995.
- [4] H.A. Hindi, B. Hassibi and S.P. Boyd Multiobjective $\mathcal{H}_2/\mathcal{H}_{\infty}$ control, via finite dimensional Q-parameterization and linear matrix inequalities. In <u>Proceedings of American Control Conf.</u>, 1998.
- [5] V. Blondel and J.N. Tsitsiklis. NP-hardness of some linear control design problems. <u>SIAM journal</u> on control and optimization, volume 35(6):2218–2127, 1997.
- [6] Y. Nesterov and A. Nemirovski. <u>Interior point polynomial methods in convex programming</u>: Theory and applications. SIAM Books, Philadelphia, 1994.
- [7] C.W. Scherer, P. Gahinet and M. Chilali. Multi-objective output-feedback control via LMI optimization. <u>IEEE Transactions on Automatic Control</u>, volume 42:896–911, 1997.
- [8] J. Maciejowski. <u>Multivariable feedback design</u>. Addison-Wesley Wokingham, 1989.
- [9] C. Stoica, P. Rodríguez-Ayerbe and D. Dumur. Off-line improvement of multivariable model predictive control robustness. In Proceedings of 46th IEEE Conf. on Decision and Control, 2007.
- [10] A. Helmersson. Model reduction using LMIs. In Proceedings of 33rd IEEE Conf. on Decision and Control, 1994.
- [11] J.V. Burke, D. Henrion, A.S. Lewis and M.L. Overton. HIFOO A Matlab package for fixedorder controller design and \mathcal{H}_{∞} optimization. In <u>Proceedings of the IFAC Symposium on Robust</u> <u>Control Design</u>, 2006.
- [12] P. Apkarian and D. Noll. Nonsmooth \mathcal{H}_{∞} synthesis <u>IEEE Transactions on Automatic Control</u>, volume 51(1):71–86, 2006.
- [13] M. Fu and Z.-Q. Luo. Computational complexity of a problem arising in fixed order output feedback design. Systems & Control Letters, volume 30:209–215, 1997.
- [14] E. Simon, P. Rodríguez-Ayerbe, C .Stoica, D .Dumur and V .Wertz. LMIs-based coordinate descent method for solving BMIs in control design. In <u>Proceedings of 18th IFAC World Congress</u>, 2011. http://hdl.handle.net/2078.1/69840

- [15] L. El Ghaoui and V. Balakrishnan. Synthesis of fixed-structure via numerical optimization. In Proceedings of 33rd IEEE Conf. on Decision and Control, 2678–2683, 1994.
- [16] L. El Ghaoui, F. Oustry and M. Ait Rami. A cone complementary linearization algorithm for static output-feedback and related problems. <u>IEEE Transactions on Automatic Control</u>, volume 42(8):1171–1176, 1997.
- [17] T. Iwasaki. The dual iteration for fixed-order control. <u>IEEE Transactions on Automatic Control</u>, volume 44(4):783–788, 1999.
- [18] Y. Yamada and S. Hara. Global optimization for \mathcal{H}_{∞} control with constant diagonal scaling. IEEE Transactions on Automatic Control, volume 43:191–203, 1998.
- [19] S. Kanev, C. Scherer, M. Verhaegen and B. De Schutter. Robust output-feedback controller design via local BMI optimization. Automatica, volume 40:1115–1127, 2004.
- [20] R.E. Wendell and A.P. Hurter. Minimization of a non-separable objective function subject to disjoint constraints. Operations Research, volume 24(4):643–657, 1976.
- [21] K-C. Goh, M. G. Safonov and G. P. Papavassilopoulos. Global optimization for the biaffine matrix inequality problem. Journal of Global Optimization, volume 7:365–380, 1995.
- [22] J. W. Helton and O. Merino. Coordinate optimization for bi-convex matrix inequalities. In Proceedings of 36th IEEE Conf. on Decision and Control, 3609–3613, 1997.
- [23] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan. <u>Linear Matrix Inequalities in system and control theory</u>. SIAM Publications, 1994. http://www.stanford.edu/~boyd/lmibook/