Technical report from Automatic Control at Linköpings universitet

A partially augmented Lagrangian method for low order H-infinity controller synthesis using rational constraints

Daniel Ankelhed, Anders Helmersson, Anders Hansson

Division of Automatic Control

E-mail: ankelhed@isy.liu.se, andersh@isy.liu.se, hansson@isy.liu.se

24th March 2011

Report no.: LiTH-ISY-R-3008

Address:
Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

WWW: http://www.control.isy.liu.se

AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Technical reports from the Automatic Control group in Linköping are available from http://www.control.isy.liu.se/publications.

Abstract

When designing robust controllers, H-infinity synthesis is a common tool to use. The controllers that result from these algorithms are typically of very high order, which complicates implementation. However, if a constraint on the maximum order of the controller is set, that is lower than the order of the (augmented) system, the problem becomes nonconvex and it is relatively hard to solve. These problems become very complex, even when the order of the system is low.

The approach used in this work is based on formulating the constraint on the maximum order of the controller as a polynomial (or rational) equation. This equality constraint is added to the optimization problem of minimizing an upper bound on the H-infinity norm of the closed loop system subject to linear matrix inequality (LMI) constraints. The problem is then solved by reformulating it as a partially augmented Lagrangian problem where the equality constraint is put into the objective function, but where the LMIs are kept as constraints.

The proposed method is evaluated together with two well-known methods from the literature. The results indicate that the proposed method has comparable performance in most cases, especially if the synthesized controller has many parameters, which is the case if the system to be controlled has many input and output signals.

Keywords: H-infinity synthesis, augmented Lagrangian, linear systems

A Partially Augmented Lagrangian Method for Low Order H-infinity Controller Synthesis using Rational Constraints

Daniel Ankelhed, Anders Helmersson and Anders Hansson
Email: {ankelhed, andersh, hansson}@isy.liu.se
Division of Automatic Control
Department of Electrical Engineering
Linköping University, Sweden

***Abstract*— When designing robust controllers, H-infinity synthesis is a common tool to use. The controllers that result from these algorithms are typically of very high order, which complicates implementation. However, if a constraint on the maximum order of the controller is set, that is lower than the order of the (augmented) system, the problem becomes nonconvex and it is relatively hard to solve. These problems become very complex, even when the order of the system is low.**

The approach used in this work is based on formulating the constraint on the maximum order of the controller as a polynomial (or rational) equation. This equality constraint is added to the optimization problem of minimizing an upper bound on the H-infinity norm of the closed loop system subject to linear matrix inequality (LMI) constraints. The problem is then solved by reformulating it as a partially augmented Lagrangian problem where the equality constraint is put into the objective function, but where the LMIs are kept as constraints.

The proposed method is evaluated together with two well-known methods from the literature. The results indicate that the proposed method has comparable performance in most cases, especially if the synthesized controller has many parameters, which is the case if the system to be controlled has many input and output signals.

I. Introduction

The development of robust control theory emerged during the 80s and and a contributory factor certainly was the fact that the robustness of Linear Quadratic Gaussian (LQG) controllers can be arbitrarily bad as reported in [1]. A few years later, in [2], an important step in the development towards a robust control theory was taken, where the concept of H_∞ theory was introduced. The H_∞ synthesis, which is an important tool when solving robust control problems, was a cumbersome problem to solve until a technique was presented in [3], which is based on solving two Riccati equations. Using this method, the robust design tools became much easier to use and gained popularity. Quite soon thereafter, linear matrix inequalities (LMIs) were found to be a suitable tool for solving these kinds of problems by using reformulations of the Riccati equations, see [4].

Typical applications for robust control include systems that have high requirements for robustness to parameter variations and for disturbance rejection. The controllers that result from these algorithms are typically of very high order, which complicates implementation. However, if a constraint on the maximum order of the controller is set, that is lower than the order of the plant, the problem is no longer convex and it is then relatively hard to solve. These problems become very complex, even when the order of the system to be controlled is low. This motivates the development of efficient algorithms that can solve these kinds of problems.

In [5], Apkarian et. al presented a method for low order H_∞ controller synthesis which relaxes only one of the constraints and is thus called a *partially* augmented Lagrangian method. In [6] the method is extended to more general robust control than H_∞ controller problems and [7] generalizes the framework to optimization problems with general matrix inequality constraints.

In this paper we will describe a method based on what is done in [5], but where the equality constraint involves coefficients of a characteristic polynomial, similarly to what is done in some of our previous work, [8], [9]. In contrast to the approach in [5], our method does not introduce additional variables when synthesizing dynamic controllers, i.e. controller of order one or higher.

Other methods for solving reduced order H_∞ problems that have gained attention recently are e.g. HIFOO and HINFSTRUCT, see [10] and [11] respectively. These methods are based on nonconvex, nonsmooth approaches for minimizing the H_∞ norm of a closed loop system that do not involve any LMIs. The advantage of these methods is that they in general reduce the number of variables of the problem, while they on the other hand introduce other difficulties due to the nonsmooth formulation of the problem.

Denote with $\mathbb{S}^n$ the set of real symmetric $n \times n$ matrices and $\mathbb{R}^{m\times n}$ is the set of real $m \times n$ matrices. The notation $A \succ 0$ ($A \succeq 0$) and $A \prec 0$ ($A \preceq 0$) means A is a positive (semi)definite matrix and negative (semi)definite matrix, respectively.

II. Preliminaries

We begin by describing a linear system, G, with state vector, $x \in \mathbb{R}^{n_x}$. The input vector contains the disturbance signal, $w \in \mathbb{R}^{n_w}$, and the control signal, $u \in \mathbb{R}^{n_u}$. The output vector contains the measurement, $y \in \mathbb{R}^{n_y}$, and the performance measure, $z \in \mathbb{R}^{n_z}$. In terms of its system

matrices, we can represent the linear system as

$$G: \begin{pmatrix} \dot{x} \\ \hline z \\ y \end{pmatrix} = \left(\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right) \begin{pmatrix} x \\ \hline w \\ u \end{pmatrix}, \quad (1)$$

where D_{22} is assumed to be zero, i.e., the system is strictly proper from u to y. If this is not the case, we can find a controller $\tilde{K}$ for the system where D_{22} is set to zero, and then construct the controller as $K = \tilde{K}(I + D_{22}\tilde{K})^{-1}$. Hence, there is no loss of generality in making this assumption. For simplicity, it is also assumed that the whole system is on minimal form, i.e., it is both observable and controllable. However, in order to find a controller, it is enough to assume detectability and stabilizability (non observable and non controllable modes are stable).

The linear controller is denoted K. It takes the system measurement, y, as input and the output vector is the control signal, u. The system matrices for the controller are defined by the equation

$$K: \begin{pmatrix} \dot{x}_K \\ u \end{pmatrix} = \begin{pmatrix} K_A & K_B \\ K_C & K_D \end{pmatrix} \begin{pmatrix} x_K \\ y \end{pmatrix}, \quad (2)$$

where $x_K \in \mathbb{R}^{n_k}$ is the state vector of the controller.

Lemma 1 (H_∞ controllers for continuous plants): The problem of finding a linear controller such that the closed loop system G_c is stable and such that $\|G_c\|_\infty < \gamma$, is solvable if and only if there exist positive definite matrices $X, Y \in \mathbb{S}^{n_x}$, which satisfy

$$\begin{pmatrix} \mathcal{N}_X & 0 \\ 0 & I \end{pmatrix}^T \begin{pmatrix} XA + A^TX & XB_1 & C_1^T \\ B_1^TX & -\gamma I & D_{11}^T \\ C_1 & D_{11} & -\gamma I \end{pmatrix} \begin{pmatrix} \mathcal{N}_X & 0 \\ 0 & I \end{pmatrix} \prec 0 \quad (3a)$$

$$\begin{pmatrix} \mathcal{N}_Y & 0 \\ 0 & I \end{pmatrix}^T \begin{pmatrix} AY + YA^T & YC_1^T & B_1 \\ C_1Y & -\gamma I & D_{11} \\ B_1^T & D_{11}^T & -\gamma I \end{pmatrix} \begin{pmatrix} \mathcal{N}_Y & 0 \\ 0 & I \end{pmatrix} \prec 0 \quad (3b)$$

$$\begin{pmatrix} X & I \\ I & Y \end{pmatrix} \succeq 0 \quad (3c)$$

$$\operatorname{rank}(XY - I) \leq n_k. \quad (3d)$$

where $\mathcal{N}_X$ and $\mathcal{N}_Y$ denote any bases of the null-spaces of $\begin{pmatrix} C_2 & D_{21} \end{pmatrix}$ and $\begin{pmatrix} B_2^T & D_{12}^T \end{pmatrix}$ respectively.

Proof: See [4]. ■

It could be desirable to replace the rank constraint in (3d) with a smooth function in order to be able to apply gradient methods for optimization. To do this, the following lemma is used.

Lemma 2: Assume that the inequality

$$\begin{pmatrix} X & I \\ I & Y \end{pmatrix} \succeq 0 \quad (4)$$

holds. Let

$$\det(\lambda I - (I - XY)) = \sum_{i=0}^{n_x} c_i(X,Y)\lambda^i = \\ = \lambda^{n_x} + c_{n_x-1}(X,Y)\lambda^{n_x-1} + \dots \\ + c_1(X,Y)\lambda + c_0(X,Y) \quad (5)$$

be the characteristic polynomial of $(I - XY)$, where the functions $c_i(X,Y)$ are its coefficients. Then the following statements are equivalent if $n_k < n_x$:

1) $\operatorname{rank}(XY - I) \leq n_k$
2) $c_{n_x-n_k-1}(X,Y) = 0$

Additionally, all coefficients are non-negative, i.e.

$$c_i(X,Y) \geq 0, \ \forall i. \quad (6)$$

Proof: See [12]. ■

How to compute $c_i(X,Y)$ and their derivatives is explained in [12] where also additional properties of the coefficients are shown.

III. Problem formulation

The problem we wish to solve is this paper is to minimize γ subject to the constraints in (3). Formally this can be stated as the following optimization problem.

$$\begin{aligned} \text{minimize} \quad & \gamma \\ \text{subject to} \quad & c_{n_x-n_k-1}(X,Y) = 0 \\ & (\gamma, X, Y) \in \mathbb{X} \end{aligned} \quad (7)$$

where $\mathbb{X}$ is a convex set defined by the three LMIs in (3a)–(3c). We have noticed that scaling the equality constraint function in (7) by the next coefficient in the characteristic polynomial in (5) makes it numerically sounder, i.e., we replace $c_{n_x-n_k-1}(X,Y)$ by $\hat{c}(X,Y) = c_{n_x-n_k-1}(X,Y)/c_{n_x-n_k}(X,Y)$. This results in the following problem.

$$\begin{aligned} \text{minimize} \quad & \gamma \\ \text{subject to} \quad & \hat{c}(X,Y) = 0 \\ & (\gamma, X, Y) \in \mathbb{X} \end{aligned} \quad (8)$$

This problem can be solved by using the *partially augmented Lagrangian* algorithm, see e.g. [11], where the equality constraint is relaxed and added to the objective function in the following way.

$$\begin{aligned} \text{minimize} \quad & \gamma + \lambda\hat{c}(X,Y) + \frac{\mu}{2}\hat{c}^2(X,Y) \\ \text{subject to} \quad & (\gamma, X, Y) \in \mathbb{X} \end{aligned} \quad (9)$$

where λ is a Lagrangian multiplier and μ is a penalty multiplier. The word "partially" refers to the fact that only the equality constraint is used in the augmentation while the LMIs are kept as they are. The solution to the original problem (8) is obtained by iteratively solving an approximation of (9) for a sequence of increasing values of μ. More details on augmented Lagrangian methods can be found in e.g. the books by Bertsekas, [13], [14] and Nocedal and Wright, [15].

IV. Reformulating the problem

To simplify the notation, let us first define the half-vectorization operator.

Definition 1 (Half-vectorization): Let

$$X = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & & \vdots \\ \vdots & & \ddots & \\ x_{n1} & x_{n2} & \dots & x_{nn} \end{pmatrix}.$$

Then

$$\mathrm{vech}(X) = (x_{11}\ x_{21}\ \dots\ x_{n1}\ x_{22}\ \dots\ x_{n2}\ x_{33}\ \dots\ x_{nn})^T,$$

i.e., vech stacks the columns of X from the principal diagonal downwards in a column vector. See [16] for properties and details.

Next, let us do a variable substitution as follows. Let

$$x = \big(\mathrm{vech}(X)^T, \mathrm{vech}(Y)^T, \gamma\big)^T.$$

By choosing b as the last unit vector such that $\gamma = b^T x$, the optimization problem (9) can be written as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \Phi_c(x,\lambda,\mu) \\ \text{subject to} \quad & x \in \mathbb{X} \end{aligned} \tag{10}$$

where

$$\Phi_c(x,\lambda,\mu) = b^T x + \lambda \hat{c}(x) + \frac{\mu}{2}\hat{c}^2(x).$$

This is a nonconvex problem, since the function $\hat{c}(x) = c_{n_x-n_k-1}(x)/c_{n_x-n_k}(x)$ is nonconvex. However $\mathbb{X}$ is a convex set which makes the problem somewhat less difficult to solve than a general nonconvex problem.

V. Calculating the search direction

The next step is to approximate $\Phi_c(x+p,\lambda,\mu)$ by a quadratic function related to the first three terms in the Taylor series expansion around the point x. Similarly to what is done in regular Newton methods, we intend to find a step direction p that minimizes this second order model, but the difference is that we also require that $x+p \in \mathbb{X}$, i.e. that the next point also lies in the feasible set. This problem can be formulated as

$$\begin{aligned} \underset{p}{\text{argmin}} \quad & \nabla_x \Phi_c(x,\lambda,\mu)^T p + \frac{1}{2} p^T H(x,\lambda,\mu) p \\ \text{subject to} \quad & x + p \in \mathbb{X} \end{aligned} \tag{11}$$

which is a conic programming problem that can be solved using e.g. Yalmip, [17] with SDPT3, [18]. The symmetric matrix $H(x,\lambda,\mu,\delta)$ is a positive definite approximation of the Hessian of $\Phi_c(x,\lambda,\mu)$. We will come back to how this approximation is calculated later.

A. Calculating the derivatives

In order to solve (11), we need to calculate the gradient and Hessian of Φ_c. Differentiating $\Phi(x,\lambda,\mu)$ with respect to x yields

$$\begin{aligned} \nabla_x \Phi_c(x,\lambda,\mu) &= b + \lambda \nabla_x \hat{c}(x) + \mu \hat{c}(x) \nabla_x \hat{c}(x) \\ \nabla_{xx} \Phi_c(x,\lambda,\mu) &= \big(\lambda + \mu \hat{c}(x)\big) \nabla^2_{xx} \hat{c}(x) + \mu \nabla_x \hat{c}(x) \nabla_x^T \hat{c}(x) \end{aligned}$$

with

$$\begin{aligned} \nabla \hat{c} &= \frac{1}{c_{n_x-n_k}} \nabla c_{n_x-n_k-1} - \frac{c_{n_x-n_k-1}}{c^2_{n_x-n_k}} \nabla c_{n_x-n_k} \\ \nabla^2 \hat{c} &= \frac{1}{c_{n_x-n_k}} \nabla^2 c_{n_x-n_k-1} - \frac{c_{n_x-n_k-1}}{c^2_{n_x-n_k}} \nabla^2 c_{n_x-n_k} \\ &\quad + \frac{2c_{n_x-n_k-1}}{c^3_{n_x-n_k}} (\nabla c_{n_x-n_k} \nabla^T c_{n_x-n_k}) \\ &\quad - \frac{1}{c^2_{n_x-n_k}} (\nabla c_{n_x-n_k-1} \nabla^T c_{n_x-n_k} \\ &\quad + \nabla c_{n_x-n_k} \nabla^T c_{n_x-n_k-1}) \end{aligned}$$

where we have omitted the dependence on x to simplify notation. Since the constraint function $\hat{c}(x)$ is nonconvex, the Hessian $\nabla^2_{xx}\hat{c}(x)$ is not always positive definite which in turn might lead to that $H(x,\lambda,\mu) = \nabla^2_{xx}\Phi_c(x,\lambda,\mu)$ is not necessarily positive definite, which has to be dealt with. Two common ways are to either use Newton methods in which the Hessian is convexified or to use Trust-region methods where the nonconvexity is dealt with by optimizing over a limited region in each iteration. The authors of [5] advice against using Trust-region methods since the complexity of such a method is too large in this case. Therefore, our choice is to convexify the Hessian $\nabla^2_{xx}\Phi_c(x,\lambda,\mu)$ as will be explained next.

B. Hessian modification

We have chosen to calculate the exact Hessian $\nabla^2_{xx}\Phi_c(x,\lambda,\mu)$, and then convexifying it using a modified indefinite symmetric factorization as described in [19]. The procedure is as follows. First calculate the indefinite symmetric factorization $\nabla^2_{xx}\Phi_c = P^T L D L^T P$, where L is lower triangular, P is a permutation matrix and D is a block diagonal matrix with block sizes of 1×1 or 2×2. Then we construct a modification matrix F such that $L(D+F)L^T$ is sufficiently positive definite. The matrix F is chosen to be

$$F = Q \operatorname{diag} \tau_i Q^T,$$

where

$$\tau_i = \begin{cases} 0, & d_i \geq \delta, \\ \delta - d_i, & d_i < \delta, \end{cases} \quad i = 1,2,\dots,n_x(n_x+1)+1, \tag{12}$$

and where $Q \operatorname{diag} d_i Q^T$ is the spectral factorization of D. We then have that the modified Hessian is

$$H(x,\lambda,\mu,\delta) = P^T L (D+F) L^T P. \tag{13}$$

The parameter δ is chosen as $10^{-4}\|\nabla^2_{xx}\Phi_c\|_\infty$, where the matrix norm $\|A\|_\infty$ denotes the largest row sum of A.

Now we are ready to outline the suggested algorithm for H_∞ synthesis.

VI. An outline of the algorithm

The algorithm can be outlined as follows.

1) **Initial phase.** (Find a starting a point.)
   Let $\Phi_c = \gamma + \mathrm{trace}(X+Y)$, and solve (10). This is a convex SDP. Denote the solution $(X^{(0)}, Y^{(0)})$. Set $k = 0$.

2) **Optimization phase.**
Set $k := k + 1$.
   a) Solve (11) for the solution $p = \left(\operatorname{vech}(p_X)^T, \operatorname{vech}(p_Y)^T, p_\gamma\right)^T$.
   b) Update the variables as

$$X^{(k)} = X^{(k-1)} + \alpha p_X,$$
$$Y^{(k)} = Y^{(k-1)} + \alpha p_Y,$$
$$\gamma^{(k)} = \gamma^{(k-1)} + \alpha p_\gamma$$

   where $\alpha = 0.98$.

3) **Update phase**.
Update Lagrangian multiplier λ and penalty multiplier μ using the following update rules.

$$\lambda^{(k)} = \lambda^{(k-1)} + \mu^{(k-1)} c_{n_x - n_k - 1}(X^{(k)}, Y^{(k)}) \quad (14)$$

If $|c_{n_x-n_k-1}(X^{(k)}, Y^{(k)})| > \text{tol}$, we update μ as follows.

$$\mu^{(k)} = \begin{cases} \rho\mu^{(k-1)} & \text{if } |c_{n_x-n_k-1}(X^{(k)}, Y^{(k)})| > \\ & \rho_0 |c_{n_x-n_k-1}(X^{(k-1)}, Y^{(k-1)})| \\ \mu^{(k-1)} & \text{if } |c_{n_x-n_k-1}(X^{(k)}, Y^{(k)})| \leq \\ & \rho_0 |c_{n_x-n_k-1}(X^{(k-1)}, Y^{(k-1)})| \end{cases} \quad (15)$$

The first option in (15) reflects our thought that the decrease in the equality constraint function value was not enough. Therefore we increase the penalty parameter. The second option reflects our content with the value of the constraint function, and we leave the penalty parameter at its current value.

4) **Terminating phase.**
If $|c_{n_x-n_k-1}(X^{(k)}, Y^{(k)})| > \text{tol}$, go to phase 2, otherwise we check the following.
   - if $\gamma^{(k)} < 0.99\gamma^{(k-1)}$ for three consequent iterates, it is quite likely we are close enough to a local optimum. Proceed to phase 5.
   - Otherwise, the objective function value is still decreasing, hence we continue the optimization, i.e., go back to phase 2.

5) **Recover controller phase**.
Recover the controller parameters (K_A, K_B, K_C, K_D) as described in [4] and verify that the closed loop system is stable and that $\|G_c\|_\infty < \gamma$ holds true. These requirements should normally be satisfied, but if there are numerical problems this might not hold true.

Remark 1: The objective function $\Phi_c = \gamma + \operatorname{trace}(X+Y)$ used in the initial phase is a combination of two objectives. The first objective is that the performance measure γ should be low and the second is that the equality constraint $\hat{c}(X, Y) = 0$ should be approximately satisfied. Minimizing $\operatorname{trace}(X + Y)$ is a heuristic for minimizing the rank of $I - XY$ that is used e.g. in [20].

Remark 2: Note that in the optimization phase, one normally choose α in the interval $0 < \alpha \leq 1$ by performing a line search. However, we noticed that very small step-lengths α were taken which resulted in bad performance that might be caused by the *Maratos effect*. A solution could be to use a *watchdog* strategy to remedy this, but we have chosen to simply use $\alpha = 0.98$ which seem to work well. For more details on the Maratos effect and watchdog strategies, see [15].

VII. Numerical experiments

All experiments were performed on a Dell Optiplex GX620 with 2GB RAM, Intel P4 640 (3.2 GHz) CPU running under Windows XP using Matlab, version 7.11 (R2010b).

Evaluation of the methods was done on examples from the benchmark problem library COMPl$_e$ib, see [21]. The suggested method was evaluated and compared to Hifoo 3.0[1] see [10] and Hinfstruct which is part of the Robust Control Toolbox in Matlab, version 7.11 (R2010b), and based on the paper [11].

The result from the evaluation is presented in Table I, where the H_∞ norms and required computation time for the respective methods are displayed. We have chosen to evaluate the methods on a couple of systems of different orders and to synthesize controllers of different orders for these. Note that the same settings were used throughout the whole evaluation for the suggested method and that default settings were used for Hifoo and Hinfstruct. Cases where the suggested method had numerical problems are marked in Table I by *.

In the upper part of the table, controllers of order either zero or three was synthesized in order to evaluate both the static output feedback controllers and reduced order feedback controllers. In cases where only the static output feedback controller is evaluated it is because the higher order controllers have the same performance.

Since the computational complexity of Hifoo and Hinfstruct is dependent on the number of parameters in the controller while ours is not, we chose to also include a system (IH) which has 11 input signals and 10 output signals in order to check if the results would differ. The number of optimization variables for Hinfstruct and Hifoo is $n_k^2 + n_k n_y + n_u n_k + n_u n_y$ while for our method it is $n_x(n_x+1)+1$, which means that our method is not affected by the number of states (n_k), inputs (n_u) or outputs (n_y) of the controller, while the other methods are. The results are shown in the lower part of Table I. For this example we also synthesized controllers of higher order than for the other examples.

Note that Hifoo was run ten times for every combination of system and controller order. This was done in order to lessen the effect of the random initialization of the method. The best H_∞ norm from these ten runs is displayed in Table I while the required time is the sum of all ten runs.

VIII. Results

The result in upper part of Table I indicates that the suggested method achieves comparable results in most cases, while Hinfstruct by far is the fastest algorithm. However

[1]Code freely available from `http://www.cs.nyu.edu/overton/software/hifoo/`.

TABLE I

RESULTS FROM EVALUATION ON A COLLECTION OF SYSTEMS FROM COMPL$_{\text{E}}$IB. THE FIRST COLUMN DISPLAYS THE SYSTEM NAME, THE ORDER OF THE SYSTEM, THE NUMBER OF INPUTS AND OUTPUTS AND THE ORDER OF THE CONTROLLER THAT WAS SYNTHESIZED. THE SECOND, THIRD AND FORTH COLUMNS SHOW THE H_∞ NORM AND REQUIRED TIME FOR THE SUGGESTED METHOD (AL), HINFSTRUCT (HS) AND HIFOO (HF) RESPECTIVELY. CASES WHERE THE SUGGESTED METHOD HAD NUMERICAL PROBLEMS ARE MARKED BY *.

Sys, (n_x, n_u, n_y, n_k)	$\|\cdot\|_\infty^{\text{AL}}$, t^{AL}	$\|\cdot\|_\infty^{\text{HS}}$, t^{HS}	$\|\cdot\|_\infty^{\text{HF}}$, t^{HF}
AC2 (5,3,3,0)	0.11, 19.1 s	0.11, 2.72 s	0.11, 168 s
AC5 (4,2,2,0)	670, 20.8 s	665, 0.73 s	669, 24.8 s
AC5 (4,2,2,3)	660*, 10.3 s	658, 1.20 s	643, 1100 s
AC18 (10,2,2,0)	14.8, 37.4 s	10.7, 1.08 s	12.6, 124 s
AC18 (10,2,2,3)	8.09, 36.9 s	6.51, 2.17 s	6.54, 3860 s
CM1 (20,1,2,0)	0.84, 278 s	8.15, 0.72 s	0.82, 125 s
EB4 (20,1,1,0)	2.46*, 460 s	2.06, 1.97 s	2.06, 10.5 s
EB4 (20,1,1,3)	2.14, 370 s	1.82, 3.66 s	1.82, 1160 s
JE3 (24,3,6,0)	8.74, 645 s	5.10, 3.14 s	5.10, 4880 s
JE3 (24,3,6,3)	2.89*, 1403 s	2.90, 3.58 s	2.89, 5910 s
IH (21,11,10,0)	1.88, 367 s	2.45, 14.2 s	1.90, 2450 s
IH (21,11,10,1)	1.86, 523 s	1.96, 13.3 s	1.80, 2410 s
IH (21,11,10,3)	1.49, 373 s	1.69, 18.5 s	1.74, 2170 s
IH (21,11,10,5)	1.39*, 868 s	1.98, 23.0 s	1.69, 2620 s
IH (21,11,10,7)	1.61*, 169 s	1.73, 28.9 s	1.72, 2450 s

it does not always find the best result of the three methods. HIFOO achieves consistent results, but keep in mind that the best result in ten runs is displayed.

The results in the lower part of Table I indicates that the suggested method has an edge when the number of controller parameters are many, which is the case when the system to be controlled has many input and output signals, i.e. the product $n_u \times n_y$ is large as in system IH. The suggested method beats HINFSTRUCT in all cases and it beats HIFOO in four out of five cases. Unfortunately, COMPl$_{\text{e}}$ib does not include more systems with comparable amount of input and output signals for us to evaluate our method on in order to make a more certain statement.

IX. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

We have presented a method for low order H_∞ controller synthesis based on the LMI formulation of the problem which is smooth. The approach is to reformulate the rank constraint as a rational equality constraint and then solve the problem by using a partial augmented Lagrangian minimization algorithm. The suggested method was evaluated and compared with two other methods from the literature. The evaluation indicates that the suggested algorithm obtains comparable results in most cases and that it has an edge in cases where the number of controller parameters are many, which is the case if the system to be controlled has many input and output signals. Overall, HINFSTRUCT is the fastest of the three compared methods.

B. Future Works

We would like to improve the numerical properties of the method so that it becomes more stable and able to handle higher order systems.

X. ACKNOWLEDGMENTS

The author would like to thank The Swedish research council for financial support under contract no. 60519401.

REFERENCES

[1] J. Doyle, "Guaranteed margins for LQG regulators," *IEEE Transactions on Automatic Control*, vol. 23, no. 4, pp. 756–757, 1978.

[2] G. Zames, "Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses," *IEEE Transactions on Automatic Control*, vol. 26, no. 2, pp. 301–320, Apr 1981.

[3] J. Doyle, K. Glover, P. Khargonekar, and B. Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 831–47, 1989.

[4] P. Gahinet and P. Apkarian, "A Linear Matrix Inequality approach to H_∞ control," *International Journal of Robust and Nonlinear Control*, vol. 4, no. 4, pp. 421–48, 1994.

[5] P. Apkarian, D. Noll, and H. Tuan, "Fixed-order H-infinity control design via a partially augmented Lagrangian method," *International Journal of Robust and Nonlinear Control*, vol. 13, no. 12, pp. 1137–1148, 2003.

[6] B. Fares, P. Apkarian, and D. Noll, "An augmented lagrangian method for a class of LMI-constrained problems in robust control theory," *International Journal of Control*, vol. 74, no. 4, pp. 248–360, 2001.

[7] D. Noll, M. Torki, and P. Apkarian, "Partially augmented lagrangian method for matrix inequality constraints," *SIAM Journal on Optimization*, vol. 15, no. 1, pp. 161–184, 2004.

[8] D. Ankelhed, A. Helmersson, and A. Hansson, "A quasi-Newton interior point method for low order H-infinity controller synthesis," *Accepted for publication in IEEE Transactions on Automatic Control*, 2011.

[9] ——, "A primal-dual method for low order H_∞ controller synthesis," in *Proceedings of the 2009 IEEE Conference on Decision and Control*, Shanghai, China, Dec 2009.

[10] S. Gumussoy, D. Henrion, M. Millstone, and M. Overton, "Multiobjective robust control with HIFOO 2.0," in *Proceedings of the IFAC Symposium on Robust Control Design*, Haifa, Israel, June 2009, pp. 144–149.

[11] P. Apkarian and D. Noll, "Nonsmooth H_∞ synthesis," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 71–86, 2006.

[12] A. Helmersson, "On polynomial coefficients and rank constraints," Department of Automatic Control, Linköping university, Sweden, Tech. Rep. LiTH-ISY-R-2878, 2009. [Online]. Available: http://www.control.isy.liu.se/publications/doc?id=2119

[13] D. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. New York, USA: Academic Press, Inc., 1982.

[14] ——, *Nonlinear Programming*. Belmont, Massachusetts, USA: Athena Scientific, 1995.

[15] J. Nocedal and S. Wright, *Numerical Optimization*, 2nd ed. Springer, 2006.

[16] H. Lütkepohl, *Handbook of Matrices*. John Wiley & Sons, Ltd, 1996.

[17] J. Löfberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004, available from `http://users.isy.liu.se/johanl/yalmip/`.

[18] K. Toh, M. Todd, and R. Tütüncü, "SDPT3 — a Matlab software package for semidefinite programming," *Optimization Methods and Software*, vol. 11, no. 1-4, pp. 545–581, 1999.

[19] S. Cheng and N. Higham, "A modified cholesky algorithm based on a symmetric indefinite factorization," *SIAM Journal on Matrix Analysis and Applications*, vol. 19, no. 4, pp. 1097–1110, 1998.

[20] M. Fazel, H. Hindi, and S. P. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proceedings of the American Control Conference*, vol. 6, Viginia, June 2001, pp. 4734–4739.

[21] F. Leibfritz, "COMPl$_e$ib: COnstrained Matrix optimization Problem library," 2006. [Online]. Available: http://www.complib.de

Avdelning, Institution
Division, Department

Division of Automatic Control
Department of Electrical Engineering

Datum
Date

2011-03-24

Språk
Language

☐ Svenska/Swedish
☒ Engelska/English
☐ _____

Rapporttyp
Report category

☐ Licentiatavhandling
☐ Examensarbete
☐ C-uppsats
☐ D-uppsats
☒ Övrig rapport
☐ _____

ISBN
—

ISRN
—

Serietitel och serienummer
Title of series, numbering

ISSN
1400-3902

LiTH-ISY-R-3008

URL för elektronisk version

http://www.control.isy.liu.se

Titel
Title

A partially augmented Lagrangian method for low order H-infinity controller synthesis using rational constraints

Författare
Author

Daniel Ankelhed, Anders Helmersson, Anders Hansson

Sammanfattning
Abstract

When designing robust controllers, H-infinity synthesis is a common tool to use. The controllers that result from these algorithms are typically of very high order, which complicates implementation. However, if a constraint on the maximum order of the controller is set, that is lower than the order of the (augmented) system, the problem becomes nonconvex and it is relatively hard to solve. These problems become very complex, even when the order of the system is low.

The approach used in this work is based on formulating the constraint on the maximum order of the controller as a polynomial (or rational) equation. This equality constraint is added to the optimization problem of minimizing an upper bound on the H-infinity norm of the closed loop system subject to linear matrix inequality (LMI) constraints. The problem is then solved by reformulating it as a partially augmented Lagrangian problem where the equality constraint is put into the objective function, but where the LMIs are kept as constraints.

The proposed method is evaluated together with two well-known methods from the literature. The results indicate that the proposed method has comparable performance in most cases, especially if the synthesized controller has many parameters, which is the case if the system to be controlled has many input and output signals.

Nyckelord
Keywords

H-infinity synthesis, augmented Lagrangian, linear systems