# An Augmented Lagrangian Method for a Class of LMI-Constrained Problems in Robust Control Theory

*B.Fares**   *P. Apkarian*‡   *D. Noll*¶

*Abstract* – **This paper presents a new approach to solve a class of non-convex LMI-constrained problems in robust control theory. These problems are recast as minimization of a linear objective subject to constraints including LMIs and non-convex constraints which are related to rank conditions. The central idea of our approach is based on an Augmented Lagrangian Technique. The Lagrangian function combines a Lagrange multiplier term and a penalty term governing the non-convex constraints while the LMI constraints, due to their special structure, are handled explicitly. Global and fast convergence is then achieved by using an LMI-constrained Newton method combined with line search strategy. This procedure may therefore be regarded as a sequential semi-definite programming (SSDP) method, inspired by the sequential quadratic programming (SQP) in nonlinear optimization. The method is conveniently implemented with available SDP interior-point solvers. We compare its performance to the well-known D-K iteration scheme in robust control. Two test problems are investigated and demonstrate the power and efficiency of our approach.**

Key words: Nonlinear Programming, Semi-Definite Programming, Robust control, Linear Matrix Inequality.

## 1 INTRODUCTION

A large variety of problems in robust control can be cast as minimizing a linear objective subject to linear matrix inequality (LMI) constraints and additional nonlinear constraints which represent rank deficiency conditions. More formally, this can be stated as

$$\text{minimize } c^T x \tag{1}$$
$$\mathcal{L}(x) \leq 0, \tag{2}$$
$$\text{Rank}\,\mathcal{A}(x) = r, \tag{3}$$

where $c$ and $r$ are given and $x$ denotes the vector of decision variables. Inequality (2) represents LMI constraints, while (3) is a rank condition on $\mathcal{A}(x)$, with both $\mathcal{A}, \mathcal{L}$ affine matrix-valued functions of $x$. Synthesis problems that can be formulated as (1) - (3) are:

- fixed-order $H_\infty$ synthesis,
- robust synthesis with different classes of scalings multipliers,
- reduced-order linear parameter-varying (LPV) synthesis.

The rank condition (3) renders these synthesis problems highly complex. Due to their practical importance, however, various heuristics and ad hoc methods have been developed in recent years to obtain solutions to these difficult problems. The $D-K$ iteration procedure is a popular example of this type, [3]. Most currently used methods are based on *coordinate descent schemes* which alternatively and iteratively fix parts of the coordinates of the decision vector, while trying to optimize the remaining indices. This is conceptually simple and easily implemented as long as the intermediate steps are convex LMI programs. The latter may often be guaranteed through an appropriate choice of the decision variables held fixed at each step. However, a major drawback of coordinate descent schemes is that they may (and often will) fail to converge even for starting points close to a local solution. As a result, solutions obtained via such methods are highly questionable and bear the risk of additional conservatism in the synthesis task.

In this paper, we follow a quite different line of attack initiated in [1]. The rank constraints (3) are incorporated into an augmented Lagrangian function with a suitably defined penalty term and a term involving Lagrange multiplier variables. The LMI constraints (2), due to their infinite character, are treated explicitely and not included in the Lagrangian. Instead, the augmented Lagrangian function is minimized subject to these LMI constraints, using an increasing sequence of penalty parameters and a first-order update rule for the Lagrange multiplier estimates. At each step, the minimization of the augmented Lagrangian is performed by a Newton type method including a line search strategy. The entire scheme may be considered as a sequential semi-definite programming (SSDP) method which at each step requires solving a convex LMI program. It therefore lends itself to currently available LMI solvers [7] based on SDP interior-point codes. Even though more sophisticated than most coordinate descent schemes, the advantages of the new approach are at hand:

*ONERA-CERT, Control Systems and Flight Mechanics Dept., 2 av. Edouard Belin, 31055 Toulouse, FRANCE - Email : fares@cict.fr

‡ONERA-CERT, Control Systems and Flight Mechanics Dept., 2 av. Edouard Belin, 31055 Toulouse, FRANCE - Email : apkarian@cert.fr - Tel : +33 5.62.25.27.84 - Fax : +33 5.62.25.27.64

¶UPS-LAO, Applied Mathematics Dept., A 118, route de Narbonne, 31062 Toulouse, FRANCE - Email : noll@dumbo.ups-tlse.fr - Tel : +33 5.61.55.67.78 - Fax : +33 5.61.55.61.83

- The decision variables do not have to be treated separately. The entire vector $x$ of decision variables is updated at each step.
- The method, being of descent type, is guaranteed to converge globally, that is, to a local minimum from any feasible, even remote, starting point. Moreover, the rate of convergence is at least linear.

From a control theory viewpoint, the first observation is important since it means that there is no need to separate Lyapunov and scaling variables from control variables. All these parameters are processed jointly during the iteration.

In this paper, we focus on the robust synthesis problem which, in a sense, is the most difficult among the problems mentioned above, with rank constraint of the form Rank $\mathcal{A}(x) = 0$. The paper is organized as follows. Section 2 recalls the setting of the robust control problem. Section 3 gives a detailed description of the augmented Lagrangian method. Numerical aspects of the algorithms are presented in Section 4.

The notation used throughout the paper is fairly standard. $\mathcal{S}^n$ denotes the set of $n \times n$ symmetric matrices. $M^T$ is the transpose of the matrix M. The notation Tr $M$ stands for the trace of $M$. For Hermitian or symmetric matrices , $M > N$ means that $M-N$ is positive definite and $M \geq N$ means that $M-N$ is positive semi-definite. The notation co$\{p_1, \cdots, p_L\}$ stands for the convex hull of the set $\{p_1, \cdots, p_L\}$. In symmetric block matrices or long matrix expressions, we use $*$ as an ellipsis for terms that are induced by symmetry. Finally, the gradient of real-valued function $\Phi(x)$ is denoted $\nabla\Phi(x)$ and its Hessian $\nabla^2\Phi(x)$.

## 2 ROBUST CONTROL SYNTHESIS

This section provides a brief review of a basic result that will be exploited throughout the paper. We are concerned with synthesis of robust controllers for uncertain plant subject to structured parametric LFT uncertainty. In other words, Consider the uncertain plant governed by:

$$\begin{pmatrix} \dot{x} \\ z_\theta \\ z \\ y \end{pmatrix} = \begin{pmatrix} A & B_\Theta & B_1 & B_2 \\ C_\Theta & D_{\Theta\Theta} & D_{\Theta 1} & D_{\Theta 2} \\ C_1 & D_{1\Theta} & D_{11} & D_{12} \\ C_2 & D_{2\Theta} & D_{21} & 0 \end{pmatrix} \begin{pmatrix} x \\ w_\theta \\ w \\ u \end{pmatrix} \tag{4}$$

with $w_\Theta = \Theta(t) z_\Theta$, where $\Theta(t)$ is a time varying matrix-valued parameter ranging over a polytopic set $\mathcal{P}$, i.e.,

$$\Theta(t) \in \mathcal{P} = \text{co}\,\{\Theta_{v_1}, \ldots, \Theta_{v_N}\}, \qquad \forall t \geq 0\,. \tag{5}$$

and $\Theta_{v_i}$ are the vertices of the polytope $\mathcal{P}$.

Hence the plant with inputs $w$ and $u$ and outputs $z$ and $y$ has state-space entries which are Fractional functions of the time-varying parameter $\Theta(t)$. The definitions of signals are as follows:

- $u$ is the control input.
- $w$ is the vector of exogenous signals.
- $z$ is the vector of regulated variables.
- $y$ is the measurement signal.

For the uncertain plant (4) the purpose of robust control is to find a LTI (Linear Time-Invariant) controller (6)

$$\begin{cases} \dot{x}_K = A_K x_K + B_K y \\ u = C_K x_K + D_K y \end{cases} \tag{6}$$

such that

- the closed-loop system is internally stable,
- the $L_2$-induced gain of closed-loop operator mapping $w$ to $z$ is bounded by $\gamma$.

Moreover, the above specifications must hold for all addmissible values of the parameter $\Theta(t)$ defined by (5).

It is now well-known that such problems can be handled via a suitable generalization of the Bounded Real Lemma which expresses as the existence of a Lyapunov matrix $X_{c\ell}$ and scalings $Q, S, R$ such that $X_{c\ell} > 0$ and

$$\begin{pmatrix} \mathcal{A}^T X_{c\ell} + X_{c\ell}\mathcal{A} & X_{c\ell}\mathcal{B}_\Theta + \mathcal{C}_\Theta^T S^T & X_{c\ell}\mathcal{B}_1 & \mathcal{C}_\Theta^T R & \mathcal{C}_1^T \\ \mathcal{B}_\Theta^T X_{c\ell} + S\mathcal{C}_\Theta & Q + S\mathcal{D}_{\Theta\Theta} + \mathcal{D}_{\Theta\Theta}^T S^T & S\mathcal{D}_{\Theta 1} & \mathcal{D}_{\Theta\Theta}^T R & \mathcal{D}_{1\Theta}^T \\ \mathcal{B}_1^T X_{c\ell} & \mathcal{D}_{\Theta 1}^T S^T & -\gamma I & \mathcal{D}_{\Theta 1}^T R & \mathcal{D}_{11}^T \\ R\mathcal{C}_\Theta & R\mathcal{D}_{\Theta\Theta} & R\mathcal{D}_{\Theta 1} & -R & 0 \\ \mathcal{C}_1 & \mathcal{D}_{1\Theta} & \mathcal{D}_{11} & 0 & -\gamma I \end{pmatrix} < 0 \tag{7}$$

where the scalings $Q < 0, R > 0$ and $S$ must satisfy the LMI constraints $\forall i = 1, \cdots, N$:

$$\begin{pmatrix} \Theta_{v_i} \\ I \end{pmatrix}^T \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} \begin{pmatrix} \Theta_{v_i} \\ I \end{pmatrix} > 0 \tag{8}$$

The state-space data $A_{c\ell}, B_{c\ell}, C_{c\ell}, D_{c\ell}$ determine the closed-loop system (4) and (6) with the loop $w_\Theta = \Theta(t) z_\Theta$ open,

The Bounded Real Lemma conditions (7) are then simplified by means of the Projection Lemma [6, 8], and the following characterization, more easily amenable to numerical computation, is obtained.

**Theorem 2.1** *Consider the LFT plant governed by (4) where $\Theta$ is ranging over the polytopic set $\mathcal{P}$ defined in (5). Let $\mathcal{N}_X$ and $\mathcal{N}_Y$ denote any bases of the null spaces of $(C_2, D_{2\Theta}, D_{21})$ and $(B_2^T, D_{\Theta 2}^T, D_{12}^T)$, respectively. Then, there exists a controller such that the closed-loop system is well posed and the Bounded Real Lemma holds for all admissible $\Theta \in \mathcal{P}$ and with $L_2$ gain performance $\gamma$, if there exist a pair of symmetric matrices $(X, Y)$ and scalings $Q, S, R, \widetilde{Q}, \widetilde{S}$ and $\widetilde{R}$ such that the LMIs (9)-(12), and the nonlinear algebraic constraints (13) below are met:*

$$\mathcal{N}_X^T \begin{pmatrix} A^T X + XA & XB_\Theta + C_\Theta^T S^T & XB_1 & C_\Theta^T R & C_1^T \\ B_\Theta^T X + SC_\Theta & Q + SD_{\Theta\Theta} + D_{\Theta\Theta}^T S^T & SD_{\Theta 1} & D_{\Theta\Theta}^T R & D_{1\Theta}^T \\ B_1^T X & D_{\Theta 1}^T S^T & -\gamma I & D_{\Theta 1}^T R & D_{11}^T \\ RC_\Theta & RD_{\Theta\Theta} & RD_{\Theta 1} & -R & 0 \\ C_1 & D_{1\Theta} & D_{11} & 0 & -\gamma I \end{pmatrix} \mathcal{N}_X < 0 \tag{9}$$

$$\mathcal{N}_Y^T \begin{pmatrix} AY+YA^T & YC_\Theta^T+B_\Theta\widetilde{S} & YC_1^T & B_\Theta\widetilde{Q} & B_1 \\ C_\Theta Y+\widetilde{S}^TB_\Theta^T & D_{\Theta\Theta}\widetilde{S}+\widetilde{S}^TD_{\Theta\Theta}-\widetilde{R} & \widetilde{S}^TD_{1\Theta}^T & D_{\Theta\Theta}\widetilde{Q} & D_{\Theta 1} \\ C_1Y & D_{1\Theta}\widetilde{S} & -\gamma I & D_{1\Theta}\widetilde{Q} & D_{11} \\ \widetilde{Q}B_\Theta^T & \widetilde{Q}D_{\Theta\Theta}^T & \widetilde{Q}D_{1\Theta}^T & \widetilde{Q} & 0 \\ B_1^T & D_{\Theta 1}^T & D_{11}^T & 0 & -\gamma I \end{pmatrix} \mathcal{N}_Y < 0 \tag{10}$$

$$\begin{pmatrix} X & I \\ I & Y \end{pmatrix} > 0\ , \quad \begin{pmatrix} -Q & 0 \\ 0 & R \end{pmatrix} > 0\ , \tag{11}$$

$$\begin{pmatrix} \Theta_{v_i} \\ I \end{pmatrix}^T \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} \begin{pmatrix} \Theta_{v_i} \\ I \end{pmatrix} > 0 \quad \forall i = 1, \cdots, N \tag{12}$$

$$\begin{pmatrix} Q & S \\ S^T & R \end{pmatrix}^{-1} = \begin{pmatrix} \widetilde{Q} & \widetilde{S} \\ \widetilde{S}^T & \widetilde{R} \end{pmatrix}. \tag{13}$$

## 3 AUGMENTED LAGRANGIAN METHOD

In this section, we present our approach to finding local solutions, in a sense to be defined later, to the robust synthesis problem in Theorem 2.1. We recast it as an optimization problem using a cost function which combines the $L_2$-gain performance index $\gamma$ and a penalty term accounting for the nonlinear constraint (13), attributing a high cost to infeasible points. The LMI-constraints, being different in nature, are not included in the objective but kept explicitely. Our approach is known in nonlinear optimization with a finite number of equalities [5] as an *augmented Lagrangian method*, and we extend it here in a natural way to include LMI constraints. The entire procedure is then a sequential semi-definite programming (SSDP) scheme inspired by the sequential quadratic programming (SQP) method in classical optimization.

In order to simplify the expressions, we shall use the following notations. Define new variables $P, \widetilde{P}$ as :

$$P = \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} \quad , \quad \widetilde{P} = \begin{pmatrix} \widetilde{Q} & \widetilde{S} \\ \widetilde{S}^T & \widetilde{R} \end{pmatrix}. \tag{14}$$

Let $\mathcal{X}$ be the convex set of LMI constraints (9)-(12), and $x$ be the complete vector of decision variables $x = (\gamma, P, \tilde{P})$. The robust control problem is equivalently formulated as :

$$\min \left\{ \gamma \ : \ P\widetilde{P} - I = 0\ ,\ x \in \mathcal{X} \right\}, \tag{15}$$

The key idea in solving (17) is now to eliminate the non-convex constraints (13) by including them into a partially augmented Lagrangian function. This allows us to break the difficult non-convex synthesis problem into a series of easier LMI subproblems. The non-convex problem (17) is now approximated by a series of new optimization problems each of which involves minimizing the augmented Lagrangian function, $\Phi_c(x, \Lambda)$, defined as:

$$\Phi_c(x,\Lambda) = \gamma + \sum_{ij} \Lambda_{ij}(P\widetilde{P} - I)_{ij} + \frac{c}{2}\sum_{ij}(P\widetilde{P} - I)_{ij}^2$$

subject to the LMI-constraints (2). In matrix form, the new objective is:

$$\Phi_c(x,\Lambda) = \gamma + \mathrm{Tr}\Big(\Lambda(P\widetilde{P} - I)\Big) + \frac{c}{2}\mathrm{Tr}\Big((P\widetilde{P} - I)^T(P\widetilde{P} - I)\Big)\,, \tag{16}$$

where $c$ is a positive penalty parameter and $\Lambda$ is a Lagrange multiplier. Each of the new optimization problems

$$\begin{array}{ll} \text{minimize} & \Phi_c(x,\Lambda) \\ \text{subject to} & x \in \mathcal{X} \end{array} \tag{17}$$

can then be solved by a sequence of SDPs. At the current position $x$, a new iterate $x^+$ is obtained by minimizing a second-order Taylor series approximation of $\Phi_c(\cdot, \Lambda)$ about the current $x$ subject to the LMI-constraints. We must keep in mind, however, that the variables $\gamma$, $P$, $\widetilde{P}$ are linked to the Lyapunov variables $X$ and $Y$ appearing in the LMI constraints (9)-(12).

Let $(x^*, \Lambda^*)$ be a local minima-lagrange multiplier pair of (17). There are now two mechanisms by which the minimization of (17) can yield points close to $x^*$ [5]. Clearly, when $\Lambda$ is close to $\Lambda^*$. If this is not the case, it is still reasonable to infer that exists a local minimizer of $\Phi_c(., \Lambda)$ close to $x^*$ if $c$ is chosen sufficiently large, say $c \geq \bar{c}$ for a certain threshold $\bar{c}$. In fact, by taking $c$ large, we attribute a high cost to infeasible points, so the local minimizer of $\Phi_c(., \Lambda)$ will be nearly feasible and we can expect $\Phi_c(x, \Lambda)$ to be close to $\gamma$ for nearly feasible $x$. This suggests that in both cases, we can obtain good approximations to $x^*$.

To ensure that $\Lambda$ tends to $\Lambda^*$ during the iteration, we consider an intelligent update of $\Lambda^j$ based on first-order Lagrange multiplier estimates [5, 4]:

$$\Lambda^{j+1} = \Lambda^j + c^j(P_{j+1}\widetilde{P}_{j+1} - I)\,.$$

This updating rule improves the convergence to a local minimizer $x^*$ even when the penalty parameter $c$ is not large [5] and thus, numerical ill conditioning is avoided.

### 3.1 Lagrangian algorithm

**Step 0. Initialization**. Initialize the algorithm by determining a feasible point of the LMI constraints: For fixed large enough $\gamma = \gamma_0$, find an initial point that renders the LMIs (9)-(12) maximally negative by solving the SDP:

$$\min \left\{ t \ : \ \text{LMIs (9)-(12)} < t\,I \right\}.$$

Then, determine $X_0$, $Y_0$, $P_0$ and $\widetilde{P}_0$ so that $P_0\widetilde{P}_0 - I$ is as close as possible to zero. This can be done using the techniques in [1, 2]. Then initialize the penalty parameter $c^0 > 0$ and the Lagrange multiplier $\Lambda^0$.

**Step 1. Lagrangian minimization**. For $j = 0, 1, \cdots$ minimize the augmented Lagrangian $\Phi_j(x) := \Phi_{c^j}(x, \Lambda^j)$ associated with $\Lambda^j$, $c^j$ subject to $x \in \mathcal{X}$. The solution so obtained is $x^{j+1} = (\gamma_{j+1}, P_{j+1}, \tilde{P}_{j+1})$.

**Step 2. Update penalty and multiplier.**

$$\Lambda^{j+1} = \Lambda^j + c^j(P_{j+1}\widetilde{P}_{j+1} - I).$$

$$c^{j+1} = \begin{cases} \rho c^j & \text{if } \|P_{j+1}\widetilde{P}_{j+1} - I\|_F > \mu\|P_j\widetilde{P}_j - I\|_F \\ c^j & \text{if } \|P_{j+1}\widetilde{P}_{j+1} - I\|_F \le \mu\|P_j\widetilde{P}_j - I\|_F \end{cases} \tag{18}$$

for given $\rho$ and $\mu$.

**Step 3. Terminating phase.** Due to non-linearity the algebraic constraint (13) is never exactly satisfied at $x^{j+1}$. It is, however, possible to terminate the program without strict satisfaction of the nonlinear constraints by a simple perturbation technique [1], which is applicable as long as the LMIs (9)-(12) are strictly satisfied. One can then replace $\widetilde{P}_{j+1}$ with $P_{j+1}^{-1}$ and check whether the LMI constraints (9)-(12) hold, possibly with new $X$ and $Y$. In this case a controller is readily obtained. Dually, we can replace $P_{j+1}$ with $\widetilde{P}_{j+1}^{-1}$ and check the LMI constraints (9)-(12), with the scaling constraint in (12) suitably replaced with its dual form

$$\begin{pmatrix} I \\ -\Theta_{v_i}^T \end{pmatrix}^T \widetilde{P}_{j+1} \begin{pmatrix} I \\ -\Theta_{v_i}^T \end{pmatrix} < 0 \quad , \quad \forall i = 1, \cdots, N.$$

If the test fails, set $j = j + 1$ and return to **Step 1**.

## 3.2 Choice of parameters

An important practical question is how to select the initial multiplier $\Lambda^0$ and the penalty parameter sequence $c^j$. Any prior knowledge should be exploited to select $\Lambda^0$ as close as possible to $\Lambda^*$, but this is generally difficult. Concerning the penalty parameter sequence $c^j$ some important remarks are in order:

- the initial value of $c^0$ should not be too large. This increases the number of steps, when the
- $c^j$ should not be increased too fast to a point where the sub-problem (17) becomes ill-conditioned.
- $c^j$ should not be increased too slowly, at least in the early steps,

A good practical scheme is to choose a moderate value $c^0$, and then during an initial phase increase $c^j$ by a factor $\rho > 1$ only if the constraint violation measured by $\|P\widetilde{P} - I\|_F$ is not decreased by a factor $0 < \mu < 1$ over the previous minimization as in (18). Typical values are $\rho = 4$ and $\mu = 0.2$.

## 3.3 Modified Newton method

We have not specified in which way the minimization of $\Phi_{c^j}(x, \Lambda^j)$ in step 1 of the algorithm should be achieved. As a first option, we propose to use a Newton type method which minimizes the second order Taylor polynomial

$$\psi(\delta) = \Phi_j(x) + \nabla\Phi_j(x)^T\delta + \frac{1}{2}\delta^T\nabla^2\Phi_j(x)\delta \tag{19}$$

of $\Phi_j(x)$ about the current iterate $x$ and subject to the constraint set $x + \delta \in \mathcal{X}$ in order to obtain the next iterate $x^+$. When combined with a line search, this provides the new iterate $x^+ = x + t\delta$ with an appropriate $t > 0$.

A difficulty with Newton's method occurs when the Hessian $\nabla^2\Phi_j(x)$ is not positive definite. In this case, modifying the inertia of $\nabla^2\Phi_j(x)$ may be advised, for instance by adding a diagonal correction matrix $\Psi$ rendering the matrix $\nabla^2\Phi_j(x) + \Psi$ positive definite and reasonably well conditioned. Different techniques have been proposed in the literature [?]. In most schemes, a modified Cholesky algorithm is used and pivot entries are sequentially introduced to meet the positive definiteness condition.

# 4 NUMERICAL EXPERIMENTS

This section provides two applications of the techniques just discussed. As compared to previously developed techniques like Frank & Wolfe [1], our approach is in many respects superior. Firstly, the Frank & Wolfe algorithm is not guaranteed to find a local optimal solution and might be subject to zigzagging in the final steps. When well implemented, however, the augmented Lagrangian method is characterized by good convergence properties. Convergence to a local solution with linear rate for any feasible starting point. Furthermore, the approach here is more general since a best $\gamma$ level is computed whereas the Frank & Wolfe technique will require a less efficient dichotomy scheme to minimize $\gamma$. Through examples, we shall also discuss the efficiency of the approach as compared to the classical $D - K$ iteration method which does not enjoy good convergence properties.

## 4.1 Autopilot robust control of missile

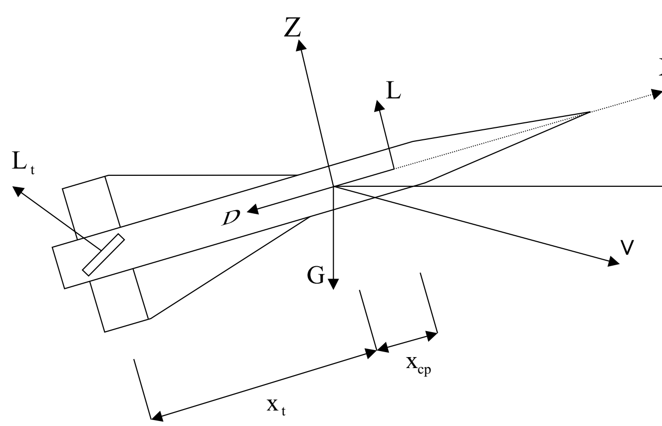


Figure 1: Aerodynamic model for air to air missile

Consider the missile-airframe control problem illustrated in Figure 1. when the vehicle is flying with an angle of attack ($\alpha$). The control problem requires that the autopilot generate the required tail deflection ($\delta$) to produce an angle of attack. corresponding to a maneuver called by the guidance law. Sensor measurements for feedback include missile rotational rates $q$ (rate gyros) and $\alpha$. For the problem considered here. it is desired to track step input commands $\alpha_c$ with a steady state accuracy of 1% and to achieve a rise time less than 0.2 second, and limit overshoot to be 2% over a wide range of angles of attack ±20 deg and variations in Mach number 2.5 to 3.5.

The numerical data subject to the LFT model (4) are given in appendix A. The optimization technique discussed in this paper is then immediately applicable and results are shown in Table 1. We note that good value of $\gamma$ is achieved after a few iterations with fast rate of convergence.

### 4.2 Comparison with $\mathfrak{D} - \mathfrak{K}$ iteration

In this section, we provide a brief comparison with $D-K$-iteration method. The general scheme is as follows.

**Step 1.** Find an initial controller that stabilizes the closed-loop system.

**Step 2.** Analysis phase : for fixed controller, find the optimal $\gamma$ subject to the LMI constraints (7)-(8).

**Step 3.** Compute the scaling $\widetilde{Q}, \widetilde{S}$ and $\widetilde{R}$ so that the nonlinear constraints (13) holds.

**Step 4.** Synthesis phase : for fixed scaling, minimize $\gamma$ subject to LMI the constraints (9)-(11).

**Step 5.** Compute the new controller and return to **Step 2**.

We observe that this coordinate descent technique fails to achieve an adequate value of $\gamma$, Table (2) as compared to the Lagrangian method in Table 1. Also, the convergence is fairly slow and exhibits a typical gradient behavior.

## 5 CONCLUDING REMARKS

In this paper, we have developed an Augmented Lagrangian technique for finding local solutions of robust control problems. The proposed technique is an extension of classical Augmented Lagrangian ideas where LMI constraints are handled explicitly in the course of the algorithm. Therefore, it is easily implemented with available SDP codes. The overall method is highly reliable as demonstrated on a set examples, and has sound convergence properties. Finally, it provides remarkable advantages in terms of efficiency and reduced conservatism over customary $D-K$-iteration schemes.

## A Missile state-space data

The LFT model for missile autopilot example are given as

$$\left(\begin{array}{cccc|ccccc|cc|c}
-0.876 & 1 & -0.1209 & 0 & 0.273 & 0.201 & 1.185 & 0 & 0 & 0 & 0 & 0 \\
8.9117 & 0 & -130.75 & 0 & 23.46 & 86.32 & 0 & 3 & 1 & 0 & 0 & 0 \\
0 & 0 & -150 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 150 \\
-1 & 0 & 0 & -0.05 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.123 & 0 & -0.017 & 0 & 0.038 & 0.028 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.495 & 0 & -7.264 & 0 & 1.303 & 4.796 & 0 & 0 & 0 & 0 & 0 & 0 \\
1.485 & 0 & -21.79 & 0 & 3.91 & 14.38 & 0 & .5 & 0 & 0 & 0 & 0 \\
\hline
-0.25 & 0 & 0 & 3.487 & 0 & 0 & 0 & 0 & 0 & 0 & .25 & 0 \\
0 & 0 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\
\hline
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .01 & 0 & 0
\end{array}\right)$$

The vector $z$ is chosen consist of two components, $z_e$ corresponds to a frequency weighted sensitivity design goal, and $z_\delta$ to limit the tail-fin actuator rate $\dot{\delta}$ . The vector $w$ content the commande $\alpha_c$ and the pitch rate sensor noise $n$.

| | *Constrained Newton* | | | | | | |
|---|---|---|---|---|---|---|---|
| step | $\gamma$ | $\Vert P\widetilde{P}-I\Vert$ | $c$ | step | $\gamma$ | $\Vert P\widetilde{P}-I\Vert$ | $c$ |
| 0 | 5 | 8.037 e-04 | 0.125 | 7 | 0.912 | 1.235 e-02 | 0.5 |
| 1 | 3.172 | 7.127 e-00 | | 8 | 0.914 | 7.742 e-04 | 2 |
| 2 | 1.875 | 9.512 e-00 | | 9 | | 4.595 e-04 | |
| 3 | 0.987 | 5.652 e-00 | | 10 | 0.915 | 9.215 e-05 | 8 |
| 4 | 0.941 | 2.751 e-00 | | 11 | 0.915 | 5.247 e-06 | 32 |
| 5 | 0.915 | 2.974 e-01 | | 12 | | 1.542 e-07 | 128 |
| 6 | 0.907 | 7.892 e-02 | | | | | |

Table 1: Behavior of Algorithm 3.1 for Missile autopilot

## References


[1] P. Apkarian and H. D. Tuan, *Robust Control via Concave Minimization - Local and Global Algorithms*, (1998). to appear in IEEE Trans. on Automatic Control.

[2] ———, *A Sequential SDP/Gauss-Newton Algorithm for Rank-Constrained LMI Problems*, in Proc. IEEE Conf. on Decision and Control, 1999.

[3] G. J. Balas, J. C. Doyle, K. Glover, A. Packard, and R. Smith, *$\mu$-Analysis and synthesis toolbox : User's Guide*, The MathWorks, Inc., april 1991.

[4] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*, Academic Press, London, 1982.

[5] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons, 1987.

[6] P. Gahinet and P. Apkarian, *A Linear Matrix Inequality Approach to $H_\infty$ Control*, Int. J. Robust and Nonlinear Control, 4 (1994), pp. 421–448.

[7] P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali, *LMI Control Toolbox* , The MathWorks Inc., 1995.

[8] C. W. Scherer, *A Full Block S-Procedure with Applications*, in Proc. IEEE Conf. on Decision and Control, San Diego,USA, 1997, pp. 2602–2607.


| | *Classical $\mathfrak{D} - \mathfrak{K}$ iteration Approach* | | | | |
|---|---|---|---|---|---|
| step | phase | $\gamma$ | step | phase | $\gamma$ |
| 1 | A | 4.871 | 8 | S | 2.280 |
| 2 | S | 3.237 | 9 | A | 2.252 |
| : | : | : | 10 | S | fail |

Table 2: Behavior of the classical $\mathfrak{D} - \mathfrak{K}$ iteration approach