Robust Control via Sequential Semidefinite Programming

B. Fares[*] *D.Noll* [*] *P. Apkarian*[‡]

Abstract

This paper discusses nonlinear optimization techniques in robust control synthesis, with special emphasis on design problems which may be cast as minimizing a linear objective function under Linear Matrix Inequality (LMI) in tandem with nonlinear matrix equality constraints. The latter type of constraints renders the design numerically and algorithmically difficult. We solve the optimization problem via *Sequential Semidefinite Programming* (SSDP), a technique which expands on Sequential Quadratic Programming (SQP) known in nonlinear optimization. Global and fast local convergence properties of SSDP are similar to those of SQP, and SSDP is conveniently implemented with available semidefinite programming (SDP) solvers. Using two test examples, we compare SSDP to the augmented Lagrangian method, another classical scheme in nonlinear optimization, and to an approach using concave optimization.

Key words: Nonlinear programming, sequential semi-definite programming, robust gain-scheduling control design, linear matrix inequalities, nonlinear matrix equalities.

1 Introduction

A variety of problems in robust control design can be cast as minimizing a linear objective subject to linear matrix inequality (LMI) constraints and additional nonlinear matrix equality constraints:

$$(D) \quad \begin{array}{ll} \text{minimize} & d^T x \\ \text{subject to} & \mathcal{A}(x) \leq 0, \\ & \mathcal{B}(x) = 0, \end{array}$$

where d is a given vector, x denotes the vector of decision variables, $\mathcal{A}(x)$ is an affine symmetric matrix function, ≤ 0 means negative semidefinite, and $\mathcal{B}(x)$ a nonlinear matrix-valued function, which in many cases is bilinear in x. In the present paper we are primarily

[*]Université Paul Sabatier, Mathématiques pour l'Industrie et la Physique, 118, route de Narbonne, 31062 Toulouse, France - Email: `fares@cict.fr` and `noll@dumbo.ups-tlse.fr` - Tel: +33 5.61.55.86.22 - Fax: +33 5.61.55.83.85

[‡]ONERA-CERT, Control Systems Department, 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

interested in robust gain-scheduling control design, but a variety of other design problems may be cast in the form (D). Without aiming at completeness, let us just mention examples like fixed or reduced-order $\mathcal{H}_2$ and $\mathcal{H}_\infty$ synthesis, robust control synthesis with different classes of scalings, robust control design with parameter-dependent Lyapunov functions, robust control of nonlinear systems with IQC-defined components, and more generally, minimization or feasibility problems with Bilinear Matrix Inequality (BMI) constraints. We discuss some of these applications of (D) at more detail.

Example 1. Observe that the reduced order $\mathcal{H}_\infty$ synthesis problem may be cast as

$$
(H_\infty) \quad \begin{array}{ll} \text{minimize} & d^T x \\ \text{subject to} & \mathcal{A}(x) \preceq 0 \\ & \text{rank}\, \mathcal{Q}(x) \leq r \end{array}
$$

where $\mathcal{A}(x)$ and $\mathcal{Q}(x)$ are symmetric affine. One way to transform (H_∞) into the form (D) is to introduce a slack matrix variable W of size $q \times r$, q the dimension of $\mathcal{Q}(x)$, let $\tilde{x} = (x, W)$ the new decision vector, and introduce the quadratic equality constraint

$$
\mathcal{B}(\tilde{x}) \;=\; \mathcal{Q}(x) - W^T W = 0.
$$

In special situations there may be better suited ways to obtain the form (D). □

Example 2. The BMI-feasibility problem is a near at hand application of our method. If the BMI appears in standard form

$$
\mathcal{B}(x) = \mathcal{A}(x) + \sum_{1 \leq i < j \leq n} B_{ij} x_i x_j
$$

for an affine symmetric matrix valued function $\mathcal{A}$ and symmetric matrices B_{ij}, we are readily led to introduce a slack variable $z_{ij} = x_i x_j$, and replace the BMI with a new LMI in tandem with the nonlinear constraints $z_{ij} - x_i x_j = 0$. In practice, we are more likely to encounter bilinear or even multilinear matrix inequalities, featuring terms of the form $X_i A X_j$ with X_i, X_j parts of the decision vector. In this event, introducing an auxiliary decision matrix variable $Z_{ij} = X_i A X_j$ will have the same effect and transform the constraint set into the form of LMIs plus algebraic equalities. □

Example 3. As a special case of a BMI problem, consider static output feedback control design, where we have to find a Lyapunov matrix variable $X > 0$ and a controller K such that for given matrices A, B, C the BMI

$$
(A + BKC)X \;+\; X(A + BKC)^T \;<\; 0
$$

is satisfied. Introducing a new variable $W = KCX$, we could readily transform this into a LMI plus a nonlinear matrix equality, $KCX - W = 0$, to obtain the program (D).

An alternative way to obtain the form (D) is to open the BMI via the projection lemma [18]. This leads to two LMIs

$$
\mathcal{N}_{B^T}^T (AX + XA^T) \mathcal{N}_{B^T} \;<\; 0, \qquad \mathcal{N}_C^T (YA + A^T Y) \mathcal{N}_C \;<\; 0,
$$

in tandem with $X = Y^{-1}$. Here $\mathcal{N}_{B^T}$, $\mathcal{N}_C$ are bases for the null spaces of B^T, C. With the nonlinear equality constraint rearranged as $XY - I = 0$, we obtain a second version of (D).

It seems appealing to include the LMI

$$\begin{pmatrix} X & I \\ I & Y \end{pmatrix} \geq 0$$

among the above, as with $Y > 0$, and via Schur complement, it is equivalent to $X - Y^{-1} \geq 0$. While becoming redundant near the optimum, the new LMI will help to stabilize the problem. Notice, however, that this idea, which has even been used to relax the static output feedback problem into a LMI problem, is no longer applicable in the more complicated robust design problem we shall present in more detail in Section 2. □

Example 4. Yet another important case is robust control design via generalized Popov multipliers (cf. [34, 28]), also known as k_m- or μ-synthesis. Here we encounter a BMI of the form

$$(P + UKV)^T S^T + S(P + UKV) \leq 0$$

to be solved for S and K for given P, U, V. By introducing a slack matrix variable $G = SUKV + (SUKV)^T$, the design problem may be cast in the form (D) as

$$\begin{array}{ll} \text{minimize} & d^T x, \ x = (S, C, G) \\ \text{subject to} & P^T S^T + SP + G \leq 0 \\ & SUKV + (SUKV)^T - G = 0 \end{array}$$

A similar situation occurs in mixed $\mathcal{H}_2/\mathcal{H}_\infty$-control design, where a BMI of the more general form

$$\Psi + (P + UKV)^T S^T \Phi + \Phi S(P + UKV) \leq 0$$

with fixed Ψ, Φ, P, U, V, to be solved for S, K, arises. This could now be handled using $G = \Phi SUKV + (\Phi SUKV)^T$. □

Example 5. For many robust control problems, Linear Fractional Transformations (LFT) are used to model plants with uncertain components, or to represent nonlinear systems as uncertain linear systems. The corresponding LFTs are often highly complex and difficult to handle numerically, and techniques to reducing the order of LFT representations are required. One way to compute a reduced-order LFT approximation of the nominal LFT is by minimizing the worst-case energy discrepancy between outputs of the nominal and the reduced plant in response to arbitrary finite-energy input signals (see Figure 1). This approach admits a formulation of the form (D). See e.g. [21] for more details. □

The nonlinear constraint $\mathcal{B}(x) = 0$ renders problem (D) highly complex and difficult to solve in practice (cf. [13]). Nonetheless, due to its importance, various heuristics and ad hoc methods have been developed over recent years to obtain suboptimal solutions to (D). Methods currently employed are usually *coordinate descent schemes*, which alternatively and iteratively fix parts of the coordinates of the decision vector, x, trying to optimize the remaining indices. The D-K iteration procedure is an example of this type, [6, 38],

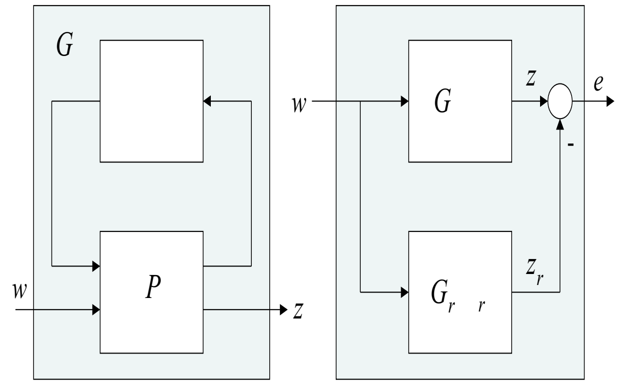

Figure 1: *Designing Reduced LFT Model*

whose popularity may be attributed to the fact that it 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 almost always fail to converge, even for starting points close to a local solution (see [22]). As a result, controllers obtained via such methods are highly questionable and bear the risk of unnecessary conservatism.

A new optimization approach to robust control design was initiated in [5], where the authors showed that reduced-order $\mathcal{H}_\infty$ control could be cast as a concave minimization problem. It was observed, however, that in a number of cases local concave minimization, which is known to be numerically difficult, produced unsatisfactory results. This occurs in particular when iterations get stalled, which is probably due to the lack of second order information.

In [16], we therefore proposed a different approach to (D), again based on nonlinear optimization techniques. The *augmented Lagrangian method* from nonlinear optimization was successfully extended to program (D). The difficult nonlinear constraints were incorporated into an augmented Lagrangian function, while the LMI constraints, due to their linear structure, were kept explicitly during optimization. A Newton type method including a line search, or alternatively a trust-region strategy, were shown to work if the penalty parameters were appropriately increased at each step, and if the so-called first-order update rule for the Lagrange multiplier estimates (cf.[9]) was used.

The disadvantage of the augmented Lagrangian method is that its convergence is at best linear if the penalty parameter c is held fixed. Superlinear convergence is guaranteed if $c \to \infty$, but the use of large c, due to the inevitable ill-conditioning, is prohibitive in practice. The present investigation therefore aims at adapting methods with better convergence properties, like sequential quadratic programming (SQP), to the case of LMI constrained problems. Minimizing at each step the second order Taylor expansion of the Lagrangian of (D) about the current iterate, defines the *tangent subproblem*, (T), whose solution will provide the next iterate. Due to the constraints $\mathcal{A}(x) \leq 0$, (T) is not a quadratic program, as in the case of SQP, but requires minimizing a quadratic objective function under LMI constraints. After convexification of the objective, (T) may be turned into a semi-definite program (SDP), conveniently solved with current LMI tools (cf. for instance [20, 37]). We refer to this approach as *Sequential Semi-Definite Programming* (SSDP). It will be discussed in Section 4, and a local convergence analysis will be presented in Section 5. Although more complex than most coordinate descent schemes, the advantages of the new approach are at hand:

- The entire vector x of decision variables is updated at each step, so for instance we do not have to separate Lyapunov and scaling variables from controller variables.
- Like SQP, SSDP is guaranteed to converge globally, which means, for an arbitrary and possibly remote initial guess, if an appropriate line search or trust region strategy is applied.
- Being of second order type, the rate of convergence of SSDP is superlinear in a neighborhood of attraction of a local optimum.

The present paper discusses and compares three nonlinear optimization techniques suited for the design problem (D), with special emphasis on SSDP since it performed best. The reader might be missing an approach via interior-point techniques – perhaps more in the spirit of the age. In fact, in a different context, Jarre [24] proposes such a method based on the log-barrier function known from the interior-point approach to the SDP-problem, but does not present any numerical evidence as to the practicality of the approach. Theoretical *and* practical results are presented by Leibfritz et al. [25, 26], who consider static output feedback control and mixed $\mathcal{H}_2/\mathcal{H}_\infty$-control. Our own numerical experiments [3] with interior-point methods for robust control design seem to indicate that those are generally less robust, and that the different parameters may be difficult to tune. We emphasize that the method proposed for robust control design is modulable in the sense that the optimization procedure featuring SSDP may be replaced by any other tool based on the user's favorite optimizer. Future investigations will show which methods work best in a given situation, and the present contribution does not claim to present the ultimate tool.

The paper is organized as follows. Section 2 presents and develops the setting of the robust gain-scheduling control, a particularly important application of (D). Even though the full robust gain-scheduling case has never been presented, let alone attacked algorithmically, we keep this part rather cursory, as the individual steps of the method are essentially known. We rely on a recent excellent exposition of the material by Scherer [36] and related texts [29, 1, 21]. We have chosen this problem as our main motivating case study, as it seems to be among the most difficult and numerically demanding cases of the scheme (D).

Section 3 aims at practical aspects. We offer more specific choices of parameter uncertainties and scaling variables which help to reduce the algorithmic complexity of the problem and, as far as our own experiments go, work well in practice.

Section 4 gives a description of the SSDP method as it naturally emerges from the classical SQP method. Local superlinear and quadratic convergence of SSDP is shown in Section 5. While several convergence proofs for the SQP method are known in the literature, (cf. [11, 12]), they all seem to depend heavily on the polyhedrality of the classical order cone, and no extension addressing the semi-definite cone seems available.

The proof we present here is fairly general and includes nonlinear programming with more general order cones.

Numerical aspects of the SSDP method are discussed in Section 7. Using two typical test examples, we compare it to the augmented Lagrangian method and to concave programming. While apparently of moderate size, these examples represent cases where classical approaches like the D-K iteration perform poorly, or are even at complete loss.

2 Robust Gain-Scheduling Control Design

We wish to design a robust gain-scheduling controller for a plant which depends rationally on the uncertain and scheduled parameters. Consider an LFT plant in standard form described by the state-space equations:

$$\left(\begin{array}{c} \dot{x} \\ \hline z_\theta \\ z \\ y \end{array}\right) = \left(\begin{array}{c|ccc} A & B_\theta & B_1 & B_2 \\ \hline C_\theta & D_{\theta\theta} & D_{\theta 1} & D_{\theta 2} \\ C_1 & D_{1\theta} & D_{11} & D_{12} \\ C_2 & D_{2\theta} & D_{21} & 0 \end{array}\right) \left(\begin{array}{c} x \\ \hline w_\theta \\ w \\ u \end{array}\right), \qquad w_\theta = \Theta z_\theta, \tag{1}$$

where $\Theta(t)$ is a time-varying matrix-valued parameter assumed to have a two-block diagonal structure

$$\Theta = \begin{pmatrix} \Theta_m & \\ & \Theta_u \end{pmatrix}. \tag{2}$$

Here $\Theta_m(t)$ represents the scheduled parameters, measured on-line, $\Theta_u(t)$ the time varying parametric uncertainties, which we allow to vary in a known compact set $\mathcal{K}$ of matrices. We call parameters Θ of this form *admissible*, and the set of admissible (scheduled and bounded uncertain) parameters is denoted $\boldsymbol{\Theta}$.

We recall that the limiting case: no Θ_u (all parameters measured), is called the LPV or gain-scheduling control problem, while the case: no Θ_m (all parameters uncertain) is referred to as the robust control problem.

The state-space entries of the plant (1) with inputs w, u and outputs z, y are rational functions of the parameters Θ_m and Θ_u. The meaning of the signals is as follows: u is the control input, y is the measurement signal, w stands for the vector of exogenous signals, while z stands for regulated variables.

The robust gain-scheduling control design requires finding a linear controller K of the form

$$\left(\begin{array}{c} \dot{\bar{x}} \\ \hline \bar{z}_\theta \\ u \end{array}\right) = \left(\begin{array}{c|cc} A_K & B_{K\theta} & B_{K1} \\ \hline C_{K\theta} & D_{K\theta\theta} & D_{K\theta 1} \\ C_{K2} & D_{K2\theta} & D_{K21} \end{array}\right) \left(\begin{array}{c} \bar{x} \\ \hline \bar{w}_\theta \\ y \end{array}\right), \quad \bar{w}_\theta = \phi(\Theta_m)\bar{z}_\theta \tag{3}$$

where ϕ is called the *scheduling function*, to be determined as part of the design, such that (3) fulfills the following requirements:

- The closed-loop system, obtained by substituting (3) into (1), is internally stable.

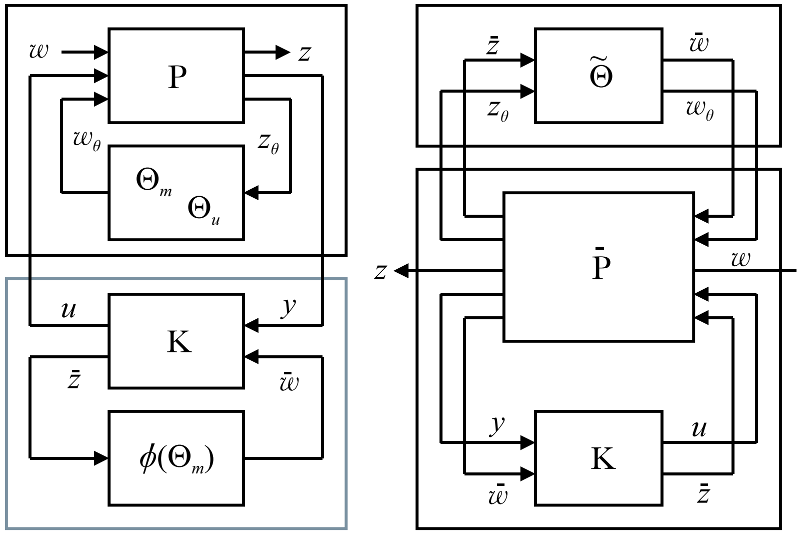

Figure 2: *Gain-scheduling robust control*

- The L_2-gain of the closed-loop operator mapping w to z is bounded by γ.
- The above specifications hold for *all* admissible parameter trajectories $\Theta \in \boldsymbol{\Theta}$.

In order to continue our analysis, we apply a convenient procedure first used in [30, 1]. We gather all parameter-dependent components into a single block which leads to an augmented plant $\bar{P}(s)$ described in the frequency domain as:

$$\begin{pmatrix} \bar{z}_\theta \\ \hline z_\theta \\ z \\ y \\ \hline \bar{w}_\theta \end{pmatrix} = \overbrace{\left(\begin{array}{c|c|c} 0 & 0 & I \\ \hline 0 & P(s) & 0 \\ \hline I & 0 & 0 \end{array}\right)}^{\bar{P}(s)} \begin{pmatrix} \bar{w}_\theta \\ \hline w_\theta \\ w \\ u \\ \hline \bar{z}_\theta \end{pmatrix} \tag{4}$$

It is easy to verify picturally that the original scheme shown on the left hand side of Figure 2 is equivalent to the one shown on the right hand side using the augmented system $\bar{P}(s)$. After closing the loop, i.e., substituting (3) into (1), respectively (4), the closed-loop systems mapping exogenous inputs w to regulated outputs z are the same on both sides.

By inspecting the left hand diagram in Figure 2, we see that the original robust gain-scheduling control problem can now be viewed as a standard robust control problem for the time-invariant plant $\bar{P}$ facing the augmented uncertain parameter matrix $\tilde{\Theta}$, where

$$\tilde{\Theta} = \begin{pmatrix} \phi(\Theta_m) & \\ & \Theta \end{pmatrix}, \quad \Theta = \begin{pmatrix} \Theta_m & \\ & \Theta_u \end{pmatrix}.$$

Based on this reformulation, sufficient conditions for the existence of a robust gain-scheduling LTI controller (3) are consequently obtained by a suitable extension of the usual procedure in robust control design: Apply the Bounded Real Lemma and the generalized

S-procedure with a suitable choice of scalings to obtain sufficient conditions for robust stability of the closed-loop system (Lemma 1). Then use the Projection Theorem [18, 36] to eliminate the state-space variables of the controller K. The sufficient conditions for solvability are now again formulated in terms of the state space entries (1) in conjunction with the Lyapunov and scaling variables (Theorem 2). They form, as we shall see, a mix of LMIs and nonlinear algebraic equalities. Now at the core of the procedure, calculate the optimal gain using the proposed optimization techniques. As a final step, extract the robust controller K from the decision parameters used during optimization using e.g. the method in [1].

We proceed to present the details of this scheme for the robust gain-scheduling control case. At the present stage, we aim at a fairly general approach, but the next section will focus on the practical aspects, where some of the theoretically possible steps will have to be reconsidered regarding their numerical performance. This concerns in particular the choice of the Lyapunov test matrix used in the Bounded Real Lemma, the S-procedure, and the choice of the scalings (structured or general).

In this section we allow for a fairly general class of scalings $\mathcal{Q}$ of the form

$$\mathcal{Q} = \begin{pmatrix} \mathcal{Q}_m & \\ & Q_u \end{pmatrix}, \quad \mathcal{Q}_m = \begin{pmatrix} Q_1 & Q_2 \\ Q_2^T & Q_m \end{pmatrix} \tag{5}$$

compatible with the block structure of $\tilde{\Theta}$. Later on we shall, at the cost of some conservatism, consider more special classes of scalings in order to reduce the numerical burden in the design.

Remark. Let us address the question of choosing the Lyapunov test function. Although parameter-dependent Lyapunov functions can be used, see [7, 17] for discussions, in the present paper, we shall restrict our attention to the more traditional single quadratic Lyapunov approach based on a parameter-independent Lyapunov matrix $\mathcal{P}_0$. This choice is at the cost of some conservatism but keeps the theoretical descriptions simple and practically useful. □

For the notation, observe that we use script matrix symbols $\mathcal{A}, \mathcal{B}_1, \mathcal{B}_\theta$ etc. for the state-space data of the closed loop-system obtained by substituting (3) into (4), see Figure 2. We have the following

Lemma 1. *Suppose there exists a Lyapunov matrix $\mathcal{P}_0 > 0$ and scalings $\mathcal{Q}, \mathcal{R}$ and $\mathcal{S}$, of the above form (5) such that the nonlinear matrix inequality*

$$\star \left(\begin{array}{cc|cc|cc} 0 & I & & & & \\ I & 0 & & & & \\ \hline & & \mathcal{Q} & \mathcal{S} & & \\ & & \mathcal{S}^T & \mathcal{R} & & \\ \hline & & & & -\gamma & 0 \\ & & & & 0 & \frac{1}{\gamma} \end{array}\right) \left(\begin{array}{ccc} \mathcal{P}_0 & 0 & 0 \\ \mathcal{A} & \mathcal{B}_\theta & \mathcal{B}_1 \\ \hline 0 & I & 0 \\ \mathcal{C}_\theta & \mathcal{D}_{\theta\theta} & \mathcal{D}_{\theta 1} \\ \hline 0 & 0 & I \\ \mathcal{C}_1 & \mathcal{D}_{1\theta} & \mathcal{D}_{11} \end{array}\right) < 0 \tag{6}$$

is satisfied. Suppose further that the scalings satisfy the condition:

$$\begin{pmatrix} \tilde{\Theta} \\ I \end{pmatrix}^T \begin{pmatrix} \mathcal{Q} & \mathcal{S} \\ \mathcal{S}^T & \mathcal{R} \end{pmatrix} \begin{pmatrix} \tilde{\Theta} \\ I \end{pmatrix} \geq 0 \tag{7}$$

for each admissible Θ. *Then the closed-loop system is robustly stable over the uncertain set* $\mathbf{\Theta}$. *Moreover, for every admissible* $\Theta \in \mathbf{\Theta}$, *the operator mapping the exogenous signal* w *into the regulated variables* z *has* $\mathcal{H}_\infty$*-norm bounded above by* γ.

Proof. The result is essentially the same as Theorem 10.4 in [36]. It consists in applying the Bounded Real Lemma in tandem with the full block S-procedure. □

Remark. The derived sufficient conditions for robust gain-scheduling control are not suited for practice as they stand. This is mainly due to the infinite constraint (7), which involves an infinity of test matrices $\tilde{\Theta}$. In the following section, we shall indicate in which way (7) may, at the cost of some conservatism, be turned into a finite condition.

A second aspect of the derived criteria is that (6) is not jointly convex in the decision variables $\mathcal{P}_0, \mathcal{Q}, \mathcal{R}, \mathcal{S}$ and K. As a consequence, using these variables in the design is a difficult problem not suited for the usual convexity techniques in control. □

As we shall see in our next step, the non-convexity of the design problem may to some extent be reduced through the Projection Lemma [18]. As a result, the solvability conditions are stated back in terms of the original state-space entries in tandem with the Lyapunov and scaling variables, whereas the controller variable K has been eliminated. The mild inconvenience of this is that the actual controller has to be obtained in an extra step using the decision variables in Theorem 2 below. This step may itself be numerically demanding if the scheduling function ϕ has some undesirable properties.

Theorem 2. *Consider the LFT plant (1) with scheduled and uncertain parameters* $\Theta \in \mathbf{\Theta}$ *as in (2). Let* $\mathcal{N}_X$ *and* $\mathcal{N}_Y$ *be bases of the null spaces of* $(C_2, D_{\theta 2}, D_{12}, 0, 0)$ *and* $(B_2^T, D_{2\theta}^T, D_{21}^T, 0, 0)$, *respectively. Suppose there exist scalings* $Q, R, S, \tilde{Q}, \tilde{R}, \tilde{S}$ *of the form*

$$Q = \begin{pmatrix} Q_m & \\ & Q_u \end{pmatrix}, \; R = \begin{pmatrix} R_m & \\ & R_u \end{pmatrix}, \; etc. \tag{8}$$

compatible with the block structure of Θ *in (2), and a pair of symmetric matrices* (X, Y) *satisfying the matrix completion conditions:*

$$\begin{pmatrix} X & I \\ I & Y \end{pmatrix} > 0\,, \tag{9}$$

such that the linear matrix inequalities (10) - (12):

$$\mathcal{N}_X^T \begin{pmatrix} A^TX + XA & XB_\theta + C_\theta^T S^T & XB_1 & C_\theta^T R & C_1^T \\ B_\theta^T X + SC_\theta & Q + SD_{\theta\theta} + D_{\theta\theta}^T S^T & SD_{\theta 1} & D_{\theta\theta}^T R & D_{1\theta}^T \\ B_1^T X & D_{\theta 1}^T S^T & -\gamma I & D_{\theta 1}^T R & D_{11}^T \\ RC_\theta & RD_{\theta\theta} & RD_{\theta 1} & -R & 0 \\ C_1 & D_{1\theta} & D_{11} & 0 & -\gamma I \end{pmatrix} \mathcal{N}_X < 0 \tag{10}$$

$$\mathcal{N}_Y^T \begin{pmatrix} AY + YA^T & YC_\theta^T + B_\theta \tilde{S} & YC_1^T & B_\theta \tilde{Q} & B_1 \\ C_\theta Y + \tilde{S}^T B_\theta^T & D_{\theta\theta}\tilde{S} + \tilde{S}^T D_{\theta\theta} - \tilde{R} & \tilde{S}^T D_{1\theta}^T & D_{\theta\theta}\tilde{Q} & D_{\theta 1} \\ C_1 Y & D_{1\theta}\tilde{S} & -\gamma I & D_{1\theta}\tilde{Q} & D_{11} \\ \tilde{Q} B_\theta^T & \tilde{Q} D_{\theta\theta}^T & \tilde{Q} D_{1\theta}^T & \tilde{Q} & 0 \\ B_1^T & D_{\theta 1}^T & D_{11}^T & 0 & -\gamma I \end{pmatrix} \mathcal{N}_Y < 0 \tag{11}$$

$$\begin{pmatrix} \Theta \\ I \end{pmatrix}^T \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} \begin{pmatrix} \Theta \\ I \end{pmatrix} \geq 0 \;\; \textit{for every} \;\; \Theta \in \boldsymbol{\Theta} \tag{12}$$

in tandem with the nonlinear algebraic equality

$$\begin{pmatrix} Q_u & S_u \\ S_u^T & R_u \end{pmatrix}^{-1} = \begin{pmatrix} \tilde{Q}_u & \tilde{S}_u \\ \tilde{S}_u^T & \tilde{R}_u \end{pmatrix} \tag{13}$$

are satisfied. Then there exists an n-th order gain-scheduling controller K, (n the order of the plant (1)), and a choice of the scheduling function ϕ such that the closed-loop system is internally and robustly stable, and the operator mapping w into z has L_2 gain bounded by γ for all admissible parameter trajectories $\Theta \in \boldsymbol{\Theta}$.

Proof. The argument is based on a solvability test for quadratic inequalities developed in [35, 1, 2]. The most recent reference is Lemma 10.2 in [36]. This result is used to eliminate the controller variable K from the solvability conditions (6) in Lemma 1.

When applying the solvability test, due to the special structure of $\bar{P}(s)$, the solvability conditions obtained simplify to (9) – (12), (13). The question which remains is how the scheduled part of the coupling condition (13) is avoided. Following Theorem 10.11 of [36], one may show that the variables $Q_1, Q_2, R_1, R_2, S_1, S_2$ in the scheduled part of the multipliers are not called for by the matrix inequalities (9) – (12), and are therefore free to be chosen to satisfy the scheduled part of (13). This also requires a special choice of the scheduling function ϕ given in [36]. □

As already mentioned, condition (12) needs to be worked on in order to become numerically tractable. This aspect is treated in the next section.

3 Choices Suited for Practice

In this section we address the practical aspects of the control design part and indicate that, at the cost of some conservatism, the difficulty of the design may be greatly reduced by accepting some restrictions in the general outline.

To begin with, let us assume that the uncertain matrix function $\Theta_u(t)$ varies in a polyhedral convex and compact set $\mathcal{K}$ of matrices, i.e., $\Theta_u(t) \in \mathcal{K} = \mathrm{co}\{\Theta_{u1}, \ldots, \Theta_{uN}\}$ at all times t. We refer to the Θ_{ui} as the vertices of the value set. Let us examine the consequence of this choice. Observe that due to the block structure of $\tilde{\Theta}$, the infinite dimensional scaling condition (12) already decouples into a scheduled part and an uncertain part. Concerning the uncertain part, we have the following

Lemma 3. *Suppose the value set of $\Theta_u(t)$ is polyhedral and the scaling satisfies $Q_u < 0$. Then the uncertain part of condition (12) is equivalent to the finite condition*

$$\begin{pmatrix} \Theta_{ui} \\ I \end{pmatrix}^T \begin{pmatrix} Q_u & S_u \\ S_u^T & R_u \end{pmatrix} \begin{pmatrix} \Theta_{ui} \\ I \end{pmatrix} \geq 0 \qquad \textit{for every} \;\; i = 1, \ldots, N. \tag{14}$$

□

The proof is in fact a straightforward convexity argument based on $Q_u < 0$ and may be found e.g. in [19, 36]. This settles the question of finiteness for the uncertain part of (12) at the slight cost of conservatism introduced by assuming $Q_u < 0$.

Remark. We mention that in practice it is sufficient to let Θ_u have a block diagonal structure of the form

$$\Theta_u(t) \;=\; \mathrm{diag}(\theta_{u1}(t)I_{p_1}, \ldots, \theta_{ur}(t)I_{p_r}) \tag{15}$$

where we may without loss assume $|\theta_{uj}| \leq 1$, so the set $\mathcal{K}$ will be a cube with the 2^r vertices $\theta_{uj}(t) = \pm 1$. □

The conservatism introduced to obtain the finite condition (14) is minor and acceptable in practice. Notice that the number N may become inconveniently large if the number of parameters θ_{ui} grows. We therefore mention another strategy to avoid the infinite scaling condition. Assuming that the uncertain parameters have the block diagonal structure (15) above, we consider what we call *structured scalings* satisfying the following conditions: (i) Q_u and S_u commute with $\Theta_u(t)$; (ii) $R_u = -Q_u$ and $R_u > 0$, and (iii) $S_u^T = -S_u$. We check that the scheduled part of condition (7) is satisfied. Developing the term gives

$$\Theta_u^2 Q_u + \Theta_u S_u + S_u^T \Theta_u + R_u = (I - \Theta_u^2) R_u \geq 0$$

as required. This choice of the scaling appears rather special and therefore bears the risk of unnecessary conservatism, but its merit is that it greatly reduces the number of decision variables and LMI constraints.

Let us now consider the corresponding questions for the scheduled part of (12). We start with the following technical Lemma, which was already used in the proof of Theorem 2, cf. [36]:

Lemma 4. *Suppose the scalings Q_m, S_m and R_m have been found such that*

$$\begin{pmatrix} \Theta_m \\ I \end{pmatrix}^T \begin{pmatrix} Q_m & S_m \\ S_m^T & R_m \end{pmatrix} \begin{pmatrix} \Theta_m \\ I \end{pmatrix} \geq 0 \quad \textit{for every } \Theta \in \boldsymbol{\Theta}. \tag{16}$$

Then there is a choice of the scheduling function ϕ along with appropriate choices of Q_1, Q_2, R_1, R_2 and S_1, S_2 such that the scheduled part of (7) is satisfied, i.e.,

$$\begin{pmatrix} \tilde{\Theta}_m \\ I \end{pmatrix}^T \begin{pmatrix} \mathcal{Q}_m & \mathcal{S}_m \\ \mathcal{S}_m^T & \mathcal{R}_m \end{pmatrix} \begin{pmatrix} \tilde{\Theta}_m \\ I \end{pmatrix} \geq 0 \quad \textit{for every } \Theta \in \boldsymbol{\Theta}. \tag{17}$$

Proof. As shown in [36], if $\Phi_m := [Q_m\, S_m; S_m^T\, R_m]$ satisfies (16), it is always possible to adjust the extended scalings $\mathcal{Q}_m = [Q_1\, Q_2; Q_2^T\, Q_m]$, $\mathcal{R}_m = [R_1\, R_2; R_2^T\, R_m]$, $\mathcal{S}_m = [S_1\, S_2; S_2^T\, S_m]$ in such a way that, with an appropriate choice of the scheduling function ϕ, the scheduled part (17) of condition (12) holds true. An explicit formula for ϕ is given in [36]. □

This means that we are left to define a class of scalings Q_m, R_m, S_m, which allows reducing the infinite set of LMIs (16) to a finite set. If Θ_m has a block diagonal structure,

$$\Theta_m \;=\; \mathrm{diag}(\theta_{m1} I_{\ell_1}, \ldots, \theta_{ms} I_{\ell_s}), \tag{18}$$

and if prior bounds $|\theta_{mj}(t)| \leq 1$ like for the uncertain parameters are available, this may be done in exactly the same way as for the uncertain part.

Assuming block diagonal structures (15), (18) for both types of parameters, we find it useful in practice to pursue different strategies for the two types of parameters. We use the vertex idea to render the uncertain part of (7) finite, and we use structured scalings for the scheduled parameters. This avoids numerical difficulties which may arise when constructing K if a complicated scheduling function ϕ is required. The use of structured scalings allows the choice $\phi(x) = x$. Notice that for the scheduled parameters, due to condition (i) above, choosing structured scalings implies that each of the sub-blocks Q_1, Q_2, Q_m of $\mathcal{Q}_m$ has the block diagonal structure with diagonal blocks of sizes $\ell_1, \ldots, \ell_s$ in (18). This option finally turned out a good compromise in our numerical tests, and we recommend its use for the type of problem under investigation.

4 Sequential Semi-definite Programming

In this section we cast the robust gain-scheduling control design problem as an optimization problem and present an algorithmic approach to its solution.

Recall from Theorem 2 that the complete vector of decision variables for design is $x = (\gamma, Q, R, S, \tilde{Q}, \tilde{R}, \tilde{S}, X, Y)$. We find it notationally useful to point to parts of the vector x by introducing the notation

$$\Phi_u = \begin{pmatrix} Q_u & S_u \\ S_u^T & R_u \end{pmatrix}, \qquad \tilde{\Phi}_u = \begin{pmatrix} \tilde{Q}_u & \tilde{S}_u \\ \tilde{S}_u^T & \tilde{R}_u \end{pmatrix}$$

involving the uncertain blocks of the scaling variables. Similarly, Φ_m, $\tilde{\Phi}_m$ regroup the scheduled parts of Q, R, S, $\tilde{Q}, \tilde{R}, \tilde{S}$.

Let $\mathcal{A}(x) \leq 0$ represent the LMI constraints (9) - (12), where (12), using one of the techniques from the previous section, has been replaced with a finite set of LMIs, along with $Q \leq 0$ and $\tilde{Q} \leq 0$ required for these procedures. Finally, let $\mathcal{B}(x) = \Phi_u \tilde{\Phi}_u - I = 0$ represent the nonlinear algebraic constraint (13). Then the robust gain-scheduling control problem may be cast in the form (D). More generally, we consider an augmented version (D_c) of (D) for a penalty parameter $c \geq 0$:

$$(D_c) \quad \begin{array}{ll} \text{minimize} & f_c(x) = \gamma + \frac{c}{2}\|\Phi_u \tilde{\Phi}_u - I\|^2 \\ \text{subject to} & \mathcal{A}(x) \leq 0 \\ & \mathcal{B}(x) = \Phi_u \tilde{\Phi}_u - I = 0 \end{array}$$

Remark. Notice that problems (D) and (D_c) are equivalent, since the penalty term $\frac{c}{2}\|\Phi_u \tilde{\Phi}_u - I\|^2$ added in (D_c) will vanish at the optimal x. Using (D_c) instead of (D) may, as we shall see, add some numerical stability. □

Remark. We observe that the variables $Q_m, R_m, S_m, \tilde{Q}_m, \tilde{R}_m, \tilde{S}_m$ and X, Y only occur in the LMI constraint, which strongly indicates that we expect redundancies in the decision parameters. In fact, our experiments indicate that this is a strong point for using structured scalings in the Θ_m block, as this tends to limit these redundancies. In general, we propose to put bounds $\|\cdot\|_\infty \leq M$ on the free variables in order to avoid degeneracy or failure of the successive LMI subproblems. As these additional constraints may be included among the LMIs, $\mathcal{A}(x) \leq 0$, we do not change the notation here. □

Remark. Notice that the trick used in Examples 1 and 2 of the introductory section does not apply in the robust synthesis case, as the matrices Φ_u, $\tilde{\Phi}_u$ are indefinite. This shows that the problem is as a rule numerically harder than e.g. static output feedback design, or reduced order design. □

Let us now extend the idea of SQP to the augmented program (D_c). As we aim at a *primal-dual* method, this requires maintaining estimates for the decision *and* Lagrange multiplier variables. Consider the Lagrangian associated with (D_c):

$$L_c(x;\Lambda,\lambda) \;=\; f_c(x) + \operatorname{trace}(\Lambda\cdot\mathcal{A}(x)) + \lambda^T \operatorname{vec}(\Phi_u\tilde{\Phi}_u - I), \tag{19}$$

where $\Lambda \geq 0$ is a positive semi-definite dual matrix variable, λ a traditional Lagrange multiplier variable whose dimension is m^2, m the size of the matrices $\Phi_u, \tilde{\Phi}_u$. Given the current iterate x and the current Lagrange multiplier estimates λ, $\Lambda \geq 0$, we define the *tangent problem*

$$(T) \quad \begin{array}{ll} \text{minimize} & \nabla f_c(x)^T\Delta x + \frac{1}{2}\Delta x^T\nabla^2 L_c(x;\Lambda,\lambda)\Delta x \\ \text{subject to} & \mathcal{A}(x+\Delta x) \leq 0 \\ & \Phi_u\tilde{\Phi}_u + \Phi_u\,\Delta\tilde{\Phi}_u + \Delta\Phi_u\,\tilde{\Phi}_u - I = 0 \end{array}$$

which consists in minimizing the second order Taylor polynomial of $L_c(x+\Delta x;\Lambda,\lambda)$ about the current x for possible steps Δx, subject to the LMI constraints, $\mathcal{A} \leq 0$, and the equality constraint $\mathcal{B} = 0$ linearized about the current $\Phi_u, \tilde{\Phi}_u$. Notice that the equality constraint above is given in matrix notation. The equivalent expression in long vector notation using the Kronecker product $\otimes$ is:

$$(\tilde{\Phi}_u \otimes I)\operatorname{vec}(\Delta\Phi) \;+\; (I\otimes\Phi_u)\operatorname{vec}(\Delta\tilde{\Phi}_u) \;-\; \operatorname{vec}(I - \Phi_u\tilde{\Phi}_u) \;=\; 0. \tag{20}$$

Here $\tilde{\Phi}_u \otimes I$ is invertible as soon as $\tilde{\Phi}_u$ has maximal rank, while $I \otimes \Phi_u$ is invertible as soon as Φ_u has maximal rank.

Remark. If either Φ_u or $\tilde{\Phi}_u$ is positive definite, we may symmetrize the equality constraint, as considered e.g. in [15]. As mentioned before, this is typically not possible in the robust synthesis case, but may help in different cases. □

The choice of (T) is understood by inspecting the necessary optimality conditions, which show that the solution Δx of (T) may be considered as the Newton step from the current point x to the new iterate $x^+ = x + \Delta x$. The Lagrange multipliers $\Lambda^+ \geq 0$ and λ^+ belonging to the linear constraints in (T) are the updates for Λ and λ. Notice that $\Lambda^+ \geq 0$ as a consequence of the Kuhn-Tucker conditions for (T). Notice further that despite the notation, Λ does not explicitly appear in the Hessian $\nabla^2 L_c(x;\Lambda,\lambda)$ of the Lagrangian, a fact which is due to the linearity of $\mathcal{A} \leq 0$. On the other hand, due to nonlinearity of the equality constraint, λ appears explicitly in the Hessian of the Lagrangian. Updating Λ is then still mandatory to obtain the update λ^+.

Remark. At this stage, we observe that due to the linearity of the LMI constraints, the iterates produced by the SSDP scheme will always satisfy the LMIs, while the nonlinear equality constraint will of course be only approximately satisfied. The fact that we iterate on decision variables satisfying the LMIs is an advantage of our method, since it may render even suboptimal solutions of the optimization problem (D_c) useful for the design (cf. the termination phase in the robust control design algorithm presented at the end of this section). □

The special structure and the moderate size of the variable $(\Phi_u, \tilde{\Phi}_u)$ occurring in the equality constraint $\mathcal{B} = 0$ suggest using a reduced Hessian technique. For fixed x respectively $\Phi_u, \tilde{\Phi}_u$, we can eliminate either $\Delta\Phi_u$ or $\Delta\tilde{\Phi}_u$ from the linearized equality constraint in (T), as long as we maintain iterates x with full rank Φ_u, $\tilde{\Phi}_u$. In that event, the matrix $B = [\tilde{\Phi}_u \otimes I \; I \otimes \Phi_u]$ has full row rank m^2, m the size of the matrix Φ_u, and eliminating the equality constraint therefore reduces the problem size by m^2.

Following the standard notation in SQP, let Z be a matrix whose columns form a basis (preferably orthogonal) of the null space of the matrix B belonging to (20), and let the columns of Y form a basis for the range of B^T. Then we may write the displacement Δx as $\Delta x = Z\Delta\tilde{x} + Yw_0 = Z\Delta\tilde{x} + p_0$ for the fixed vector $p_0 = Y(BY)^{-1}\mathrm{vec}(I - \Phi_u\tilde{\Phi}_u)$, where $\Delta\tilde{x}$ is now the reduced decision vector.

With this notation, the reduced tangent problem is

$$(\tilde{T}) \quad \begin{array}{ll} \text{minimize} & (\nabla f_c(x)^T Z + p_0^T \nabla^2 L_c Z)\Delta\tilde{x} + \frac{1}{2}\Delta\tilde{x} Z^T \nabla^2 L_c Z \Delta\tilde{x} \\ \text{subject to} & \mathcal{A}_* \circ Z(\Delta\tilde{x}) \leq -\mathcal{A}(x + p_0) \end{array}$$

where $\mathcal{A}_*$ is the linear part of $\mathcal{A}$. Notice that in general $(\tilde{T})$ is not yet an SDP, since the reduced Hessian $Z^T\nabla^2 L_c Z$ may be indefinite. In order to obtain a convex program, we have to convexify the reduced Hessian, which may be done in several ways. We comment on these at the end of the section.

The correction done, the subproblem is convex and may easily be transformed into a semi-definite programming problem. Ideally, the solution $\Delta\tilde{x}$ gives rise to a step Δx in the original tangent problem, and the new iterate x^+ is obtained as $x + \Delta x$, but in practice a line search using an appropriate merit function is required. For appropriate choices avoiding the Maratos effect we refer to the vast literature on the subject (see e.g. [10], [12]).

In order to obtain the Lagrange multiplier updates, we have to inspect the necessary optimality conditions for $(\tilde{T})$. Let $\tilde{\Lambda}^+ \geq 0$ be the Lagrange multiplier matrix variable in $(\tilde{T})$ associated with the constraint $\mathcal{A} \leq 0$, and let $\Delta\tilde{x}$ be the optimal solution of $(\tilde{T})$. Then the optimal Δx is readily obtained via (20), Λ^+ is chosen as $\tilde{\Lambda}^+$, while λ^+ is found through

$$Y^T\nabla f_c(x) + Y^T\nabla^2 L_c(Z\Delta\tilde{x} + p_0) + Y^T\mathcal{A}_*^T\Lambda^+ + Y^T B^T\lambda^+ = 0 \tag{21}$$

which determines λ^+ uniquely if B has full rank. Conceptually, the SSDP algorithm proposed to solve (D) may be described as follows:

SSDP - Algorithm

1. Find an initial point x^0, such that $\mathcal{A}(x^0) \leq 0$ and such that Φ_u^0, $\tilde{\Phi}_u^0$ are full rank. Select Lagrange multiplier estimates λ^0 and $\Lambda^0 \geq 0$ using formula (21).

2. Given the iterate x^k with Φ_u^k and $\tilde{\Phi}_u^k$ nonsingular, and multiplier estimates $\Lambda^k \geq 0$, λ^k, form the reduced tangent problem $(\tilde{T}_k)$ about the current data. Render the reduced Hessian positive definite if required. Obtain the reduced step $\Delta\tilde{x}^k$ as a solution to the SDP, and let $\Delta x^k = Z_k \Delta\tilde{x}^k + p_0^k$. Obtain Lagrange multipliers $\Lambda^\sharp \geq 0$ and $\lambda^\sharp$ from $(\tilde{T}_k)$ using (21).

3. Do a line search in direction Δx^k using an appropriate merit function and determine the new iterate $x^{k+1} = x^k + \alpha_k \Delta x^k$. Set $\Lambda^{k+1} = \Lambda^k + \alpha_k(\Lambda^\sharp - \Lambda^k)$ and $\lambda^{k+1} = \lambda^k + \alpha_k(\lambda^\sharp - \lambda^k)$. Choose α_k so that Φ_u^{k+1} and $\tilde{\Phi}_u^{k+1}$ are nonsingular.

4. Check the stopping criteria. Either halt, or replace k by $k+1$ and go back to step 2.

In order to compute the Hessian $\nabla^2 L(x; \Lambda, \lambda)$ of the Lagrangian in step 2, only second-order derivatives with respect to Φ_u and $\tilde{\Phi}_u$ are required, as $f_c(x)$ is linear in γ and does not depend on the other decision variables. Using the Kronecker product $\otimes$, we have the following formulae (cf. also [16]):

Lemma 5.

$$\nabla^2_{\Phi_u \Phi_u} L_c = c\,(\widetilde{\Phi}_u \otimes I)^T (\widetilde{\Phi}_u \otimes I), \;\; \nabla^2_{\tilde{\Phi}_u \tilde{\Phi}_u} L_c = c\,(I \otimes \Phi_u)^T (I \otimes \Phi_u)\ ,$$
$$\nabla^2_{\tilde{\Phi}_u \Phi_u} L_c = (I \otimes \mathrm{mat}(\lambda))^T + c((\Phi_u \widetilde{\Phi}_u - I)^T \otimes I + (I \otimes \Phi_u)^T (\widetilde{\Phi}_u \otimes I))\ ,$$

□

Remark. Let us comment on the convexification of the reduced tangent problem $(\tilde{T})$, required to obtain an SDP. Recent trends in optimization indicate that one should dispense with this procedure. It is considered important to take the directions of negative curvature of the (reduced) Hessian into account, e.g. by using a trust region strategy, or by doing sophisticated line searches which combine the Newton direction and the dominant direction of negative curvature. While the second idea could be at least partially realized, a trust region approach is not feasible as yet in the presence of LMI constraints, as optimizing a non-convex quadratic function subject to LMIs is presently too difficult numerically to become a functional scheme. We therefore have to use the well-known convexification methods used in nonlinear optimization over many years, and we refer to [9, 23] for several such strategies.

In our numerical experiments, we tested Powell's idea of doing a Cholesky factorization, and adding correction terms as soon as negative square roots appear, and a direct method which used the QR-factorization to correct negative eigenvalues of the reduced Hessian. A third method adapted to the structure of the problem which we found even more efficient consisted in a Gauss-Newton type idea. We neglect the term $\Phi_u \widetilde{\Phi}_u - I$ in the Hessian matrix (22), performing the modified Cholesky factorization on the remaining term. This is motivated by the fact that dropping this term leaves a positive semi-definite matrix, which is still close to the correct Hessian as long as the neglected term $\Phi_u \widetilde{\Phi}_u - I$ is small. This is the case when the nonlinear constraint (13) is approximately satisfied, and the matrix is therefore asymptotically close to the correct (reduced) Hessian. As a consequence, and in contrast with the true Gauss-Newton method, this procedure therefore does not destroy the superlinear quadratic convergence of the scheme.

Observe that in all these procedures, the augmented form (D_c) of the program helps. In fact, the penalty term renders the Hessian more convex than in the original form (D), so the corrections are often very mild in practice, and according to the theory in polyhedral programming are not even required asymptotically (cf. [8, 10, 12]). This observation is corroborated in our experiments with LMI-constraints. □

We summarize the results of this section by presenting the following algorithmic approach to the robust gain-scheduling design problem:

Algorithm for Robust Gain-scheduling Control Design

Step 1. Initialization. Locate a strictly feasible decision vector x^0 for the LMI constraints: For fixed large enough $\gamma = \gamma_0$, render 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, Φ_u^0 and $\widetilde{\Phi}_u^0$ so that $\Phi_u^0\widetilde{\Phi}_u^0 - I$ is as close as possible to zero. Then initialize the Lagrange multiplier estimates λ_0 and $\Lambda_0 \geq 0$.

Step2. Optimization. Solve the optimization problem (D_c) via SSDP, using $(x^0, \Lambda^0, \lambda^0)$ as primal-dual starting point. The primal solution is x.

Step 3. Terminating phase. Due to non-linearity, the algebraic constraints (13) is never exactly satisfied at the solution x. It is, however, possible to terminate the program without strict satisfaction of the nonlinear constraints by a simple *perturbation technique* [5], which is applicable as long as the LMIs (9) - (12) are strictly satisfied. One can then replace Φ_u with $\widetilde{\Phi}_u^{-1}$ 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 $\widetilde{\Phi}_u$ with Φ_u^{-1} and check the LMI constraints (9) - (11), with (7) respectively (17) suitably replaced with its dual form

$$\left(\begin{array}{c} I \\ -\Theta_i^T \end{array}\right)^T \widetilde{\Phi}_u \left(\begin{array}{c} I \\ -\Theta_i^T \end{array}\right) < 0 \quad \forall i = 1, \cdots, N.$$

If the test fails, the numerical solution to (D_c) is unsatisfactory and has to be improved, e.g. by changing the stopping criteria, or by increasing the penalty constant c and re-running step 2.

Remark. Notice that strict feasibility < 0 is a priori not guaranteed by SSDP, but may easily be forced if we replace ≤ 0 in the corresponding LMIs by the stronger $\leq -\varepsilon I$ for a small $\varepsilon > 0$. Moreover, if the SDP subproblem is solved by the notorious interior point techniques, the LMIs are automatically strictly satisfied, and the above perturbation argument is applicable. □

5 Fast Local Convergence of SSDP

In this section we prove local superlinear and quadratic convergence of the SSDP method under mild regularity hypotheses. It is interesting to recall the history of the SQP method, which became already popular during the late 1970ties, even though the first proof of superlinear and quadratic convergence under realistic assumptions was published as late as 1994 by Bonnans [11]. A more compact version of that proof is published in [12]. Both versions are based on techniques introduced by Robinson in the 1980ties.

The time interval is the more remarkable, as the equality constrained case was settled much earlier, apparently first by Boggs and Tolle [10] around 1982. Early proofs of the general case existed but always reduced the situation to the equality constrained case under the (unrealistic) assumption of strict complementarity at the optimal pair.

Inspecting the convergence proofs for Newton's method in [11, 12] shows that they heavily depend on the polyhedrality of the order cone in classical nonlinear programming, so a natural extension to the present case of SSDP does not seem near at hand. Our present approach is nevertheless inspired by Bonnans's paper [11]. It turns out that our method of proof applies even to more general situations, and we present the method in a fairly general context.

We consider the nonlinear programming problem of the form

$$(P) \quad \begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_E(x) = 0 \\ & g_I(x) \in K^0 \end{array}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $g_E : \mathbb{R}^n \to \mathbb{R}^m$ and $g_I : \mathbb{R}^n \to \mathbb{R}^p$ are $\mathcal{C}^2$-functions, K is a cone in $\mathbb{R}^p$, and K^0 is its polar cone defined as

$$K^0 = \{y \in \mathbb{R}^p : \langle x, y\rangle \leq 0 \text{ for each } x \in K\}.$$

In the classical nonlinear programming case $K = \mathbb{R}^p_+$, $K^0 = \mathbb{R}^p_-$, the constraint $g_I(x) \in K^0$ becomes $g_i(x)_leq0$ componentwise, while in the semi-definite case $\mathbb{R}^p \cong \mathbb{S}^r$, (with $p = r(r+1)/2$), the space of symmetric $r \times r$-matrices, $K = \mathbb{S}^r_+$, (with $K^0 = \mathbb{S}^r_-$), the cone of positive semi-definite matrices, the constraint $g_i(x) \in K^0$ means that the matrix $g_i(x)$ is negative semidefinite. We use the notation $\langle\cdot,\cdot\rangle$ for the scalar product employed, since this may include the classical case $\langle x, y\rangle = \sum_i x_i y_i$, as well as $\langle x, y\rangle = \text{trace}(x \cdot y)$ in the semi-definite case. The adjoint of an operator A with respect to this scalar product is denoted A^*, derivatives with respect to $\langle\cdot,\cdot\rangle$ in the x-variable are indicated by primes. Notice also that $g_I(x)$ was an affine matrix valued function in our applications, but we prefer to include the general nonlinear case, as applications of this type are eminent.

We suppose that $\bar{x}$ is a local minimum of (P) and that there exists a Lagrange multiplier $\bar{\lambda} = (\bar{\lambda}_E, \bar{\lambda}_I)$ satisfying the necessary optimality condition

$$(KT) \quad \begin{array}{ll} (1) & f'(\bar{x}) + g'(\bar{x})^*\bar{\lambda} = 0; \\ (2) & g_E(\bar{x}) = 0; \\ (3) & g_I(\bar{x}) \in K^0,\ \lambda_I \in K,\ \langle g_I(x), \bar{\lambda}_I\rangle = 0. \end{array}$$

Observe that the existence of $\bar{\lambda}$ is guaranteed under a weak regularity assumption, like for instance Robinson's constraint qualification hypothesis (cf. [33]). The Lagrangian associated with (P) is

$$L(x;\lambda) \;=\; f(x) + \langle g(x),\lambda\rangle \;=\; f(x) + \langle g_E(x),\lambda_E\rangle + \langle g_I(x),\lambda_I\rangle. \tag{22}$$

We consider Newton's method for solving the Kuhn-Tucker system (KT) which generates a sequence (x^k,λ^k) approximating the optimal pair $(\bar{x},\bar{\lambda})$. Given the kth iterate (x^k,λ^k), the $(k+1)$st iterate is obtained by solving the tangent problem

$$(T_k)\quad \begin{array}{ll} \text{minimize} & \langle f'(x^k),\Delta x\rangle + \frac{1}{2}\langle \Delta x, L''(x^k;\lambda^k)\Delta x\rangle \\ \text{subject to} & g_E(x^k) + g_E'(x^k)\Delta x \;=\; 0 \\ & g_I(x^k) + g_I'(x^k)\Delta x \in K^0 \end{array}$$

If Δx is the solution to (T_k), then $x^{k+1} \;=\; x^k + \Delta x$. The Lagrange multiplier update $\lambda^{k+1} = (\lambda_E^{k+1},\lambda_I^{k+1})$ is just the Lagrange multiplier belonging to the linearized constraints in (T_k). The Kuhn-Tucker conditions for (T_k) are the following:

$$(KT_k)\quad \begin{array}{l} L'(x^k;\lambda^{k+1}) + L''(x^k;\lambda^k)(x^{k+1}-x^k) \;=\; 0 \\ g_E(x^k) + g_E'(x^k)(x^{k+1}-x^k) \;=\; 0 \\ g_I(x^k) + g_I'(x^k)(x^{k+1}-x^k) \in K^0,\; \lambda_I^{k+1}\in K, \\ \qquad \langle g_I(x^k) + g_I'(x^k)(x^{k+1}-x^k),\lambda_I^{k+1}\rangle = 0. \end{array}$$

The aim of the following analysis is to give sufficient conditions for local quadratic or superlinear convergence of the sequence (x^k,λ^k).

Remark. The usual choice of quasi Newton methods is easily obtained from our scheme by approximating the Hessian $L''(x^k;\lambda^k)$ of the Lagrangian of (P) by a matrix M^k. In order to account for modifications of $L''(x^k;\lambda^k)$ like convexifications as proposed in our experimental section, we include the quasi-Newton approach into our convergence analysis. We shall use the notation $(T_k(M^k))$ for the modified tangent problem with M^k replacing the Hessian of the Lagrangian. □

Inspecting classical approaches for the usual polyhedral cone in nonlinear programming shows that local convergence of Newton's method usually requires two types of hypothesis, (a) the second order sufficient optimality condition, and (b) a constraint qualification. As we mentioned before, a third type of condition, strict complementarity, is often used but should be avoided, since it is artificial as a rule. At the core is the second order sufficient optimality condition, saying that the Hessian of the Lagrangian $L''(\bar{x};\bar{\lambda})$ is positive definite along critical directions. We adopt the definition of critical directions from [33, 11], which in the presence of a multiplier leads to the following:

Definition. The direction $h \neq 0$ is critical at $\bar{x}$ with respect to the Lagrange multiplier $\bar{\lambda}$ if

1. $g_E'(\bar{x})h = 0$;

2. There exists $h^k \to h$, $x^k = \bar{x} + t_k h^k$ with $t_k \to 0^+$ in tandem with $\lambda_I^k \to \bar{\lambda}_I$, $\lambda_I^k \in K$ such that for some v^k with $v^k = \mathrm{o}(t_k)$, $g_I(x^k) - v^k \in K^0$ and $\langle g_I(x^k) - v^k, \lambda_I^k\rangle = 0$.

□

Remark. Recall that in the case of the polyhedral cone $K = \mathbb{R}^p_+$ in nonlinear programming, a critical direction h satisfies condition (1) for the equality constraints along with the following condition (2′) for inequalities: $g_i'(\bar{x})h = 0$ for active constraints $i \in I$ having multiplier $\bar{\lambda}_i > 0$, and $g_i'(\bar{x})h \leq 0$ for active constraints $i \in I$ where $\bar{\lambda}_i = 0$. It is an easy exercise to show that in this case, (2) is equivalent to this classical definition of criticality (2′). □

Let us now start upon analyzing the Newton step for (P) via the following perturbation result.

Lemma 6. *Suppose there exist sequences $x^k \to \bar{x}$, $\lambda^k \to \bar{\lambda}$, $\delta_k \to 0^+$ and $u^k, v^k = (v_E^k, v_I^k)$ satisfying $u^k = \mathcal{O}(\delta_k)$, $v^k = \mathcal{O}(\delta_k)$, such that*

1. $L'(x^k; \lambda^k) \; = \; u^k$;
2. $g_E(x^k) \; = \; v_E^k$;
3. $g_I(x^k) - v_I^k \in K^0, \; \langle g_I(x^k) - v_I^k, \lambda_I^k \rangle = 0, \; \lambda_I^k \in K.$

Suppose further that the second order sufficient optimality condition is satisfied at $(\bar{x}, \bar{\lambda})$, i.e., $\langle h, L''(\bar{x}; \bar{\lambda})h \rangle > 0$ for every critical direction $h \neq 0$, and that $g'(\bar{x})$ has maximal rank. Then $x^k - \bar{x} = \mathcal{O}(\delta_k)$ and $\lambda^k - \bar{\lambda} = \mathcal{O}(\delta_k)$.

Proof. Subtracting equation (1) in the Kuhn-Tucker equations from the perturbed equation (1) above gives

$$L'(x^k; \bar{\lambda}) - L'(\bar{x}; \bar{\lambda}) + g'(x^k)^*(\lambda^k - \bar{\lambda}) \; = \; u^k. \tag{23}$$

Now it suffices to show $x^k - \bar{x} = \mathcal{O}(\delta_k)$, for then the first term $L'(x^k; \bar{\lambda}) - L'(\bar{x}; \bar{\lambda})$ on the left hand side of (23) is $\mathcal{O}(\delta_k)$, hence so is the second term. Since $g'(x^k) = g'(\bar{x}) + \mathcal{O}(x^k - \bar{x})$, this implies $g'(\bar{x})^*(\lambda^k - \bar{\lambda}) = \mathcal{O}(x^k - \bar{x})$, and since $g'(\bar{x})$ has maximal rank, we conclude $\lambda^k - \bar{\lambda} = \mathcal{O}(x^k - \bar{x}) = \mathcal{O}(\delta_k)$.

Suppose now that the result is incorrect, so $u^k/\|x^k - \bar{x}\| \to 0$, $v^k/\|x^k - \bar{x}\| \to 0$. Picking a subsequence if necessary, we may assume that $(x^k - \bar{x})/\|x^k - \bar{x}\| \to h$ with $\|h\| = 1$. We show that h is a critical direction.

Notice first that subtracting the perturbed condition (2) from condition (2) in the Kuhn-Tucker ensemble gives

$$\frac{g_E(x^k) - g_E(\bar{x})}{\|x^k - \bar{x}\|} \; = \; \frac{v_E^k}{\|x^k - \bar{x}\|} \; \to 0,$$

hence the equality part (1) of criticality is satisfied. As for the inequality part, observe that the perturbed conditions (1) - (3) just match the second part of the definition of criticality if we use the standing hypothesis that $x^k \to \bar{x}$ slower than $\delta_k \to 0$. Hence h is critical.

To conclude the proof, let us multiply equation (23) above by $x^k - \bar{x}$, and divide by $\|x^k - \bar{x}\|^2$. The right hand term of the modified equation is then

$$\frac{\langle u^k, x^k - \bar{x} \rangle}{\|x^k - \bar{x}\|^2} \to 0$$

by our standing hypothesis, so the left hand side of the modified equation also has to converge to 0. The first term on the left hand side of the modified equation is

$$\frac{\langle L'(x^k;\bar\lambda) - L'(\bar x;\bar\lambda), x^k - \bar x\rangle}{\|x^k - \bar x\|^2},$$

which converges to $\langle L''(\bar x;\bar\lambda)h, h\rangle$. Since the direction h was seen to be critical, this term is strictly positive. We shall now obtain the sought for contradiction by showing that the remaining term on the left hand side of the modified equation is asymptotically nonnegative. This is verified by splitting this term into its equality and inequality part.

The equality part of the term in question is

$$\frac{\langle g_E'(x^k)(x^k - \bar x), \lambda_E^k - \bar\lambda_E\rangle}{\|x^k - \bar x\|^2},$$

which due to $g_E'(\bar x)h = 0$ tends to 0. This argument uses the fact that $\lambda_E^k - \bar\lambda_E = \mathcal{O}(x^k - \bar x)$, which itself is a consequence of the standing hypothesis, equation (23), and the constraint qualification.

Inspecting the inequality term remains. Via Taylor expansion, the latter is

$$\frac{\langle g_I(x^k) - g_I(\bar x), \lambda_I^k - \bar\lambda_I\rangle}{\|x^k - \bar x\|^2} + \mathrm{o}(1), \tag{24}$$

again using $\lambda^k - \bar\lambda = \mathcal{O}(x^k - \bar x)$. The left hand term of (24) is recast as

$$\frac{\langle g_I(x^k) - v_I^k - g_I(\bar x), \lambda_I^k - \bar\lambda_I\rangle}{\|x^k - \bar x\|^2} + \frac{\langle v_I^k, \lambda_I^k - \bar\lambda_I\rangle}{\|x^k - \bar x\|^2}, \tag{25}$$

and the second term in (25) tends to 0 due to the standing hypothesis. The first term in (25) is nonnegative, for expanding its nominator gives

$$\langle g_I(x^k) - v_I^k, \lambda_I^k\rangle - \langle g_I(x^k) - v_I^k, \bar\lambda_I\rangle - \langle g_I(\bar x), \lambda_I^k\rangle + \langle g_I(\bar x), \bar\lambda_I\rangle.$$

Here the first and the last term vanish as a consequence of the complementarity condition (3) in the Kuhn-Tucker ensemble (KT), and the perturbed condition (3) above, while the two terms with the negative signs are themselves negative, again due to the conditions (3) above and in (KT). Indeed, $\lambda_I^k, \bar\lambda_I \in K$ and $g_I(\bar x) \in K^0$, $g_I(x^k) - v_I^k \in K^0$, imply $\langle g_I(x^k) - v_I^k, \bar\lambda\rangle \le 0$ and $\langle g_I(\bar x), \lambda_I^k\rangle \le 0$. This settles the case by providing the desired contradiction. □

With this observation, we are now ready to state our first result.

Lemma 7. *Suppose Newton's method for solving (P) via successive solution of $(T_k(M^k))$ with a choice of matrices M^k generates a sequence of iterates (x^k, λ^k) which converges to the Kuhn-Tucker pair $(\bar x, \bar\lambda)$. Suppose further that $g'(\bar x)$ has maximal rank, and that the second order sufficient optimality condition is satisfied at $(\bar x, \bar\lambda)$.*

1. *If $M^k \to L''(\bar x;\bar\lambda)$, convergence $(x^k, \lambda^k) \to (\bar x, \bar\lambda)$ is superlinear.*
2. *If $M^k - L''(\bar x;\bar\lambda) = \mathcal{O}(x^k - \bar x)$, then convergence $(x^k, \lambda^k) \to (\bar x, \bar\lambda)$ is even quadratic.*

Proof. We observe that with $x^{k+1} = x^k + \Delta x$, and λ^{k+1} the Lagrange multiplier in $(T_k(M^k))$, the quasi-Newton step about the current iterate (x^k, λ^k) may be represented as

$$\begin{array}{ll} (i) & L'(x^{k+1}; \lambda^{k+1}) \;=\; u^k; \\ (ii) & g_E(x^{k+1}) = v_E^k; \\ (iii) & g_I(x^{k+1}) - v_I^k \in K^0,\ \lambda_I^{k+1} \in K,\ \langle g_I(x^{k+1}) - v_I^k, \lambda_I^{k+1}\rangle = 0. \end{array}$$

where the perturbation terms u^k and $v^k = (v_E^k, v_I^k)$ are as follows:

$$\begin{aligned} u^k \;=&\; L'(x^{k+1}; \bar\lambda) - L'(x^k; \bar\lambda) - L''(x^k; \bar\lambda)(x^{k+1} - x^k) + (L''(\bar x; \bar\lambda) - M^k)(x^{k+1} - x^k) \\ &+ (L''(x^k; \bar\lambda) - L''(\bar x; \bar\lambda))(x^{k+1} - x^k) + (g'(x^{k+1}) - g'(x^k))^*(\lambda^{k+1} - \bar\lambda) \\ v^k \;=&\; -g(x^{k+1}) + g(x^k) + g'(x^k)(x^{k+1} - x^k). \end{aligned}$$

As we wish to bring in the perturbation Lemma 6 above, we let $\delta_k \to 0$ the speed of convergence of $(u^k, v^k) \to (0,0)$, then $(x^k, \lambda^k) - (\bar x, \bar\lambda) = \mathcal{O}(\delta_k)$ as a consequence of that lemma.

Now observe that $v^k = \mathrm{o}(x^{k+1} - x^k)$, and similarly $u^k = \mathrm{o}(x^{k+1} - x^k)$ if we use the hypothesis $M^k - L''(\bar x; \bar\lambda) = \mathrm{o}(1)$. Altogether, $\delta_k = \mathrm{o}(x^{k+1} - x^k)$. The perturbation Lemma therefore implies

$$x^{k+1} - \bar x = \mathcal{O}(\delta_k) = \mathrm{o}(x^{k+1} - x^k) = \mathrm{o}(\|x^{k+1} - \bar x\| + \|x^k - \bar x\|).$$

Similarly, as $g'(\bar x)$ has maximal rank,

$$\lambda^{k+1} - \bar\lambda = \mathcal{O}(\delta_k) = \mathrm{o}(\|x^{k+1} - \bar x\| + \|x^k - \bar x\|).$$

These estimates prove superlinear convergence.

The argument giving quadratic convergence under the stronger hypothesis in (2) is standard and left to the reader (see for instance [11]). □

As a consequence of Lemma 7, what remains to be checked is mere convergence of Newton's method under the same regularity hypotheses. Here we shall be able to follow a known line of argument already present in Robinson's approach [33]. Let us consider the *limiting tangent problem*

$$(T_\infty) \quad \begin{array}{ll} \text{minimize} & \langle f'(\bar x), d\rangle + \frac{1}{2}\langle d, L''(\bar x; \bar\lambda) d\rangle \\ \text{subject to} & g_E(\bar x) + \langle g_E'(\bar x), d\rangle \;=\; 0 \\ & g_I(\bar x) + \langle g_I'(\bar x), d\rangle \;\in\; K^0 \end{array}$$

whose optimal solution is $\bar d = 0$, and for which $\bar\lambda$ is a Lagrange multiplier. Observe that the second order optimality conditions for (T_∞) are identical with those of (P), so if we adopt the constraint qualification from before and the second order sufficient optimality condition for (P), they also hold for (T_∞). Using a result obtained by Robinson [33, Theorems 2.3, 3.1], we have the following

Lemma 8. *Suppose the second order sufficient optimality condition for (P) is satisfied at the optimal pair $(\bar x, \bar\lambda)$. Suppose further that $g'(\bar x)$ has maximal rank. Then, given $\varepsilon > 0$, there exists $\delta > 0$ such that if $\|x^k - \bar x\| < \delta$, $\|\lambda^k - \bar\lambda\| < \delta$ and $\|M^k - L''(\bar x; \bar\lambda)\| < \delta$,* *the tangent problem* $(T_k(M^k))$ *has a local minimum* x^{k+1} *and an associated Lagrange multiplier* λ^{k+1} *satisfying* $\|x^{k+1} - \bar{x}\| < \varepsilon$ *and* $\|\lambda^{k+1} - \bar{\lambda}\| < \varepsilon$.

Proof. Notice that the tangent subproblem $(T_k(M^k))$ may be considered a perturbed version of the ideal tangent problem (T_∞) in the sense of [33, (2.7)]. Now by assumption $g'(\bar{x})$ has maximal rank, and hence (T_∞) is regular in the sense of [33]. Secondly, since (P) satisfies the second order sufficient optimality condition at $(\bar{x}, \bar{\lambda})$, so does (T_∞) at the optimal pair $(0, \bar{\lambda})$. Using [33, Thm. 3.1], there exist neighborhoods N_1 of $\bar{x}$, N_2 of $\bar{\lambda}$, and N_3 of $L''(\bar{x}; \bar{\lambda})$ such that for $x^k \in N_1$, $\lambda^k \in N_2$, and $M^k \in N_3$, the tangent problem $(T_k(M^k))$ has a solution x^{k+1}. We may in addition choose N_1 small enough to guarantee that $g'(x^k)$ has maximal rank, and therefore $(T_k(M^k))$ also admits Lagrange multipliers λ^{k+1}.

Now using Theorem 2.3 of the same paper, the set-valued operator mapping the datum (x^k, λ^k, M^k) of $(T_k(M^k))$ into the set of possible optimal pairs (x^{k+1}, λ^{k+1}) is upper semi-continuous. By second order sufficient optimality, $(\bar{x}, \bar{\lambda})$ is locally unique. Therefore upper semi-continuity translates into the following statement: Given $\varepsilon > 0$, there exists $\delta > 0$ such that if (x^k, λ^k, M^k) is in the δ-neighborhood of $(\bar{x}, \bar{\lambda}, L''(\bar{x}; \bar{\lambda}))$, then any (x^{k+1}, λ^{k+1}) lies in the ε-neighborhood of $(\bar{x}, \bar{\lambda})$. This is just what we claimed. □

With these auxiliary results, we are now ready to state our local convergence theorem for Newton's method.

Theorem 9. *Let* $(\bar{x}, \bar{\lambda})$ *be a Kuhn-Tucker pair for* (P) *satisfying the second order sufficient optimality condition, and suppose* $g'(\bar{x})$ *has maximal rank. Then there exists* $\delta > 0$ *such that if* $\|x^0 - \bar{x}\| < \delta$, $\|\lambda^0 - \bar{\lambda}\| < \delta$, $\|M^k - L''(\bar{x}; \bar{\lambda})\| < \delta$ *for every* k, *and* $M^k \to L''(\bar{x}; \bar{\lambda})$, *the sequence* (x^k, λ^k) *obtained by successive solution of the tangent subproblems* $(T_k(M^k))$ *is well-defined and converges superlinearly to* $(\bar{x}, \bar{\lambda})$. *Convergence is even quadratic if* $M^k - L''(\bar{x}; \bar{\lambda}) = \mathcal{O}(\|x^k - \bar{x}\| + \|\lambda^k - \bar{\lambda}\|)$.

Proof. 1) Observe that the perturbation Lemma 6 tells that due to second order sufficient optimality, the Kuhn-Tucker conditions for (P) follow a Lipschitz type behavior with respect to specific perturbations u^k, v^k. Let us quantify this: There exist $\delta_1 > 0$ and $\alpha > 0$, $\beta > 0$ such that if u^k, v^k are sufficiently small in the sense that $\|v^k\|$, $\|u^k\| < \delta_1$, and if x^k, λ^k along with u^k, v^k satisfy (1) - (3) in the perturbation Lemma 6, then $\|x^k - \bar{x}\| + \|\lambda^k - \bar{\lambda}\| < \alpha$, and $\|x^k - \bar{x}\| + \|\lambda^k - \bar{\lambda}\| < \beta(\|u^k\| + \|v^k\|)$.

2) Let $\delta_3 \le \min(\alpha, \frac{1}{3\beta})$. According to Lemma 8 above, there exists $\delta_2 > 0$ such that whenever $\|\hat{x} - \bar{x}\| < \delta_2$, $\|\hat{\lambda} - \bar{\lambda}\| < \delta_2$, and $\|\hat{M} - L''(\bar{x}; \bar{\lambda})\| < \delta_2$, the result (x, λ) of the Newton step with datum $(\hat{x}, \hat{\lambda}, \hat{M})$ satisfies $\|(x, \lambda) - (\bar{x}, \bar{\lambda})\| < \delta_3$.

3) Choose $\delta_4 > 0$ such that the following 5 conditions are satisfied. First,

$$\|g(x^1) - g(x^2) - g'(x^2)(x^1 - x^2)\| \le \frac{1}{24\beta}\|x^1 - x^2\|$$

whenever $x^1, x^2 \in B(\bar{x}, \delta_4)$. Secondly,

$$\|L'(x^1; \bar{\lambda}) - L'(x^2; \bar{\lambda}) - L''(x^2; \bar{\lambda})(x^1 - x^2)\| \le \frac{1}{24\beta}\|x^1 - x^2\|$$

whenever $x^1, x^2 \in B(\bar{x}, \delta_4)$. Thirdly, $\delta_4 < 1/24\beta$, fourthly,

$$\|L''(x^1; \bar{\lambda}) - L''(\bar{x}; \bar{\lambda})\| \leq \frac{1}{24\beta}$$

whenever $x^1 \in B(\bar{x}, \delta_4)$. Finally,

$$\|(g'(x^2) - g'(x^1))(\lambda^1 - \lambda^2)\| \leq \frac{1}{24\beta}\|\lambda^1 - \lambda^2\|$$

whenever $\lambda^1, \lambda^2 \in B(\bar{\lambda}, \delta_4)$ and $x^1, x^2 \in B(\bar{x}, \delta_4)$.

4) Now choose $\delta = \min(\delta_1, \delta_2, \delta_3, \delta_4)$, then the conclusion of the theorem holds. In fact, let (x^k, λ^k, M^k) be the datum of the kth Newton step. As $\delta \leq \delta_1$, an optimal pair (x^{k+1}, λ^{k+1}) exists and satisfies $\|(x^{k+1}, \lambda^{k+1}) - (\bar{x}, \bar{\lambda})\| < \delta_3$.

As in the proof of Lemma 7, let us write the Newton step in the form

$$\begin{array}{ll} (i) & L(x^{k+1}; \lambda^{k+1}) = u^k \\ (ii) & g_E(x^{k+1}) = v_E^k \\ (iii) & g_I(x^{k+1}) - v_I^k \in K^0,\ \lambda_I^{k+1} \in K,\ \langle g_I(x^{k+1}) - v_I^k, , \lambda_I^{k+1}\rangle = 0, \end{array}$$

where u^k, v^k have the meaning given there. Then $\delta \leq \delta_2$ and $\delta \leq \delta_1$ and step 1) imply $\|u^k\| \leq \frac{1}{6\beta}\|x^{k+1} - x^k\|$ and $\|v^k\| \leq \frac{1}{6\beta}(\|x^{k+1} - x^k\| + \|\lambda^{k+1} - \lambda^k\|)$. Therefore step 1) implies

$$\|x^{k+1} - \bar{x}\| \leq \beta \frac{1}{6\beta}(\|x^{k+1} - x^k\| + \|\lambda^{k+1} - \lambda^k\|) \leq$$
$$\frac{1}{6}(\|x^k - \bar{x}\| + \|x^{k+1} - \bar{x}\| + \|\lambda^k - \bar{\lambda}\| + \|\lambda^{k+1} - \bar{\lambda}\|),$$

and similarly for $\|\lambda^{k+1} - \bar{\lambda}\|$. Adding both estimates gives

$$\|(x^{k+1}, \lambda^{k+1}) - (\bar{x}, \bar{\lambda})\| \leq \frac{1}{3}(\|(x^k, \lambda^k) - (\bar{x}, \bar{\lambda})\| + \|(x^{k+1}, \lambda^{k+1}) - (\bar{x}, \bar{\lambda})\|).$$

Therefore,

$$\|(x^{k+1}, \lambda^{k+1}) - (\bar{x}, \bar{\lambda})\| \leq \frac{1}{2}\|(x^k, \lambda^k) - (\bar{x}, \bar{\lambda})\|,$$

and this proves linear convergence of the sequence. This settles the case, since it proves in particular that the situation needed to start this argument is reproduced at each step. □

Remarks. 1) Notice that Newton's method $M^k = L(x^k; \lambda^k)$ satisfies hypothesis (2) and therefore converges quadratically.

2) As is well-known, superlinear and quadratic convergence of the primal-dual pair (x^k, λ^k) does not imply superlinear or quadratic convergence of the primal sequence x^k. In order to establish primal superlinear convergence, an extra argument is needed, and this leads to a result in the style of the classical Dennis-Moré characterization of superlinear convergence for unconstrained optimization. The result is similar to [12, Théorème 11.5], and we do not present the details here.

In the same vein, a more thorough analysis of the SSDP method will have to address the following elements not considered here due to the lack of space. A global convergence analysis based on an appropriate merit function, and an extension of the known result in polyhedral programming saying that the Hessian of the augmented Lagrangian is positive definite at the optimal pair if the penalty parameter c is properly chosen (cf. for instance [12, Proposition 12.2]). In particular, in our numerical tests we observed this effect, and it should be proven rigorously in order to fully justify our approach via SSDP.

3) Let us point to the main difference of our approach to the setting [11]. Following Robinson's methods, Bonnans embeds the Newton step for (P) into the Newton step of a suitably formulated variational inequality, and then establishes a perturbation Lemma in the style of Lemma 6, but with perturbations based on the variational formulation. Consequently, more (and in fact, too many) perturbations are allowed, and the Lipschitz type behavior is then only established under polyhedrality.

6 Existing Techniques and Comparison

In principle, the optimization step in our control design algorithm may be replaced with any optimization technique adapted to deal with LMI constraints. Here we shall compare SSDP to two other methods which we have previously used in robust control design, the augmented Lagrangian method, and an approach via concave programming. Numerical experiments based on interior-point methods have been reported by Leibfritz et al. [25, 26] for a different but related type of application. Our own experiments with the interior-point approach will be presented in [3].

A thorough investigation of nonlinear optimization techniques in robust control synthesis should include comparison with existing techniques like the D-K iteration scheme. This has already been addressed in [16], where test examples similar to the ones here were used to compare these approaches. As a result, we observed that D-K was not at ease with these seemingly innocent cases, and very often could not even be started due to lack of a useful initial controller.

A partially augmented Lagrangian scheme for solving (D) was discussed in [16], and we reproduce it here for the convenience of the reader:

Augmented Lagrangian Method

1. Select an initial penalty parameter $c_0 > 0$, a Lagrange multiplier estimate λ^0, and an initial decision vector x^0 satisfying the LMIs, $\mathcal{A}(x^0) \leq 0$.

2. For given c_k, λ^k and x^k, solve

$$\begin{array}{ll} \text{minimize} & L_{c_k}(x; 0, \lambda^k) \\ \text{subject to} & \mathcal{A}(x) \leq 0 \end{array} \tag{26}$$

   and let x^{k+1} the solution to (26).

3. Update the Lagrange multiplier using the first order update formula

$$\lambda^{k+1} = \lambda^k + c_k \mathcal{B}(x^{k+1}) \tag{27}$$

4. Update the penalty parameter such that $c_{k+1} \geq c_k$, increase k, and go back to step 2.

This scheme is often called *first-order method of multipliers.* It takes the constraint set $\{x : \mathcal{A}(x) \leq 0\}$ as an unstructured set and does not attempt to exploit its special LMI structure, which would require attaching a matrix Lagrange multiplier variable $\Lambda \geq 0$ to the LMI. As a consequence, its rate of convergence is only linear if the penalty parameter $c_k = c$ is held fixed, while super-linear convergence is guaranteed if $c_k \to \infty$. The latter is of minor practical importance due to the inevitable ill-conditioning for large c.

Remark. Aiming at good theoretical local convergence properties, we should certainly avoid the augmented Lagrangian method. To ensure super-linear convergence with fixed large enough c, we have to use second-order methods like the proposed SSDP. Nevertheless, the augmented Lagrangian method has some merits as it is robust in practice and, similar to the case of SSDP, may be tackled by a series of SDP subproblems if the Newton step called for to solve (26) is suitably convexified. In contrast with the tangent subproblem (T) in SSDP, these SDP subproblems may be solved by primal methods, as Lagrange multipliers Λ are not required. This may be an advantage of the augmented Lagrangian approach, since for instance in our experiments a well-implemented primal SDP solver like [20] often outperformed existing primal-dual software, even though the latter is preferred by theory. □

Let us finally recall an approach to (D) discussed in [4]. Primarily, this scheme is suited for the feasibility problem (find x such that $\mathcal{B}(x) = 0$, $\mathcal{A}(x) \leq 0$), but may be modified to apply to (D).

Consider (D) with a nonlinear equality constraint of the form $\mathcal{B}(x) = P\tilde{P} - I = 0$ as encountered in our applications. Introducing a slack matrix variable Z, problem (D) may be replaced by the concave program (cf. [4] for a proof):

$$
(C) \qquad
\begin{array}{ll}
\text{minimize} & f_c(x) = \gamma + c\,\mathrm{trace}(Z_1 - Z_3^T Z_2^{-1} Z_3) \\
\text{subject to} & \mathcal{A}(x) \leq 0, \\
& \mathcal{L}(x) = \begin{pmatrix} Z_1 & Z_3^T & P & I \\ Z_3 & Z_2 & I & \tilde{P} \\ * & * & I & 0 \\ * & * & 0 & I \end{pmatrix} \geq 0
\end{array}
$$

We may solve (C) by a sequence of subproblems each of which minimizes the first order Taylor polynomial of $f_c(x)$ about the inner current iterate x and over the convex set $\{\mathcal{A} \leq 0, \mathcal{L} \geq 0\}$. This procedure is known as the conditional gradient or Frank and Wolfe method. In order to improve its performance, second order information is at least partially included by approximating the concave second order term of the objective $f_c(x)$ by a linear underestimate (see [31]). This modification improves convergence but still has the inconvenience of a high CPU cost. Altogether, concave methods cannot compete with the SSDP or augmented Lagrangian techniques, as they are very slow and, due to the slack variable Z, lead to large size problems. We use the concave programming approach in order to check on the quality of our local optimal solutions. In a reasonable number of tests, SSDP did in fact terminate with values of γ close to the global optimum. Yet another way to testing the quality of the gain γ is to establish a lower bound γ_ℓ for the optimal γ_{opt} by solving (D) without the nonlinear constraint $\mathcal{B} = 0$.

7 Numerical Experiments

In this section two typical test examples are used to compare SSDP to the augmented Lagrangian method proposed in [16] for a related situation, and a concave programming approach.

7.1 Robust Control of a Flexible Actuator

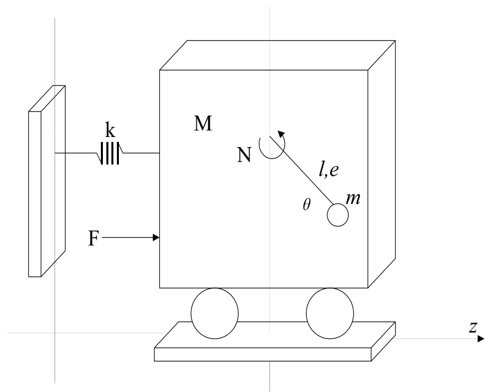

Figure 3: Flexible Actuator

Consider the unbalanced oscillator described in Figure 3. The plant is built with a cart of weight M, fixed to a vertical plane by a linear spring k and constrained to move only along the z axis. An embedded pendulum with mass m and moment of inertia I is attached to the center of mass of the cart and can be rotated in the vertical plane. The cart is submitted to an external disturbance F, and a control torque N is applied to the pendulum to stabilize its movement. The nonlinear equations of motion are:

$$(M+m)\ddot{Z} + me\ddot{\vartheta}\cos\vartheta = me\dot{\vartheta}^2\sin\vartheta - kZ + F \qquad me\ddot{Z}\cos\vartheta + (I+me^2)\ddot{\vartheta} = N$$

where ϑ and $\dot{\vartheta}$ denote the angular position and velocity of the pendulum, and $Z, \dot{Z}$ denote the position and velocity of the cart. We normalize these equations as in [14]:

$$\ddot{\zeta} + \varepsilon\ddot{\vartheta}\cos\vartheta = \varepsilon\dot{\vartheta}^2\sin\vartheta - \zeta + w, \qquad \varepsilon\ddot{\zeta}\cos\vartheta + \ddot{\vartheta} = u$$

where $[\zeta\ \dot{\zeta}\ \vartheta\ \dot{\vartheta}]^T$ is the new state vector. We assume $\theta_m = \cos\vartheta$ measured, and we express the remaining nonlinear term in the left hand equation through the uncertain parameter $\theta_u = \dot{\vartheta}\sin\vartheta$. The parameter block becomes $\Theta = \text{diag}(\theta_m, \theta_u I_3)$. The LFT model of the plant is then derived and numerical data are given below in order to allow testing of our results with different approaches. Table 1 displays the behavior of the SSDP-Algorithm. We can see that SSDP achieves good values of γ already after a few iterations. The nonlinear constraints decrease with an approximately linear rate. In practice, one may stop the algorithm whenever γ is no longer reduced over a certain number of iteration and the nonlinear constraint is sufficiently small, say smaller that 10^{-6} or 10^{-7}. The final steps in the table are only for illustration of the asymptotic behavior of the method. Note that the number of decision variables in this example was 94. The gain $\gamma_{\text{opt}} = 1.262$ obtained by SSDP was close to the lower estimate $\gamma_\ell = 1.18$ obtained by solving (D) without the constraint $\mathcal{B} = 0$.

The numerical data for the flexible actuator LFT plant are:

$$
P(s) \cong \left(\begin{array}{cccc|cccc|c|c}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 5\varepsilon & \varepsilon & -\varepsilon & -\varepsilon & 1 & -0.2 \\
0 & 0 & 0 & 1.02 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.2 & 0 & 0 & 0 & -\varepsilon & \varepsilon & \varepsilon & 0 & -0.2 & 1 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.84 & 0 & 0 & 0 & -4\varepsilon & -\varepsilon & 0 & \varepsilon & -0.84 & 0.16 \\
1.23 & 0 & 0 & 0 & -6\varepsilon & 0 & 2\varepsilon & 2\varepsilon & -1.23 & 0.23 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
\hline
0.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right)
$$

where ε, a coupling parameter, is chosen in this problem to equal 0.01. The vector of regulated variables z consists of three components: z_ζ, z_ϑ are the damping specifications on ζ, ϑ , and z_u serves to limit the control activity. The exogenous input w is the external force F.

	SSDP method			*Augmented Lagrangian*		
step	γ	$\|P\tilde{P} - I\|_F^2$	c	γ	$\|P\tilde{P} - I\|_F^2$	c
0	7	1.152 e+002	0.5	7	1.152 e+002	0.5
1	4.429	1.935 e-000		3.771	1.085 e+001	
2	2.976	1.529 e+001		2.870	1.156 e+001	
3	1.795	1.717 e-000		2.083	1.297 e+001	
4	1.287	6.214 e-000		1.849	1.415 e+001	
5	1.262	1.762 e-000		1.276	7.169 e-000	
6	1.259	7.276 e-001		1.245	2.615 e-000	
7	1.261	4.679 e-001		1.246	4.716 e-001	
8	1.262	1.526 e-002		1.249	1.274 e-001	2
9	——	2.647 e-004	2	1.251	4.247 e-002	
10		1.796 e-006		1.254	1.676 e-002	
11				——	7.462 e-003	8
12					1.179 e-003	
13					9.584 e-005	32
14					2.145 e-005	
15					1.217 e-006	128

Table 1: Behavior of SSDP for the Flexible Actuator
Computations on PC with CPU Pentium II
333 MHz.

7.2 Robust Autopilot of a Missile

Consider the missile-airframe control problem illustrated in Figure 4, where the missile is flying with an angle of attack α. The control problem requires that the autopilot generate

	Modified conditional gradient						
step	γ	$\|P\tilde{P} - I\|_F^2$	c	step	γ	$\|P\tilde{P} - I\|_F^2$	c
0	7	1.152 e+002	0.5	11		1.725 e-002	512
1	1.295	2.169 e+001		12	1.315	7.642 e-003	
2	1.292	2.576 e+001		13	1.319	2.764 e-003	1024
3	1.302	1.145 e+001	2	14	1.321	9.476 e-004	
4	1.307	5.872 e+000	8	15	—	6.125 e-004	2048
5	1.309	2.057 e+000		16		4.679 e-004	
6	1.311	8.451 e-001		17	1.325	2.762 e-004	
7	—	4.251 e-001	32	18	—	1.927 e-004	
8	1.312	2.745 e-001		19	1.322	2.169 e-004	
9	—	7.567 e-002	128	20	1.324	1.742 e-004	
10		4.571 e-002					

Table 2: Behavior of Modified Conditional Gradient Algorithm for Flexible Actuator

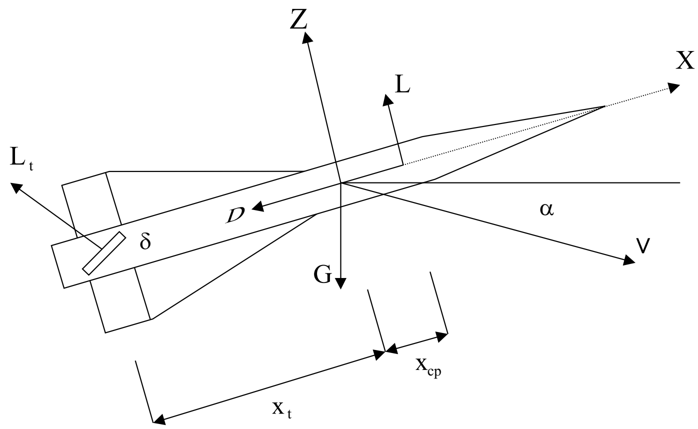

Figure 4: Aerodynamic model for air to air missile

the elevator deflection δ to maintain the angle of attack α_c called for by the guidance law. The tail-fin actuator is modeled as a first order system

$$\dot{\delta} = \tau(u - \delta)$$

with time constant $\tau = 1/150$ seconds, so δ becomes itself a state of the system. The nonlinear dynamics of the missile are adopted from [32]:

$$\dot{\alpha} = f\,\frac{\cos(\alpha/f)}{mV}\,Z + q, \qquad \dot{q} = fM_\ell/I_y$$

where m is the mass, $V = M/V_s$ is speed , I_y the pitch moment of inertia, $Z = C_Z(\alpha, \delta, M)QS$ the normal force, $M_\ell = C_m(\alpha, \delta, M)QSd$ the pitch moment, Q is dynamic pressure, and S, d reference area and diameter. The normal force and pitch moment aerodynamic coefficients are approximated by third-order polynomials in α and first oder polynomials in δ and M.

Sensor measurements y for feedback include the pitch rate q and α, while the state of the actuator deflection δ is unobserved. The robust control scheme for the missile autopilot is shown in Figure 5. The time-varying matrix-valued parameter is $\Theta = \mathrm{diag}(\theta_m I_4, \theta_u)$,

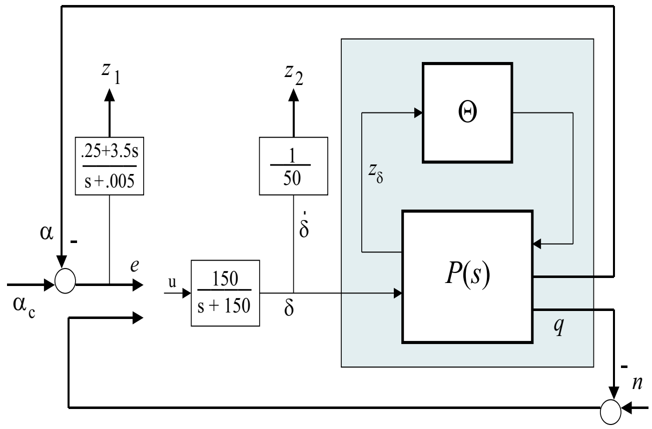

Figure 5: *Configuration for synthesis*

where θ_u is used to translate the nonlinearity in α in the left hand equation into an LFT with uncertainty. The scheduled parameter θ_m is the variation in Mach number M about nominal flight conditions, $M_0 = 3$. The Mach number is slowly time-varying and easily measured on-line.

The vector of regulated variables z consists of two components (see Figure 5). The first, z_1, corresponds to a frequency-weighted sensitivity design goal, for tracking error accuracy, while $z_2 = c_\delta \dot{\delta} = c_\delta \tau(u - \delta)$ serves to limit the tail-fin actuator rate $\dot{\delta}$, and indirectly, to bound the controller bandwidth in order to avoid trouble with unmodeled flexible modes. The vector of exogenous inputs w includes the command α_c and the pitch rate sensor noise n.

For the problem considered, it is desired to track step input commands $\alpha_c = 20^0$ with a steady state accuracy of 2%, to achieve a rise time of less than 0.3 seconds, and to limit overshoot to 5% for a wide range of angles of attack ± 20 deg and under variations in Mach number ranging from 2 to 4.

For comparison, the numerical data for this LFT model of the missile are reproduced below. Our optimization techniques are then readily applicable, and the results are shown in Table 3. SSDP achieves good values of γ after a few iterations, with a similar rate of decrease for the nonlinear constraints. The autopilot example involved 132 decision variables. Again the gain $\gamma_{\text{opt}} = 0.597$ observed at the optimum was close to the lower estimate $\gamma_\ell = 0.57$. The optimal controller obtained by SSDP was then tested in a time domain simulation based on the nonlinear model shown in Figure 6. The upper curve shows the tracking in α, the lower curve shows the corresponding elevator deflection δ.

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

	SSDP method			*Augmented Lagrangian*		
step	γ	$\|\|P\tilde{P}-I\|\|_F^2$	c	γ	$\|\|P\tilde{P}-I\|\|_F^2$	c
0	7	6.254 e+003	0.25	7	6.254 e+003	0.25
1	1.124	0.476 e-001		1.552	2.147 e+002	
2	0.854	1.876 e-000		1.467	6.345 e-001	
3	0.762	1.287 e-000		0.745	1.827 e-000	
4	0.631	2.655 e-001		0.642	1.845 e-000	
5	0.609	1.425 e-002		0.598	1.475 e-000	
6	0.597	1.721 e-005	2	0.617	7.857 e-001	2
7				0.607	5.749 e-003	8
8				0.596	4.671 e-003	
9				0.597	2.612 e-005	64
10				—	1.682 e-006	

Table 3: Behavior of SSDP for the Missile autopilot
Computations on PC with CPU Pentium II
333 MHz

Remark. Computational experience with a larger set of typical design examples indicates that the number of iterations (in terms of SDPs) required by SSDP is almost independent of the problem dimension, whereas the CPU of course strongly depends on the efficiency of the SDP solver. As it turns out, in its actual state, the bottleneck of SSDP *is* the SDP-solver. The public domain software for SDP we tested could be reliably used to problem sizes of up to 500 − 1000 decision variables. For larger sizes, the method may fail due to failure of the SDP-solver, often already at the stage of finding feasible starting values, or while trying to solve one of the LMI-subproblems. Solvers exploiting at best the structure of the problem under consideration may then be required. □

Remark. A special type of LMI-solver which replaces the SDP by an eigenvalue optimization and uses the bundle method from nonsmooth optimization was presented by Lemaréchal et al. [27], and reported to work well for certain large size LMI problems. On the other hand, for large size problems where most SDP solvers are at ill, the direct approach via interior-point methods may turn out preferable. □

8 Concluding Remarks

In this paper we have developed *sequential semi-definite programming* (SSDP), a technique for finding local solutions to robust control design problems. SSDP is an extension of (and inspired by) sequential quadratic programming (SQP), a method in nonlinear optimization known since the late 1970s. Expanding on SQP, SSDP comprises LMI-constraints, which are handled explicitly in the course of the algorithm. The method is comfortably implemented with available SDP codes if the Hessian or reduced Hessian are suitably convexified. We found the approach highly reliable (as we demonstrated on a set of test examples), exhibiting local super-linear convergence properties, and applicable to a rich list of problems in robust control theory.

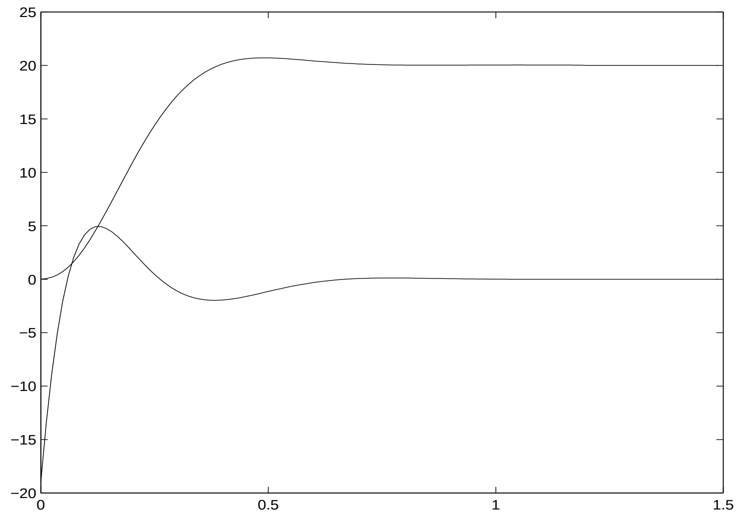

Figure 6: Nonlinear simulation for missile autopilot example

References

[1] P. Apkarian and P. Gahinet, *A Convex Characterization of Gain-Scheduled H_∞ Controllers*, IEEE Trans. Aut. Control, 40 (1995), pp. 853–864. See also pp. 1681.

[2] P. Apkarian, P. Gahinet, and G. Becker, *Self-Scheduled H_∞ Control of Linear Parameter-Varying Systems: A Design Example*, Automatica, 31 (1995), pp. 1251–1261.

[3] P. Apkarian and D. Noll, *A prototype primal-dual LMI-interior point algorithm for nonconvex robust control problems*, (2000). Submitted.

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

[5] ——, *Concave Programming in Control Theory*, Journal of Global Optimization, 15 (1999), pp. 343–370.

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

[7] G. Becker, *Parameter-Dependent Control of an Under-Actuated Mechanical System*, in Proc. IEEE Conf. on Decision and Control, LA, 1995.

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

[9] ——, *Nonlinear Programming*, Athena Scientific, USA, Belmont, Mass., 1995.

[10] P. T. Boggs and J. W. Tolle, *Sequential Quadratic Programming*, Acta Numerica, (1995), pp. 1–52.

[11] J. Bonnans, *Local analysis of Newton-type methods for variational inequalitites and nonlinear programming*, Appl. Math. Optim., 29 (1994), pp. 161–186.

[12] J. Bonnans, J.-C. Gilbert, C. Lemaréchal, and C. Sagastizábal, *Optimisation Numérique*, vol. 27, Series: Mathématiques et Applications, Springer Verlag, 1997.

[13] J. David, *Algorithms for Analysis and Design of Robust Controllers*, ph.d., Dept. of Elect. Eng., K. U. Leuven, Belgium, 1994.

[14] S. Dussy and L. ELGhaoui, *Measurement-scheduled Control for RATC problem*, Int. J. Robust and Nonlinear Control, (1997).

[15] F.Alizadeh, J. Haeberly, and M. Overton, *Primal-dual interior-point methods for semi-definite programming: convergence rates, stability, and numerical results*, In revision for J. of Optimization, (1997).

[16] B. Fares, P. Apkarian, and D. Noll, *An Augmented Lagrangian Method for a Class of LMI-Constrained Problems in Robust Control Theory*, (1999). To appear: International Journal of Control.

[17] E. Feron, P. Apkarian, and P. Gahinet, *Analysis and Synthesis of Robust Control Systems via Parameter-Dependent Lyapunov Functions*, IEEE Trans. Aut. Control, 41 (1996), pp. 1041–1046.

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

[19] P. Gahinet, P. Apkarian, and M. Chilali, *Parameter-Dependent Lyapunov Functions for Real Parametric Uncertainty*, IEEE Trans. Aut. Control, 41 (1996), pp. 436–442.

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

[21] A. Helmersson, *Methods for Robust Gain-Scheduling*, Ph. D. Thesis, Linkoping University, Sweden, 1995.

[22] J. W. Helton and O. Merino, *Coordinate optimization for bi-convex matrix inequalitites*, Proc. 36th IEEE CDC, San Diego, CA, 4 (1997), pp. 3609 – 13.

[23] N. J. Higham and S. H. Chenge, *Modifying the inertia of matrices arising in optimization*, Linear Algebra and Appl., (1998).

[24] F. Jarre, *A QQP-Minimization Method for Semidefinite and Smooth Nonconvex Programs*, Technical Report, University of Notre Dame, (1999).

[25] F. Leibfritz and E. S. Mostafa, *An interior point constrained trust region method for a special class of nonlinear semi-definite programming problems*, SIAM J. on Control and Optimization, (2000). Submitted.

[26] ——, *Trust region methods for solving the optimal output feedback design problem*, Forschungsbericht Nr. 00-01, (2000). Universität Trier.

[27] C. Lemaréchal and F. Oustry, *Nonsmooth algorithms to solve semidefinite programs, SIAM Advances in Linear Matrix Inequality Methods in Control series, ed. L. El Ghaoui & S.-I. Niculescu*, (2000).

[28] J. Ly, M. G. Safonov, and R. Y. Chiang, *Real/complex multivariable stability margin computation via generalized popov multiplier - lmi approach*, in Proc. American Control Conf., Baltimore, Ma, June 1994, pp. 425–429.

[29] A. Packard, *Gain scheduling via linear fractional transformations*, Syst. Control Letters, 22 (1994), pp. 79–92.

[30] ——, *Gain Scheduling via Linear Fractional Transformations*, Syst. Control Letters, 22 (1994), pp. 79–92.

[31] P. Pardalos and J. Rosen, *Constrained Global Optimization : Algorithms and Applications*, Lecture Notes in Computer Science, Springer-Verlag, Berlin Heidelberg, 1987.

[32] R. T. Reichert, *Robust Autopilot Design Using μ-Synthesis*, in Proc. American Control Conf., may 1990, pp. 2368–2373.

[33] S. Robinson, *Generalized equations and their solution: part II: Applications to nonlinear programming*, Mathematical Programming Studies, 19 (1982), pp. 200–221.

[34] M. G. Safonov, K. C. Goh, and J. H. Ly, *Control System Synthesis via Bilinear Matrix Inequalities*, in Proc. American Control Conf., 1994, pp. 45–49.

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

[36] ——, *Robust mixed control and linear parameter-varying control with full block scaling, SIAM Advances in Linear Matrix Inequality Methods in Control series, ed. L. El Ghaoui & S.-I. Niculescu*, (2000).

[37] S. Wu and S. Boyd, *Software for Semidefinite Programming and Determinant Maximization Problems with Matrix Structure, User's Guide*, Stanford University, may 1996.

[38] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice Hall, 1996.