SIAM J. OPTIM.
Vol. 15, No. 1, pp. 161–184

PARTIALLY AUGMENTED LAGRANGIAN METHOD FOR MATRIX INEQUALITY CONSTRAINTS*

DOMINIKUS NOLL†, MOUNIR TORKI‡, AND PIERRE APKARIAN§

Abstract. We discuss a partially augmented Lagrangian method for optimization programs with matrix inequality constraints. A global convergence result is obtained. Applications to hard problems in feedback control are presented to validate the method numerically.

Key words. augmented Lagrangian, linear matrix inequalities, bilinear matrix inequalities, semidefinite programming

AMS subject classifications. 49N35, 90C22, 93B51

DOI. 10.1137/S1052623402413963

1. Introduction.

The augmented Lagrangian method was proposed independently by Hestenes [34] and Powell [47] in 1969 and, since its inauguration, continues to be an important option in numerical optimization. With the introduction of successive quadratic programming (SQP) in the 1970s and the rise of interior point methods in the 1980s, the interest in the augmented Lagrangian somewhat declined but never completely ceased. For instance, in the 1980s, some authors proposed to combine SQP with augmented Lagrangian merit functions, and today the idea of the augmented Lagrangian is revived in the context of interior point methods, where it is one possible way to deal with nonlinear equality constraints. A history of the augmented Lagrangian from its beginning to the early 1990s is presented in [20].

Here we are concerned with a *partially augmented Lagrangian* method, a natural variation of the original theme. *Partial* refers to when some of the constraints are not included in the augmentation process but kept explicitly in order to exploit their structure. Surprisingly enough, this natural idea appears to have been overlooked before 1990. In a series of papers [20, 21, 22, 23] starting in the early 1990s, Conn et al. finally examined this approach, and a rather comprehensive convergence analysis for traditional nonlinear programming problems has been obtained in [23, 49].

In the present work we discuss optimization programs featuring matrix inequality constraints in addition to the traditional equality and inequality constraints. Such programs arise quite naturally in feedback control and have a large number of interesting applications. We propose a partially augmented Lagrangian strategy as one possible way to deal with these programs.

Semidefinite programming (SDP) is the most prominent example of a matrix inequality constrained program. With its link to integer programming [32] and because of a large number of applications in control [12], SDP has become one of the most

*Received by the editors September 3, 2002; accepted for publication (in revised form) March 8, 2004; published electronically October 14, 2004.
http://www.siam.org/journals/siopt/15-1/41396.html
†Université Paul Sabatier, Mathématiques pour l'Industrie et la Physique, CNRS UMR 5640, 118, route de Narbonne, 31062 Toulouse, France (noll@mip.ups-tlse.fr).
‡Université d'Avignon, Laboratoire d'Analyse non linéaire et Géométrie, Institut Universitaire Professionnalisé, 339 chm. des Meinajariés, Agroparc BP 1228, 84911 Avignon, France (mounir.torki@iup.univ-avignon.fr).
§ONERA-CERT, Centre d'études et de recherche de Toulouse, Control System Department, 2 av. Edouard Belin, 31055 Toulouse, France, and Université Paul Sabatier, Mathématiques pour l'Industrie et la Physique, CNRS UMR 5640, 118, route de Narbonne, 31062 Toulouse, France (apkarian@cert.fr).

active research topics in nonlinear optimization. During the 1990s, problems like H_2- or H_∞-synthesis, linear parameter varying (LPV) synthesis, robustness analysis, and analysis under integral quadratic constraints (IQCs), among others, have been identified as linear matrix inequality (LMI) feasibility or optimization problems, solvable therefore by SDP [2, 35, 50, 40, 30, 12].

It needs to be stressed, however, that the most important problems in feedback control *cannot* be solved by SDP. Challenging problems like parametric robust H_2- or H_∞-output feedback synthesis, reduced or fixed-order output feedback design, static output feedback control, multimodel design or synthesis under IQC-constraints, synthesis with parameter-dependent Lyapunov functions, robust controller design with generalized Popov multipliers, and stabilization of delayed systems are all known to be NP-hard problems, which are beyond convexity methods, and the list could be extended.

Most of these hard problems in control have been deemed largely inaccessible only a couple of years ago [6, 18, 48, 42]. In response to this challenge, we have proposed three different strategies beyond SDP which address these hard problems [28, 29, 4, 5], and one of the most promising approaches is the partially augmented Lagrangian discussed here. In this work we will mainly consider convergence issues, but several numerical test examples in reduced order H_∞-synthesis and in robust H_∞-control synthesis are included in order to validate the approach numerically. We mention related work on reduced order synthesis by Leibfritz and Mostafa [37, 38], and a very different algorithmic approach by Burke, Lewis, and Overton [15, 16] based on nonsmooth analysis techniques. The appealing aspect of their strategy is that it seems better adapted to large-size problems.

A general feature of the mentioned hard problems in feedback control is the fact that they may all be cast as minimizing a convex or even linear objective function subject to bilinear matrix inequality (BMI) constraints:

$$(B)\qquad \begin{array}{ll} \text{minimize} & c^T x,\ x \in \mathbb{R}^n, \\ \text{subject to} & A_0 + \displaystyle\sum_{i=1}^n x_i A_i + \sum_{1\le i<j\le n} x_i x_j B_{ij} \preceq 0, \end{array}$$

where $\mathcal{B}(x) := A_0 + \sum_{i=1}^n x_i A_i + \sum_{1\le i<j\le n} x_i x_j B_{ij}$ is a bilinear matrix function with values in a space $\mathbb{S}_p$ of symmetric $p \times p$ matrices, and where $\preceq 0$ means negative semidefinite. Such a program may be transformed to minimizing a linear objective subject to LMI-constraints in tandem with nonlinear equality constraints:

$$(S)\qquad \begin{array}{ll} \text{minimize} & c^T x,\ x \in \mathbb{R}^n, \\ \text{subject to} & g_j(x) = 0,\ j = 1, \dots, m, \\ & A_0 + \displaystyle\sum_{i=1}^n x_i A_i \preceq 0, \end{array}$$

where $\mathcal{A}(x) := A_0 + \sum_{i=1}^n x_i A_i$ is now an affine matrix function, and where the nonconvexity in (B) has been shifted to the equality constraints $g(x) = 0$. Notice, however, that the way in which the cast (S) is obtained from (B) is usually critical for a successful numerical solution.

Once a suitable form (S) has been found, the following partially augmented Lagrangian strategy seems near at hand. Augmenting the nonlinear equality constraints, $g(x) = 0$, and keeping the LMI-constraints, $\mathcal{A}(x) \preceq 0$, we expect to solve the difficult problem (S) through a succession of easier SDPs. This is convenient from a practical point of view, as existing software for SDP may be exploited. We have successfully applied this strategy in [28, 5, 4, 29] to problems in robust control and static output feedback control. Here we shall corroborate our experience by presenting two applications in robust control classified as difficult.

The paper is structured as follows. Section 2 presents the partially augmented Lagrangian method, sections 3–5 provide convergence results, followed by a discussion in section 6. Applications are presented in section 7.

Our presentation is inspired by the work of Conn et al. [20, 21, 22, 23], even though their techniques strongly rely on the polyhedral nature of the constraint set C. It will become clear at which points new ideas are required to account for the more complicated boundary structure of LMI or BMI constrained sets.

2. Problem setting.

Our convergence analysis applies to more general situations than program (S). We consider the program

$$(P)\quad \begin{array}{ll} \text{minimize} & f(x), x \in \mathbb{R}^n, \\ \text{subject to} & g_j(x) = 0,\ j = 1, \dots, m, \\ & x \in C, \end{array}$$

where C is a closed convex set, and the data f, g_j are of class $\mathcal{C}^2$. For later use, let $g(x) = [g_1(x), \dots, g_m(x)]^T$, and let $J(x)$ be the $m \times n$ Jacobian of $g(x)$, that is,

$$J(x)^T \;=\; [\nabla g_1(x), \dots, \nabla g_m(x)].$$

For a given penalty parameter $\mu > 0$ and a Lagrange multiplier estimate $\lambda \in \mathbb{R}^m$, we define the *partially augmented Lagrangian* of program (P) as

$$\Phi(x; \lambda, \mu) \;=\; f(x) + \sum_{j=1}^{m} \lambda_j g_j(x) + \frac{1}{2\mu} \sum_{j=1}^{m} g_j(x)^2. \tag{1}$$

Following the classical idea of the augmented Lagrangian method, we replace program (P) by the following approximation:

$$(P_{\lambda,\mu})\quad \begin{array}{ll} \text{minimize} & \Phi(x; \lambda, \mu), x \in \mathbb{R}^n, \\ \text{subject to} & x \in C. \end{array}$$

The rationale of this choice is that $(P_{\lambda,\mu})$ is easier to solve than (P) and that, with appropriate λ and μ, a solution to $(P_{\lambda,\mu})$ will be close to a solution of the original program (P). The first condition is in particular met for polyhedral sets $C = \{x \in \mathbb{R}^n : A^{(1)}x \le b^{(1)},\ A^{(2)}x = b^{(2)}\}$. The application we have in mind is when C is an LMI-constrained set,

$$C = \{x \in \mathbb{R}^n : \mathcal{A}(x) \preceq 0\}, \qquad \text{where } \mathcal{A}(x) = A_0 + \sum_{i=1}^{n} x_i A_i \tag{2}$$

with $A_i \in \mathbb{S}_p$, the space of $p \times p$ symmetric matrices, and where $\preceq 0$ means negative semidefinite.

For later use, let us fix some notation. Along with the augmented Lagrangian (1), we also consider the traditional Lagrangian of program (P), defined as

$$L(x; \lambda) = f(x) + \lambda^T g(x). \tag{3}$$

The first order Lagrange multiplier update rule is now defined as

$$\lambda^\sharp(x;\lambda,\mu) := \lambda + \frac{g(x)}{\mu} \tag{4}$$

and the useful relation

$$\nabla\Phi(x;\lambda,\mu) = \nabla L(x;\lambda^\sharp(x;\lambda,\mu)) \tag{5}$$

is satisfied. Here and later on, whenever a gradient symbol as in $\nabla\Phi(x;\lambda,\mu)$ or $\nabla L(x;\lambda)$ occurs, it is applied to the variable x. For simplicity we will exclusively use the Euclidean norms on the spaces $\mathbb{R}^n$ and $\mathbb{S}_p$. The scalar product on $\mathbb{S}_p$ is $X \bullet Y = \mathrm{tr}(XY)$.

We need some more notation from nonsmooth analysis. For $x \in C$ let $N(C,x)$ be the normal cone, and let $T(C,x)$ be the tangent cone of C at x. When there is no ambiguity as to the meaning of the set C, the orthogonal projection $P_{T(C,x)}$ onto the tangent cone $T(C,x)$ will be written as $P(x)$. Let $V(C,x)$ denote the largest linear subspace or *lineality space* of $T(C,x)$, that is, $V(C,x) = T(C,x) \cap -T(C,x)$ (see [19]). Again, if the meaning of the set C is clear, the orthogonal projection onto $V(C,x)$ will be denoted by $\Pi(x)$.

During our analysis we assume that the nonsmoothness in program (P) has been shifted to the set C, whereas the other program data are smooth. Throughout we make the following hypothesis.

$(\mathcal{H}_1)$ The functions f, g_j are of class $\mathcal{C}^1$.

The method we investigate is now the following, built along the models by Conn et al. [20, 21, 22, 23].

PARTIALLY AUGMENTED LAGRANGIAN

1. (Preparation) Fix an initial Lagrange multiplier estimate λ and an initial penalty $\mu > 0$, $(\mu < 1)$, and let x be an initial guess of the solution. Fix the tolerance parameters $\omega, \eta > 0$ at the initial values $\omega_0, \eta_0 > 0$, and choose final tolerance values $\omega_* \ll \omega_0$, $\eta_* \ll \eta_0$. Let $0 < \tau < 1$ and $\alpha > 0$, $\beta > 0$. Let `success` be a boolean variable with the values `yes` and `no`, and initialize `success` = `no`.
2. (Stopping test) Stop the algorithm if `success` == `yes` and if ω and η are sufficiently small, that is, $\omega \le \omega_*$ and $\eta \le \eta_*$.
3. (Optimization step) Given the current λ and $\mu > 0$, $\omega > 0$, approximately solve program $(P_{\lambda,\mu})$, possibly using x as a starting value. Stop the optimization $(P_{\lambda,\mu})$ as soon as an iterate x^+ close to the true solution of $(P_{\lambda,\mu})$ has been found: Stop if the solution d^+ of

$$\inf\left\{\| -\nabla\Phi(x^+;\lambda,\mu) - d\| : d \in T(C,x^+)\right\} \tag{6}$$

   satisfies $\|d^+\| \le \omega$.
4. (Decision step) If $\|g(x^+)\| \le \eta$, put `success` = `yes` and do a multiplier update step:

$$\begin{aligned}\mu^+ &= \mu,\\ \lambda^+ &= \lambda + g(x^+)/\mu^+,\\ \omega^+ &= \omega\mu^\beta,\\ \eta^+ &= \eta\mu^\beta,\end{aligned}$$

else put `success` = `no` and do a constraint reduction step:

$$\begin{aligned}\lambda^+ &= \lambda,\\ \mu^+ &= \tau\mu,\\ \omega^+ &= \omega_0(\mu^+)^\alpha,\\ \eta^+ &= \eta_0(\mu^+)^\alpha.\end{aligned}$$

5. Go back to step 2.

The mechanism is the following. Having solved the approximate program $(P_{\lambda,\mu})$ within the allowed tolerance ω, we check whether the approximate solution x^+ in step 3 satisfies the constraints $g(x^+) = 0$ within the currently acceptable tolerance level η. If this is the case, we consider this step as successful and proceed to a new instance of $(P_{\lambda,\mu})$ with λ updated according to the first order multiplier update rule (4). On the other hand, if $g(x) = 0$ is significantly violated, the solution of $(P_{\lambda,\mu})$ is considered unsuccessful. Here we reduce μ and perform $(P_{\lambda,\mu})$ again with λ unchanged. The choice of the term *successful* versus *unsuccessful* is understood from the perspective that we want to update λ according to the first order rule in order to drive it toward an optimal Lagrange multiplier λ_*. Our convergence theorems, Theorems 4.4 and 5.1, will clarify in which sense the first order updates $\lambda^\sharp$ may be expected to converge.

3. Multiplier estimates. Let us suppose that $x_* \in C$ is a Karush–Kuhn–Tucker (KKT) point of program (P) in the sense that $g(x_*) = 0$ and there exist a Lagrange multiplier $\lambda_* \in \mathbb{R}^m$ and an exterior normal vector $y_* \in N(C, x_*)$ such that

$$\nabla f(x_*) + J(x_*)^T \lambda_* + y_* = 0. \tag{7}$$

Let us further assume that the linear subspace $V(C, x_*)$ has dimension $r \geq 1$, and let $\Pi_* : \mathbb{R}^n \to \mathbb{R}^n$ be the orthogonal projection onto that subspace. Then Π_* may be decomposed as $\Pi_* = Z_* Z_*^T$, where the columns of the $n \times r$ matrix Z_* form an orthonormal basis of $V(C, x_*)$. Notice that $Z_*^T Z_* = I_r$. Since $y_* \in N(C, x_*)$, we have $\Pi_* y_* = 0$, and by the orthogonality of Z_* this gives $Z_*^T y_* = 0$. Hence from (7) we derive

$$Z_*^T \nabla f(x_*) + Z_*^T J(x_*)^T \lambda_* = 0,$$

which gives rise to the relation

$$\lambda_* = -\left(J(x_*) Z_* Z_*^T J(x_*)^T\right)^{-1} J(x_*) Z_* Z_*^T \nabla f(x_*), \tag{8}$$

valid as soon as $J(x_*)Z_*$ has column rank $\geq m$. This suggests that for vectors x in a neighborhood of x_*, where $J(x)Z_*$ also has column rank $\geq m$, a natural Lagrange multiplier estimate would be

$$\lambda_*(x) := -\left(J(x) Z_* Z_*^T J(x)^T\right)^{-1} J(x) Z_* Z_*^T \nabla f(x). \tag{9}$$

This estimate is indeed used by Conn et al. as the main analytical tool to analyze convergence of the partially augmented Lagrangian method for polyhedral sets C. In the case of LMI-constrained sets C, it encounters problems related to the more complicated boundary structure, and a better suited construction will be elaborated on below. First, let us observe that the following hypothesis was needed to introduce $\lambda_*(x)$.

$(\mathcal{H}_2)$ $J(x_*)Z_*$ has column rank $\geq m$.

During the following, assume that $(\mathcal{H}_1)$, $(\mathcal{H}_2)$ are satisfied.

Remark. Notice that $(\mathcal{H}_2)$ is a constraint qualification hypothesis. To see this consider the case where C is described by a finite set of inequality constraints, $h_1(x) \leq 0, \dots, h_s(x) \leq 0$, each of which is active at x. Moreover, assume that the active gradients at x, $\nabla h_1(x), \dots, \nabla h_s(x)$, are linearly independent.

LEMMA 3.1. *Under these circumstances, the validity of $(\mathcal{H}_2)$ at x is equivalent to linear independence of the set $\nabla h_1(x), \dots, \nabla h_s(x), \nabla g_1(x), \dots, \nabla g_m(x)$ of equality and active inequality constraint gradients at x.*

Proof. (1) To see that $(\mathcal{H}_2)$ implies linear independence of the active gradients, suppose these vectors were linearly dependent. Then $p := \sum_{i=1}^m \mu_i \nabla g_i(x) = \sum_{j=1}^s \nu_j \nabla h_j(x)$, where $\mu \neq 0$. Hence $v^T p = 0$ for every $v \in V(C,x)$, these v being orthogonal to all $\nabla h_j(x)$. However, then the image of $V(C,x)$ under the operator $J(x)$ could no longer have dimension m, as required by $(\mathcal{H}_2)$, because $J(x)p = 0$ with $p \neq 0$.

(2) Conversely, suppose $\nabla h_1(x), \dots, \nabla h_s(x), \nabla g_1(x), \dots, \nabla g_m(x)$ are linearly independent. Suppose $J(x)Z(x)$ is not of rank m; then it is not surjective, so $Z(x)^T J(x)^T$ is not injective. Therefore there exists $\lambda \neq 0$ such that $Z(x)^T J(x)^T \lambda = 0$, but linear independence of $\nabla g_1(x), \dots, \nabla g_m(x)$ means $J(x)^T$ is injective, so $\mu = J(x)^T \lambda \neq 0$ with $Z(x)^T \mu = 0$. This means $\mu \perp V(C,x)$, so μ is a linear combination of the $\nabla h_i(x)$, a consequence of the special boundary structure of C at x. However, $\mu - J(x)^T \lambda = 0$, so $\nabla h_i(x), \nabla g_j(x)$ are linearly dependent, which is a contradiction. ∎

Let us now resume our line of investigation and see in which way trouble with (9) could be avoided for a reasonable rich class of sets C. Suppose that for every x in a neighborhood $U(x_*)$ of x_* there exists a linear subspace $L(C,x)$ of $V(C,x)$ which depends smoothly on x and coincides with $V(C,x_*)$ at x_*. This means that dim $L(C,x) = r$, and that the orthogonal projector $\tilde{\Pi}(x)$ onto $L(C,x)$ varies smoothly with x. We may represent $\tilde{\Pi}(x) = \tilde{Z}(x)\tilde{Z}(x)^T$, with an orthonormal $n \times r$ matrix $\tilde{Z}(x)$ varying also smoothly with x. Then we define

$$\tilde{\lambda}(x) := -\big(J(x)\tilde{Z}(x)\tilde{Z}(x)^T J(x)^T\big)^{-1} J(x)\tilde{Z}(x)\tilde{Z}(x)^T \nabla f(x), \tag{10}$$

which is now Lipschitz in a neighborhood of x_*. Moreover, $\tilde{\lambda}(x_*) = \lambda_*$. We observe that as a consequence of $(\mathcal{H}_2)$, the matrix $J(x)\tilde{Z}(x)\tilde{Z}(x)^T J(x)^T$ is invertible in a neighborhood $U(x_*)$ of x_*.

DEFINITION 1. *A closed convex set C is said to admit a* stratification into differentiable layers *at $x \in \partial C$ if for $x' \in C$ in a neighborhood of x there exists a linear subspace $L(C,x')$ of the tangent cone $T(C,x')$ varying smoothly with x' such that at $x' = x$, $L(C,x)$ coincides with the lineality space $V(C,x)$ of the tangent cone at x.*

Example 1. Let $C = \mathbb{S}_p^-$, the negative semidefinite cone. Let A be in the boundary of $\mathbb{S}_p^-$; then $V(\mathbb{S}_p^-, A) = \{Z \in \mathbb{S}_p : Y_1^T Z Y_1 = 0\}$, where the columns of the $p \times r$ matrix Y_1 form an orthonormal basis of the eigenspace of the leading eigenvalue $\lambda_1(A) = 0$ of A, whose multiplicity is r.

For a perturbation E of A, there exists a matrix $Y_1(A+E)$ whose columns form an orthonormal basis of the invariant subspace associated with the first r eigenvalues of $A+E$. Then (cf. [53])

$$Y_1(A+E) = Y_1 + (\lambda_1(A)I_p - A)^\dagger E Y_1 + \mathrm{o}(\|E\|),$$

where $M^\dagger$ denotes the pseudoinverse of M. Then we define the subspace $L(C, A+E)$ as $L(C, A+E) = \{Z \in \mathbb{S}_p : Y_1(A+E)^T Z Y_1(A+E) = 0\}$. This means that the semidefinite order cone $\mathbb{S}_p^-$ has a differentiable stratification in the sense of Definition 1. In this example the layers or strata are the sets $\mathcal{S}_r = \{A \in \mathbb{S}_p^- : \lambda_1(A)$ has multiplicity $r\}$.

Example 2. Now let C be an LMI-constrained set given by (2). Since C is the preimage of $\mathbb{S}_p^-$ under an affine operator $\mathcal{A} : \mathbb{R}^n \to \mathbb{S}_p$, the elements $V(C,x)$ and $L(C,x)$ are just the preimages of $V(\mathbb{S}_p^-, \mathcal{A}(x))$ and $L(\mathbb{S}_p^-, \mathcal{A}(x))$ under the linear part $\mathcal{A}_*$ of $\mathcal{A}$. Therefore, LMI-sets satisfy the condition in Definition 1.

4. Convergence. Consider a sequence of iterates x_k generated by our algorithm. Let λ_k be the corresponding multiplier estimates, μ_k be the penalty parameters, and ω_k, η_k be the tolerance parameters. Suppose $\omega_k \to 0$. Suppose x_* is an accumulation point of the sequence x_k, and select a subsequence $\mathcal{K} \subset \mathbb{N}$ such that x_k, $k \in \mathcal{K}$, converges to x_*. Suppose hypotheses $(\mathcal{H}_1)$, $(\mathcal{H}_2)$ are met at x_*. Moreover, suppose $x_* \in C$ admits a stratification into differentiable layers as in Definition 1.

LEMMA 4.1. *Suppose the x_k satisfy the stopping test* (6) *in step* 3 *of the algorithm. Then*

$$\|\tilde{Z}(x_k)^T \nabla\Phi(x_k; \lambda_k, \mu_k)\| \le \omega_k. \tag{11}$$

Proof. Since $\tilde{\Pi}(x_k) = \tilde{Z}(x_k)\tilde{Z}(x_k)^T$ is the projection onto $L(C, x_k)$, and since $L(C,x) \subset T(C,x)$, we have $\|\tilde{\Pi}(x_k)\nabla\Phi(x_k;\lambda_k,\mu_k)\| \le \|P(x_k)(-\nabla\Phi(x_k;\lambda_k,\mu_k))\|$, where $P(x)$ is the orthogonal projector onto the tangent cone $T(C,x)$ at x. However, now the stopping test (6) gives $\|P(x_k)(-\nabla\Phi(x_k;\lambda_k,\mu_k))\| \le \omega_k$.

To conclude, observe that $\|\tilde{Z}(x_k)^T\nabla\Phi(x_k;\lambda_k,\mu_k)\| = \|\tilde{\Pi}(x_k)\nabla\Phi(x_k;\lambda_k,\mu_k)\|$, since $\tilde{Z}(x)$ is orthogonal. ∎

LEMMA 4.2. *Under the same assumptions,*

1. $\lambda_k^\sharp := \lambda^\sharp(x_k;\lambda_k,\mu_k)$, $k \in \mathcal{K}$, *converges to* $\lambda_* = \tilde{\lambda}(x_*)$.
2. *There exists a constant* $K > 0$ *such that*
$$\|\lambda_k^\sharp - \lambda_*\| \le K(\omega_k + \|x_k - x_*\|)$$
for every $k \in \mathcal{K}$.
3. $\nabla\Phi(x_k;\lambda_k,\mu_k) \to \nabla L(x_*;\lambda_*)$, $k \in \mathcal{K}$.
4. *There exists a constant* $K' > 0$ *such that for every* $k \in \mathcal{K}$,
$$\|g(x_k)\| \le K'\mu_k\big(\omega_k + \|\lambda_k - \lambda_*\| + \|x_k - x_*\|\big).$$

Proof. (1) Starting out with

$$\|\lambda_k^\sharp - \lambda_*\| \le \|\lambda_k^\sharp - \tilde{\lambda}(x_k)\| + \|\tilde{\lambda}(x_k) - \lambda_*\|,$$

we observe that since $\lambda_* = \tilde{\lambda}(x_*)$, the second term on the right-hand side is of the order $\mathcal{O}(\|x_k - x_*\|)$, since $\tilde{\lambda}$ is Lipschitz on $U(x_*)$. Let us say $\|\tilde{\lambda}(x_k) - \lambda_*\| \le K_0\|x_k - x_*\|$ for some $K_0 > 0$. So in order to establish items 1 and 2, it remains to estimate the first term on the right-hand side. We have

$$\begin{aligned}\|\lambda_k^\sharp - \tilde{\lambda}(x_k)\| &= \|[J(x_k)\tilde{Z}(x_k)\tilde{Z}(x_k)^T J(x_k)^T]^{-1} J(x_k)\tilde{Z}(x_k)\tilde{Z}(x_k)^T\nabla f(x_k) + \lambda_k^\sharp\| \\ &\le K_1\,\|J(x_k)\tilde{Z}(x_k)\tilde{Z}(x_k)^T\nabla f(x_k) + J(x_k)\tilde{Z}(x_k)\tilde{Z}(x_k)^T J(x_k)^T\lambda_k^\sharp\| \\ &\le K_1\,K_2\,\|\tilde{Z}(x_k)^T\nabla f(x_k) + \tilde{Z}(x_k)^T J(x_k)^T\lambda_k^\sharp\| \\ &= K_3\,\|\tilde{Z}(x_k)^T\nabla\Phi(x_k;\lambda_k,\mu_k)\| \\ &\le K_3\,\omega_k.\end{aligned}$$

Here the second line comes from $\|[J(x_k)\tilde{Z}(x_k)\tilde{Z}(x_k)^T J(x_k)^T]^{-1}\| \leq K_1$ on a neighborhood $U(x_*)$, which is guaranteed by the rank hypothesis $(\mathcal{H}_2)$. From the same reason, in line 3, $\|J(x_k)\tilde{Z}(x_k)\| \leq K_2$ on $U(x_*)$ for some $K_2 > 0$. We let $K_3 = K_1 K_2$ and use the definition of $\lambda^\sharp$, which gives line 4. Finally, the last line follows from Lemma 4.1. Altogether, we obtain the estimate in item 2 with $K = \max\{K_0, K_3\}$.

(2) Now consider item 3. Observe that by our assumptions $x_k \to x_*$, $(k \in \mathcal{K})$, and $\omega_k \to 0$, so item 2 gives $\lambda_k^\sharp \to \lambda_*$. Therefore, $\nabla\Phi(x_k; \lambda_k, \mu_k) = \nabla L(x_k; \lambda_k^\sharp)$ converges to $\nabla L(x_*; \lambda_*)$, $k \in \mathcal{K}$.

(3) Finally, to see estimate 4 we multiply (4) by μ_k and take norms, which gives

$$\|g(x_k)\| = \mu_k \|\lambda_k^\sharp - \lambda_k\| \leq \mu_k \big(K(\omega_k + \|x_k - x_*\|) + \|\lambda_k - \lambda_*\|\big).$$

This is just the desired estimate in item 4 with $K' = \max\{K, 1\}$. ∎

LEMMA 4.3. *With the same hypotheses, suppose $g(x_*) = 0$; then x_* is a KKT point, with corresponding Lagrange multiplier λ_*.*

Proof. To prove that x_* is a KKT point, we must show $P(x_*)(-\nabla L(x_*; \lambda_*)) = 0$, i.e., that $-\nabla L(x_*; \lambda_*)$ is in the normal cone to C at x_*. Since C is convex, this is equivalent to proving that for every test point $y \in C$, the angle between $-\nabla L(x_*; \lambda_*)$ and $y - x_*$ is at least 90°, i.e., that $-\nabla L(x_*; \lambda_*)^T (y - x_*) \leq 0$. Writing $\nabla\Phi_k = \nabla\Phi(x_k; \lambda_k, \mu_k)$, we first observe that by the stopping test (6),

$$\|P(x_k)(-\nabla\Phi_k)\| \leq \omega_k \to 0.$$

Let us now decompose the vector $-\nabla\Phi_k$ into its normal and tangential components at x_k, that is,

$$-\nabla\Phi_k = P(x_k)(-\nabla\Phi_k) + P^+(x_k)(-\nabla\Phi_k),$$

where $P^+(x_k)$ denotes the orthogonal projection onto $N(C, x_k)$, $P(x_k)$ as before the orthogonal projection onto $T(C, x_k)$. Such a decomposition is possible because the normal and tangent cones are polar cones of each other. Using this decomposition gives

$$\begin{aligned} -\nabla\Phi_k^T (y - x_k) &= P(x_k)(-\nabla\Phi_k)^T (y - x_k) + P^+(x_k)(-\nabla\Phi_k)^T (y - x_k) \\ &\leq P(x_k)(-\nabla\Phi_k)^T (y - x_k) \\ &\leq \omega_k \|y - x_k\| \to 0, \ (k \in \mathcal{K}), \end{aligned}$$

where the last line uses the stopping test and Cauchy–Schwarz, while the second line comes from $P^+(x_k)(-\nabla\Phi)^T (y - x_k) \leq 0$, which is a consequence of the definition of $P^+(x_k)$ and the convexity of C. Altogether the term $-\nabla\Phi_k^T (y - x_k)$ converges to a quantity ≤ 0, but by item 3 in Lemma 4.2, the same term also converges to $-\nabla L(x_*; \lambda_*)^T (y - x_*)$, $(k \in \mathcal{K})$. This proves $-\nabla L(x_*; \lambda_*)^T (y - x_*) \leq 0$. ∎

THEOREM 4.4. *Let x_* be an accumulation point of a sequence x_k generated by the partially augmented Lagrangian algorithm such that hypotheses $(\mathcal{H}_1)$, $(\mathcal{H}_2)$ are satisfied at x_*. Suppose further that C admits a stratification into differentiable layers at x_*. Let $\mathcal{K} \subset \mathbb{N}$ be the index set of a subsequence converging to x_*. Let $\lambda_* := \tilde{\lambda}(x_*)$. Then we have the following:*

1. *$\lambda_k^\sharp$, $k \in \mathcal{K}$, converges to λ_*. In particular, there exists a constant $K > 0$ such that $\|\lambda_k^\sharp - \lambda_*\| \leq K(\omega_k + \|x_k - x_*\|)$ for every $k \in \mathcal{K}$.*
2. *x_* is a KKT point, and λ_* is an associated Lagrange multiplier.*
3. *$\nabla\Phi(x_k; \lambda_k, \mu_k)$, $k \in \mathcal{K}$, converges to $\nabla L(x_*; \lambda_*)$.*

Proof. Suppose first that μ_k is bounded away from 0. Then the algorithm eventually decides to do a first order update step at each iteration. Then $\|g(x_k)\| \leq \eta_k$, eventually, and $\eta_{k+1} = \mu^\beta \eta_k$ with $\mu^\beta < 1$ implies $\eta_k \to 0$. Therefore $g(x_*) = 0$. However, now the assumptions of Lemma 4.3 are all met, so we have the correct conclusions.

Now assume μ_k is not bounded away from 0. Assume $\mu_k \to 0$ for a subsequence. Then the construction of the parameters μ_k ensures that $\mu_k\|\lambda_k - \lambda_*\| \to 0$. This is exactly the argument from [20, Lemma 4.2], whose statement we reproduce below for the reader's convenience. So we arrive at the same conclusions, because now estimate 4 in Lemma 4.2 implies $g(x_*) = 0$. □

LEMMA 4.5 (see [20, Lemma 4.2]). *Suppose μ_k, $k \in \mathcal{K}$, converges to* 0. *Then $\mu_k\|\lambda_k\|$, $k \in \mathcal{K}$, also converges to* 0.

The proof of Lemma 4.5 uses the specific form of the parameter updates in step 4 of the augmented Lagrangian algorithm. Any other update $\mu \to \mu^+$ for which the statement of Lemma 4.5 remains correct gives the same convergence result.

Remark. Notice that the weak convergence statement of Theorem 4.4 in terms of subsequences is the best we can hope to achieve in general. Reference [20] gives an example where the sequence x_k generated by the augmented Lagrangian algorithm has two accumulation points. A strict convergence result requires strong additional assumptions, like, for instance, convexity, which is not satisfied in cases we are interested in. On the other hand, in our experiments the method often converges nicely even without these hypotheses, so we consider Theorem 4.4 a satisfactory result.

5. SDP-representable sets. In this section we indicate in which way Theorem 4.4 may be extended to a larger class of convex constraint sets C. The motivating example are *SDP-representable sets*, a natural extension of LMI-sets as in (2). Recall that a closed convex set C is SDP-representable [10, 11] if it may be written in the form

$$C = \{x \in \mathbb{R}^n : \mathcal{A}(x,u) \preceq 0 \text{ for some } u \in \mathbb{R}^q\},$$

where $\mathcal{A} : \mathbb{R}^n \times \mathbb{R}^q \to \mathbb{S}_p$ is an affine operator. In other terms, SDP-representable sets are orthogonal projections of LMI-sets and may be considered the natural class of sets described by semidefinite programs. Notice that despite the similarity to LMI-sets, SDP-representable sets are a much larger class, including very interesting examples (see [10, 11]).

More generally, we may consider the class of closed convex sets C which are orthogonal projections of sets $\widetilde{C}$ admitting a stratification into differentiable layers according to Definition 1. It is not clear whether Definition 1 is invariant under projections, which means that sets C of this type do not necessarily inherit this structure, and we cannot apply Theorem 4.4 directly to this class. Nonetheless, there is an easy way in which the partially augmented Lagrangian method can be extended to this larger class of sets C.

Consider program (P) with C the orthogonal projection of a set $\widetilde{C}$, which admits a stratification into differentiable layers. Suppose without loss that C is the set of $x \in \mathbb{R}^n$ such that there exists $u \in \mathbb{R}^q$ with $(x,u) \in \widetilde{C}$. It seems natural to consider the following program $(\widetilde{P})$, which contains u as a slack variable and is equivalent to (P):

$$(\widetilde{P}) \quad \begin{array}{ll} \text{minimize} & f(x),\, x \in \mathbb{R}^n,\, u \in \mathbb{R}^q \\ \text{subject to} & g_j(x) = 0,\ j = 1, \ldots, m, \\ & (x,u) \in \widetilde{C}. \end{array}$$

This program is amenable to our convergence theorem as soon as the corresponding constraint qualification hypothesis is satisfied. At first sight, replacing (P) by $(\widetilde{P})$ does not seem attractive, because we have introduced a slack variable. On second sight, however, we see that the impact of adding u is moderate.

Suppose we apply the partially augmented Lagrangian algorithm to program (P), generating iterates $x_k \in C$, so that $(x_k, u_k) \in \widetilde{C}$ for suitable $u_k \in \mathbb{R}^q$. Can we interpret (x_k, u_k) as a sequence of iterates generated by the same algorithm, but running for program $(\widetilde{P})$ in (x, u)-space? If so, then convergence could be proved in (x, u)-space and would immediately imply convergence in x-space. This idea requires that we analyze the different steps of the algorithm in both settings.

Let us begin with the augmented version $(\widetilde{P}_{\lambda,\mu})$ of program $(P_{\lambda,\mu})$. Since the partially augmented Lagrangian $\Phi(x, \lambda, \mu)$ does not depend on u, we realize that these two programs are exactly the same. This is good news, because on solving $(P_{\lambda,\mu})$ in x-space, as we naturally plan to do, we also implicitly solve $(\widetilde{P}_{\lambda,\mu})$ in (x, u)-space.

What really needs to be done in (x, u)-space and not in x-space is the stopping test (6) in step 3 of our algorithm. What we propose to do is to modify the augmented Lagrangian scheme and accept $x^+ \in C$ as an approximate solution of $(P_{\lambda,\mu})$, and hence as the new iterate in x-space, if there exists u^+ such that $(x^+, u^+) \in \widetilde{C}$ satisfies the stopping test (6) for the lifted program $(\widetilde{P}_{\lambda,\mu})$. Explicitly this leads to the following test. Accept x^+ as soon as the solution (d_x, d_u) of

$$\inf \left\{ \| \left(-\nabla\Phi(x^+, \lambda, \mu), 0\right) - (d_x, d_u) \| : (d_x, d_u) \in T(\widetilde{C}, (x^+, u^+)) \right\} \tag{12}$$

satisfies $\|(d_x, d_u)\| \leq \omega$. For definiteness, we may require here that u^+ be the smallest element in norm satisfying $(x^+, u^+) \in \widetilde{C}$.

The last element of the algorithm to analyze concerns the parameter updates in step 4, and in particular the first order update rule. This is again identical in both settings, because the variable u does not intervene.

Altogether we have the following consequence of Theorem 4.4.

THEOREM 5.1. *Let C be a closed convex set which is the orthogonal projection of a closed convex set $\widetilde{C}$ admitting a stratification into differentiable layers. Generate sequences $x_k \in C$, ω_k, η_k, λ_k, $\lambda_k^\sharp$, μ_k according to the partially augmented Lagrangian algorithm, with the difference that the stopping test* (12) *is applied at the point $(x_k, u_k) \in \widetilde{C}$. Suppose (x_*, u_*) is an accumulation point of (x_k, u_k) such that hypotheses $(\mathcal{H}_1)$, $(\mathcal{H}_2)$ are satisfied at (x_*, u_*). Let $\mathcal{K} \subset \mathbb{N}$ the index set of a convergent subsequence. Let $\lambda_* = \tilde{\lambda}(x_*)$; then*

1. *$\lambda_k^\sharp$, $k \in \mathcal{K}$, converge to λ_*. In particular, there exists a constant $K > 0$ such that $\|\lambda_k^\sharp - \lambda_*\| \leq K(\omega_k + \|x_k - x_*\|)$ for every $k \in \mathcal{K}$.*
2. *x_* is a KKT point for (P), and λ_* is an associated Lagrange multiplier.*
3. *$\nabla\Phi(x_k; \lambda_k, \mu_k)$, $k \in \mathcal{K}$, converges to $\nabla L(x_*; \lambda_*)$.*

One may wonder whether it is really necessary to solve the stopping test in (x, u)-space all the time. Obviously, as soon as the orthogonal projection of $T(\widetilde{C}, (x^+, u^+))$ is identical with $T(C, x^+)$, solving (6) and (12) is equivalent. In general, however, this is not the case. We have only the trivial inclusion $\pi(T(\widetilde{C}, (x^+, u^+))) \subset T(C, x^+)$, where π denotes the projection $(x, u) \mapsto x$, which also shows that the stopping test (12) is stronger than (6).

A particular case where equality holds is when (x^+, u^+) is a smooth point of the boundary of $\tilde{C}$, because then x^+ is also smooth for C. Since almost all points in the boundary of a convex set are smooth points, this is quite satisfactory.

6. Discussion. In this section we briefly discuss the hypotheses in Theorems 4.4 and 5.1 and then pass to practical aspects of the algorithm. Both results use the constraint qualification hypothesis $(\mathcal{H}_2)$, which as we have seen reduces to a familiar condition in the case of classical programming. Notice that for $m \geq 1$, $(\mathcal{H}_2)$ excludes in particular corner points x of the constraint set C, which would have $V(C, x) = \{0\}$. An assumption like $(\mathcal{H}_2)$ is already required to obtain suitable KKT conditions.

The additional hypothesis of boundedness of the gradients $\nabla\Phi(x_k; \lambda_k, \mu_k)$ has been made in several approaches (see [20]). Our present approach shows that this hypothesis can be avoided.

We recall that the original idea of the augmented Lagrangian method [47] was to improve on pure penalty methods insofar as the penalty parameter μ_k no longer needed to be driven to 0 to yield convergence—a major advantage because ill-conditioning is avoided. For the partially augmented Lagrangian method with polyhedral sets, a similar result is proved in [23]. We can establish such a result for matrix inequality constraints if a second order sufficient optimality condition stronger than the no-gap condition in [13] is satisfied. Details will be presented elsewhere. The phenomenon is confirmed by experiments, where μ_k is very often frozen at a moderately small size.

Let us now consider some practical aspects of the partially augmented Lagrangian for LMI constrained sets $C = \{x \in \mathbb{R}^n : \mathcal{A}(x) \preceq 0\}$. Observe that the stopping test (6) may be computed by solving an SDP. According to [52], the tangent cone at $x_0 \in C$ is $T(C, x_0) = \{d \in \mathbb{R}^n : Y_1^T(\mathcal{A}_* d)Y_1 \preceq 0\}$, where the columns of Y_1 form an orthonormal basis of the eigenspace of $\lambda_1(\mathcal{A}(x_0))$, and where $\mathcal{A}_*$ is the linear part of $\mathcal{A}$, i.e., $\mathcal{A}_* d = \sum_{i=1}^n A_i d_i$. Letting $\nabla\Phi := \nabla\Phi(x^+; \lambda, \mu)$, the stopping test (6) leads to the LMI constrained least squares program

$$\min\{\| - \nabla\Phi - d\| : Y_1^T(\mathcal{A}_* d)Y_1 \preceq 0,\ d \in \mathbb{R}^n\}. \tag{13}$$

An equivalent cast as an SDP is

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \begin{pmatrix} I_n & \nabla\Phi + d \\ * & t \end{pmatrix} \succeq 0, \qquad Y_1^T(\mathcal{A}_* d)Y_1 \preceq 0, \end{array} \tag{14}$$

where the decision variable is now $(t, d) \in \mathbb{R} \times \mathbb{R}^n$. Notice that in general the column rank r of Y_1 is much smaller than the size of $\mathcal{A}$, so a full spectral decomposition of $\mathcal{A}(x_0)$ is not required and the program data of (13) or (14) are obtained efficiently. For large-dimension n, it may therefore be interesting to solve the dual of (13), which is readily obtained as

$$\min\left\{\frac{1}{2}\left\|\mathcal{A}_*^T\left(Y_1 Z Y_1^T\right)\right\|^2 + \mathcal{A}_*\nabla\Phi \bullet \left(Y_1 Z Y_1^T\right) : Z \succeq 0,\ Z \in \mathbb{S}_r\right\},$$

with return formula $d = -\nabla\Phi - \mathcal{A}_*^T\left(Y_1 Z Y_1^T\right)$ relating dual and primal optimal solutions.

Most of the time the multiplicity r of $\lambda_1(\mathcal{A}(x_0))$ even equals 1. Then the LMI-constraint $Y_1^T(\mathcal{A}_* d)Y_1 \preceq 0$ in (13) and (14) becomes the scalar constraint $e_1^T(\mathcal{A}_* d)e_1 \leq 0$, where e_1 is the normalized eigenvector of $\lambda_1(\mathcal{A}(x_0))$. This may also be written as $[\mathcal{A}_*^T e_1 e_1^T]^T d \leq 0$, where the adjoint $\mathcal{A}_*^T$ of the linear part of $\mathcal{A}(x)$ is defined as $\mathcal{A}_*^T Z = (A_1 \bullet Z, \ldots, A_n \bullet Z)$, and where $e_1 e_1^T$ is of rank 1. Then (13) is an inequality constrained least squares program, $\min\{\|g - d\| : d \in \mathbb{R}^n,\ h^T d \leq 0\}$, which has an explicit solution:

$$d = \begin{cases} g - \dfrac{g^T h}{\|h\|^2} h & \text{if } g^T h \geq 0, \\ g & \text{if } g^T h \leq 0, \end{cases} \qquad \text{where } g := -\nabla\Phi,\ h := \mathcal{A}_*^T e_1 e_1^T.$$

In practice $g = -\nabla\Phi$ clearly points away from the half space $h^T d \leq 0$, so that the first case occurs, which we recognize as the projection of g onto the hyperplane $h^T d = 0$.

To conclude, recall that the partially augmented Lagrangian scheme clearly hinges on the possibility of solving the approximate programs $(P_{\lambda,\mu})$ much faster than the full program (P). To this end, the structure of C should be sufficiently simple, since $(P_{\lambda,\mu})$ has to be solved many times.

7. Applications. In our experimental section we test the augmented Lagrangian method on two typical applications of program (S) in feedback control synthesis. We start with static output-feedback H_∞-synthesis in section 7.1 and present numerical tests in sections 7.2 and 7.3. A second application is parametric robust control design, which is considered in section 7.4. A case study in section 7.5 concludes the experimental part.

7.1. Static H_∞-synthesis. Static H_∞-control design is an NP-hard problem. Due to its great practical importance many heuristic approaches have been proposed; see, e.g., [8, 24, 41, 26]. Solutions based on nonlinear optimization are, for instance, [37, 38] or [16]. We have proposed several optimization-based approaches in [5, 4, 28, 29]. Here we show how this problem may be solved with the help of our augmented Lagrangian algorithm.

A detailed description of the static H_∞-problem and a comprehensive discussion are presented in [5, 6]. Here we only briefly recall the outset. Consider a linear time-invariant plant described in standard form by its state-space equations:

$$P(s): \qquad \begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix}, \tag{15}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^{m_2}$ are the control inputs, $w \in \mathbb{R}^{m_1}$ is an exogenous signal, $y \in \mathbb{R}^{p_2}$ is the vector of measurements, and $z \in \mathbb{R}^{p_1}$ is the vector of controlled or performance variables. After substitution into (15), any static output feedback control law $u = Ky$ induces a closed-loop transfer function $T_{w,z}(s)$ from w to z, called the *performance channel*. Our aim is now to compute a static controller K which meets the following design requirements:

Stability. It stabilizes the plant.

Performance. Among all stabilizing controllers, K minimizes the H_∞-norm $\|T_{w,z}(s)\|_\infty$.

The closed-loop system is first transformed into a matrix inequality using the Bounded Real Lemma [1]. Then the Projection Lemma [31] is used to eliminate the unknown controller data K from the cast. We obtain the following.

PROPOSITION 7.1. *A stabilizing static output feedback controller K with H_∞-gain $\|T_{w,z}(s)\|_\infty \leq \gamma$ exists provided there exist $X, Y \in \mathbb{S}^n$ such that*

TABLE 1
Problem dimensions.

pb.	n	m_2	p_2	m_1	p_1	var	LMI	const
pb5	5	2	2	2	2	31	25	25
pb10	10	2	3	3	3	111	48	100
pb15	15	3	3	3	3	241	67	225
pb20	20	3	4	5	5	421	94	400
pb25	25	3	4	5	5	651	114	625
pb30	30	5	6	6	7	931	136	900
pb35	35	5	6	6	7	1261	156	1225

$$\mathcal{N}_Q^T \begin{bmatrix} A^TX + XA & XB_1 & C_1^T \\ B_1^TX & -\gamma I & D_{11}^T \\ C_1 & D_{11} & -\gamma I \end{bmatrix} \mathcal{N}_Q \prec 0, \tag{16}$$

$$\mathcal{N}_P^T \begin{bmatrix} YA^T + AY & B_1 & YC_1^T \\ B_1^T & -\gamma I & D_{11}^T \\ C_1Y & D_{11} & -\gamma I \end{bmatrix} \mathcal{N}_P \prec 0, \tag{17}$$

$$X \succ 0, \qquad Y \succ 0, \qquad XY - I = 0, \tag{18}$$

where $\mathcal{N}_Q$ *and* $\mathcal{N}_P$ *denote bases of the null spaces of* $Q := [\,C_1 \quad D_{21} \quad 0\,]$ *and* $P := [\,B_1^T \quad D_{12}^T \quad 0\,]$.

It is convenient [26] to replace positive definiteness in (18) by

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} \succ 0. \tag{19}$$

On the other hand, the nonlinear equality $XY - I = 0$ cannot be removed and renders the problem difficult. The cast (16)–(18) and (19) is now of the form (S) if we replace strict inequalities $\prec 0$ by suitable $\preceq -\varepsilon$. The objective to be minimized is γ. The dimension of the decision variable $x = (X, Y, \gamma)$ is $1 + n(n+1)$, displayed as var in Table 1. The size of the LMIs is displayed in the column labeled LMI. It depends on the dimensions of $\mathcal{N}_P$ and $\mathcal{N}_Q$ and due to possible rank deficiency cannot be computed in advance. The nonlinear equality constraint in the terminology of (S) corresponds to a function $g : \mathbb{R}^{n(n+1)} \to \mathbb{R}^{n^2}$. The last column const in Table 1 therefore displays n^2.

Once solved via the augmented Lagrangian method, this procedure requires an additional step, where the controller K, which has been eliminated from the cast, needs to be restored from the decision parameters of (S). This last step may be based on the method in [31] and, as a rule, does not present any numerical difficulties.

7.2. Numerical experiment I. In our first experiment we solve a series of static output-feedback H_∞-synthesis problems randomly generated via the procedure in [43] at different sizes n ranging from 5 to 35. In each case it is known that a stabilizing static controller K exists, but the global optimal gain $\gamma = \|T_{w,z}(s)\|_\infty$ is not known. Dimensions of our test problems are described in Table 1.

While n, m_2, p_2, m_1, p_1 refer to the plant (15), columns var, LMI, and const display for each problem the number of decision variables, the LMI size, and the number of nonlinear equality constraints in $g(x) = 0$.

In Table 2, the column $P_{\lambda,\mu}$ gives the number of instances of the augmented Lagrangian subproblem. Each of these programs is solved by a succession of SDPs, and the column labeled SDP therefore gives the total number of SDPs needed to solve

Table 2
Results of static H_∞-synthesis.

pb.	$P_{\lambda,\mu}$	SDP	μ	ω	$\|g\|_\infty$	Full/static
pb5	16	20	$1.42e{-}2$	$1.12e{-}2$	$5.71e{-}6$	$9.63e{-}5$; 3.49
pb10	20	21	$5.58e{-}5$	$0.63e{-}2$	$2.08e{-}7$	3.44; 3.48
pb15	21	27	$2.18e{-}4$	$3.28e{-}2$	$7.55e{-}6$	3.44; 3.48
pb20	16	21	$9.07e{-}4$	$8.05e{-}3$	$9.14e{-}6$	3.25; 3.74
pb25	21	22	$5.50e{-}5$	$5.35e{-}2$	$5.90e{-}6$	4.60; 4.61
pb30	26	32	$3.34e{-}6$	$4.23e{-}2$	$9.77e{-}6$	1.099; 1.317
pb35	28	35	$8.26e{-}6$	$2.07e{-}2$	$4.43e{-}6$	6.47; 8.46

(P). As a rule, only between one and two SDPs per subproblem $(P_{\lambda,\mu})$ are needed. The number of SDPs needed to solve the augmented Lagrangian problem (P) may be considered the crucial parameter to judge the speed of our approach.

In our tests, SDPs are solved with an alpha version of our own spectral SDP code, which minimizes convex quadratic objectives subject to LMI-constraints

$$\begin{array}{ll} \text{minimize} & c^T x + \frac{1}{2} x^T Q x \\ \text{subject to} & \mathcal{A}(x) \preceq 0. \end{array} \tag{20}$$

In contrast, currently available SDP solvers are often based on the cast

$$\min\{c^\top x : \mathcal{A}(x) \preceq 0\}.$$

We have observed that those run into numerical problems very early, since the quadratic term $x^\top Qx$ in the objective of (20) has to be converted into an LMI via Schur complement. This leads to large-size LMIs very quickly. For the problems in Table 1 the corresponding augmented LMIs are of size 57×57 in pb5, 160×160 in pb10, 309×309 in pb15, 516×516 in pb20, 766×766 in pb25, 1068×1068 in pb30, and 1418×1418 in pb35.

The remaining entries in Table 2 are as follows. Column μ gives the final value of the penalty parameter, while $\|g\|_\infty$ gives the final precision in the equality constraint. In each of our test cases this precision was small enough in order to enable the procedure in [31] to find a controller K meeting both design specifications, stability and H_∞-performance. This may be regarded as the ultimate test of success of the method. The column ω gives the final value $\|P(-\nabla\Phi)\|$ used in the stopping test (6). We have observed that (6) should be employed rather tolerantly, which suggests using a comparatively large stopping tolerance ω_* in step 2 of the augmented Lagrangian algorithm. (This is also reflected by the fact that the covering sequence ω_k converges to 0 fairly slowly.)

The column full/static should be interpreted with care. It compares the performance $\gamma = \|T_{w,z}(s)\|_\infty$ achieved by the solution of (S) to the lower bound γ_∞ of the full H_∞-controller, computed by the usual SDP or Riccati method. In general γ_∞ cannot be a tight lower bound for the best possible γ in (S), but in a considerable number of cases both gains are fairly close. This indicates that our method, as a rule, gets close to the global minimum of (S), even though theoretical evidence for this is lacking. Notice here that even cases with a large gap between γ and γ_∞ do not contradict this supposition. One may always artificially arrange a large gap by creating a poorly actuated system, that is, a system where the number of control inputs is much smaller than the state of the system, $m_2 << n$.

For the set of test examples (S) in automatic control we observed that the partially augmented Lagrangian method required a rather limited number of SDP subproblems.

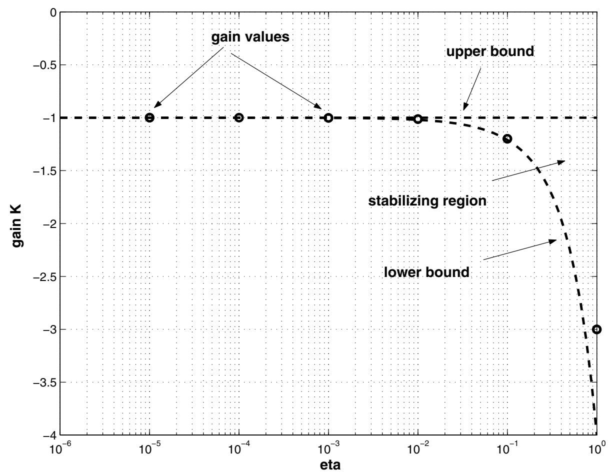

FIG. 1. *Stabilizing static gains and stabilizing region.*

This makes our approach based on a succession of SDPs attractive. Moreover, we observed that a solution to the control problem is practically always found provided the SDP subproblems can be solved in a reasonable time. The bottleneck of our approach *is* the SDP solver. The alpha version of our own solver performed well up to systems of size $n = 35$, which means 1261 decision variables and LMIs of size 156×156 (or of size 1418×1418 if the quadratic term is Schur complemented into an LMI).

7.3. Numerical experiment II. In this section we present an experiment imported from [41]. Numerical data are

$$A = \begin{bmatrix} 1 & 1+\eta \\ -(1+\eta) & 0 \end{bmatrix}, \qquad B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \qquad C_2 = [\,0 \quad 1\,],$$

where η is some small positive parameter. We consider the static stabilization problem for various values of η, so this example does not include performance, meaning $B_1 = 0$, $D_{11} = 0$, $C_1 = 0$, $D_{12} = 0$. The attractive feature of this example is that feasible values for the gain reduce to a small interval $K \in (-(1+\eta)^2,\ -1)$ whose length decreases quickly as η tends to zero. As a consequence, most existing algorithms fail when the interval shrinks significantly. Also, this example provides an indication on the accuracy and reliability of the proposed augmented Lagrangian algorithm.

Using the proposed technique, we computed a stabilizing gain K for a set of values $\eta \in \{1,\ 10^{-1},\ 10^{-2},\ 10^{-3},\ 10^{-4},\ 10^{-5}\}$. The results are shown in Figure 1, where in logarithmic scale the tube of admissible values K appears as the zone between the horizontal line at $K = -1$ and the curved lower bound. The achieved values of K are indicated through small circles and, as expected, lie within the stabilizing region. Note that the stabilizing region shrinks dramatically as η gets smaller. For $\eta = 10^{-6}$ the algorithm fails since the system can be regarded as numerically unstabilizable. The admissible interval length has then been reduced to about $2.0 \cdot 10^{-6}$. The problem has 13 decision variables.

In parallel with the experiments presented in sections 7.2 and 7.3, we point the reader to the testing in [5], where reduced order synthesis (including the static case) is examined from various other points of view.

7.4. Robust synthesis with time-varying uncertainties. In this section we consider a second class of automatic control applications of program (S), the robust control problem for an uncertain plant subject to parametric uncertainties. These system uncertainties may be described by so-called linear fractional transforms (LFTs):

$$\begin{bmatrix} \dot{x} \\ z_\Delta \\ z \\ y \end{bmatrix} = \begin{bmatrix} A & B_\Delta & B_1 & B_2 \\ C_\Delta & D_{\Delta\Delta} & D_{\Delta 1} & D_{\Delta 2} \\ C_1 & D_{1\Delta} & D_{11} & D_{12} \\ C_2 & D_{2\Delta} & D_{21} & 0 \end{bmatrix} \begin{bmatrix} x \\ w_\Delta \\ w \\ u \end{bmatrix}, \quad w_\Delta = \Delta(t)\, z_\Delta. \tag{21}$$

Here $\Delta(t)$ is a time-varying matrix-valued parameter, usually assumed to have a block-diagonal structure

$$\Delta(t) = \operatorname{diag}(\ldots, \delta_i(t)I, \ldots, \Delta_j(t), \ldots) \in \mathbf{R}^{N\times N} \tag{22}$$

normalized such that

$$\Delta(t)^T\Delta(t) \leq I \qquad \forall\, t \geq 0. \tag{23}$$

According to the μ analysis and synthesis literature [27, 24], blocks $\delta_i I$ and Δ_j are referred to as repeated-scalar blocks and full blocks, respectively. Straightforward computations lead to the state-space representation

$$\begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} = \left\{ \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & 0 \end{bmatrix} + \begin{bmatrix} B_\Delta \\ D_{1\Delta} \\ D_{2\Delta} \end{bmatrix} \Delta(t)(I - D_{\Delta\Delta}\Delta(t))^{-1} \left[C_\Delta \quad D_{\Delta 1} \quad D_{\Delta 2} \right] \right\} \times \begin{bmatrix} x \\ w \\ u \end{bmatrix},$$

which means that the state-space data of the plant with inputs w and u and outputs z and y are fractional functions of the time-varying uncertain parameter $\Delta(t)$. Hence we have the name *LFT*. In (21), u is the control input, w denotes the exogenous input, z denotes the controlled performance variables, and y denotes the measurement signal or output. Given the uncertain plant (21)–(23), the robust control problem requires finding a linear time-invariant (LTI) controller

$$\begin{aligned} \dot{x}_K &= A_K x_K + B_K y, \\ u &= C_K x_K + D_K y, \end{aligned} \tag{24}$$

such that for *all* admissible parameter trajectories (22), (23),

- the closed-loop system, obtained by substituting (24) into (21)–(23), is internally stable.
- the L_2-induced gain of the operator connecting w to z is bounded by γ.

Note that the performance specification says that

$$\int_0^\infty z(t)^T z(t)\mathrm{d}t \leq \gamma^2 \int_0^\infty w(t)^T w(t)\mathrm{d}t \;\forall w \in L_2,\; \forall\Delta(t) \text{ as in (22), (23).}$$

It has been shown that such problems may be handled via a version of the Bounded Real Lemma [1], which translates these conditions into matrix inequalities. The resulting sufficient conditions are the following: Find a Lyapunov matrix $X_{c\ell}$, $X_{c\ell} \succ 0$ and appropriately structured scalings S and T such that

$$\begin{bmatrix} A_{c\ell}^T X_{c\ell} + X_{c\ell} A_{c\ell} & * & * \\ B_{c\ell}^T X_{c\ell} + \begin{bmatrix} T & 0 \\ 0 & 0 \end{bmatrix} C_{c\ell} & \begin{array}{c} -\begin{bmatrix} S & 0 \\ 0 & \gamma I \end{bmatrix} \\ +\begin{bmatrix} T & 0 \\ 0 & 0 \end{bmatrix} D_{c\ell} + D_{c\ell}^T \begin{bmatrix} T & 0 \\ 0 & 0 \end{bmatrix}^T \end{array} & * \\ C_{c\ell} & D_{c\ell} & -\begin{bmatrix} S & 0 \\ 0 & \gamma I \end{bmatrix}^{-1} \end{bmatrix} \prec 0. \tag{25}$$

Here $A_{c\ell}$, $B_{c\ell}$, $C_{c\ell}$, and $D_{c\ell}$ are the state-space data of the closed-loop system (21)–(24) with the loop $w_\Delta = \Delta(t)\, z_\Delta$ still open. *Appropriately structured* means that the scalings are compatible with the uncertainty structure (22), (23), for which we use the shorthand notation $\Delta \in \mathbf{\Delta}$. The set of symmetric scalings associated with the parameter structure $\mathbf{\Delta}$ is defined as

$$S_{\mathbf{\Delta}} := \left\{ S : S^T = S, \quad S\Delta = \Delta S \quad \forall \Delta \in \mathbf{\Delta} \right\}.$$

Similarly, the set of skew-symmetric scalings associated with the parameter structure $\mathbf{\Delta}$ is defined as

$$T_{\mathbf{\Delta}} := \left\{ T : T^T = -T, \quad T\Delta = \Delta^T T \quad \forall \Delta \in \mathbf{\Delta} \right\}.$$

The idea here is that as soon as $\Delta \in \mathbf{\Delta}$ and $S \succ 0$, $S \in S_{\mathbf{\Delta}}$, $T \in T_{\mathbf{\Delta}}$, the quadratic constraint

$$\begin{bmatrix} \Delta \\ I \end{bmatrix}^T \begin{bmatrix} -S & T^T \\ T & S \end{bmatrix} \begin{bmatrix} \Delta \\ I \end{bmatrix} \succeq 0 \tag{26}$$

is automatically satisfied. With the above definitions and notation, the following algebraically constrained LMI characterization for the solvability of the problem can be established. The reader is referred to [45, 44, 2, 3, 33, 51, 25] for more details and additional results.

THEOREM 7.2. *Consider the plant* (21) *with uncertain trajectories* $\Delta(t)$ *as in* (22), (23). *The following conditions guarantee the existence of an LTI output-feedback controller* (24), *which stabilizes the closed-loop system* (21)–(24) *for all admissible trajectories* $\Delta \in \mathbf{\Delta}$ *such that, in addition, the* L_2 *gain performance* γ *is achieved: There exist symmetric matrices* X, Y, S, Σ *and skew-symmetric matrices* T, Γ *such that*

$$S,\ \Sigma \in S_{\mathbf{\Delta}} \qquad \textit{and} \qquad T,\, \Gamma \in T_{\mathbf{\Delta}}, \tag{27}$$

and such that the matrix inequalities

$$\mathcal{N}_1^T \begin{bmatrix} A^TX+XA & XB_\Delta + C_\Delta^T T^T & XB_1 & C_\Delta^T S & C_1^T \\ B_\Delta^T X + TC_\Delta & -S+TD_{\Delta\Delta}+D_{\Delta\Delta}^T T^T & TD_{\Delta 1} & D_{\Delta\Delta}^T S & D_{1\Delta}^T \\ B_1^T X & D_{\Delta 1}^T T^T & -\gamma I & D_{\Delta 1}^T S & D_{11}^T \\ SC_\Delta & SD_{\Delta\Delta} & SD_{\Delta 1} & -S & 0 \\ C_1 & D_{1\Delta} & D_{11} & 0 & -\gamma I \end{bmatrix} \mathcal{N}_1 \prec 0, \tag{28}$$

$$\mathcal{N}_2^T \begin{bmatrix} AY+YA^T & YC_\Delta^T + B_\Delta \Gamma^T & YC_1^T & B_\Delta \Sigma & B_1 \\ C_\Delta Y + \Gamma B_\Delta^T & -\Sigma+\Gamma D_{\Delta\Delta}^T + D_{\Delta\Delta}\Gamma^T & \Gamma D_{1\Delta}^T & D_{\Delta\Delta}\Sigma & D_{\Delta 1} \\ C_1 Y & D_{1\Delta}\Gamma^T & -\gamma I & D_{1\Delta}\Sigma & D_{11} \\ \Sigma B_\Delta^T & \Sigma D_{\Delta\Delta}^T & \Sigma D_{1\Delta}^T & -\Sigma & 0 \\ B_1^T & D_{\Delta 1}^T & D_{11}^T & 0 & -\gamma I \end{bmatrix} \mathcal{N}_2 \prec 0, \tag{29}$$

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} \succ 0, \tag{30}$$

$$\begin{bmatrix} S & 0 \\ 0 & \Sigma \end{bmatrix} \succ 0 \tag{31}$$

in tandem with the algebraic constraints

$$(S+T)^{-1} = (\Sigma+\Gamma), \qquad \textit{or equivalently,} \qquad \begin{bmatrix} S & T \\ T^T & -S \end{bmatrix}^{-1} = \begin{bmatrix} \Sigma & \Gamma^T \\ \Gamma & -\Sigma \end{bmatrix}, \tag{32}$$

are satisfied. Here $\mathcal{N}_1$ *is a basis of the null space of* $[C_2, D_{2\Delta}, D_{21}, 0]$*, and* $\mathcal{N}_2$ *is a basis of the null spaces of* $[B_2^T, D_{\Delta 2}^T, D_{12}^T, 0]$*.* □

Note that due to the algebraic constraints (32), the problem under consideration is NP-hard [9] and not solvable via SDP. Even simpler instances of this problem like those considered in [39] are already NP-hard. (This is in sharp contrast to the associated nominal H_∞-synthesis problem, which reduces to a standard SDP since the nonlinear conditions (32) fully disappear.) This is our second example of how a program of type (S) may be obtained in lieu of a program of type (B), based directly on (25). Once again this rests on a diligent use of the Projection Lemma.

The explicit form (S) is obtained through the following steps. As previously done, replace strict inequalities $\prec 0$ by $\preceq -\varepsilon I$. For the structure (22), conditions (27) imply a typical block structure for the matrices S, Σ, T, Γ, so the conditions (31) reduce to blocks of LMIs, to which the nonlinear equality constraints (32) have to be added. The cost function to be minimized is γ, and the decision vector is $x = (X, Y, \gamma, S, T, \Sigma, \Gamma)$, which regroups the gain γ, the multiplier variables S, Σ, T, Γ, and the Lyapunov matrix variables X, Y. As before, due to the Projection Lemma, the controller data do not directly enter the decision vector x and have to be recovered from the optimal x through the procedure in [31]. With these elements, the problem is directly open to our partially augmented Lagrangian algorithm, and the numerical tests presented below have been obtained accordingly.

7.5. Numerical experiment III: Flexible satellite. We consider the design of a robust attitude control system for a flexible satellite, adopted from [14, 17]. Despite its seemingly moderate size, this problem has been identified as a difficult case, where nominal H_∞-synthesis fails and robust techniques are required. We confine our study to the yaw axis of the satellite, whose dynamics are of the form

$$M\ddot{\phi} + D\dot{\phi} + K\phi = Tu,$$

where $\phi = [\phi_1\ \phi_2]^T$ and ϕ_1 is the yaw angle displacement in radians, ϕ_2 is the modal displacement used to represent the flexible dynamics, and u is the control torque (inch-pounds). Numerical data are given as

$$M = \begin{bmatrix} 77511 & 248.1 \\ 248.1 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0.002288 \end{bmatrix},$$
(33)
$$K = \begin{bmatrix} 0 & 0 \\ 0 & k_0 \end{bmatrix}, \qquad T = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

with $k_0 = 0.104124$. As is often convenient in applications, we augment the model by the (redundant) integral of ϕ_1. This introduces a new variable $\dot{\phi}_3 = \phi_1$, whose role will become clear when performance variables will be specified. The system can then be rewritten in first order form as

$$\dot{x} = Ax + Bu,$$

where $x = [\phi_1,\ \phi_2,\ \dot{\phi}_1, \dot{\phi}_2,\ \int \phi_1 \mathrm{dt}] \in \mathbb{R}^5$ and

$$A = \begin{bmatrix} 0_{2\times 2} & I_2 & 0_{2\times 1} \\ -M^{-1}K & -M^{-1}D & 0_{2\times 1} \\ [1 \quad 0] & 0_{1\times 2} & 0 \end{bmatrix}, \ B = \begin{bmatrix} 0_{2\times 1} \\ M^{-1}T \\ 0 \end{bmatrix}.$$

The measured variables are $y = [\phi_1,\ \dot{\phi}_1,\ \int \phi_1 \mathrm{dt}] \in \mathbb{R}^3$. Specifications in this design problem are twofold:

- We wish to maintain a pointing accuracy of $\leq 4.0 \cdot 10^{-4}$ radian (0.023 degree) in the yaw angle.
- This pointing accuracy must be guaranteed in the presence of 25% variation in the structural frequency due to uncertainties in the stiffness matrix M.

It was shown in [17] that the requested 25% variation in the structural frequency corresponds to the uncertainty $0.053296 \leq k \leq 0.154953$ in the parameter k_0. This in turn leads to the uncertain model $k = k_0 + W_\delta \delta$ with $k_0 = 0.104124$, the nominal value, $\delta \in [-1,\ 1]$, and $W_\delta = 0.0050829$. The satellite model incorporating uncertainty is now obtained as

$$\begin{bmatrix} \dot{x} \\ z_\Delta \\ y \end{bmatrix} = \begin{bmatrix} A & B_\Delta & B \\ C_\Delta & D_{\Delta\Delta} & D_{\Delta 2} \\ C & D_{2\Delta} & D_{22} \end{bmatrix} \begin{bmatrix} x \\ w_\Delta \\ u \end{bmatrix},$$
$$w_\Delta = \delta z_\Delta,\ \delta \in [-1,\ 1],$$

where A, B are as before and

$$\begin{aligned} B_\Delta &= [0 \quad 0 \quad m_{12} \quad m_{22} \quad 0], \\ C_\Delta &= W_\delta\, [0 \quad 1 \quad 0 \quad 0 \quad 0], \\ D_{\Delta\Delta} &= 0,\ D_{\Delta 2} = 0,\ D_{2\Delta} = 0,\ D_{22} = 0, \end{aligned}$$

and where m_{12} and m_{22} are the $(1,2)$ and $(2,2)$ elements of $-M^{-1}$, respectively. The matrix $C \in \mathbb{R}^{3\times 5}$ is such that $y = Cx$.

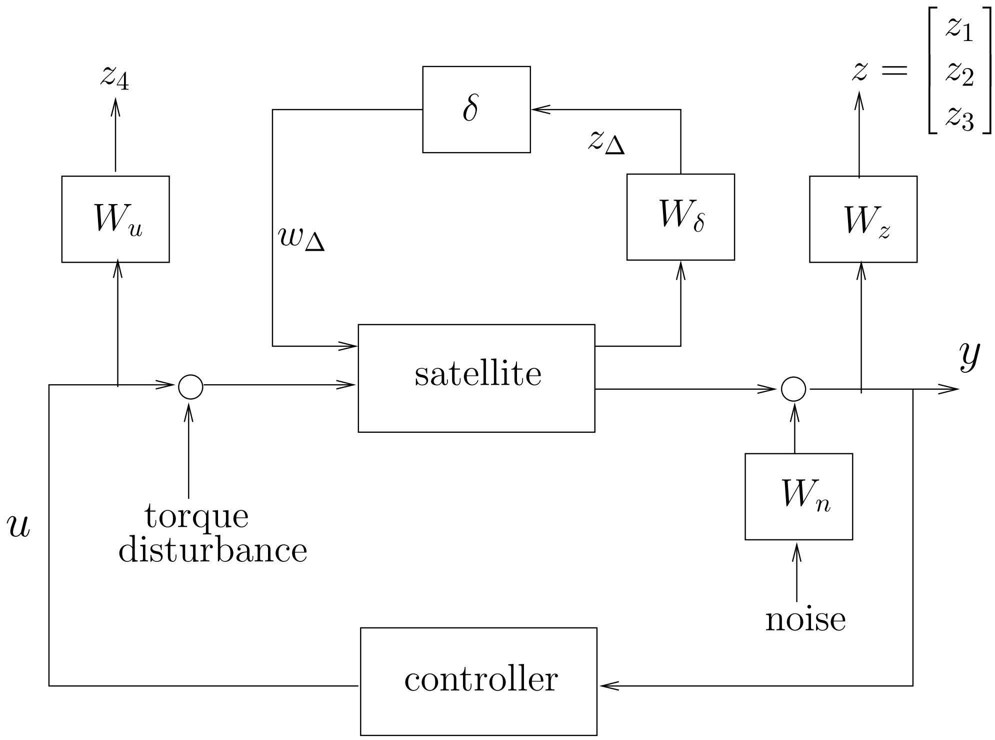

FIG. 2. *Synthesis architecture.*

This uncertain model has now to be completed by specifying exogenous input w and controlled output signals z. The data of the synthesis structure were all taken from [14] except for the noise weighting, W_n, which we have increased from 10^{-6} to 10^{-4} in order to comply with the increased performance request. This modification will highlight the differences between nominal and robust syntheses.

Inspecting the overall synthesis architecture in Figure 2, we see in which way the controller interacts with the satellite, and also how exogenous signals and performance signals are specified. We note that as is common in control system design, the measurements y are corrupted by noise. The noise magnitude is determined through the so-called weighting filter $W_n = 10^{-4}$ as discussed above, and this represents our first exogenous signal noise $= (w_1, w_2, w_3) \in \mathbb{R}^3$. The satellite is also subject to torque disturbance, which tends to deviate the yaw angle offset from its nominal zero level. This disturbance represents the fourth exogenous input w_4.

The loop featuring the δ block corresponds to the diagram representation of the LFT uncertainty in the structural frequency. The magnitude of the uncertainty is specified by the weighting $W_\delta = 0.0050829$ introduced to obtain the normalization $|\delta| \leq 1$.

A performance vector $(z_1, z_2, z_3) \in \mathbb{R}^3$ is introduced to reduce the impact of both noise and torque disturbance on the yaw angle offset. The corresponding channel is $(z_1, z_2, z_3) = W_z([\phi_1; \dot{\phi}_1; \int \phi_1 \mathrm{dt}] + W_n \text{ noise})$, where the weighting is defined as $W_z = \mathrm{diag}(400; 1; 20)$ and serves to specify the relative importance of the entries z_1, z_2, z_3 in the minimization. Indeed, z_1 specifies high-, z_2 specifies medium-, and z_3 specifies low-frequency parts of the yaw angle offset, and W_z allows us to address these components

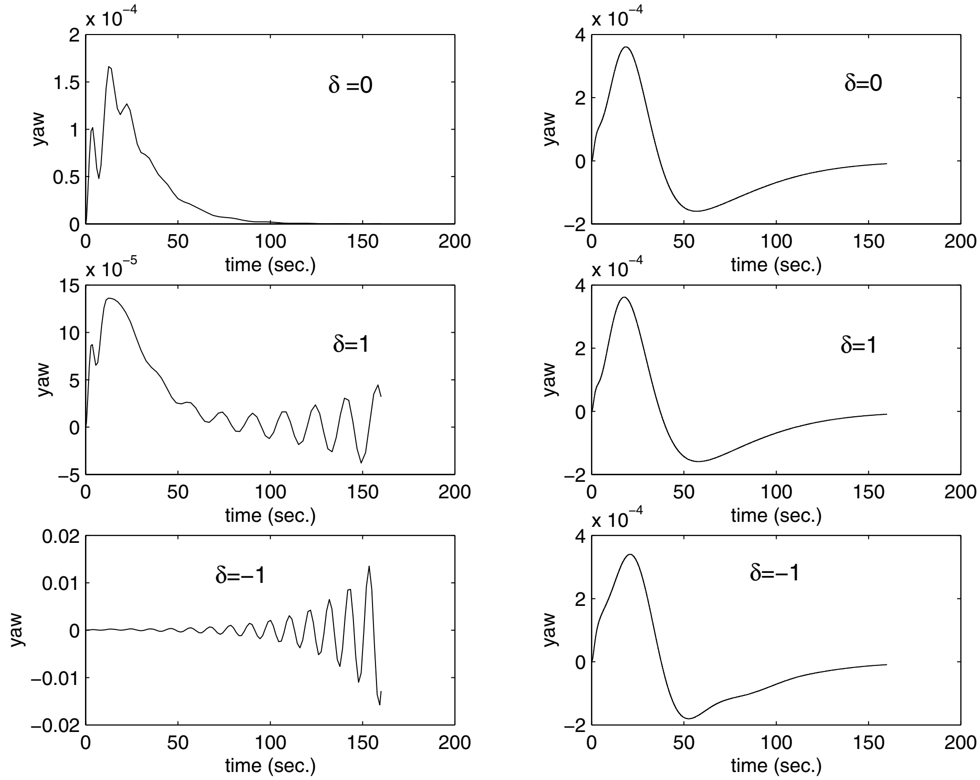

FIG. 3. *Simulations of yaw angle for* $\delta = 0;\, 1;\, -1$. *Left column: nominal controller. Right column: robust controller.*

individually. The effect of controlling (z_1, z_2, z_3) on the torque disturbance w_4 is indirect, as w_4 acts on the entry signal u to the satellite (see Figure 2).

Finally, in order to prevent unrealistic controller gains, the controller output u is given an additional cost by introducing a performance variable $z_4 = W_u u$. The weighting associated with this specification has been set to $W_u = 0.1$. We stress that choosing the weighting matrices reflects practical engineering specifications and is not a trivial task, involving both engineering insight and trial and error at this stage.

Note that two controllers were designed in this example. A nominal H_∞-controller was synthesized (corresponding to $\delta = 0$ above). A second robust H_∞-controller was synthesized using our augmented Lagrangian algorithm which explicitly accounts for the uncertainty in the flexible dynamics. Simulations were then performed with a torque step disturbance of 1 in-lbf at the plant input. The results are compared in Figure 3 for a set of values of δ. We observe that as expected the nominal controller (left-column simulations) is satisfactory in the nominal case ($\delta = 0$) but exhibits significant loss of performance and even of stability in the nonnominal situations $\delta = \pm 1$. This is in strong contrast with the robust controller obtained by our method, which meets the prescribed performance requirements despite the uncertainties in the flexible dynamics (right-column simulations).

REFERENCES

[1] B. Anderson and S. Vongpanitlerd, *Network Analysis and Synthesis: A Modern System Theory Approach*, Prentice–Hall, Englewood Cliffs, NJ, 1973.

[2] P. Apkarian and P. Gahinet, *A convex characterization of gain-scheduled $\mathcal{H}_\infty$ controllers*, IEEE Trans. Automat. Control, 40 (1995), pp. 853–864.

[3] P. Apkarian, P. Gahinet, and G. Becker, *Self-scheduled H_∞ control of linear parameter-varying systems: A design example*, Automatica J. IFAC, 31 (1995), pp. 1251–1261.

[4] P. Apkarian, D. Noll, and H. D. Tuan, *A Prototype Primal-Dual LMI-Interior Algorithm for Nonconvex Robust Control Problems*, Rapport Interne 01-08, MIP, UMR 5640, Toulouse, France, 2001. Available online at http://mip.ups-tlse.fr/publi/publi.html.

[5] P. Apkarian, D. Noll, and H. D. Tuan, *Fixed-order $\mathcal{H}_\infty$ control design via an augmented Lagrangian method*, Internat. J. Robust Nonlinear Control, 13 (2003), pp. 1137–1148.

[6] P. Apkarian and H. D. Tuan, *Concave programming in control theory*, J. Global Optim., 15 (1999), pp. 343–370.

[7] V. I. Arnold, *On matrices depending on parameters*, Russian Math. Surveys, 26 (1971), pp. 29–43.

[8] G. J. Balas, J. C. Doyle, K. Glover, A. Packard, and R. Smith, *μ-Analysis and Synthesis Toolbox: User's Guide*, The MathWorks, Inc., Natick, MA, 1991.

[9] K. P. Bennett and O. L. Mangasarian, *Bilinear separation of two sets in n-space*, Comput. Optim. Appl., 2 (1993), pp. 207–227.

[10] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization*, MPS/SIAM Ser. Optim., SIAM, Philadelphia, 2001.

[11] A. Ben-Tal, *Conic and Robust Optimization*, Lecture Notes, Università di Roma La Sapienzia, Rome, Italy, 2002.

[12] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, SIAM Stud. Appl. Math. 15, SIAM, Philadelphia, 1994.

[13] J. F. Bonnans and A. Shapiro, *Perturbation Analysis of Optimization Problems*, Springer-Verlag, New York, 2000.

[14] H. Buschek and H. A. J. Calise, *Mu controllers: Mixed and fixed*, AIAA J. Guidance Control Dyn., 20 (1995), pp. 43–41.

[15] J. Burke, A. S. Lewis, and M. L. Overton, *Optimizing matrix stability*, Proc. Amer. Math. Soc., 129 (2000), pp. 1635–1642.

[16] J. Burke, A. S. Lewis, and M. L. Overton, *Two numerical methods for optimizing matrix stability*, Linear Algebra Appl., 351/352 (2002), pp. 117–145.

[17] H. A. J. Calise and E. V. Byrn, *Parameter sensitivity reduction in fixed-order dynamic compensation*, AIAA J. Guidance Control Dyn., 15 (1992), pp. 440–447.

[18] R. Y. Chiang and M. G. Safonov, *Real K_m-synthesis via generalized Popov multipliers*, in Proceedings of the American Control Conference, Chicago, IL, 1992, pp. 2417–2418.

[19] F. H. Clarke, *Optimization and Nonsmooth Analysis*, SIAM Class. Appl. Math. 5, SIAM, Philadelphia, 1990.

[20] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds*, SIAM J. Numer. Anal., 28 (1991), pp. 545–572.

[21] A. R. Conn, N. I. M. Gould, A. Sartenaer, and Ph. L. Toint, *Global Convergence of Two Augmented Lagrangian Algorithms for Optimization with a Combination of General Equality and Linear Constraints*, Technical report TR/PA/93/26, Toulouse, France, 1993.

[22] A. R. Conn, N. I. M. Gould, A. Sartenaer, and Ph. L. Toint, *Local Convergence Properties of Two Augmented Lagrangian Algorithms for Optimization with a Combination of General Equality and Linear Constraints*, Technical report TR/PA/93/27, Toulouse, France, 1993.

[23] A. R. Conn, N. I. M. Gould, A. Sartenaer, and Ph. L. Toint, *Convergence properties of an augmented Lagrangian algorithm for optimization with a combination of general equality and linear constraints*, SIAM J. Optim., 6 (1996), pp. 674–703.

[24] J. C. Doyle, A. Packard, and K. Zhou, *Review of LFT's, LMI's and μ*, in Proceedings of the IEEE Conference on Decision and Control, Vol. 2, Brighton, UK, 1991, pp. 1227–1232.

[25] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory: A Convex Approach*, Springer Texts Appl. Math. 36, Springer-Verlag, New York, 2000.

[26] L. El Ghaoui, F. Oustry, and M. Ait Rami, *An algorithm for static output-feedback and related problems*, IEEE Trans. Automat. Control, 42 (1997), pp. 1171–1176.

[27] M. K. H. Fan, A. L. Tits, and J. C. Doyle, *Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics*, IEEE Trans. Automat. Control, 36 (1991), pp. 25–38.

[28] B. Fares, P. Apkarian, and D. Noll, *An augmented Lagrangian method for a class of LMI constrained problems in robust control theory*, Internat. J. Control, 74 (2001), pp. 348–360.

[29] B. Fares, D. Noll, and P. Apkarian, *Robust control via sequential semidefinite programming*, SIAM J. Control Optim., 40 (2002), pp. 1791–1820.

[30] E. Feron, P. Apkarian, and P. Gahinet, *Analysis and synthesis of robust control systems via parameter-dependent Lyapunov functions*, IEEE Trans. Automat. Control, 41 (1996), pp. 1041–1046.

[31] P. Gahinet and P. Apkarian, *A linear matrix inequality approach to H_∞ control*, Internat. J. Nonlinear Control, 4 (1994), pp. 421–448.

[32] M. X. Goemans and D. P. Williamson, *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*, J. ACM, 42 (1995), pp. 1115–1145.

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

[34] M. R. Hestenes, *Multiplier and gradient methods*, J. Optim. Theory Appl., 4 (1969), pp. 303–320.

[35] T. Iwasaki and R. E. Skelton, *All controllers for the general $\mathcal{H}_\infty$ control problem: LMI existence conditions and state space formulas*, Automatica J. IFAC, 30 (1994), pp. 1307–1317.

[36] J. LaSalle and S. Lefschetz, *Stability by Lyapunov's Direct Method*, Academic Press, New York, 1961.

[37] F. Leibfritz and E. M. E. Mostafa, *Trust region methods for solving the optimal output feedback design problem*, Int. J. Control, 76 (2003), pp. 501–519.

[38] F. Leibfritz and E. M. E. Mostafa, *An interior point constrained trust region method for a special class of nonlinear semidefinite programming problems*, SIAM J. Optim., 12 (2002), pp. 1048–1074.

[39] O. L. Mangasarian and J. S. Pang, *The extended linear complementarity problem*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 359–368.

[40] A. Megretzki and A. Rantzer, *System analysis via integral quadratic constraints*, IEEE Trans. Automat. Control, 42 (1997), pp. 819–830.

[41] M. Mesbashi, *A semi-definite programming solution of the least order dynamic output feedback synthesis problem*, in Proceedings of the IEEE Conference on Decision and Control, Phoenix, AZ, 1999.

[42] A. Nemirovskii, *Several NP-hard problems arising in robust stability analysis*, Math. Control Signals Systems, 6 (1994), pp. 99–105.

[43] M. C. de Oliveira and J. C. Geromel, *Numerical comparison of output feedback design methods*, in Proceedings of the American Control Conference, Albuquerque, NM, 1997, pp. 72–76.

[44] A. Packard, *Gain scheduling via linear fractional transformations*, Systems Control Lett., 22 (1994), pp. 79–92.

[45] A. Packard and G. Becker, *Quadratic stabilization of parametrically-dependent linear systems using parametrically-dependent linear dynamic feedback*, Adv. Robust Nonlinear Control Syst., 43 (1992), pp. 29–36.

[46] A. Packard, K. Zhou, P. Pandey, and G. Becker, *A collection of robust control problems leading to LMI's*, in Proceedings of the IEEE Conference on Decision and Control, Brighton, UK, 1991, pp. 1245–1250.

[47] M. J. D. Powell, *A method for nonlinear constraints in minimization problems*, in Optimization, R. Fletcher, ed., Academic Press, London, New York, 1969.

[48] M. G. Safonov and G. P. Papavassilopoulos, *The diameter of an intersection of ellipsoids and BMI robust synthesis*, in Proceedings of the IFAC Symposium on Robust Control Design, Rio de Janeiro, Brazil, 1994.

[49] A. Sartenaer, *Développement d'algorithmes en programmation non-linéaire: de l'étude théorique à l'expérimentation numérique*, Habilitation à diriger des recherches, Université Paul Sabatier, Toulouse, France, 1999.

[50] C. W. Scherer, *Robust mixed control and linear parameter-varying control with full block scaling*, in Advances in Linear Matrix Inequality Methods in Control, Adv. Des. Control 2, L. El Ghaoui and S.-I. Niculescu, eds., SIAM, Philadelphia, 1999, pp. 187–207.

[51] G. Scorletti and L. El Ghaoui, *Improved linear matrix inequality conditions for gain-scheduling*, in Proceedings of the IEEE Conference on Decision and Control, New Orleans, 1995, pp. 3626–3631.
[52] A. Shapiro, *First and second order analysis of nonlinear semidefinite programs*, Math. Program., 77 (1997), pp. 301–320.
[53] M. Torki, *First- and second-order epi-differentiability in eigenvalue optimization*, J. Math. Anal. Appl., 234 (1999), pp. 391–416.
[54] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice–Hall, Upper Saddle River, NJ, 1996.