2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Anti-windup design via nonsmooth multi-objective H_∞ optimization

Jean-Marc Biannic and Pierre Apkarian

Abstract— **This paper investigates constrained control design problems involving mixed magnitude and rate saturations. We show that simultaneous design of structured feedback and anti-windup gains can be recast as minimizing two H_∞-norm objectives. One is associated with nominal system operation while the other also captures the saturations effects. Because of the two-block structure of the controller, the problem is inherently difficult. We use recently available tools in nonsmooth optimization to compute controller components in a single run thereby alleviating the conservatism of more conventional schemes. The proposed methodology is applied to an aircraft control problem for which a PID feedback controller and a second-order anti-windup compensator are optimized simultaneously.**

keywords: Anti-windup, magnitude and rate saturations, structured controllers, `hinfstruct`, nonsmooth programming.

I. Introduction

New direct nonsmooth optimization techniques for controller design have have been developed over the past 10 years [1], [2]. The main thrust of these new tools lie in their flexibility to optimize arbitrary single- or multi-loop controller architectures built from linear time-invariant elements including static gains, PIDs, transfer functions, state-space models as well as custom components to suit most practical needs. We refer the reader to the routine `hinfstruct` from the Robust Control Toolbox, Matlab Release 2010b or higher. See [3] for details. Controller design with architecture/structure constraints is a complex problem and global solutions are not generally accessible. As demonstrated in a number of applications, nonsmooth H_∞ synthesis techniques turn out very effective to find good practical solutions if one exists. Also, since these techniques work in the reduced space of tunable controller elements they are only mildly sensitive to the plant dimension. Anti-windup control is another context where nonsmooth optimization can prove useful. Introduced in the $80's$, this specific control structure aims at preserving the nominal behavior of the plant while minimizing the adverse effects of magnitude and rate saturations of the control signals. In most standard schemes (see the tutorial papers [4], [5] and references therein), the anti-windup involves a two-block architecture consisting of a feedback controller and of an anti-windup component. The anti-windup component is optimized after the feedback controller has been tuned in the conventional approach which incurs some conservatism. In this paper, the focus is on simultaneous design of both the feedback controller and the anti-windup compensator using `hinfstruct`. In the special case of full-order feedback controllers and static anti-windup gains, an LMI-based solution was recently proposed in [6] for the restricted instance of magnitude saturations. In the general case, based on the formalism from [7], we show that the problem can be recast as a multi-objective synthesis problem involving two H_∞ constraints with a two-block structured controller. We then compare the conventional two-step approach with the nonsmooth method for a longitudinal aircraft model where the feedback controller is a simple PID and the anti-windup compensator is subject to order limitations.

The paper is organized as follows. The anti-windup control structure is first detailed in section II with a particular attention to magnitude and rate saturations models. Simplifying [8], we propose a model where the two nonlinearities are captured by a single parameter. Then, following a standard approach, a multi-objective H_∞ design formulation is proposed in section III. A brief overview of the nonsmooth technique is given in section IV. Finally, section V addresses a realistic illustration of the proposed methodology.

Notation. The notation used throughout the paper is standard. The normalized saturation operator is denoted by $\mathrm{sat}(.)$ *i.e.* for $u \in \mathbb{R}$, $\mathrm{sat}(u) = u$ if $|u| \leq 1$ and $\mathrm{sat}(u) = \mathrm{sign}(u)$ otherwise. Given two matrices M and N of appropriate dimensions, the (assumed well-posed) lower LFT is defined as $\mathcal{F}_l(M,N) := M_{11} + M_{12}N(I - M_{22}N)^{-1}M_{21}$, where the matrices M_{ij} define a suitable partition for M. For any (of arbitrary sizes) matrices M and N, $\mathrm{diag}(M,N)$ denotes the block-diagonal matrix. Given a square matrix $M \in \mathbb{C}^{n\times n}$, $\lambda_i(M)$ is the i^{th} eigenvalue of M and $\sigma_i(M)$ denotes the i^{th} singular value. As usual, $\overline{\sigma}(M) = \max_{i=1\ldots n}\sigma_i(M)$. Finally, for any locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}$, $\partial f(x)$ denotes its Clarke subdifferential at x [9].

II. On the anti-windup control structure

Although they are not necessarily limited to that case, anti-windup schemes are most often used to minimize the adverse effects of magnitude and rate limitations of control input signals (see Figure 1).

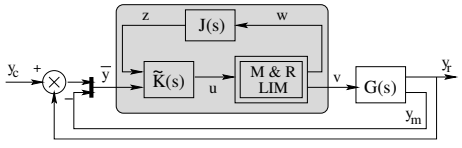

Fig. 1. Anti-windup structure

Authors are with the Systems Control and Flight Dynamics Department (DCSD) of ONERA, 2 Avenue Edouard Belin, BP 74025, 31055 Toulouse Cedex, FRANCE.

Consider a linear plant $G(s)$ to be controlled via rate and magnitude saturated inputs. With the notation of Figure 1, if no saturation occurs then the anti-windup signal verifies $w = v - u = 0$ and the control input u is described by its nominal expression:

$$u = \tilde{K}(s) \begin{bmatrix} \mathbf{0} \\ \bar{y} \end{bmatrix} = K(s)\bar{y} \tag{1}$$

Otherwise, the anti-windup loop becomes active:

$$u = \tilde{K}(s) \begin{bmatrix} z \\ \bar{y} \end{bmatrix}, \; z = J(s)w \neq 0 \tag{2}$$

where the LTI operator $J(s)$ denotes the anti-windup gain. In standard approaches, the latter is optimized for a fixed controller $K(s)$. In this contribution, as is also proposed in [6], we will focus on the simultaneous design of $J(s)$ and $K(s)$.

Remark 1: In the proposed structure, the nonlinear bounding operator is integrated into the global controller which eventually results from the interconnection of three elements as shown by the shaded region in Figure 1. Hence, the controller output v will never saturate, so that the actuator at the plant input can be accurately described by linear dynamics. The latter are included in the LTI model $G(s)$. Not only for design purposes, but also for implementation aspects, simple and accurate descriptions of the nonlinear operator (*MR-LIM*) are required.

A. Exact and approximated representations of mixed magnitude and rate limitations

An exact representation of the mixed magnitude & rate limiter is described by the diagram in Figure 2. Let denote L_M and L_R the maximum magnitude and rate limitations respectively, which for simplicity of the presentation are assumed symmetrical. In the scalar case, the two-input static nonlinear operator $\Phi(.,.)$ verifies:

$$\Phi(v,w) = \begin{cases} L_R \,\mathrm{sign}(w) & \text{if } \mathrm{sign}(w)v < L_M \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

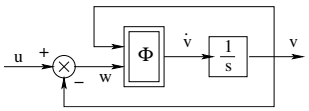

Fig. 2. Exact representation of *MR-LIM*

The above description is of high interest since the two nonlinearities are captured by a single and static operator Φ. However, because of the discontinuities in Φ, it cannot be implemented as depicted in Figure 1. Let us now rewrite Φ in a factorized form:

$$\Phi(v,w) = \Phi_R(w)\Phi_M(v,w) \tag{4}$$

where $\Phi_R(w) = L_R \mathrm{sign}(w) \in \{0, \pm L_R\}$ is related to the rate limitation, while $\Phi_M(v,w) \in \{0,1\}$ reflects the magnitude constraint. Using the approximation $\mathrm{sign}(w) \approx \mathrm{sat}(w/\varepsilon)$, and replacing $\mathrm{sat}(.)$ by an input-dependent gain one obtains:

$$\Phi(v,w) \approx \Phi_M(v,w) L_R \frac{\mathrm{sat}(w/\varepsilon)}{w} = \lambda(v,w)\, w \tag{5}$$

from which the diagram of Figure 3 is readily deduced.

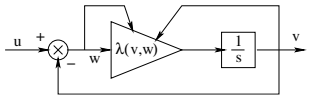

Fig. 3. Approximation of *MR-LIM*

Remark 2: It is easily checked that $\forall v, \forall w,\ \lambda(v,w) \in [0, L_R/\varepsilon]$. The lower bound corresponds to cases for which the magnitude constraint is activated, while the upper bound can be associated to the nominal situation for which $|v-u| \leq \varepsilon$ and no saturation is active. The proposed approximation can be tightened by reducing ε. But this will also shrink the linear region and induce chattering problems in the control laws (see [10]). In the context of this paper, since it can be observed that:

$$\forall \varepsilon > 0,\ \forall t \geq 0,\ |v(t)| \leq L_M,\ |\dot{v}(t)| \leq L_R \tag{6}$$

the choice of "large" values for ε will have no dramatic effects outside a (possibly significant) loss of performances. In practice, the designer will be guided by the actuators' dynamics. Let us denote for example ω_a the pulsation of a second-order actuator model. It should then be checked that $\omega_a \varepsilon \ll L_R$, so that the interactions between the actuators' dynamics and those introduced by the magnitude-and-rate limiter (in the linear region) remains negligible.

B. A design-oriented representation

From the diagram of Figure 3, a design-oriented representation of the magnitude-and-rate limiter can be derived by simply rewriting $\lambda(v,w)$ as a linear function of a normalized time-varying parameter $\delta(t)$:

$$\lambda(v,w) = \lambda_2(1 - \delta(t))\lambda_1 \tag{7}$$

where λ_1 and λ_2 are tuned so that the nominal case is recovered for $\delta(t) = 0$, which implies $\lambda_1\lambda_2 = L_R/\varepsilon$. From (7), it readily follows that rate-limited cases correspond to $\delta(t) \in]0,1[$ while the magnitude-limited behavior is obtained with $\delta(t) = 1$. The new model, illustrated on Figure 4.a, can be redrawn in a compact LFT format, as shown in Figure 4.b.

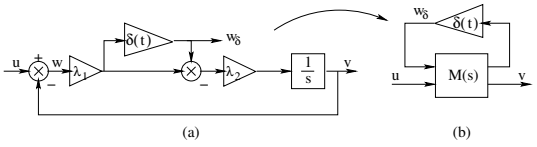

Fig. 4. Design-oriented version of *MR-LIM*

III. Formulation as an H_∞ design problem

Using the above ingredients and the standard upper-LFT notation for $J(s)$ and $K(s)$:

$$J(s) = \mathcal{F}_u(\Omega_J, I_{n_J}/s)\ ,\ K(s) = \mathcal{F}_u(\Omega_K, I_{n_K}/s) \tag{8}$$

the initial diagram of Figure 1 is readily redrawn in a design-oriented format as illustrated by Figure 5.

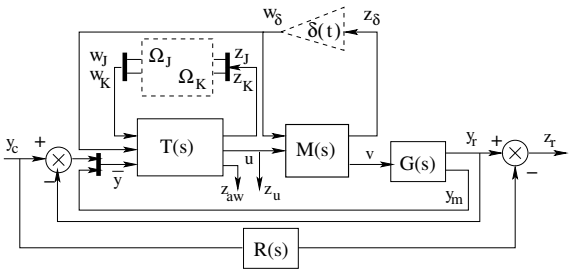

Fig. 5. H_∞ synthesis model for simultaneous feedback and anti-windup design

The LTI operator $T(s)$ is computed by straightforward LFT manipulations such that:

$$u = \tilde{K}(s)\begin{bmatrix} J(s)w_\delta \\ \bar{y} \end{bmatrix} = \mathcal{F}_u(T(s), \mathrm{diag}(\Omega_J, \Omega_K)) \tag{9}$$

and is then augmented to incorporate a weighting signal z_{aw} penalizing unrealistic high-frequency behavior in $J(s)$. The LTI operator $R(s)$ denotes the reference model to be tracked. Based on a small gain argument, following [11] or [12], a standard approach to preserve stability despite saturations effects consists in minimizing the H_∞-norm of the transfer function from w_δ to z_δ. Including nominal and robust performance leads to a multi-objective H_∞ control design problem:

$$\begin{cases} \|\mathcal{F}_l(P_{nom}(s), \Omega_K)\|_\infty \le c_1 \\ \|\mathcal{F}_l(P_{rob}(s), \mathrm{diag}(\Omega_J, \Omega_K))\|_\infty \le c_2 \end{cases} \tag{10}$$

where $P_{nom}(s)$ denotes the augmented plant translating nominal performance requirements and $P_{rob}(s)$ plays a similar role for robustness against possibly active magnitude and rate saturations. Note that only the second index depends on both $K(s)$ and $J(s)$ since saturations are not active in the nominal case. More precisely, with the notation of Figure 5, the nominal interconnection $P_{nom}(s)$ denotes the weighted transfer from (y_c, w_K) to (z_r, z_u, z_K), while $P_{rob}(s)$ also include anti-windup related signals (w_δ, w_J) and (z_δ, z_{aw}, z_J) at the inputs and outputs, respectively.

Interestingly, thanks to the general *LFT* format we have adopted for the description of $J(s)$ and $K(s)$, structural and order constraints are easily incorporated. This will be exploited in turn by the nonsmooth optimization technique to compute a simple PID controller for $K(s)$. This is easily achieved by setting the upper-left part of Ω_K to zero.

IV. Resolution via nonsmooth optimization

Anti-windup problems discussed in this paper are solved using tailored nonsmooth optimization techniques. Obviously the full details of these techniques are outside the scope of this paper, and we refer the reader to [1], [2] for comprehensive expositions on line-search-based methods on which `hinfstruct` is based. Also, a variety of problem studies as well as applications using `hinfstruct` as core tool can be found at [13]. Note that related nonsmooth techniques and software for control applications are also discussed in [14], [15] and references therein. In the sequel we recall the salient nonsmooth programming features of the Robust Control Toolbox routine `hinfstruct` [3] as it will be used in the application section V. The nonsmooth programming software for structured H_∞ synthesis is distributed by MathWorks as part of the Robust Control Toolbox 3.5, Matlab Release R2010b or higher [3].

In sharp contrast with standard approaches and generalizing [6] to highly structured and fixed-order architecture, the proposed anti-windup scheme computes both the feedback controller $K(s)$ and the anti-windup compensator $J(s)$ **simultaneously**. This leads to the bi-objective problem as stated above. Defining

$$\mathcal{H}(\Omega_J, \Omega_K, s) = \begin{bmatrix} \frac{1}{c_1}\mathcal{F}_l(P_{nom}(s), \Omega_K) & 0 \\ 0 & \frac{1}{c_2}\mathcal{F}_l(P_{rob}(s), \mathrm{diag}(\Omega_J, \Omega_K)) \end{bmatrix} \tag{11}$$

constraints in (10) are readily recast into the following nonsmooth program:

$$\begin{array}{ll} \underset{\Omega_K, \Omega_J}{\text{minimize}} & \max\limits_{\omega\in[0,\infty]} \overline{\sigma}(\mathcal{H}(\Omega_J, \Omega_K, j\omega)) \\ \text{subject to} & \text{nominal stability}. \end{array} \tag{12}$$

It is important to notice that constraints in (10) will be met as soon as a value less than one is achieved for program (12). In general, it is neither necessary nor advisable to run the code to completion and a quick exit is triggered when the unit target value is reached. Constants c_1 and c_2 are tuning parameters which serve to weigh the relative importance of nominal performance against robustness. Clearly, some trial-and-error is needed here till both nominal performance and behavior subject to saturations are both acceptable.

In the above expressions, internal nominal stability is easily captured using a constraint on the closed-loop spectral abscissa. We therefore include a constraint $\alpha\{A(G(s), \Omega_K)\} \le -\varepsilon$ with ε small enough and where $A(G(s), \Omega_K)$ stands for the nominal closed-loop dynamics in the nominal case and where α denotes the spectral abscissa, i.e., $\alpha(A) := \max_i \Re\lambda_i(A)$.

Expression (12) highlights the composite nature of the objective. The outer function $\max_{[0,\infty]} \overline{\sigma}(.)$ is nonsmooth but convex while the inner function $(\Omega_K, \Omega_J) \to \mathrm{diag}(\frac{1}{c_1}\mathcal{F}_l(P_{nom}, \Omega_K), \frac{1}{c_2}\mathcal{F}_l(P_{rob}, \mathrm{diag}(\Omega_J, \Omega_K)))$ is nonconvex but differentiable. Such functions are Clarke regular according to the terminology [9], which means that a lossless

description of the Clarke gradient is numerically accessible using chain rules and thereby good resolution algorithms can be built based on this information.

For simplicity of the discussion, program (12) is rewritten as:

$$\underset{\kappa\in\mathbb{R}^k}{\text{minimize}}\, f_\infty(\kappa) := \max_{\omega\in[0,\infty]} f(\kappa,\omega)\,, \tag{13}$$

where f_∞ aggregates both frequency domain constraints as well as the spectral abscissa constraint through a progress function discussed at length in [16]. Vector $\kappa\in\mathbb{R}^k$ gathers all tunable parameters in the controllers $K(s)$ and $J(s)$. By nature these programs represents challenging mathematical programming problems but their special composite structure allows us to distinguish between critical points including local minima κ^*, that is, those with $0\in\partial f_\infty(\kappa^*)$ from points κ that must be discarded, i.e., $0\notin\partial f_\infty(\kappa)$.

Solving (13) is carried out computationally by constructing a tangent model around the current iterate κ which is a quadratic first-order local approximation of the original problem (13). We introduce

$$\widehat{f}_\infty(\kappa+h,\kappa) := \max_{(\phi_f,\Phi_f)\in W_f} \phi_f - f_\infty(\kappa) + \Phi_f^T h + \frac{1}{2}h^T Q h\,, \tag{14}$$

where h is the displacement in the κ-space of controller parameters. For a given κ, the set W_f collects function values $\phi_f := f(\kappa,\omega)$ and subgradients $\Phi_f\in\partial f(\kappa,\omega)$ over an extended set of frequencies $\omega\in\Omega_f$. This can be implemented in different ways [1] but a minimal requirement is that Ω_f should at least contain active frequencies ω_a, that is, those achieving the maximum value in (13), we have that:

$$f_\infty(\kappa) = f(\kappa,\omega_a)\,.$$

The term extension set comes from the fact that Ω_f may include additional frequencies to improve our tangent model in (14) and promote convergence. See figure 6.

A key ingredient here is that selecting active frequencies (global peaks) is enough for the algorithm to converge. But adding a few more frequencies can accelerate convergence because the tangent program (14) will provide a better local approximation of the nonsmooth function and better steps will be performed at every iteration. Extensive numerical testing indicates that including frequencies bracketing global maxima and frequencies corresponding to secondary peaks (local maxima) dramatically enhances convergence [1].

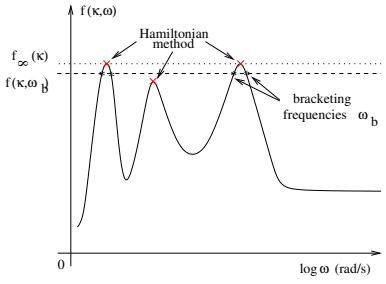

Fig. 6. Frequency selection to build tangent program

Program (14) is a standard convex quadratic program (CQP), and can be efficiently solved using currently available codes. Note the terminology "quadratic first-order local approximation" marks a subtle difference with the usual "quadratic second-order local approximation". Tangent program (14) indeed does extract first-order information from (Clarke) subgradients whereas the quadratic term is not a (second-order) Hessian in the usual sense. The latter is out of reach because functions are typically nonsmooth. Matrix Q is obtained through some sort of quasi-Newton estimate of the function local curvature. The way matrix Q evolves in the course of the algorithm is discussed in various papers along with the convergence theory. See [17] and references therein. The algorithm used in section V is a line search algorithm which updates matrix Q using a quasi-Newton BFGS technique where gradients of the BFGS formula correspond to the most active frequency. The key facts about (14) have been established in [1] and we re-state them here without proof:

- The fact that extension sets Ω_f contain active frequencies ensures that the solution h to (14) is a descent direction of f_∞ at κ. If it happens that $h=0$, then $0\in\partial f_\infty(\kappa)$, and we are done. Clearly, a stopping test may be based on the solution to the tangent program.
- The direction h can be used in an Armijo or Wolfe line search [18] which terminates after finitely many steps.

We then have all components to build a line search descent algorithm where at each iteration, a descent direction is computed, a line search is performed until a termination test holds. This algorithm has a sound convergence certificate which means that it converges to a critical point (a local minimum in practice) for any even remote starting point. Our point is that this strategy is generally preferable to biconvex $D-K$-iteration schemes or alike since they are prone to stall at points that are not local minima [19]. Finally, the proposed technique runs in the κ-space of free controller parameters without recourse to auxiliary variables such as Lyapunov variables which result in significant size inflation in BMI- or LMI-based methods.

V. Illustration

The proposed method is now applied to an aircraft control problem extracted from [20].

A. Description of the aircraft control problem

We then consider a longitudinal control design problem for a fighter aircraft with reduced control efficiency. A critical point in the flight domain is selected (medium altitude and low speed) accordingly. The linearized short term dynamics about this point read:

$$\begin{pmatrix}\dot\alpha\\ \dot q\end{pmatrix} = \begin{pmatrix}-0.5 & 1\\ 0.8 & -0.4\end{pmatrix}\begin{pmatrix}\alpha\\ q\end{pmatrix} + \begin{pmatrix}-0.2\\ -5\end{pmatrix}\delta_{e_r} \tag{15}$$

where α, q and δ_{e_r} denote the angle-of-attack, the pitch rate, and the effective elevator deflection respectively. During normal operations, the transfer from the commanded deflection

δ_{e_c} to δ_{e_r} verifies:

$$\frac{\delta_{e_r}}{\delta_{e_c}}(s) = \frac{\omega_a^2}{s^2 + 2\eta_a\omega_a s + \omega_a^2} \tag{16}$$

with $\omega_a = 60\,rad/s$ and $\eta_a = 0.6$. With reference to Figure 1, we have a fourth-order unstable plant model $G(s)$ for this application, with $y_r = \alpha$, $y_m = [\alpha, q]'$ and $v = \delta_{e_c}$. The objective is to track $y_r = \alpha$ without loss of stability on a large operating domain up to $45\,deg$ despite saturations of the control signals. These are fixed to $L_M = 20\,deg$ and $L_R = 80\,\mathrm{deg}/s$ for the magnitude and rate constraints respectively. Beyond stability requirements, the behavior of the closed-loop plant should remain as close as possible to that of a second-order reference model $R(s)$:

$$R(s) = \frac{1}{(0.25s+1)^2} \tag{17}$$

for which a nominal (without saturations) *PID* solution is known to exist [20].

B. Adaptation of the design model

With the above data, the design model is readily obtained from Figure 5. The interconnection $T(s)$ is calculated so as to enforce the following structure inspired by historical *PID* architectures with limited integrators:

$$\begin{cases} u = K_i \int(\alpha_c - \alpha + z)dt - K_a\alpha - K_q q \\ z = J(s)w_\delta \end{cases} \tag{18}$$

from which we infer:

$$\Omega_K = \mathrm{diag}(K_i, K_a, K_q)\,,\ \Omega_J = \begin{pmatrix} A_J & B_J \\ C_J & D_J \end{pmatrix} \tag{19}$$

Next, with reference to Figure 4, the operator $M(s)$ is built with $\lambda_1 = 100/L_R$ and $\lambda_2 = L_R$. For this choice, during normal operation, $M(s)$ behaves as a first-order linear model with a short time constant $\tau_M = 0.01\,s$ which is consistent with the actuators dynamics. The last – but also the most involving – step to complete the design models $P_{nom}(s)$ and $P_{rob}(s)$ requires appropriate choice of weighting functions in order to reflect the design constraints. This point is further discussed in the next subsection.

C. Weighting selection and resolution aspects

Given the *PID* structure constraint imposed to $K(s)$, we essentially need two weighting functions here. The first one – $W_{perf}(s)$ – weighs the error z_r between the plant output α and the reference trajectory. The second one – $W_J(s)$ – weighs the signal z_{aw} which we defined earlier in section III, in order to penalize high-frequency dynamics in the anti-windup controller. A few trials-and-error yield the following weighting filters:

$$\begin{cases} W_{perf}(s) = \frac{1}{1+0.5s} \\ W_J(s) = \tau\left(\frac{1+s}{1+\tau s}\right)^2 \text{ with } \tau = 10^{-4} \end{cases} \tag{20}$$

Remark 3: In any standard H_∞ design scheme, weighting functions $W_K(s)$ (typically high-pass filters) are also required on the controller outputs (denoted z_u in Figure 5). The reason why such functions are not present here ($W_K = 0$) is directly related to the imposed *PID* structure, for which an acceptable solution (with reasonable gains) is known to exist. For a similar reason, $W_J(s) = 0$ can be used when the anti-windup compensator is restricted to be static.

With the above filters, the constraints in (10) have been solved in the following two cases:

- case I : static anti-windup gain ($W_J = 0$)
- case II : second-order anti-windup gain ($W_J \neq 0$)

where the tuning parameters c_1 and c_2 on the right-hand side in (10) are chosen such that $c_1 \leq 0.01c_2$ which emphasizes satisfaction of nominal performance as a primary specification. Note that the H_∞ norm of the latter will necessarily verify $c_2 \geq 1$. This property is readily deduced from the diagram of Figure 4.a. In order to minimize the effects of saturations, the objective is then to reduce c_2 until it gets close to 1.

The results are summarized in Table I. Note that the first column corresponds to an initial tuning which was obtained from the classical two-step approach.

	init	case I	case II
c_1	$5.45\,10^{-4}$	$9.95\,10^{-3}$	$1.05\,10^{-2}$
c_2	1.143	1.094	1.057

TABLE I
OPTIMIZATION RESULTS

As expected, simultaneous optimization of $J(s)$ and $K(s)$ with `hinfstruct` outperforms separate design of the compensators as far as robustness is concerned. Moreover, the trade-off between nominal performance and robust performance is better optimized when the order of $J(s)$ increases. The numerical values of the anti-windup and *PID* feedback gains associated to each of the above cases are given in Table II.

	init	case I	case II
K_i	30	24	24
K_a	15	12.2	12.3
K_q	2.5	2.1	1.95
$J(s)$	23	13.5	$\frac{10.8-2.5\,10^{-2}s+1.4\,10^{-6}s^2}{1+3.8\,10^{-2}s+6\,10^{-4}s^2}$

TABLE II
FEEDBACK AND ANTI-WINDUP GAINS

D. Implementation and simulation results

The above solutions are now implemented in a SIMULINK^{TM}-based simulation diagram following the general picture of Figure 1. Simulations are performed with a fixed-step solver ($\tau = 0.02\,s$), so as to reproduce the behavior of embedded calculators.

A first simulation is performed without any anti-windup device. As clearly illustrated by Figure 7, an unstable behavior appears even for small values of the commanded angle-of-attack ($\alpha_c = 7\,deg$).

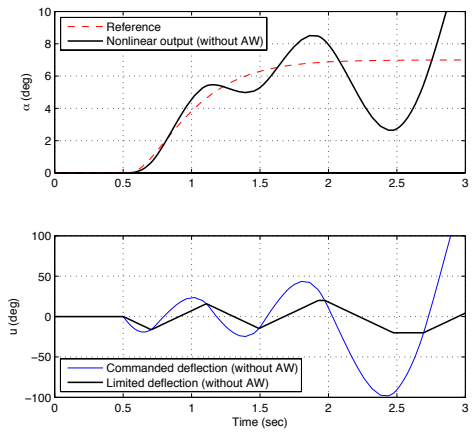

Fig. 7. Simulation results without anti-windup

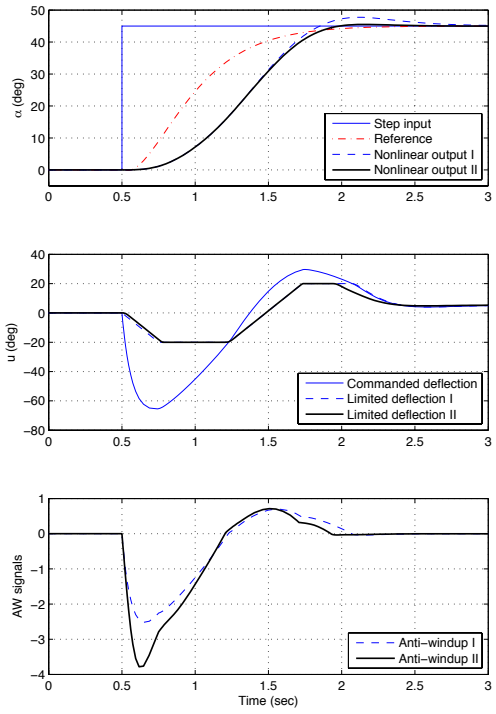

Fig. 8. Simulation results with static (dashed lines) and dynamic (thick & solid lines) anti-windup

In a next phase, the solutions associated to case I and II are tested. As expected, the unstable behavior has disappeared even for very large commanded values ($\alpha_c = 45\,deg$) and despite the saturations which are both actives during the simulations (see Figure 8). It is also interesting to note that the dynamic anti-windup (case II) performs slightly better than its static counterpart. The overshoot has disappeared.

VI. Conclusion

Based on the standard anti-windup structure, mixed magnitude and rate limited control design problems have been reformulated as a two-block structured multi-objective H_∞ synthesis. Exploiting recently available nonsmooth optimization techniques, it has been shown that structured and fixed-order feedback and anti-windup gains could be computed simultaneously. The advantage of this study is twofold. First, simple structure controllers are accessible using nonsmooth optimization which facilitates implementation if realistic applications is the aim. Secondly, simultaneous design of all components improves over conventional schemes as demonstrated for a longitudinal aircraft model.

References

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

[2] ——, "Nonsmooth optimization for multiband frequency domain control design," *Automatica*, vol. 43, no. 4, pp. 724–731, April 2007.

[3] MATLAB R2010b, *Robust Control Toolbox*. Natick, MA: The MathWorks Inc., 2010.

[4] S. Galeani, S. Tarbouriech, M. Turner, and L. Zaccarian, "A tutorial on modern anti-windup design," *European J. of Control*, vol. 15, no. 3-4, pp. 418–440, May-August 2009.

[5] S. Tarbouriech and M. Turner, "Anti-windup design: an overview of some recent advances and open problems," *IET Control Theory & Applications*, vol. 3, no. 1, pp. 1–19, January 2009.

[6] E. Mulder, P. Tiwari, and M. Kothare, "Simultaneous linear and anti-windup controller synthesis using multiobjective convex optimization," *Automatica*, vol. 45, no. 3, pp. 805–811, March 2009.

[7] M. Kothare and M. Morari, "Multivariable anti-windup controller synthesis using multiobjective optimization," in *Proc. American Control Conf.*, Albuquerque, New Mexico, June 1997, pp. 3093–3097.

[8] S. Galeani, S. Onori, A. Teel, and L. Zaccarian, "A magnitude and rate saturation model and its use in the solution of a static anti-windup problem." *Syst. Control Letters*, vol. 57, no. 1, pp. 1–9, Jan. 2008.

[9] F. H. Clarke, *Optimization and Nonsmooth Analysis*, ser. Canadian Math. Soc. Series. New York: John Wiley & Sons, 1983.

[10] I. Boiko and L. Fridman, "Analysis of chattering in continuous sliding-mode controllers," *IEEE Trans. Aut. Control*, vol. 50, no. 9, pp. 1442–1446, Sept. 2005.

[11] C. Edwards and I. Postlethwaite, "An anti-windup scheme with closed-loop stability considerations," *Automatica*, vol. 35, no. 4, pp. 761–765, April 1999.

[12] G. Ferreres and J.-M. Biannic, "Convex design of a robust anti-windup controller for an LFT model." *IEEE Trans. Aut. Control*, vol. 52, no. 11, pp. 2173–2177, November 2007.

[13] P. Apkarian, Internet Pages, http://pierre.apkarian.free.fr, 2010.

[14] S. Gumussoy, M. Millstone, and M. L. Overton, "H_∞ strong stabilization via HIFOO, a package for fixed-order controller design," in *Proc. IEEE Conf. on Decision and Control*, Cancun, Mexico, 2008, pp. 4135–4140.

[15] W. Michiels and S. I. Niculescu, *Stability and stabilization of time-delay systems. An eigenvalue based approach*, ser. Advances in Design and Control. SIAM Publications, 2007, vol. 12.

[16] P. Apkarian, D. Noll, and A. Rondepierre, "Mixed H_2/H_∞ control via nonsmooth optimization," *SIAM J. on Control and Optimization*, vol. 47, no. 3, pp. 1516–1546, 2008.

[17] V. Bompart, P. Apkarian, and D. Noll, "Nonsmooth techniques for stabilizing linear systems," in *Proc. American Control Conf.*, New York, NY, July 2007, pp. 1245–1250.

[18] D. P. Bertsekas, *Nonlinear Programming*. Belmont, Mass.: Athena Scientific, USA, 1995.

[19] J. W. Helton and O. Merino, "Conditions for optimality over H_∞," *SIAM J. on Control*, vol. 31, no. 6, pp. 1379–1415, 1993.

[20] J.-M. Biannic and S. Tarbouriech, "Optimization and implementation of dynamic anti-windup compensators in aircraft control systems with multiple saturations." *Control Eng. Practice*, vol. 17, no. 6, pp. 703–713, June 2009.