

# Model Predictive Control with Prioritised Actuators\*

Marco Gallieri<sup>1</sup> and Jan M. Maciejowski<sup>2</sup>

**Abstract**—This paper deals with the control of systems for which there is a clear distinction between preferred and auxiliary actuators, the latter to be used only when the control error is large. Explicit MPC and exact penalty functions are used to show how  $\ell_{asso}$ -MPC can implement this idea. Two  $\ell_{asso}$ -MPC versions are reviewed, that allow the designer to impose a certain nominal operations zone, namely, a neighbourhood of the set-point in which the auxiliary actuators are never used. For the sake of brevity, the required procedures are shown only for version 1, but it is also discussed how they can be extended to version 2. Limitations due to the presence of constraints are also formalised. The  $\ell_{asso}$ -MPC version 1 can be used to embed an existing linear quadratic MPC, while  $\ell_{asso}$ -MPC version 2 can be used to obtain multiple levels of priority. The paradigm is demonstrated for version 1 through the control of the linearised lateral dynamics of a Boeing 747. In particular, the approach uses the spoilers only when the control error is larger than a desired threshold.

## I. INTRODUCTION

This paper addresses the use of  $\ell_{asso}$ -MPC [1], [2], [3], [4] for control scenarios in which actuators can be clearly divided in two groups: *preferred actuators* and *auxiliary actuators*. A *pre-existing stabilising MPC* is assumed to be given, that considers only the former. These preferred actuators are set to be used for the most of the time, and are meant to stabilise the system. On the other hand, the auxiliary actuators are meant to be used only when the control error is outside a nominal region,  $\mathcal{X}_{nom}$ , containing the origin.

This paper provides a set of tools to construct a  $\ell_{asso}$ -MPC that includes the pre-existing controller as well as the auxiliary actuators. The designer can specify (to a certain extent) the region of the error space  $\mathcal{X}_{nom}$ , in which auxiliary actuators are guaranteed not to be used by the resulting  $\ell_{asso}$ -MPC. The proposed technique allows as well to restrict the use of auxiliary actuators to the times when the main actuators are saturated, or when certain constraints become active. Moreover, the original MPC controller is recovered in  $\mathcal{X}_{nom}$ . Constraints in MPC limit the achievable region of attraction. Similar limitations occur in the achievement of a certain  $\mathcal{X}_{nom}$ , as it will be detailed in this paper. The proposed procedure makes use of multi-parametric programming and the theory of exact penalty functions (Theorem 14.3.1 of [5]).

\*Research supported by the EPSRC grant “Control for Energy and Sustainability”, EP/G066477/1.

<sup>1</sup> Marco Gallieri is with McLaren Racing Limited, Chertsey Road, Woking GU21 4YH, UK mgallieri.ac@gmail.com

<sup>2</sup>Jan M. Maciejowski is with the Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PX, UK jmm1@cam.ac.uk

## II. CONSIDERED FRAMEWORK

This paper concerns the control of Linear Time-Invariant (LTI) systems of the form

$$x(k+1) = f(x(k), u(k)) = Ax(k) + Bu(k) \quad (1)$$

subject to constraints of the form

$$u \in \mathbb{U} \subset \mathbb{R}^m, \quad x \in \mathbb{X} \subset \mathbb{R}^n. \quad (2)$$

### A. The $\ell_{asso}$ -MPC

**Definition II.1.** ( $\ell_{asso}$ -MPC)

Consider the following finite-horizon constrained optimal control problem (FHCOCP)

$$V_N^o(x) = \min_{\underline{\mathbf{u}}} \left\{ V_N(x, \underline{\mathbf{u}}) \triangleq F(x_N) + \sum_{j=0}^{N-1} \ell(x_j, u_j) \right\}$$

$$s.t. \quad x_{j+1} = f(x_j, u_j),$$

$$u_j \in \mathbb{U}, \quad x_j \in \mathbb{X}, \quad \text{for } j = 0, 1, \dots, N-1,$$

$$x_0 = x(k), \quad x_N \in \mathbb{X}_f \subseteq \mathbb{X}, \quad (3)$$

with stage cost

$$\ell(x, u) = x^T Q x + u^T R u + \|S u\|_1, \quad (4)$$

where the terminal cost  $F(x)$  is strongly convex,  $\underline{\mathbf{u}}^T = [u_0^T \ \dots \ u_{N-1}^T]$ ,  $S \in \mathbb{R}^{m_s \times m}$ ,  $m_s \in \mathbb{I}_{>0}$ <sup>1</sup>. At each iteration  $k$ , the  $\ell_{asso}$ -MPC applies to the plant the first move of the optimal policy,  $u(k) = u_0^*$ , obtained by online solution of (3-4), at the current state,  $x(k)$ . The generated implicit control law is referred to as  $K_N(x) \equiv u_0^*(x)$ .

The following is assumed throughout the paper

**Assumption 1. (A2)** For system (1)

(H0)  $(A, B)$  is stabilisable,

(H1)  $Q \succ 0$ ,  $R \succ 0$ ,  $S \in \mathbb{R}^{n_s \times m}$ ,

(H2)  $\mathbb{X}$ ,  $\mathbb{U}$ ,  $\mathbb{X}_f$  are polytopic C-sets (convex, compact).

### B. Version 1: “Partial Regularisation”

This particular formulation features the terminal cost

$$F(x) = x^T P x, \quad P \succ 0. \quad (5)$$

If the prediction model  $f$  is LTI, this choice of  $F$  will result in a *strongly convex* Quadratic programme (QP). Stability of the origin of the closed-loop system is investigated in Chapter 4 of [6], where it is also shown that the explicit control law [7] is a continuous Piece-Wise Affine (PWA) function of  $x$ , with a *dead-zone* around the origin<sup>2</sup> (as in

<sup>1</sup>In this paper  $\mathbb{I}_{>0}$  denotes positive integers.

<sup>2</sup>This happens when  $S$  penalises all actuators.

Figure 1). Differently from MPC based on 1-norm costs, in the solution of  $\ell_{asso}$ -MPC is unique, thus avoiding actuators chattering. In order to obtain closed-loop asymptotic stability, in [6] it is assumed that only some actuators are regularised, namely, that the 1-norm penalty in (4) involves only a subset of actuators, labelled  $u^{(ii)}$ . The remaining, non-regularised actuators (used to stabilise the plant), are instead labelled  $u^{(i)}$ . This means, for instance, that

$$S = \begin{bmatrix} 0 & S_{\{ii\}} \end{bmatrix}, \quad (6)$$

for some matrix,  $S_{\{ii\}}$ . Note that, in (6), the actuators have been grouped as  $u = (u^{(i)}, u^{(ii)})$ , the former being the *main* or *preferred* actuators, used to stabilise the plant, and the latter being the *auxiliary* or *backup actuators*. Recall that we wish not to use the backup actuators when the control error is in a given neighbourhood of the origin,  $\mathcal{X}_{\text{nom}}$ . To do so, a suitable  $S_{\{ii\}}$  will be determined.

### C. Version 2: General form

This formulation features the terminal cost

$$F(x) = \beta \psi_{\mathbb{X}_f}(x)^2 + \alpha \psi_{\mathbb{X}_f}(x), \quad \beta, \alpha > 0, \quad (7)$$

where  $\psi_{\mathbb{X}_f}$  is the Minkowski function of the terminal set  $\mathbb{X}_f$ . This formulation is used to provide closed-loop asymptotic stability guarantee, for any regularisation penalty, in Chapter 5 of [6]. For this purpose,  $\mathbb{X}_f$  is taken to be  $\lambda$ -contractive [8]. If the prediction model  $f$  is LTI, then this choice of  $F$  gives again a *strongly convex* QP, with  $K_N(x)$  unique, continuous and PWA.

For reasons of limited space, this paper focuses only on the partial regularisation case. For the general case, a similar formulation is proposed in Chapter 6 of [6]. In particular, the fact that the preferred actuators are also regularised allows for an additional level of priority.

### D. Example 1: Soft-thresholding and dead-zone

Consider the LTI system with  $A = B = Q = R = 1$ ,  $N = 3$ ,  $S = 100$ ,  $|x| \leq 5$ ,  $|u| \leq 1$ , and  $P = 1.6180$ . The terminal constraint is  $|x_N| \leq 1.6180$ . The explicit solution of  $\ell_{asso}$ -MPC is computed using the Matlab Multi-Parametric Toolbox (MPT) [9]. As it will be shown in the next Section, the input is divided in 2 positive components,  $u = u_+ - u_-$ , providing a multi-parametric QP (mpQP) with positive definite Hessian (required by this version of MPT). The PWA control law, in Figure 1, features a *dead-zone*. In particular, when  $x$  is in the terminal set, the plant is open-loop. This is what happens to the regularised actuators.

The solution for different  $S$  can also be computed, by means of an mpQP. The control law, in Figure 2, is in fact PWA in both  $x$ ,  $S$ . It can be noticed, in Figure 2, how initially the *dead-zone* varies linearly with  $S$ , to then stop for approximately  $S \geq 11$ . This is due to the terminal constraint. If constraints are removed, then the dead zone will keep increasing linearly for increasing  $S$ .

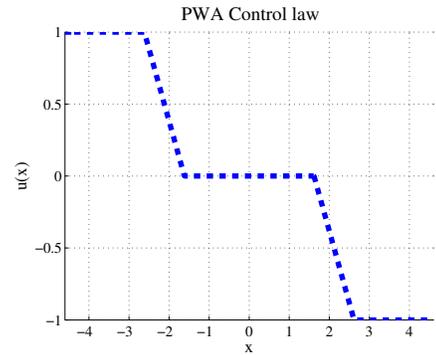


Fig. 1. Explicit  $\ell_{asso}$ -MPC for a scalar system: Control law is PWA with a *dead-zone*.

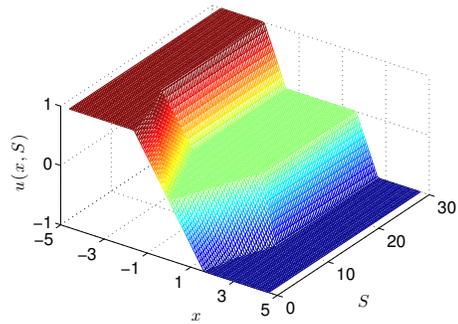


Fig. 2.  $\ell_{asso}$ -MPC solution path for a scalar system: The *dead-zone* varies with  $S$ .

## III. MULTI-PARAMETRIC FORMULATION

The considered problem is now formulated as a QP, in order to compute the required regularisation penalty. The purpose is to obtain a mpQP that can be solved offline using standard tools [7]. If the MPC is instead solved online, then more efficient formulations exist, see for instance [10] and references therein.

Suppose to have the polytopic constraints,  $\mathbb{X} = \{x \mid Lx \leq \underline{1}\}$ ,  $\mathbb{U} = \{u \mid Eu \leq \underline{1}\}$ , and  $\mathbb{X}_f = \{x \mid L_f x \leq \underline{1}\}$ . Then, a constrained  $\ell_{asso}$ -MPC for LTI systems can be implemented by solving, for instance, the strongly convex non-smooth problem (see Chapter 4 of [6])

$$\begin{aligned} \underline{u}^* &= \arg \min_{\underline{u}} \underline{u}^T H \underline{u} + 2\underline{u}^T \Gamma x + \|\underline{W}\underline{u}\|_1 \\ \text{s.t. } \Omega \underline{u} &\leq \underline{1} - Mx, \quad x = x(k), \end{aligned} \quad (8)$$

where  $\underline{1}$  is a vector of ones and

$$\Omega = \begin{bmatrix} (I_N \otimes L)\Theta \\ L_f \phi \\ I_N \otimes E \end{bmatrix}, \quad M = \begin{bmatrix} (I_N \otimes L)\Psi \\ L_f A^N \\ 0 \end{bmatrix}, \quad (9)$$

for appropriate matrices  $H$ ,  $\Gamma$ ,  $\Theta$ ,  $\Psi$ ,  $\phi$ , with  $W = I_N \otimes S$ .

**Assumption 2.** Throughout the paper, we assume that  $R = \text{BlockDiag}(R_1, R_2) \succ 0$ , with  $R_2$  and  $S_{\{ii\}}$  diagonal.

A unitary permutation,  $\Upsilon$ , is used to separate the

predictions of  $u^{(i)}$  from the ones of  $u^{(ii)}$ , namely,

$$\Upsilon \underline{u} = \begin{bmatrix} \underline{u}_{\{i\}} \\ \underline{u}_{\{ii\}} \end{bmatrix}, \quad \Theta \Upsilon^T = [\Theta_1, \Theta_2]. \quad (10)$$

The permutation can be computed as follows. Define  $\mathbb{I}^{(i)}$  and  $\mathbb{I}^{(ii)}$  as, respectively, the indices of the columns of  $B$  corresponding to the columns of actuator set (i) and (ii). Define  $\mathbb{I}^{(I)}$  and  $\mathbb{I}^{(II)}$  as

$$\begin{aligned} \mathbb{I}^{(I)} &= \{\mathbb{I}^{(i)}, \dots, \mathbb{I}^{(i)} + m(k-1), \dots, \mathbb{I}^{(i)} + m(N-1)\}, \\ \mathbb{I}^{(II)} &= \{\mathbb{I}^{(ii)}, \dots, \mathbb{I}^{(ii)} + m(k-1), \dots, \mathbb{I}^{(ii)} + m(N-1)\}, \end{aligned} \quad (11)$$

where  $k \in [1, N]$  is the *block index*. Suppose that an index  $j \in \mathbb{I}^{(I)}$ , with  $\bar{j}$  being the corresponding index in the list, that is  $j = \mathbb{I}_{\bar{j}}^{(I)}$ , with  $\bar{j} \in [1, m^{(i)}N]$  (or respectively,  $\bar{j} \in [1, (m - m^{(ii)}N)]$ ). Then,  $\Upsilon$  is constructed as follows

$$\Upsilon(i, j) = \begin{cases} 1 & \text{if } j \in \mathbb{I}^{(I)} \text{ and } i = \bar{j}, \\ 1 & \text{if } j \in \mathbb{I}^{(II)} \text{ and } i = m^{(i)}N + \bar{j}, \\ 0 & \text{otherwise.} \end{cases} \quad (12)$$

#### A. Splitting the actuators for strict convexity

The regularised variables are split into positive and negative part, namely,  $u_+$ ,  $u_-$  with  $u = u_+ - u_-$ . Then, the additional constraints  $u_+ \geq 0$ ,  $u_- \geq 0$  are added to the original problem. Then, problem (8) can be formulated as a strictly convex mpQP in condensed formulation

$$\begin{aligned} \chi^* &= \arg \min_{\chi} \frac{1}{2} \chi^T \bar{H} \chi + \chi^T \bar{G}(x) \\ \text{s.t. } \bar{\Omega} \chi &\leq b(x), \quad x = x(k). \end{aligned} \quad (13)$$

where

$$\chi = \begin{bmatrix} \underline{u}_{\{i\}} \\ \underline{u}_{\{ii\}+} \\ \underline{u}_{\{ii\}-} \end{bmatrix}, \quad \bar{G}(x) = 2\bar{\Gamma}x + \begin{bmatrix} 0 \\ W_{\{ii\}} \underline{1} \\ W_{\{ii\}} \underline{1} \end{bmatrix}, \quad (14)$$

and with

$$\begin{aligned} \bar{H} &= 2(\bar{\Theta}^T \mathbf{Q} \bar{\Theta} + \bar{\mathbf{R}}), \\ \bar{\mathbf{R}} &= \text{BlockDiag}(I_N \otimes R_1, I_N \otimes R_2, I_N \otimes R_2), \\ \bar{\Gamma} &= \bar{\Theta}^T \mathbf{Q} \Psi, \\ \bar{\Theta} &= [\Theta_1, \Theta_2, -\Theta_2]. \end{aligned} \quad (15)$$

Constraint matrices for (13) are

$$\bar{\Omega} = \begin{bmatrix} \Omega_1 & \Omega_2 & -\Omega_2 \\ & \Pi & \end{bmatrix}, \quad (16)$$

$$b(x) = \begin{bmatrix} \underline{1} - Mx \\ 0_{(2m^{(ii)}N \times 1)} \end{bmatrix}, \quad (17)$$

$$\Pi = \begin{bmatrix} 0 & -I_{m^{(ii)}N} & 0 \\ 0 & 0 & -I_{m^{(ii)}N} \end{bmatrix}, \quad (18)$$

where  $\Omega \Upsilon^T = [\Omega_1, \Omega_2]$ . At each time  $k$ , the signal applied to the plant is given by  $u(k) = K_N(x(k))$ , with

$$K_N(x(k)) = [I \ 0 \ \dots \ 0] \Upsilon^T \text{BlockDiag}(I, [I, -I]) \chi^*. \quad (19)$$

## IV. COMPUTATION OF THE REGULARISATION PENALTY

The regularisation penalty is computed by means of similar arguments to the ones for soft-constraints in [11]. The theory of exact penalty functions [5] is used. For this purpose, an mpQP is solved for all  $x \in \mathbb{X} \cap \mathcal{X}_{\text{nom}}$ . In particular, given (13) we impose that  $\|\underline{u}_{\{ii\}}\|_1 = 0$ , and then compute the corresponding Lagrange multipliers.

#### A. Limitations due to constraints

Define  $\mathcal{Q}^{(i)}$  to be the 1-step operator for  $u^{(ii)} = 0$ , namely

$$\mathcal{Q}^{(i)}(\mathcal{S}) = \{x \in \mathbb{R}^n : \exists u \in \mathbb{U} \mid u^{(ii)} = 0, Ax + Bu \in \mathcal{S}\} \quad (20)$$

and recall  $\mathcal{K}_j(\mathbb{X}, \mathbb{X}_f)$  to be the  $j$ -step controllable set from  $\mathbb{X}$  to  $\mathbb{X}_f$  [8], [12]. Similarly, define  $\mathcal{K}_N^{\{i\}}(\mathbb{X}, \mathbb{X}_f)$  as the  $N$ -step controllable set under  $u^{(ii)} = 0$ , from  $\mathbb{X}$  to  $\mathbb{X}_f$ . By definition,  $\mathbb{X}_N^{\{i\}} \equiv \mathcal{K}_N^{\{i\}}(\mathbb{X}, \mathbb{X}_f)$ , where  $\mathbb{X}_N^{\{i\}}$  is the feasible region, as well as the Domain Of Attraction (DOA) for the formulated  $\ell_{\text{asso}}$ -MPC, under the constraint  $\underline{u}_{\{ii\}} = 0$ .

**Theorem IV.1.** The maximal feasible set for which it is possible to have  $u^{\{ii\}} = 0$  is

$$\tilde{\mathbb{X}}_N^{(i)} = \mathcal{Q}^{(i)}(\mathcal{K}_{N-1}(\mathbb{X}, \mathbb{X}_f)) \cap \mathbb{X}. \quad (21)$$

**Proof IV.1.** The set  $\mathcal{K}_{N-1}(\mathbb{X}, \mathbb{X}_f)$  is the set of states in  $\mathbb{X}$  for which there exists an admissible control policy,  $\underline{u}_{[N-2]} = \{u(k)\}_0^{N-2} \in \mathbb{U}^{N-1}$ , such that  $x_{N-1} \in \mathbb{X}_f$  under this policy. Therefore, if the state  $x(k)$  is steerable to this set in 1 step by means of only the preferred actuators, then we also have  $x(k) \in \mathbb{X}_N = \mathcal{K}_N(\mathbb{X}, \mathbb{X}_f)$ , from (20). This means that  $\tilde{\mathbb{X}}_N^{(i)} \subseteq \mathbb{X}_N$ , and that, at the optimum,  $u^{(ii)} \neq 0, \forall x \in \mathbb{X}_N / \tilde{\mathbb{X}}_N^{(i)}$ . To prove that considered set is maximal, consider that  $\mathcal{K}_j(\mathbb{X}, \mathbb{X}_f) \subseteq \mathcal{K}_j^{\{i\}}(\mathbb{X}, \mathbb{X}_f), \forall j$  and  $\mathcal{K}_j(\mathbb{X}, \mathbb{X}_f) \subseteq \mathcal{K}_j^{\{ii\}}(\mathbb{X}, \mathbb{X}_f), \forall j$ , as well as  $\mathcal{X}_1 \subseteq \mathcal{X}_2 \Rightarrow \mathcal{K}_j(\mathbb{X}, \mathcal{X}_1) \subseteq \mathcal{K}_j(\mathbb{X}, \mathcal{X}_2), \forall j$ .

**Corollary IV.1.** The maximal set in which  $\underline{u}_{\{ii\}}^* = 0$  can occur is  $\mathbb{X}_N^{\{i\}} = \mathcal{K}_N^{\{i\}}(\mathbb{X}, \mathbb{X}_f)$ .

In this paper, we focus our attention on  $\mathbb{X}_N^{\{i\}}$ . In particular, a method is provided so that  $\underline{u}_{\{ii\}}^* = 0$  for  $\mathcal{X}_{\text{nom}} \subseteq \mathbb{X}_N^{\{i\}}$ . Unfortunately, the method does not guarantee that  $u^{(ii)} = 0$  is verified  $\forall x \in \tilde{\mathbb{X}}_N^{(i)}$ . The reason for this is essentially technical, and due to the fact that our 1-nom penalty is common to all future predictions of  $u^{(ii)}$ . However, as seen in this Section, a tuning for  $S$  could exist so that  $u^{(ii)} = 0$  in  $\tilde{\mathbb{X}}_N^{(i)}$ . This could be obtained, for instance, by up-scaling the results of the strategy proposed next.

#### B. Exact penalty

Problem (13) is strictly convex, and it can be solved by means of standard mpQP methods (for instance the method of [7]). We proceed in a similar way to [11]. In order to compute the exact penalty, consider a new mpQP, that is the same as (13) with the additional constraint  $(\underline{u}_{\{ii\}+}, \underline{u}_{\{ii\}-}) = 0$ . Recall the definitions in (14)–(16). For a given  $x$ , the

optimality conditions for (13) subject to  $(\underline{u}_{\{ii\}+}, \underline{u}_{\{ii\}-}) = 0$  are

$$\begin{aligned} \bar{H} \chi + \bar{G}(x) + [\Omega_1, \Omega_2, -\Omega_2]^T \nu &= -\Pi^T \gamma \\ -[\Omega_1, 0, 0] \chi - s^* &= -\mathbf{1} + Mx \\ \Pi \chi &= 0 \\ (s^*)^T \nu &= 0 \\ s^* &\geq 0 \\ \nu &\geq 0 \\ \gamma &\geq 0. \end{aligned} \quad (22)$$

Considering (13), we have

$$\mathbb{X}_N^{\{i\}} = \pi_n \left( \left\{ (x, \chi) \mid \begin{array}{l} \bar{\Omega} \chi \leq b(x) \\ \Pi \chi = 0 \\ Lx \leq \mathbf{1} \end{array} \right\} \right), \quad (23)$$

where  $\pi_n$  is the polytope projection on the first  $n$ -coordinates [12], namely, on  $x$ . Notice that constructing these region explicitly can be computationally demanding, for large systems or long horizon length.

The following result is obtained

**Theorem IV.2.** The exact penalty weight for  $\|\underline{u}_{\{ii\}}\|_1 = \sum_{j=1}^{N-1} \|u_j^{\{ii\}}\|_1$ , providing  $\underline{u}_{\{ii\}} = 0, \forall x \in \mathbb{X}_N^{\{i\}} \cap \mathcal{X}_{\text{nom}}$ , is

$$\begin{aligned} \gamma^* &= \max_{\chi, \nu, \gamma, x} \|\gamma\|_\infty \\ \text{s.t. (22), } x &\in \mathbb{X} \cap \mathcal{X}_{\text{nom}}. \end{aligned} \quad (24)$$

**Proof IV.2.** Since the  $\infty$ -norm is the dual of the 1-norm, the exact penalty functions Theorem can be applied (Theorem 14.3.1 of [5]), by computing the maximum  $\infty$ -norm of the Lagrange multipliers associated with the constraint  $\|\underline{u}_{\{ii\}}\|_1 = 0$ . This corresponds to  $\Pi \chi = 0$ .

This leads to the desired result

**Theorem IV.3.** Assume  $S_{\{i\}} = 0, S_{\{ii\}} = \text{diag}\{s_i\}$ . Take  $\min_i s_i \geq \gamma^*$ , with  $\gamma^*$  solving (24). Then, for the  $\ell_{\text{asso}}$ -MPC solving (13) it follows that  $u^{\{ii\}} = 0, \forall x \in \mathbb{X}_N^{\{i\}} \cap \mathcal{X}_{\text{nom}}$ .

**Proof IV.3.** The Theorem is a simple application of the previous result. The  $\ell_{\text{asso}}$ -MPC is feasible, under the constraint  $\|\underline{u}_{\{ii\}}\|_1 = 0, \forall x \in \mathbb{X}_N^{\{i\}} \cap \mathcal{X}_{\text{nom}}$ . From Lemma IV.3,  $\gamma^*$  is the lower bound for the exact penalty on  $\|\underline{u}_{\{ii\}}\|_1$ , namely,  $\min_i s_i \geq \gamma^*$  provides the considered to have the same solution of (13) subject to  $\underline{u}_{\{ii\}} = 0, \forall x \in \mathbb{X}_N^{\{i\}} \cap \mathcal{X}_{\text{nom}}$ .

It can be noticed that conditions (22) provide the same solution of the LQ-MPC with only  $u^{\{i\}}$  available. We refer to this problem as the *pre-existing LQ-MPC*, since it can be designed a priori. Then, solving this smaller mpQP provides the active sets and the critical regions,  $\mathcal{P}_j$ , needed by the proposed procedure. These ingredients are finally used to compute the solution of (13), subject to  $(\underline{u}_{\{ii\}+}, \underline{u}_{\{ii\}-}) = 0$  and the Lagrange multipliers,  $\gamma, \forall x \in \mathcal{X}_{\text{nom}} \cap \mathbb{X}$ . In order to proceed further with the computation, define  $H^{\{ii,i\}} =$

$\Theta_2^T \mathbf{Q} \Theta_1$  and partition  $\Upsilon \Gamma$  into  $(F^{\{i\}}, F^{\{ii\}})$ , with

$$F^{\{i\}} = \sum_{i=0}^{N-1} B_{\{i\}}^T (A^i)^T Q_{i+1} A^{i+1}, \quad (25)$$

$$F^{\{ii\}} = \sum_{i=0}^{N-1} B_{\{ii\}}^T (A^i)^T Q_{i+1} A^{i+1}, \quad (26)$$

The following Linear Programme (LP) can be used to compute a candidate penalty,  $\bar{\gamma}_j$ , for each region  $\mathcal{R}_j = \mathcal{P}_j \cap \mathcal{X}_{\text{nom}} \neq \emptyset$ :

$$\begin{aligned} \bar{\gamma}_j &= \max_x \|2F^{\{ii\}} x + 2H^{\{ii,i\}} \underline{u}_{\{i\}}^*(x) + [\bar{\Omega}_2^T, 0, 0] v^*(x)\|_\infty \\ \text{s.t. } x &\in \mathcal{P}_j \cap \mathcal{X}_{\text{nom}}, \end{aligned} \quad (27)$$

where  $\bar{\Omega}_2$  (similarly  $\bar{\Omega}_1$ ) contains the rows of  $\Omega_2$ , from (16), corresponding to the active set in  $\mathcal{P}_j$ . The PWA functions are computed as

$$\begin{aligned} v^*(x) &= -(\Omega_{\text{act}} \bar{H}^{-1} \Omega_{\text{act}}^T)^{-1} (b_{\text{act}} \\ &\quad + (2\Omega_{\text{act}} \bar{H}^{-1} \bar{\Gamma} - M_{\text{act}}) x), \end{aligned} \quad (28)$$

$$\underline{u}^* = (\underline{u}_{\{i\}}^*, 0, 0) = -2\bar{H}^{-1} \bar{\Gamma} x - \bar{H}^{-1} \Omega_{\text{act}}^T v^*(x), \quad (29)$$

$$\Omega_{\text{act}} = \begin{bmatrix} \bar{\Omega}_1 & \bar{\Omega}_2 & -\bar{\Omega}_2 \\ & \Pi & \end{bmatrix}, \quad (30)$$

where  $\Pi$  is defined in (16) and  $\bar{H}, \bar{\Gamma}$  are from (15). The constant term  $b_{\text{act}}$  and the gain  $M_{\text{act}}$  are extrapolated from the rows of  $b(x)$ , which is also defined in (16), according to the indices of active set and of  $\Pi$ . Equivalently, one can directly solve for the multipliers, by solving the LP

$$\begin{aligned} \bar{\gamma}_j &= \max_x \|\Pi v^*(x)\|_\infty \\ \text{s.t. } x &\in \mathcal{P}_j \cap \mathcal{X}_{\text{nom}}, \end{aligned} \quad (31)$$

which avoids the computation of  $\underline{u}_{\{i\}}^*(x)$  and of several components of  $v^*(x)$ . The exact penalty is then obtained by taking the largest  $\bar{\gamma}_j$  over all feasible partitions.

The following result is obtained

**Theorem IV.4.** Consider the partition  $\text{part}\mathcal{P} = \{\mathcal{P}_j\}_{j=1}^{n_p}$  of the feasible region of (13) subject to  $(\underline{u}_{\{ii\}+}, \underline{u}_{\{ii\}-}) = 0$ , obtained from an mpQP solver, namely, with each  $\mathcal{P}_j$  corresponding to an optimal active set of the *pre-existing MPC*. Then, the regularisation penalty  $\gamma^*$ , satisfying Theorem IV.3, is given by

$$\begin{aligned} \gamma^* &= \max_j \bar{\gamma}_j, \\ \text{s.t. } \mathcal{P}_j &\in \text{part}\mathcal{P}, \\ \mathcal{P}_j \cap \mathcal{X}_{\text{nom}} &\neq \emptyset, \end{aligned} \quad (32)$$

where  $\bar{\gamma}_j$  is computed by, either, (31) or (27), subject to (28), (29), and (30).

**Proof IV.4.** Simar to [7], the Theorem can be verified by elaborating (22). Thanks to strict convexity ( $\bar{H} \succ 0$ ), from the first equation of (22) we can obtain (29). Then, substituting (29) into the second equality of (22), we can obtain (28), given the active set (since constraints are not degenerate). Then, again from the first line of (22) it can be easily verified that (31) or (27) are equivalent. Finally, in order to obtain the result, the multipliers are to be evaluated for each region  $\mathcal{P}_j$  for which  $\mathcal{P}_j \cap \mathcal{X}_{\text{nom}} \neq \emptyset$ .

### C. Example 2: Computation of the exact penalty

The proposed methodology is demonstrated on the following system

$$A = \begin{bmatrix} 0.2 & 0.1 \\ 0 & 1.1 \end{bmatrix}, B = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix},$$

$$\mathbb{I}^{(i)} = \{2\}, Q = I, R = I, S_{\{i\}} = 0, N = 3,$$

$$\mathbb{X} = \{x \mid \|x\|_{\infty} \leq 20\}, \mathbb{U} = \{u \mid \|u\|_{\infty} \leq 5\}. \quad (33)$$

The terminal controller,  $K^T = [0, K_{\{i\}}^T]$ , places the closed loop eigenvalues at 0.2, 0.9. and it is used to compute an invariant terminal set. The matrix  $P$  solves a Lyapunov equation for the system in closed-loop with  $K$ . All plots are made using the Matlab MPT toolbox [9]. The control law for the main actuator, in Figure 3, stabilises the system and resembles the one of a standard quadratic MPC. Figure 4

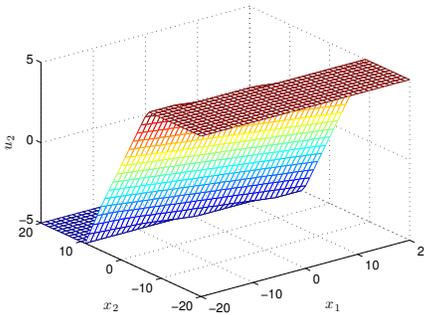


Fig. 3. Non-regularised actuator command as a function  $x$ .

shows the control law for the auxiliary actuator resulting from the procedure, with  $\mathcal{X}_{nom} = \{x \mid \pm x \leq \bar{x}\}$  for, respectively,  $\bar{x} = [5 \ 5]^T$  and  $\bar{x} = [10 \ 8]^T$ . In this case the algorithm explores only the first partition, where no constraints are active for the original problem. A few closed-loop trajectories are also shown, the origin being asymptotically stable.

### V. AIRCRAFT ROLL CONTROL

A 6-DOF model of a Boeing 747 [13] is linearised around different altitudes ( $z$  [m]) and values of airspeed ( $V_{TAS}$  [m/s]). The objective of the control design is to track a roll rate ( $p$ ) command from the pilot,  $p_{ref}$ , while keeping the magnitude of yaw rate ( $r$ ) and sideslip angle ( $\zeta$ ) moderate. The roll angle is  $\phi$ . A  $\ell_{asso}$ -MPC controller is designed using partial regularisation. The preferred actuators to perform the task are the ailerons ( $A1$ – $A4$ ), the upper and lower rudder ( $UR$ ,  $LR$ ). A nominal quadratic MPC is designed with actuators input magnitude and rate constraints. Then, the spoilers ( $S1$ – $S12$ ) are introduced in the design as auxiliary actuators, with the aim of helping out the ailerons when they are close to saturation or stall, at the possible expense of additional drag. The controller is tuned by trial and improvement.

The control is simulated on the linearisation at the operating point ( $z = 5000$ ,  $V_{TAS} = 180$ ). Figure 5 shows the

results for a moderate command (dash line). In particular,  $p$  and  $\phi$  are tracked with a good accuracy by means of the ailerons and the rudders. At the same time,  $\zeta$  and  $r$  are always less than 1 degree. Ailerons on opposite wings have opposite directions, and the rudders operate together. Noticeably, the spoilers are at zero for the whole time. A larger command

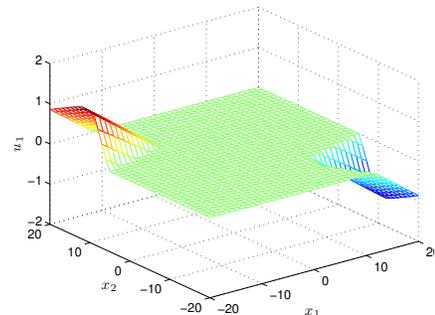
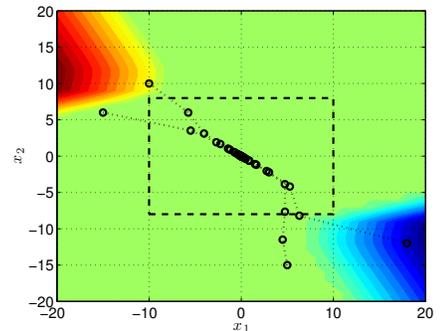
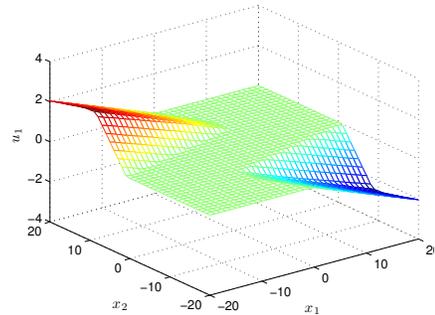
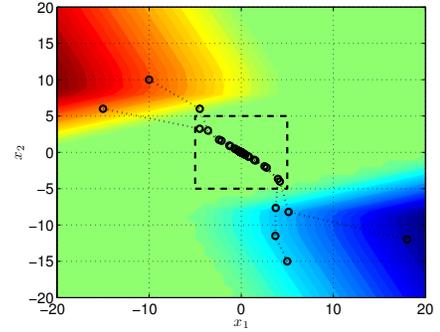


Fig. 4. Regularised actuator command as a function of  $x$  (above:  $\bar{x} = [5 \ 5]^T$ ,  $\gamma^* = 7.75$ , below:  $\bar{x} = [10 \ 8]^T$ ,  $\gamma^* = 13.07$ ).

is then applied, in Figure 6, which causes some spoilers to be set to operation. Note that the spoilers command is

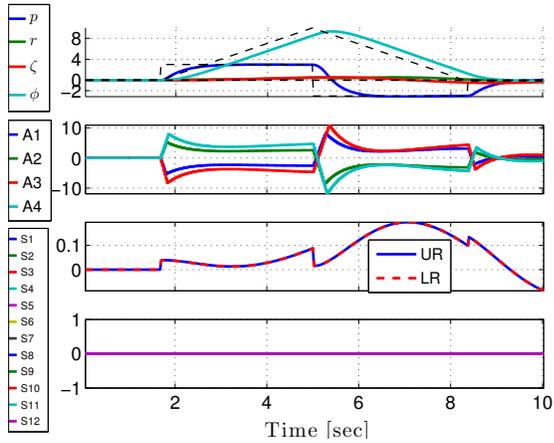


Fig. 5. Aircraft roll control. Nominal case. Moderate roll rate command ( $\pm 3$  [deg/s]).

small at the beginning, and slightly larger when the command changes in sign, confirming the control law is a continuous function. At the same time, when the control error becomes less than 5.5, the spoilers command is flat, and they literally disappear from the loop. The same behaviour can be noticed in Figure 7, with an even larger reference from the pilot. The spoilers commands are increased proportionally to the reference signal (roughly speaking, they are tripled), and a third peak appears at 8.5 seconds. Overall, the spoilers never extend for more than 15 degrees, and are not used for longer than 1.5 seconds, over a 10 seconds simulation.

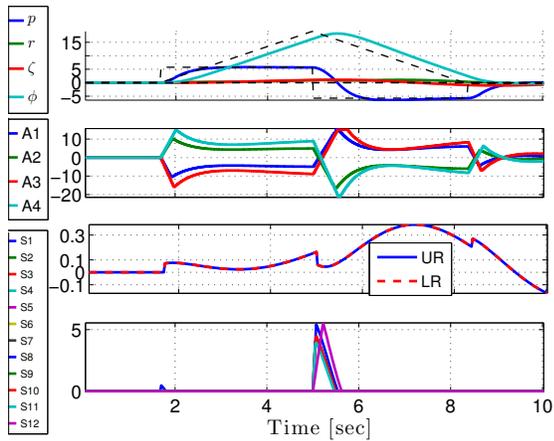


Fig. 6. Aircraft roll control. Nominal case. Large roll rate command ( $\pm 6$  [deg/s]).

## VI. SUMMARY

This paper presented a  $\ell_{asso}$ -MPC formulation for systems with preferred and auxiliary actuators. A design procedure has been proposed for automatic weights generation, so that the auxiliary actuators are not used when the control error is in a given neighbourhood of the origin. The procedure uses multi-parametric quadratic programming and the theory of exact penalty functions. Simulations of the control of a

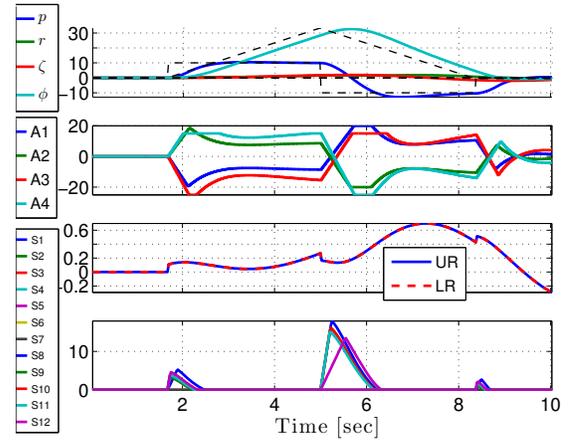


Fig. 7. Aircraft roll control. Nominal case. Very large roll rate command ( $\pm 10$  [deg/s]).

linearised Boeing 747 aircraft lateral dynamics have shown the proposed controller to be able to confine the spoiler activities to the case where the commands of the pilot are outside the nominal range. The study has shown that  $\ell_{asso}$ -MPC can be a powerful tool for this class of control problems.

## REFERENCES

- [1] H. Ohlsson, F. Gustafsson, L. Ljung, and S. Boyd, "Trajectory generation using sum-of-norms regularization," in *Conf. Decision & Control (CDC)*, 2010.
- [2] M. Nagahara and D. E. Quevedo, "Sparse representations for packetized predictive networked control," in *IFAC 18th World Congress*, Milan, Italy, 2011, pp. 84–89.
- [3] M. Gallieri and J. Maciejowski, " $\ell_{asso}$  MPC: Smart regulation of overactuated systems," in *Proc. of the American Control Conference*, Montréal, Canada, June 2012, p. 1217:1222.
- [4] —, "Stabilising terminal cost and terminal controller for  $\ell_{asso}$ -MPC: Enhanced optimality and region of attraction," in *Proc. European Control Conf. (ECC)*, Zurich., 2013, pp. 524 – 529.
- [5] R. Fletcher, *Practical Methods of Optimization*, 2nd edition. Wiley, 1987.
- [6] M. Gallieri, " $\ell_{asso}$ -MPC: Predictive Control with  $\ell_1$ -regularised Least Squares," Ph.D. dissertation, University of Cambridge, 2014.
- [7] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, pp. 3–20, 2002.
- [8] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, pp. 1747–1767, 1999.
- [9] M. Kvasnica, P. Grieder, and M. Baotić, "Multi-Parametric Toolbox (MPT)," 2004. [Online]. Available: <http://control.ee.ethz.ch/~mpt/>
- [10] J. L. Jerez, E. C. Kerrigan, and G. A. Constantinides, "A Condensed and Sparse QP Formulation for Predictive Control," in *Proc. Conf. on Decision & Control and Europ. Control (CDC-ECC)*, ser. 5217-5222, 2011.
- [11] E. C. Kerrigan and J. M. Maciejowski, "Soft constraints and exact penalty functions in model predictive control," in *Proc. UKACC International Conference (Control 2000)*, Cambridge, UK, September 2000.
- [12] E. Kerrigan and J. Maciejowski, "Invariant sets for constrained nonlinear discrete-time systems with application to feasibility in model predictive control," in *Proceedings of the 39th IEEE Conference on Decision and Control.*, vol. 5, 2000, p. 4951:4956.
- [13] C. Edwards, T. Lombaert, and S. Hafid, *Fault Tolerant Flight Control A Benchmark Challenge*. Springer-Verlag Berlin Heidelberg, 2010.