International Journal of Control, 2014
Vol. 87, No. 12, 2583–2593, http://dx.doi.org/10.1080/00207179.2014.935958

Taylor & Francis
Taylor & Francis Group

Robust explicit model predictive control via regular piecewise-affine approximation

Matteo Rubagotti[a,*], Davide Barcelli[b] and Alberto Bemporad[c]

[a]*Department of Robotics, Nazarbayev University, Kabanbay Batyr Ave 56, 010000 Astana, Kazakhstan;* [b]*Department of Information Engineering, University of Siena, Via Roma 56, 53100 Siena, Italy;* [c]*IMT Institute for Advanced Studies, Piazza S. Ponziano 6, 55100 Lucca, Italy*

(*Received 9 September 2013; accepted 13 June 2014*)

This paper proposes an explicit model predictive control design approach for regulation of linear time-invariant systems subject to both state and control constraints, in the presence of additive disturbances. The proposed control law is implemented as a piecewise-affine function defined on a regular simplicial partition, and has two main positive features. First, the regularity of the simplicial partition allows one to efficiently implement the control law on digital circuits, thus achieving extremely fast computation times. Moreover, the asymptotic stability (or the convergence to a set including the origin) of the closed-loop system can be enforced *a priori*, rather than checked *a posteriori* via Lyapunov analysis.

Keywords: model predictive control; uncertain systems; piecewise-affine functions

1. Introduction

Model predictive control (MPC) is becoming increasingly popular both in academia and in industry due to its ability to solve control problems while satisfying constraints on state and control variables (Rawlings & Mayne, 2009). The main drawback of MPC is the computation time required for solving an optimisation problem on line, which has historically prevented its application to fast processes. To circumvent this problem, two main research directions were pursued in the last decade (we limit our overview to the control of linear time-invariant (LTI) systems that are the subject of this paper). The first relates to fast algorithms for online optimisation (Ferreau, Bock, & Diehl, 2008; Patrinos & Bemporad, 2014; Richter, Morari, & Jones, 2011; Rubagotti, Patrinos, & Bemporad, 2014; Wang & Boyd, 2010). The second regards computing the control law offline as an explicit piecewise-affine (PWA) function of the state vector (Bemporad, Morari, Dua, & Pistikopoulos, 2002): the offline computation employs a multi-parametric programming solver, and leads to the same solution obtained by solving the optimisation problem online. The online computation in explicit MPC relies on determining the region of the PWA partition where the current state is located (usually referred to as the *point location* problem, which typically takes a high percentage of the overall online computation time), and then on evaluating an affine function from a pre-stored lookup table. To simplify the complexity of explicit MPC controllers, *approximate* explicit MPC, in which optimality is sacrificed for a control law defined over a smaller number of regions, has been considered in the last decade (see, e.g., Grieder, Kvasnica, Baotić, & Morari, 2005; Jones & Morari, 2010; Kvasnica & Fikar, 2012; Kvasnica, Löfberg, & Fikar, 2011, and the references therein).

In a recent work (Bemporad, Oliveri, Poggi, & Storace, 2011), an approximate MPC controller for LTI systems was proposed, based on a special class of functions, hereafter referred to as *piecewise-affine simplicial (PWAS) functions*, proposed by Julián, Desages, and D'Amico (2000). The choice of PWAS functions leads to a regular partition, so that the point-location problem is solved with a negligible effort compared to explicit MPC defined on generic PWA partitions (the reader is also referred to Oliveri et al., 2012, for the practical implementation). The control law proposed by Bemporad et al. (2011) presents feasibility and local optimality properties, but the asymptotic stability of the origin of the closed-loop system and the evaluation of its domain of attraction can be determined only *a posteriori* (see, e.g., Rubagotti, Trimboli, & Bemporad, 2013, and the references therein). We would like to remark that PWAS functions are not the only choice for approximation of explicit MPC aimed at hardware implementation: for example, two different approaches based on the use of PWA hyper-rectangular partitions have been recently proposed by Genuit, Lu, and Heemels (2011) and Lu, Heemels, and Bemporad (2011). In all of these approaches (Bemporad et al., 2011; Genuit et al., 2011; Lu et al., 2011), the possible presence of disturbance terms acting on the system is not taken into account. Note that all the proposed techniques for approximation of explicit MPC lead to a reduction of the computation time,

*Corresponding author. Email: matteo.rubagotti@nu.edu.kz

but are applicable only to relatively small-sized problems, which is an inherent limitation of explicit MPC.

In this paper, we propose an approximation method for explicit MPC based on PWAS functions, which can be implemented on digital circuits as in Bemporad et al. (2011). However, in addition to that, we guarantee *a priori* the convergence to a minimal set including the origin for the resulting closed-loop system (also obtaining the domain of attraction in which hard constraints on state and input variables are satisfied), in the presence of external disturbances.

More specifically, two different methods are hereafter proposed to design a robust MPC control law $u^*(x)$, based on tightened constraints: an approximation procedure is carried out, in order to find an approximate PWAS control law $u(x)$, such that the approximation error $u(x) - u^*(x)$ satisfies the previously defined bounds. As a drawback, $u^*(x)$ must be explicitly computed in order to obtain $u(x)$. Also, the proposed method, like all explicit MPC techniques, can only be applied to small-sized problems, due to the exponential increase of the problem complexity as the prediction horizon or the number of states/inputs increases. However, we can obtain a considerable decrease in the time needed to compute the control law if compared to directly applying $u^*(x)$, mainly due to the strong simplification of the point-location problem. A preliminary version of the theoretical development in this paper is presented in Rubagotti, Barcelli, and Bemporad (2012), where one of the two synthesis methods considered here is proposed in the case of systems without disturbances.

The paper is organised as follows. The main notation used throughout the paper and the formulation of the control problem are introduced in Sections 2 and 3, respectively, while Section 4 describes the structure of the PWAS control law. In Section 5, the synthesis of the robustly stabilising MPC control law is described, while Section 6 deals with the approximation procedure leading to the stabilising PWAS control law. In Section 7, two simulation examples are shown. Finally, conclusions are drawn in Section 8.

2. Notation

Let $\mathbb{Z}_{>0}$, $\mathbb{Z}_{\geq 0}$, $\mathbb{R}$, and $\mathbb{R}_{>0}$ denote the sets of positive integers, non-negative integers, real, and positive real numbers, respectively. Given a set $\mathcal{A} \subset \mathbb{R}^n$, its interior is referred to as int($\mathcal{A}$). Given two sets $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A} \oplus \mathcal{B} \triangleq \{a + b \,:\, a \in \mathcal{A},\ b \in \mathcal{B}\}$ and $\mathcal{A} \sim \mathcal{B} = \{a \,:\, a + b \in \mathcal{A},\ \forall b \in \mathcal{B}\}$ are their Minkowski addition and Pontryagin difference, respectively. Also, given $\lambda \in \mathbb{R}_{\geq 0}$, we define $\lambda\mathcal{A} \triangleq \{x \in \mathbb{R}^n \,:\, x = \lambda a,\ a \in \mathcal{A}\}$. We denote by $\|v\|_1$ and $\|v\|_\infty$ the 1-norm and the ∞-norm of v, respectively. Given two vectors $u, v \in \mathbb{R}^n$, the notation $u \leq v$ refers to componentwise inequalities. Given a square matrix $H \in \mathbb{R}^{n\times n}$, its trace is tr($H$), its Cholesky factor is $H^{\frac{1}{2}}$, and its positive definiteness is referred to as $H \succ 0$. The symbol $\mathbb{I}_n$ represents the identity matrix in $\mathbb{R}^{n\times n}$. Given a vector $v \in \mathbb{R}^n$ and a matrix $H \in \mathbb{R}^{n\times n}$, $\|v\|_M^2 \triangleq v'Mv$. Given a matrix $H \in \mathbb{R}^{n\times m}$ and a compact set $\mathcal{W} \subset \mathbb{R}^m$, the product $H\mathcal{W}$ denotes the image of $\mathcal{W}$ under the mapping defined by H, $H\mathcal{W} \triangleq \{v \in \mathbb{R}^n \,:\, v = Hw,\ \forall\, w \in \mathcal{W}\}$. When convenient, the explicit dependence on time of the dynamic variables will be omitted for the sake of readability.

3. Problem statement

The controlled plant is described by the following discrete-time LTI state space model:

$$x(t+1) = Ax(t) + Bu(t) + d(t), \tag{1}$$

where $t \in \mathbb{Z}_{\geq 0}$, $x, d \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$. The whole state vector x is available for feedback, while u and d represent the control input and an unknown and unmeasurable disturbance term, respectively. The state and input values are required to satisfy

$$x \in \mathcal{X},\ \mathcal{X} \triangleq \{x \in \mathbb{R}^n \,:\, C_x x \leq g_x\} \tag{2}$$

$$u \in \mathcal{U}, \mathcal{U} \triangleq \{u \in \mathbb{R}^m \,:\, C_u u \leq g_u\} \tag{3}$$

with $C_x \in \mathbb{R}^{s_x\times n}$, $C_u \in \mathbb{R}^{s_u\times m}$, $g_x \in \mathbb{R}^{s_x}$, $g_u \in \mathbb{R}^{s_u}$, while the disturbance term is assumed to be such that

$$d \in \mathcal{D},\ \mathcal{D} \triangleq \{d \in \mathbb{R}^n \,:\, C_d d \leq g_d\} \tag{4}$$

with $C_d \in \mathbb{R}^{s_d\times n}$, $g_d \in \mathbb{R}^{s_d}$.

Assumption 3.1: *The following holds for system* (1)*:*

(*i*) *The pair* (A, B) *is stabilisable.*
(*ii*) $\mathcal{X}$ *and* $\mathcal{U}$ *are non-empty, compact, and contain the origin in their interiors.*
(*iii*) $\mathcal{D}$ *is non-empty, compact, and contains the origin.*

The objective of the control law is to solve a regulation problem to the smallest possible set containing the origin, without violating the constraints (2) and (3). The control variable $u(x)$ is a state-feedback control law defined on a PWAS partition, whose structure is described in the next section.

4. Control law on a simplicial partition

The function $u(x)$ is defined on a closed hyper-rectangle $\mathcal{S} = \{x \in \mathbb{R}^n :\ x_{\min} \leq x \leq x_{\max}\}$, which is partitioned as $\mathcal{S} = \bigcup_{i=0}^{L_S-1} S_i$, where $\{S_i\}_{i=0}^{L-1}$ are simplices, i.e., polytopes given by the convex hull of their $n+1$ vertices $x_i^0, x_i^1, \ldots, x_i^n \in \mathbb{R}^n$. The partitioning of $\mathcal{S}$ is performed as follows:

(1) Every dimensional component x_j of $\mathcal{S}$ is divided into p_j subintervals of length $(x_{\max,j} - x_{\min,j})/p_j$. These intervals define a number $\prod_{j=1}^{n} p_j$ of hyper-rectangles, and $\mathcal{S}$ contains $N_v \triangleq \prod_{j=1}^{n}(p_j + 1)$ vertices v_k, collected into a set named $\mathcal{V}_S$.
(2) Every rectangle is partitioned into $n!$ simplices with non-overlapping interiors. The set $\mathcal{S}$ contains $L_S \triangleq n! \prod_{j=1}^{n} p_j$ simplices S_i, such that $\mathcal{S} = \cup_{i=0}^{L_S-1} S_i$ and $\text{int}(S_i) \cap \text{int}(S_j) = \emptyset, \forall i, j = 0, \ldots, L_S - 1$.

Note that, since the partitioning of the hyper-rectangles into simplices is univocally determined, the resulting number of simplices is determined by $p_1, \ldots, p_n$. After defining the sets S_i, it is possible to introduce the related PWAS function. We choose to define each component of $u(x)$, namely $u_j(x)$, $j = 1, \ldots, m$, as the weighted sum of N_v linearly independent α-basis functions (Julián et al., 2000). Every element of the jth basis is affine over each simplex and satisfies

$$\alpha_{j,k}(v_h) = \begin{cases} 1 & \text{if } h = k \\ 0 & \text{if } h \neq k. \end{cases}$$

After ordering the functions of the α-basis, we can consider them as an N_v-length vector $\phi(x)$. Then, each component of $u(x)$, namely $u_j(x)$, is a scalar PWAS function defined as

$$u_j(x) \triangleq \sum_{k=1}^{N_v} \theta_{j,k}\, \phi_k(x) = \phi(x)'\theta_j, \tag{5}$$

where $\theta_j = [\theta_{j,1} \ \ldots \ \theta_{j,N_v}]' \in \mathbb{R}^{N_v}$ is the weight vector. Note that the coefficients $\theta_{j,k}$ coincide with the values of the PWAS function $u_j(x)$ at the vertices of the simplicial partition. The PWAS vector function $u : \mathbb{R}^n \to \mathbb{R}^m$ is defined by the weight vector $\theta = [\theta_1' \ \theta_2' \ \ldots \ \theta_m']' \in \mathbb{R}^{mN_v}$, as

$$u(x) = \begin{bmatrix} u_1(x) \\ \vdots \\ u_m(x) \end{bmatrix} \triangleq \begin{bmatrix} \phi(x)'\theta_1 \\ \vdots \\ \phi(x)'\theta_m \end{bmatrix} = \begin{bmatrix} \phi'(x) & 0 & \cdots & 0 \\ 0 & \phi'(x) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \phi'(x) \end{bmatrix} \theta = \Phi(x)\theta. \tag{6}$$

The main reason for defining $u(x)$ as in Equation (6) is that PWAS functions can be implemented in digital circuits using linear interpolators. In fact, by exploiting the regularity of the partition, the point-location problem becomes much easier than for the case of generic PWA partitions. The value of $u(x)$ can be obtained, for any $x \in \mathcal{S}$, as a linear interpolation of the values of u at the $n + 1$ vertices $x_{i,0}, \ldots, x_{i,n}$ of the simplex S_i that contains x. For a summary of the actual field programmable gate array (FPGA) implementation (also employed to design virtual sensors in Poggi, Rubagotti, Bemporad, & Storace, 2012), the interested reader is referred to Storace and Poggi (2010).

5. Robustly stabilising optimal MPC

The next step is to obtain a function $u(x)$ as in Equation (6) using a procedure that leads to asymptotic convergence to a set containing the origin for the closed-loop system. The proposed approach consists of expressing the control variable $u(x)$ as

$$u(x) = u^*(x) + w(x),$$

where $u^*(x)$ is an optimal control law which satisfies

$$u^* \in \mathcal{U}, \tag{7}$$

while $w(x)$ represents an approximation error (*a-priori* unknown), and is considered as a bounded disturbance. System (1) can therefore be expressed as

$$x(t+1) = Ax(t) + Bu^*(t) + Bw(t) + d(t). \tag{8}$$

5.1 *Definition of the auxiliary control laws*

In order to formulate the MPC control law $u^*(x)$, we first need to define an auxiliary control law, for which we introduce two alternative choices.

The first control law is synthesised on the nominal system as follows.

Statement 5.1: The auxiliary control law is defined as $u^*(x) = K_n x$, where K_n is the solution for the nominal system

$$x(t+1) = Ax(t) + Bu^*(t) \tag{9}$$

of the infinite-horizon linear quadratic regulator (IH-LQR), given the weight matrices $Q = Q' \in \mathbb{R}^{n\times n}$ on the state and $R = R' \in \mathbb{R}^{m\times m}$ on the input, with $Q, R \succ 0$.

Remark 1: Note that, by classical results of LQR theory, the closed-loop system obtained by imposing $u^*(x) = K_n x$ in Equation (9) is asymptotically stable.

The second choice concerns an auxiliary control law which is robustly stabilising for

$$x(t+1) = Ax(t) + B\left(u^*(t) + w(x(t))\right), \tag{10}$$

where each component w_i of w is such that

$$|w_i(x)| \leq \alpha \|x\|_1, \ i = 1, \ldots, m, \tag{11}$$

and α is a tuning parameter. This formulation of the uncertainty can be shown to be a *structured feedback uncertainty*, as in Kothare, Balakrishnan, and Morari

(1996). To this purpose, let $\mathbf{1} \in \mathbb{R}^{m\times n}$ be a matrix of ones, and $\Delta = \text{diag}(\delta_1, \delta_2, \dots, \delta_n)$ be a matrix of uncertain parameters such that $|\delta_i| \le 1$ for all $i \in 1, \dots, n$. Then, Equation (11) can be equivalently formulated as $w(x) = \alpha\mathbf{1}\Delta x$. More precisely, this latter expression is equivalent to $w_i = \alpha(\sum_{j=1}^{n} \delta_j x_j)$ for all $i = 1, \dots, m$, which leads to Equation (11).

Statement 5.2: The auxiliary control law is defined as $u^*(x) = K_p x$, where $K_p = Y\Pi^{-1}$, $\Pi = \Pi' \succ 0$ and Y are the solution of the following semi-definite programme:

$$\min_{\gamma,\Lambda,\Pi,Y} \quad \gamma \tag{12a}$$

$$\text{s.t.} \quad \Lambda > 0 \tag{12b}$$

$$\text{tr}(\Pi) = 1 \tag{12c}$$

$$\begin{bmatrix} \Pi & Y'R^{\frac{1}{2}} & \Pi Q^{\frac{1}{2}} & \Pi & (A\Pi + BY)' \\ R^{\frac{1}{2}}Y & \gamma I & 0 & 0 & 0 \\ Q^{\frac{1}{2}}\Pi & 0 & \gamma I & 0 & 0 \\ \Pi & 0 & 0 & \Lambda & 0 \\ A\Pi + BY & 0 & 0 & 0 & \Pi - B_p\Lambda B_p' \end{bmatrix} \succeq 0 \tag{12d}$$

where $B_p \triangleq \alpha B\mathbf{1}$ and $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$.

Remark 2: The control law obtained in Statement 5.2 is related to the result in Theorem 1 in Kothare et al. (1996). While in Kothare et al. (1996), a semi-definite programme is solved online, we here fix the gain K_p offline. Also, as $\alpha \to 0$ (i.e., the dynamics of the uncertain system (10) approaches the nominal dynamics (9)), the gain K_p tends to the gain K_n (with the same weight matrices Q, R) defined in Statement 5.1 (Kothare et al., 1996, Remark 4). Every possible realisation of matrix $A + B(K_p + \alpha\mathbf{1}\Delta)$ has eigenvalues strictly inside the unit circle, which means that the closed-loop system (10) with $u^* = K_p x$ is *absolutely asymptotically stable* as defined by Gurvits (1995).

In conclusion, two different auxiliary control laws have been defined, both assuming no external disturbance d, the first assuming no approximation error w, and the second assuming that w is vanishing as x approaches the origin as described in Equation (11). By defining either $K \triangleq K_n$ or $K \triangleq K_p$, the resulting closed-loop system is

$$x(t+1) = A_\kappa x(t) + Bw(t) + d(t), \tag{13}$$

where $A_\kappa \triangleq A + BK$. Both of these control laws will be used in the remainder of the paper as baseline to design the MPC controller.

5.2 *Preliminary concepts for the definition of the MPC control law*

A robust MPC control law is now described, which leads to robust convergence of the state to the origin without violating constraints (2) and (7). Let $\bar{w} \in \mathbb{R}_{\ge 0}$ be a fixed scalar such that

$$w(x) \in \mathcal{W} \triangleq \{w \in \mathbb{R}^m : \|w\|_\infty \le \bar{w}\}, \ \forall x \in \mathcal{X}, \tag{14}$$

which represents a requirement on the maximum approximation error. At this point, two additive disturbances are present in the system. We define

$$\xi(t) \triangleq Bw(t) + d(t)$$

from which it follows that $\xi \in \Xi = B\mathcal{W} \oplus \mathcal{D}$, and rewrite system (8) as

$$x(t+1) = Ax(t) + Bu^*(t) + \xi(t). \tag{15}$$

The following standard definition is used as follows.

Definition 5.3: A set $\mathcal{P}$ is *robust positively invariant* (RPI) for system (15), if $x(0) \in \mathcal{P}$ implies $x(t) \in \mathcal{P}$ for all $\xi(t) \in \Xi$ and for all $t \in \mathbb{Z}_{\ge 0}$.

First of all, we define

$$\mathcal{R}_k \triangleq \bigoplus_{i=0}^{k-1} A_\kappa^i \Xi, \tag{16}$$

which is the set of states reachable by system (13) in k time steps from the origin. Then, we compute the minimal RPI set $\mathcal{R}_\infty$ for the closed-loop system (13), assuming that Equation (14) holds. The minimal RPI set for system (13) with $\xi \in \Xi$ is defined as $\mathcal{R}_\infty \triangleq \lim_{k\to\infty} \mathcal{R}_k$. Considering that this set can only be computed exactly under very restrictive assumptions, one usually needs to compute a polytopic over-approximation (not necessarily RPI) $\hat{\mathcal{R}}_\infty$ such that $\mathcal{R}_\infty \subseteq \hat{\mathcal{R}}_\infty$. Details on the characterisation and computation of $\mathcal{R}_\infty$ and $\hat{\mathcal{R}}_\infty$ as compact polytopes are given in Appendix 1.

Referring to a generic gain K, which can be determined equal to K_n or to K_p, let the MPC control law acting on system (1) be

$$u^*(x) \triangleq Kx + \mu^*(x), \tag{17}$$

so that Equation (1) becomes $x(t+1) = A_\kappa x(t) + B\mu^*(t) + \xi(t)$. Note from Equation (17) that $\mu^*(x)$ represents the difference between the MPC control move and the baseline linear control law Kx. In the following, we will make use of tightened constraints on the nominal evolution of Equation (15) to ensure the fulfilment of the actual constraints for the perturbed system. Starting from the initial condition $x(t) = x$ at time t, the nominal evolution of Equation (15) at time $t + k$ is denoted by $\hat{x}(t+k|t)$, while the evolution of the actual system with the same initial condition by $x(t + k|t)$. Both evolutions are obtained by applying the corresponding control sequence denoted by $\mu^*(t|t), \ldots, \mu^*(t+k-1|t)$. It is well known from the set-theoretical analysis in Chisci, Rossiter, and Zappa (2001) and Kolmanovsky and Gilbert (1998) that, given $\mathcal{X}_k \triangleq \mathcal{X} \sim \mathcal{R}_k$ and $\mathcal{U}_k \triangleq \mathcal{U} \sim K\mathcal{R}_k$, one has that, for all $k \in \mathbb{Z}_{\geq 0}$, $\hat{x}(t+k|t) \in \mathcal{X}_k \Leftrightarrow x(t+k|t) \in \mathcal{X}$ and $K\hat{x}(t+k|t) \in \mathcal{U}_k \Leftrightarrow Kx(t+k|t) \in \mathcal{U}$, for all $\xi \in \Xi$.

The next step is to find the maximal output admissible robust set for system (13), defined as

$$\mathcal{X}_f \triangleq \{x(0) \in \mathbb{R}^n : x(k|0) \in \mathcal{X},\ Kx(k|0) \in \mathcal{U},\ \forall k \in \mathbb{Z}_{\geq 0},\ \forall \xi \in \Xi\}. \tag{18}$$

Details on the computation of $\mathcal{X}_f$ as a compact polytope are given in Appendix 1.

Assumption 5.4: *It is supposed that* $0 \in \text{int}(\mathcal{X} \sim \mathcal{R}_\infty)$ *and* $0 \in \text{int}(\mathcal{U} \sim K\mathcal{R}_\infty)$ (*which ensures the computability of* $\mathcal{X}_f$*, see Appendix* 1). *Moreover, we assume that* $\hat{\mathcal{R}}_\infty \subset \text{int}(\mathcal{X}_f)$.

Remark 3: The condition $\hat{\mathcal{R}}_\infty \subset \text{int}(\mathcal{X}_f)$ represents only a slightly stronger requirement with respect to condition $\mathcal{R}_\infty \subseteq \mathcal{X}_f$, which always holds. Note that, if $\hat{\mathcal{R}}_\infty \subset \text{int}(\mathcal{X}_f)$, being $\hat{\mathcal{R}}_\infty$ a closed set, any state trajectory that converges to $\hat{\mathcal{R}}_\infty$ asymptotically, converges to $\mathcal{X}_f$ in finite time.

Recalling the sets S_i defined in Section 4, we introduce the set

$$\mathcal{S}_f \triangleq \bigcup S_i \ :\ S_i \subseteq \mathcal{X}_f,\ i = 0, \ldots, L-1, \tag{19}$$

which will be useful to formulate the subsequent results. Being $\mathcal{X}_f$ a convex set, $\mathcal{S}_f$ is connected, but not necessarily convex.

5.3 *MPC with tightened constraints*

For the proposed robust MPC strategies, the prediction of the system trajectory on the finite *prediction horizon* $N \in \mathbb{Z}_{>0}$ will make use of the nominal model of the system and of tightened constraints, as in Chisci et al. (2001). The vector of optimisation variables (inputs) to be determined at time t is $M \triangleq [\mu'(t|t)\ \ldots\ \mu'(t|t+N-1)]' \in \mathbb{R}^{mN}$. The definition of the optimal sequence $\mu^*(x)$ is based on the solution of the following finite-horizon optimal control problem (FHOCP) at each time t, with $x(t) = x$:

$$M^*(x) = \arg\min_{M} \sum_{k=0}^{N-1} \|\mu(k)\|^2_\Psi,\ \Psi = \Psi' \succ 0 \tag{20a}$$

$$\text{s.t. } \hat{x}(k) \in \mathcal{X}_k, \qquad k = 0, \ldots, N-1 \tag{20b}$$

$$K\hat{x}(k) + \mu(k) \in \mathcal{U}_k,\ k = 0, \ldots, N-1 \tag{20c}$$

$$\hat{x}(N) \in \mathcal{X}_f \sim \mathcal{R}_N. \tag{20d}$$

For ease of notation, implying that the solution of the FHOCP is computed at time t, we set $\mu(k) \triangleq \mu(t + k|t)$ and $\hat{x}(k) \triangleq \hat{x}(t+k|t)$. Note that Equations (20b) and (20c) lead to the fulfilment of Equations (2) and (7), respectively, along the prediction horizon. Finally, Equation (20d) guarantees that $x(t + k|t) \in \mathcal{X}_f$ for all possible disturbance sequences.

The FHOCP (20) is quadratic with respect to the decision variable M, and is subject to linear constraints. Also, the current state x can be considered as a parameter. Therefore, Problem (20) can be recast as a multi-parametric quadratic programme (mpQP), where the set of parameters x, for which a feasible solution exists, is called $\mathcal{F}_N$. Since $\mathcal{X}$, $\mathcal{U}$, and Ξ are convex polyhedra, $\mathcal{F}_N$ is a convex polytope and can be easily computed using linear programming and projections (Chisci et al., 2001). Also, an increase of the prediction horizon leads to a larger set $\mathcal{F}_N$, i.e. $\mathcal{F}_N \supseteq \mathcal{F}_{N-1} \supseteq \cdots \supseteq \mathcal{F}_1 \supseteq \mathcal{X}_f$. The nominal case (i.e., $\Xi = 0$) can be seen as a limit of the robust case, and $\mathcal{F}_N$ is always included in the corresponding set obtained for the nominal case.

Recalling Remark 3 in Chisci et al. (2001), matrix Ψ can be chosen such that Problem (20) coincides with the solution of the constrained IH-LQR associated to the weight matrices Q and R.

The application of the *receding horizon* principle leads to defining the MPC control law $\mu^*(x)$ as $\mu^*(x) \triangleq [I\ 0\ \ldots\ 0]M^*(x)$. Following the development in Bemporad et al. (2002), explicit expressions for the optimal value of the cost function in Equation (20a), namely $J^*(x)$, and for $M^*(x)$, can be obtained solving an mpQP. In particular, both $J^*(x)$ and $M^*(x)$ are Lipschitz continuous, and more precisely $J^*(x)$ is piecewise-quadratic, while $M^*(x)$ is PWA. This also implies that $\mu^*(x)$ and $u^*(x)$ are PWA functions defined in $\mathcal{F}_N$. The set $\mathcal{F}_N$ is then partitioned as $\mathcal{F}_N = \bigcup_{i=0}^{L_F-1} F_i$, where $\{F_i\}_{i=0}^{L_F-1}$ are polytopes (not necessarily simplices) with non-overlapping interiors.

Next, define the following two sets: $\mathcal{R}^n_\infty$, the minimal RPI set for the closed-loop system

$$x(t+1) = (A + BK_n)x(t) + d(t), \tag{21}$$

and $\mathcal{R}^p_\infty$, the minimal RPI set for the closed-loop system

$$x(t+1) = (A + BK_p + \alpha B\mathbf{1}\Delta)x(t) + d(t). \tag{22}$$

In both cases, the presence of the disturbance $w(t)$ is not taken into account. The computation of sets $\mathcal{R}^n_\infty$ and $\mathcal{R}^p_\infty$, and of their over-approximations $\hat{\mathcal{R}}^n_\infty$ and $\hat{\mathcal{R}}^p_\infty$ are described in Appendix 1.

We are now ready to state the first main result of the paper.

Theorem 5.5: *Let Assumptions 3.1 and 5.4 hold for system* (8) *with $\xi \in \Xi$, and let $u^*(x)$ be defined in Equation* (17)*.*

(*I*) *Let the MPC control law in Equation* (17) *be designed with $K = K_n$, $w(x)$ such that*

$$w(x) = 0, \ \forall\, x \in \mathcal{S}_f, \tag{23}$$

with $0 \in \text{int}(\mathcal{S}_f)$ (the latter being defined in Equation (19)*), and $\hat{\mathcal{R}}_\infty \subset \text{int}(\mathcal{S}_f)$. Then, for all possible realisations of the disturbance term $d(t)$, if $x(0) \in \mathcal{F}_N$, then $x(t) \in \mathcal{X}$ and $u^*(t) \in \mathcal{U}$ for all $t \geq 0$, and moreover, $x(t) \to \mathcal{R}^n_\infty \subseteq \mathcal{R}_\infty$ as $t \to \infty$.*

(*II*) *Let the MPC control law in Equation* (17) *be designed with $K = K_p$ for a given $\alpha > 0$. Moreover,*

$$|w_i(x)| \leq \alpha \|x\|_1, \ \forall i = 1, \ldots, m, \ \forall x \in \mathcal{X}_f, \tag{24}$$

i.e., condition (11) *be satisfied for all $x \in \mathcal{X}_f$. Then, for all possible realisations of the disturbance term $d(t)$, if $x(0) \in \mathcal{F}_N$, then $x(t) \in \mathcal{X}$ and $u^*(t) \in \mathcal{U}$ for all $t \geq 0$, and moreover, $x(t) \to \mathcal{R}^p_\infty \cap \mathcal{R}_\infty$ as $t \to \infty$.*

In both cases, if $\mathcal{D} = \{0\}$, then $\mathcal{R}^n_\infty = \mathcal{R}^p_\infty = \{0\}$, i.e., the origin is an asymptotically stable equilibrium for system (1)*, with domain of attraction $\mathcal{F}_N$.*

Proof. See Appendix 2.1. □

6. PWAS approximation

In this section, we describe how to obtain the control law $u(x)$ defined on a PWAS partition as in Equation (6) approximating the control law $u^*(x)$ in Equation (17), in order to obtain asymptotic stability and constraint satisfaction for system (1).

6.1 *Approximation procedure*

Assume that a control law $u^*(x)$ has been computed for system (1) with domain of attraction $\mathcal{F}_N$. Let $\mathcal{S}$ be defined as the smallest hyper-rectangle such that $\mathcal{F}_N \subseteq \mathcal{S}$, as described in Section 4. Then, we partition the (not necessarily convex) set $\mathcal{S} \setminus \mathcal{F}_N$ as $\mathcal{S} \setminus \mathcal{F}_N = \bigcup_{i=0}^{\hat{L}_F - 1} F_i$, where $\{F_i\}_{i=0}^{\hat{L}_F - 1}$ are polytopes with non-overlapping interiors. In this way, a generic partition of $\mathcal{S}$ as $\mathcal{S} = \bigcup_{i=0}^{\tilde{L}_F - 1} F_i$ is obtained, where $\tilde{L}_F \triangleq L_F + \hat{L}_F$, while we denote its set of vertices as $\tilde{\mathcal{V}}_F$. In order to introduce the used approximation procedure, we use the concept of *mixed partition* (see, e.g., Bemporad et al., 2011) as the partition of $\mathcal{S}$ induced by the facets of both simplicial (S_i) and generic (F_i) partitions. As a result, $\mathcal{S}$ is further partitioned into convex polytopes, and the partition is completely defined by the sets of vertices $\mathcal{V}_S$, $\tilde{\mathcal{V}}_F$, and $\mathcal{V}_M$, the latter representing the set of vertices given by the intersection of the two partitions and belonging neither to $\mathcal{V}_S$ nor to $\tilde{\mathcal{V}}_F$. Finally, let $\mathcal{V}_I \triangleq \{v \in (\mathcal{V}_S \cup \tilde{\mathcal{V}}_F \cup \mathcal{V}_M) \ : \ v \in \mathcal{F}_N\}$, and note that $\mathcal{F}_N$ is the convex hull of all $v \in \mathcal{V}_I$.

Let $u(x)$ be defined as the control law that minimises the maximum discrepancy with respect to $u^*(x)$ for all $x \in \mathcal{F}_N$ (note that $u^*(x)$ is not defined on $\mathcal{S} \setminus \mathcal{F}_N$), i.e.,

$$F_\infty \triangleq \max_{j=1,\ldots,m} \sup_{x \in \mathcal{F}_N} \left\{ \left| u_j(x) - u_j^*(x) \right| \right\}. \tag{25}$$

When minimising F_∞ in Equation (25), some constraints have to be imposed for all $x \in \mathcal{F}_N$. Since the minima and maxima of the PWA function $w(x) = u(x) - u^*(x)$ on any of the regions of the mixed partition are on vertices, it is sufficient to impose constraints only on the vertices of $\mathcal{V}_I$. In particular:

(1) The control law $u(x)$ must satisfy the constraint (3), which is already satisfied by $u^*(x)$. This can be done imposing $C_u u(v) \leq g_u$ for all $v \in \mathcal{V}_I$, which implies $C_u u(x) \leq g_u$ for all $x \in \mathcal{F}_N$.

(2) The value of $u(x)$ must be computed such that $\|u(x) - u^*(x)\|_\infty \leq \bar{w}$, in order for system (1) to satisfy Equation (14). This can be obtained by simply imposing $\|u(v) - u^*(v)\|_\infty \leq \bar{w}$ for all $v \in \mathcal{V}_I$.

(3)
- If $K = K_n$, in order to obtain Equation (23), we impose that $u(v) = u^*(v)$ for all $v \in \mathcal{V}_I \cap \mathcal{S}_f$;
- If $K = K_p$, in order for system (1) to fulfil Equation (24), we require that $|w_i(x)| \leq \alpha \|x\|_1$ for all $i = 1, \ldots, m$ and all $x \in \mathcal{F}_N$, which is obtained by forcing $|u_i(v) - u_i^*(v)| \leq \alpha \|v\|_1$ for all $v \in \mathcal{V}_I \cap \mathcal{X}_f$.

Therefore, after recalling the relationship between vector θ and the control variable $u(x)$ in Equations (5) and (6), we obtain $u(x)$ by solving the following linear programme:

$$\min_{\theta,\eta} \ \ \eta \tag{26a}$$

$$\text{s.t.} \ \ \eta \geq \pm \left(\phi(v)'\theta_j - u_j^*(v) \right), \ v \in \mathcal{V}_I, \ j = 1, \ldots, m \tag{26b}$$

$$C_u \Phi(v)\theta \leq g_u, \ v \in \mathcal{V}_I \tag{26c}$$

$$\begin{cases} \Phi(v)\theta = u^*(v), & v \in \mathcal{V}_I \cap \mathcal{S}_f, \text{ if } K = K_n \\ |u_i(v) - u_i^*(v)| \leq \alpha \|v\|_1, & \\ \quad i = 1, \ldots, m, & v \in \mathcal{V}_I \cap \mathcal{X}_f, \text{ if } K = K_p \end{cases} \tag{26d}$$

$$\eta \leq \bar{w}. \tag{26e}$$

The formulation of the cost function (26a) together with the constraint (26b) leads to finding the vector θ that minimises the maximum difference between $u_j(x)$ and $u_j^*(x)$ for all $x \in \mathcal{F}_N$ and all components j. Conditions (26c) and (26d) lead to the fulfilment of Equations (3) and (23) (or Equation (24)), respectively. Condition (26e) ensures the fulfilment of Equation (14). Once a feasible solution to Equation (26) has been found, vector θ determines the control law $u(x)$ for all $x \in \mathcal{S}$.

6.2 *Properties of the PWAS control law*

The following result holds when the approximate control law $u(x)$ is applied to system (1).

Theorem 6.1: *Let Equation* (14) *and Assumptions* 3.1 *and* 5.4 *holds for system* (1). *Assume that a feasible solution for the FHOCP* (20) *exists, and define* $u^*(x)$ *as in Equation* (17). *Finally, suppose that one of the following holds:*

- (i) *The MPC control law in Equation* (17) *is designed with* $K = K_n$, *all the assumptions in case* (I) *of Theorem* 5.5 *are satisfied, and there exists a realisation of* $u(x)$ *obtained through a feasible solution of Equation* (26).
- (ii) *The MPC control law in Equation* (17) *is designed with* $K = K_p$. *Moreover, all the assumptions in case* (II) *of Theorem* 5.5 *are satisfied, and there exists a realisation of* $u(x)$ *obtained through a feasible solution of Equation* (26).

Then, if $x(0) \in \mathcal{F}_N$, *one has* $x(t) \in \mathcal{X}$ *and* $u(t) \in \mathcal{U}$ *for all* $t \geq 0$. *Moreover, the state is asymptotically driven to* $\mathcal{R}_\infty^n$ *in case* (i), *or to* $\mathcal{R}_\infty^p \cap \mathcal{R}_\infty$ *in case* (ii). *Finally, if* $\mathcal{D} = \{0\}$, *in both cases* (i) *and* (ii), *the origin is an asymptotically stable equilibrium point for system* (1), *with domain of attraction equal to* $\mathcal{F}_N$.

Proof. See Appendix 2.2. □

Remark 4: Due to the properties of the α-basis chosen to formulate $u(x)$, Equation (26) imposes conditions only on a subset of the components of θ related to the vertices $v \in \mathcal{V}_S$. In particular, if $v \in S_i$ with $S_i \cap \mathcal{F}_N = \emptyset$, then any value assigned to the corresponding component of θ is not influencing the solution of Equation (26), because their values do not affect $u(x)$ in $\mathcal{F}_N$.

6.3 *Parameter tuning*

Considering that the feasibility of Equations (12), (20), and (26) is not guaranteed *a priori*, we give some guidelines on choosing the design parameters of the proposed approach. We assume that the number of vertices N_v of the simplicial partition is fixed, which fixes the memory occupation and latency time on the digital circuit implementing the control law, since these quantities only depend on the structure of the chosen PWAS structure, and not on its values. Given the sets $\mathcal{X}$ and $\mathcal{U}$, the tuning parameters on which the designer can act to design $u^*(x)$ are $\bar{w}$ (if $K = K_n$) or both $\bar{w}$ and α (if $K = K_p$). In case $K = K_n$, we can fix a value of $\bar{w}$, compute $\hat{\mathcal{R}}_\infty$ and $\mathcal{X}_f$ checking if Assumption 3.1 is satisfied, check if $0 \in \text{int}(\mathcal{S}_f)$ and $\hat{\mathcal{R}}_\infty \subset \text{int}(\mathcal{S}_f)$, and then solve Problem (20). If Problem (20) is feasible and all the required assumptions are satisfied for a given $\bar{w} = \bar{w}_1$, then the same will hold for any $\bar{w} \leq \bar{w}_1$. Then, one can find by bisection the maximum feasible value of $\bar{w}$, namely $\bar{w}_{\max}$, and then Problem (20) will be feasible for all $\bar{w}$ such that $0 \leq \bar{w} \leq \bar{w}_{\max}$.

In case $K = K_p$, one can choose a sufficiently small value for the parameter α (such that Problem (12) be feasible), and compute K_p. Then, one would act on the value of $\bar{w}$ as in the previous case, but without checking the condition relative to the set $\mathcal{S}_f$, since the equality constraints in Equation (26d) are not imposed if $K = K_p$.

In any case, we know that, once all the other parameters are fixed, a smaller value of $\bar{w}$ leads to a larger set $\mathcal{F}_N$. On the other hand, a small value of $\bar{w}$ could impose a too tight approximation in problem (26), making it infeasible. In conclusion, the designer can start obtaining a feasible realisation of the PWAS control for a value of $\bar{w}$ close to $\bar{w}_{\max}$. Then, this value can be decreased in order to enlarge the set $\mathcal{F}_N$ and obtain the desired performance.

7. Simulation examples

7.1 *Example 1*

As a first example, we consider the problem of regulating to the origin the LTI discrete-time system proposed in Bemporad et al. (2011), where system (1) is defined by

$$A = \begin{bmatrix} 1.2 & 1 \\ 0 & 1.1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \tag{27}$$

with the sets in Equations (2)–(4) defined as $\mathcal{X} = \{x \in \mathbb{R}^2 : \|x\|_\infty \leq 2\}$, $\mathcal{U} = \{u \in \mathbb{R}^2 : |u_1| \leq 0.5, |u_2| \leq 0.6\}$, $\mathcal{D} = \{d \in \mathbb{R}^2 : \|d\|_\infty \leq 5 \cdot 10^{-2}\}$. In this case, we decide to design the auxiliary control law with $K = K_n = \left[\begin{smallmatrix} 0.9337 & -0.1540 \\ -1.0333 & -0.9373 \end{smallmatrix}\right]$, obtained using the weight matrices $Q = \mathbb{I}_2$ and $R = 0.1\mathbb{I}_2$, and we set $\bar{w} = 0.1$. The MPC control law $u^*(x)$ in Equation (17) is computed with $\Psi = \mathbb{I}_2$ and $N = 4$, and its domain of attraction $\mathcal{F}_N$ is shown in Figure 1, together with sets $\hat{\mathcal{R}}_\infty$, $\mathcal{X}_f$, and $\hat{\mathcal{R}}_\infty^n$. The control law $u^*(x)$ is composed of 83 irregular regions, and the point-location problem (see, e.g., Bemporad et al. (2002)) uses a binary search tree with 427 nodes, and a depth between 6 and 9. The approximate control law $u(x)$ is computed with $p_1 = p_2 = 50$ (defined in Section 4), obtaining $N_v = 2601$ vertices and $L_s = 5000$ simplices, with a maximum approximation error $\eta = 0.0373$. In Figure 1, the set $\mathcal{S}_f$ is also shown, and

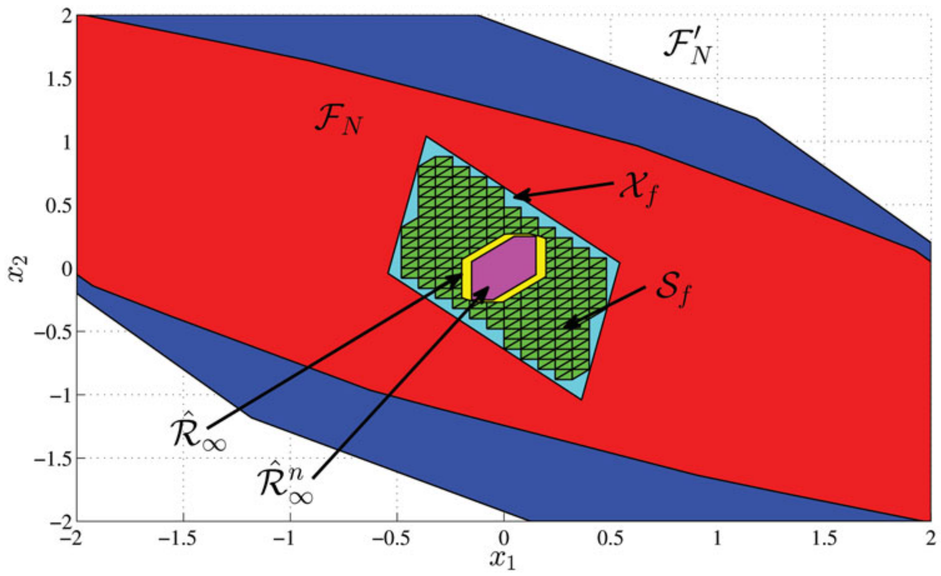

Figure 1. Sets $\mathcal{F}'_N$, $\mathcal{F}_N$, $\mathcal{X}_f$, $\hat{\mathcal{R}}_\infty$, $\hat{\mathcal{R}}^n_\infty$ for the obtained robust MPC control law $u^*(x)$ in Example 1.

it is possible to verify that all the assumptions required in case (i) of Theorem 6.1 are satisfied. The PWAS control law so obtained is shown in Figure 2. In Figure 1, the set $\mathcal{F}'_N$ of feasible states using the optimal MPC control law (designed with $\mathcal{W} = \{0\}$) is shown, and one can note a reasonably contained reduction of the region of attraction with respect to the direct employment of $u^*(x)$.

7.2 *Example 2*

As a second example, we design the approximate MPC controller for the same system in form (1) described by matrices A and B in Equation (27), in case $\mathcal{D} = \{0\}$. In this case, we set $\alpha = 0.05$, and design the auxiliary control law with $K = K_p = \begin{bmatrix} 0.9385 & -0.1696 \\ -1.0387 & -0.9570 \end{bmatrix}$, which is obtained using the same weight matrices as in Example 1. The MPC control law $u^*(x)$ in Equation (17) is computed with $\Psi = \mathbb{I}_2$ and $N = 4$, and its domain of attraction $\mathcal{F}_N$ is shown in Figure 3, together with the other sets related to this example. The approximate control law $u(x)$ is computed with $p_1 = p_2 = 50$, obtaining $N_v = 2601$ vertices and $L_s = 5000$ simplices, with a maximum approximation error $\eta = 0.0297$. In this case, the control law $u^*(x)$ is composed of 104 irregular regions, thus the point-location problem uses a binary search tree

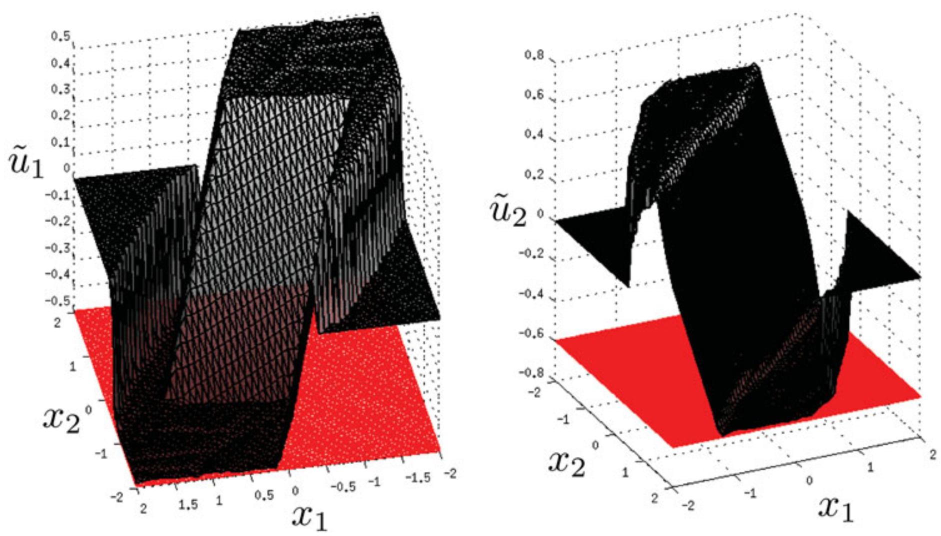

Figure 2. Control function $u(x)$ on the simplicial partition of the set $\mathcal{S}$ in Example 1.

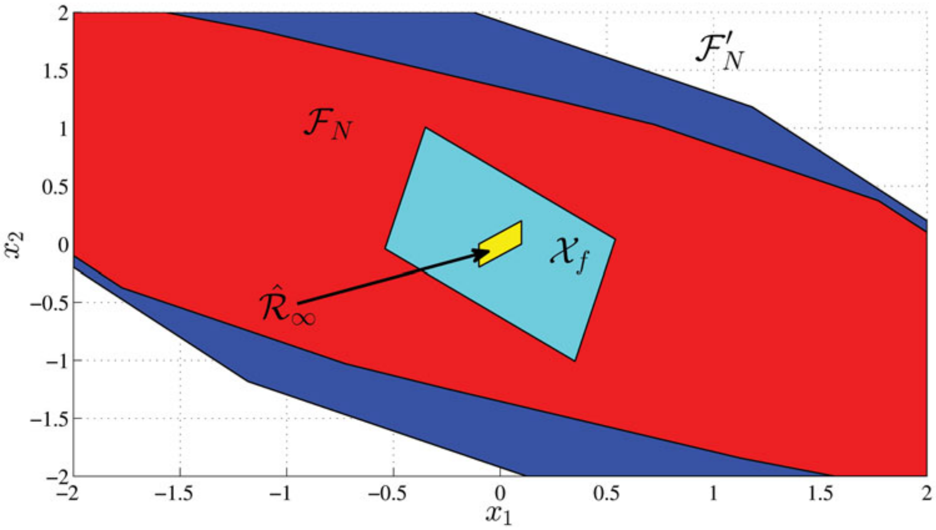

Figure 3. Sets $\mathcal{F}'_N$, $\mathcal{F}_N$, $\mathcal{X}_f$, $\hat{\mathcal{R}}_\infty$ for the obtained robust MPC control law $u^*(x)$ in Example 2.

with 701 nodes, and a depth between 7 and 10. Since all the conditions required in case (ii) of Theorem 6.1 are satisfied, the asymptotic stability of the origin is guaranteed for all initial conditions in $\mathcal{F}_N$. In Figure 4, the time evolution of the state and control variables are shown starting from the initial condition $x(0) = [\,0.88\ -0.2\,]'$.

7.3 *Circuit performance comparison*

In order to test the performance of the proposed control laws on real circuits, we used a Xilinx Spartan 3 FPGA (xc3s200) board to implement the PWAS law of Example 2, coding the state variables (circuit inputs) with 12-bit words. The employment of architecture B (serial) in Storace and Poggi (2010) for the simplicial approximation uses 7.8 KB of RAM, 165 slices, and 1 multiplier, allowing the control law computation to occur in 12 clock cycles. The simplified circuit design allows an effective circuit frequency of 80 MHz, which leads to a sampling time interval of 150 ns. Note that implementing the PWAS control law of Example 1 requires exactly the same circuit specification on the FPGA.

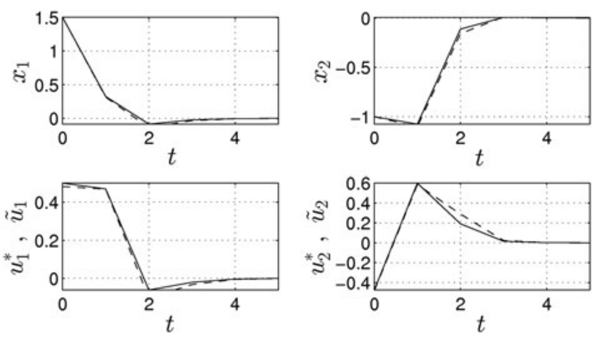

Figure 4. Time evolution of the state and control variables in Example 2 (solid line for optimal values, dashed line for approximate PWAS solution).

The optimal MPC control law, which is a generic PWA function described in Equation (17), in case $\mathcal{W} = \{0\}$ (i.e., no approximation error) and serial implementation, uses 1.012 KB of RAM, 1684 slices, and 1 multiplier, allowing the computation to occur in 49 clock cycles. Using the parallel architecture, the circuit uses 1.012 KB of RAM, 1267 slices, and 2 multipliers, allowing the computation to occur in 25 clock cycles. Both architectures can push the circuit frequency to 60 MHz, leading to latencies of 813 and 415 ns for serial and parallel implementations, respectively.

As one would expect given the more involved hardware architecture, the generic PWA implementations have greater computation latency. Moreover, the number of used slices is increased by a factor of 10 with respect to the simplicial approximation, which, however, requires more RAM to store data relative to the greater number of regions. Notice, however, that the ongoing trend in computer hardware technology is not, as few decades ago, to push on frequency, but to increase the number of processing units and RAM memory. Evidently, an upper limit to the execution efficiency of hardly parallelisable algorithms (such as optimization) is reached, suggesting the manufacturers to invest in quantity rather than in pure speed. As a consequence, the trade-off between time and space resources has changed, making RAM occupation an increasingly negligible issue when compared to online computation power (as required by online MPC). The parameters of the described circuits are summed up in Table 1.

Table 1 Parameters relative to the FPGA implementation of the described control laws.

Control law	PWAS	PWA (serial)	PWA (parallel)
RAM (kB)	7.800	1.012	1.012
No. of slices	165	1684	1267
No. of multipliers	1	1	2
No. of clock cycles	12	49	25
Frequency (MHz)	80	60	60
Latency (ns)	150	813	415

8. Conclusions

In this paper, an approximate MPC control law for uncertain LTI systems based on PWAS functions has been proposed, which can be efficiently implemented on digital hardware. The proposed synthesis methods guarantees *a priori* the asymptotic convergence of the closed-loop system to a terminal set (or its asymptotic stability in case no external disturbance is present). In particular, the approach with $K = K_n$ does not require the introduction of the additional tuning parameter α, but can be applied only if the simplicial partition is dense enough to obtain a non-empty set $\mathcal{S}_f$. The approach with $K = K_p$, instead, requires the introduction of α, but can be applied also with a coarser simplicial partition. The applicability of the proposed control strategy is effective for the case of small-sized systems, similarly to standard explicit MPC. The theoretical properties of the control law have been proved based on robust MPC synthesis, and the simulation results have confirmed the expected results, both for the theoretical properties of the PWAS controller and for the performance of the related FPGA implementation.

Funding

This work was partially supported by the European Commission under project 'MOBY-DIC – model-based synthesis of digital electronic circuits for embedded control' [grant number FP7-INFSO-ICT-248858] (http://www.mobydic-project.eu/).

References

Bemporad, A., Morari, M., Dua, V., & Pistikopoulos, E.N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, *38*, 3–20.

Bemporad, A., Oliveri, A., Poggi, T., & Storace, M. (2011). Ultra-fast stabilizing model predictive control via canonical piecewise affine approximations. *IEEE Transactions on Automatic Control*, *56*, 2883–2897.

Blanchini, F. (1999). Set invariance in control. *Automatica*, *35*, 1747–1767.

Blanchini, F., & Miani, S. (2008). *Set-theoretic methods in control*. Boston, MA: Birkhauser.

Chisci, L., Rossiter, J.A., & Zappa, G. (2001). Systems with persistent disturbances: Predictive control with restricted constraints. *Automatica*, *37*, 1019–1028.

Ferreau, H.J., Bock, H.G., & Diehl, M. (2008). An online active set strategy to overcome the limitations of explicit MPC. *International Journal of Robust and Nonlinear Control*, *18*, 816–830.

Genuit, B.A.G., Lu, L., & Heemels, W.P.M.H. (2011). Approximation of PWA control laws using regular partitions: An ISS approach. In *IFAC World Congress* (pp. 4540–4545). Milan.

Grieder, P., Kvasnica, M., Baotić, M., & Morari, M. (2005). Stabilizing low complexity feedback control of constrained piecewise affine systems. *Automatica*, *41*, 1683–1694.

Gurvits, L. (1995). Stability of discrete-time difference inclusions. *Linear Algebra and its Applications*, *231*, 47–85.

Herceg, M., Kvasnica, M., Jones, C., & Morari, M. (2013, July). Multi-parametric toolbox 3.0. *Proceedings of the European Control Conference* (pp. 502–510). Zürich. Retrieved from http://control.ee.ethz.ch/∼mpt

Jones, C.N., & Morari, M. (2010). Polytopic approximation of explicit model predictive controllers. *IEEE Transactions on Automatic Control*, *55*, 2542–2553.

Julián, P., Desages, A., & D'Amico, B. (2000). Orthonormal high level canonical PWL functions with applications to model reduction. *IEEE Transactions on Circuits and Systems I*, *47*, 702–712.

Kolmanovsky, I., & Gilbert, E. (1998). Theory and computation of disturbance invariant sets for discrete-time linear systems. *Mathematical Problems in Engineering*, *4*, 317–363.

Kothare, M., Balakrishnan, V., & Morari, M. (1996). Robust constrained model predictive control using linear matrix inequalities. *Automatica*, *32*, 1361–1379.

Kouramas, K.I., Rakovic, S.V., Kerrigan, E.C., Allwright, J.C., & Mayne, D.Q. (2005). On the minimal robust positively invariant set for linear difference inclusions. In *IEEE Conference on Decision and Control* (pp. 2296–2301). Seville.

Kvasnica, M., & Fikar, M. (2012). Clipping-based complexity reduction in explicit MPC. *IEEE Transactions on Automatic Control*, *57*, 1878–1883.

Kvasnica, M., Löfberg, J., & Fikar, M. (2011). Stabilizing polynomial approximation of explicit MPC. *Automatica*, *47*, 2292–2297.

Lu, L., Heemels, W.P.M.H., & Bemporad, A. (2011). Synthesis of low-complexity stabilizing piecewise affine controllers: A control-Lyapunov function approach. In *IEEE Conference on Decision and Control* (pp. 1227–1232). Orlando, FL.

Mayne, D., & Schroeder, W. (1997). Robust time-optimal control of constrained linear systems. *Automatica*, *33*, 2103–2118.

Oliveri, A., Barcelli, D., Bemporad, A., Genuit, B.A.G., Heemels, W.P.M.H., Poggi, T., ...Storace, M. (2012, August). MOBY-DIC: A Matlab toolbox for the circuit design of explicit MPC. In *IFAC Conference on Nonlinear Model Predictive Control* (pp. 218–225). Noordwijkerhout.

Patrinos, P., & Bemporad, A. (2014). An accelerated dual gradient-projection algorithm for embedded linear model predictive control. *IEEE Transactions on Automatic Control*, *59*, 18–33.

Poggi, T., Rubagotti, M., Bemporad, A., & Storace, M. (2012). High-speed piecewise affine virtual sensors. *IEEE Transactions on Industrial Electronics*, *59*, 1228–1237.

Rakovic, S., Kerrigan, E., Kouramas, K., & Mayne, D. (2005). Invariant approximations of the minimal robust positively invariant set. *IEEE Transactions on Automatic Control*, *50*, 406–410.

Rawlings, J.B., & Mayne, D.Q. (2009). *Model predictive control: Theory and design*. Madison, WI: Nob Hill.

Richter, S., Morari, M., & Jones, C.N. (2011). Towards computational complexity certification for constrained MPC based on Lagrange relaxation and the fast gradient method. In

IEEE Conference on Decision and Control (pp. 5223–5229). Orlando, FL.
Riverso, S., Battocchio, A., & Ferrari-Trecate, G. (2013). *PnPMPC Toolbox v. 0.9 – user manual.* Retrieved from http://sisdin.unipv.it/pnpmpc/pnpmpc.php
Rubagotti, M., Barcelli, D., & Bemporad, A. (2012). Approximate explicit MPC on simplicial partitions with guaranteed stability for constrained linear systems. In *IFAC Symposium on Nonlinear MPC* (pp. 119–125). Noordwijkerhout.
Rubagotti, M., Patrinos, P., & Bemporad, A. (2014). Stabilizing linear model predictive control under inexact numerical optimization. *IEEE Transactions on Automatic Control*, *59*, 1660–1666.
Rubagotti, M., Trimboli, S., & Bemporad, A. (2013). Stability and invariance analysis of uncertain discrete-time piecewise affine systems. *IEEE Transactions on Automatic Control*, *58*, 2359–2365.
Storace, M., & Poggi, T. (2010). Digital architectures realizing piecewise-linear multi-variate functions: Two FPGA implementations. *International Journal of Circuit Theory and Applications*, *37*, 1–15.
Wang, Y., & Boyd, S. (2010). Fast model predictive control using online optimization. *IEEE Transactions on Control Systems Technology*, *18*, 267–278.

Appendix 1. Characterisation and computation of RPI sets

Relying, for instance, on Blanchini and Miani (2008, Proposition 6.9), one can prove that $\mathcal{R}_\infty$ is a polytope in our case. Nonetheless, an explicit computation of $\mathcal{R}_\infty$ is, in general, impossible (apart from the very specific case of A_κ nilpotent, as stated by Mayne & Schroeder, 1997). The (not necessarily RPI) polytopic over-approximation $\hat{\mathcal{R}}_\infty$ can be computed using various numerical algorithms: the reader is referred to Blanchini (1999, Sections 6.4 and 6.5) and Rakovic, Kerrigan, Kouramas, and Mayne (2005) for an overview (an implementation of the algorithm described in the latter paper is also available, see Riverso, Battocchio, & Ferrari-Trecate, 2013). The same procedure, in the particular case $\mathcal{W} = 0$, leads to the computation of $\mathcal{R}^n_\infty$ and $\hat{\mathcal{R}}^n_\infty$. Analogous considerations are valid for the characterisation of $\mathcal{R}^p_\infty$ and for the computation of $\hat{\mathcal{R}}^p_\infty$, which can be obtained as a polytope after a finite number of iterations of the numerical algorithm described by Kouramas, Rakovic, Kerrigan, Allwright, and Mayne (2005) (note that the system with structured feedback uncertainty (22) is equivalent to a linear time-varying system, as highlighted by Kothare et al., 1996).

The set $\mathcal{X}_f$ in Equation (18) can also be conveniently expressed, using tightened constraints, as

$$\mathcal{X}_f = \{x : A_\kappa^k x \in \mathcal{X}_k,\ K A_\kappa^k x \in \mathcal{U}_k,\ \forall k \in \mathbb{Z}_{\geq 0}\}, \tag{A1}$$

and can be computed by Algorithm 6.1 in Kolmanovsky and Gilbert (1998) using linear programming. In particular, exploiting the results in Theorems 6.2 and 6.3 in Kolmanovsky and Gilbert (1998), $\mathcal{X}_f$ is finitely generated, if $0 \in \text{int}(\mathcal{X} \sim \mathcal{R}_\infty)$ and $0 \in \text{int}(\mathcal{U} \sim K\mathcal{R}_\infty)$. If $\mathcal{R}_\infty$ is not computable, one can use the above-mentioned over-approximation $\hat{\mathcal{R}}_\infty$ instead. Efficient methods for the computation of $\mathcal{X}_f$ are implemented in the MPT Toolbox for MATLAB (Herceg, Kvasnica, Jones, & Morari, 2013).

Appendix 2. Proofs

2.1 *Proof of Theorem 5.5*

The first part of the proof holds for both choices of K. We recall that Assumptions (A1)–(A5) in Chisci et al. (2001) are automatically satisfied if Assumptions 3.1 and 5.4 hold, together with Equation (14). Therefore, according to Lemma 7 and Theorem 8 in Chisci et al. (2001), recursive feasibility is ensured, if $x(0) \in \mathcal{F}_N$. Therefore, $x(t) \in \mathcal{X}$ and $u^*(t) \in \mathcal{U}$ for all $t \in \mathbb{Z}_{\geq 0}$. Also, $x(t) \to \mathcal{R}_\infty$ as $t \to \infty$, for all choices of K that are stabilising for the nominal system (i.e., both K_n and K_p). On the other hand, according to the expression of $\mathcal{X}_f$ in Equation (A1), the evolution of the nominal system given by $\hat{x}(k)$ with initial condition $x \in \mathcal{X}_f$ and $\mu(t + k|t) = 0, \forall k = 1, \dots, N-1$, fulfils the constraints (20b) and (20c). Also, as noticed in Chisci et al. (2001), the constraints $\hat{x}(k) \in \mathcal{X}_k$ and $K\hat{x}(k) \in \mathcal{U}_k$ for $k \geq N$ are equivalent to the terminal constraint (20d). Then, we conclude that $v = [0 \cdots 0]'$ is a feasible solution for Problem (20) whenever $x \in \mathcal{X}_f$, and is the minimiser of Problem (20), since it is the global minimum of the objective function, i.e., $x \in \mathcal{X}_f \implies M^*(x) = [0\ \dots\ 0]'$.

Consider now case (I). Since $\hat{\mathcal{R}}_\infty \subset \text{int}(\mathcal{S}_f)$, then there exists $\epsilon \in \mathbb{R}_{>0}$ arbitrary small, such that $(1+\epsilon)\hat{\mathcal{R}}_\infty \subseteq \text{int}(\mathcal{S}_f)$. Considering that $\hat{\mathcal{R}}_\infty$ is an RPI set for system (13), it is an RPI set for system (21) as well. Therefore, by linearity of the system, $(1+\epsilon)\hat{\mathcal{R}}_\infty$ is also an RPI set for system (21). Considering now the actual dynamics (8), from the trivial relation $\hat{\mathcal{R}}_\infty \subset (1+\epsilon)\hat{\mathcal{R}}_\infty$, it follows that, for all initial conditions $x(0) \in \mathcal{F}_N$, there exists $t_1 \in \mathbb{Z}_{\geq 0}$ such that $x(t_1) \in (1+\epsilon)\hat{\mathcal{R}}_\infty$. Since it is assumed that $w(x) = 0$ for all $x \in \mathcal{S}_f$, and $(1+\epsilon)\hat{\mathcal{R}}_\infty$ is positively invariant for system (21), one has that the system dynamics is given by Equation (21) for all $t \geq t_1$, which leads to the asymptotic convergence of the state of system (8) to $\mathcal{R}^n_\infty$ for all $x(0) \in \mathcal{F}_N$.

Consider now case (II). By Assumption 5.4, for any initial condition $x(0) \in \mathcal{F}_N$, there exists $t_2 \in \mathbb{Z}_{\geq 0}$ such that, applying dynamics (8), $x(t_2) \in \mathcal{X}_f$. Considering that $\mathcal{X}_f$ is by definition an RPI set for system (13), we get $u^*(x) = K_p x$ for all $t \geq t_2$. As a consequence, since, given $x(0) \in \mathcal{X}_f$, both $x(t) \to \mathcal{R}_\infty$ and $x(t) \to \mathcal{R}^p_\infty$ as $t \to \infty$, it follows that $x(t) \to \mathcal{R}_\infty \cap \mathcal{R}^p_\infty$ as $t \to \infty$ for all $x(0) \in \mathcal{F}_N$.

In both cases (I) and (II), if $\mathcal{D} = \{0\}$, it is immediate to see that the asymptotic stability of system (21) or (22) implies that $\mathcal{R}^n_\infty = \mathcal{R}^p_\infty = \{0\}$. Therefore, the origin would be an asymptotically stable equilibrium point with region of attraction equal to $\mathcal{F}_N$.

2.2 *Proof of Theorem 6.1*

In both cases (i) and (ii), conditions (26d) and (26e) allow one to consider $w(x) = u(x) - u^*(x)$ as a disturbance term that satisfies all the requirements to synthesise $u^*(x)$ in Equation (17). Therefore, by application of Theorem 5.5, all the mentioned properties are proved.