Delft University of Technology

Delft Center for Systems and Control

Technical report 04-004

Model predictive control for discrete-event and hybrid systems – Part II: Hybrid systems*

B. De Schutter and T.J.J. van den Boom

If you want to cite this report, please use the following reference instead:
B. De Schutter and T.J.J. van den Boom, "Model predictive control for discrete-event and hybrid systems – Part II: Hybrid systems," *Proceedings of the 16th International Symposium on Mathematical Theory of Networks and Systems (MTNS 2004)*, Leuven, Belgium, 10 pp., July 2004. Paper 313.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

Model Predictive Control for Discrete-Event and Hybrid Systems
Part II: Hybrid Systems

B. De Schutter, T.J.J. van den Boom
Delft Center for Systems and Control, Delft University of Technology
Mekelweg 2, 2628 CD Delft, The Netherlands
{b.deschutter,t.j.j.vandenboom}@dcsc.tudelft.nl

Abstract— **Model predictive control (MPC) is a very popular controller design method in the process industry. A key advantage of MPC is that it can accommodate constraints on the inputs and outputs. Usually MPC uses linear or nonlinear discrete-time models. In this paper and its companion paper ("Part I: Discrete-Event Systems") we give an overview of some results in connection with MPC approaches for some tractable classes of discrete-event systems and hybrid systems. In general the resulting optimization problems are nonlinear and nonconvex. However, for some classes tractable solution methods exist.**

After having discussed MPC for max-plus-linear discrete-event systems in the companion paper, we now discuss MPC for some classes of hybrid systems, viz. mixed logical dynamical systems, max-min-plus-scaling systems, and continuous piecewise-affine systems.

I. Introduction

We assume that the reader either has read the introduction to model predictive control (MPC) given in the companion paper [26], or is familiar with the basics of MPC, i.e., a model-based on-line control approach with the following ingredients: a prediction horizon, a receding horizon procedure, and a regular update of the model and re-computation of the optimal control input. More extensive information on MPC can be found in [1], [8], [16], [20], [32], [46] and the references therein.

Conventional MPC uses discrete-time models (i.e., models consisting of a system of difference equations). In the companion paper [26] we have already discussed MPC for some tractable classes of discrete-event systems. In this paper we propose some extensions and adaptations of the MPC framework to classes of hybrid systems that ultimately result in "tractable" control approaches.

This paper is organized as follows. In Section II we present hybrid systems and some specific subclasses of hybrid systems: piecewise affine systems, mixed logical dynamical systems, and max-min-plus-scaling systems. An MPC approach for these classes of hybrid systems is then presented in Sections III and IV.

For easy reference we list the abbreviations used in this paper and the companion paper [26]:

ELCP	: extended linear complementarity problem
LP	: linear programming
MIQP	: mixed integer quadratic programming
MLD	: mixed logical dynamical
MMPS	: max-min-plus-scaling
MPC	: model predictive control
MPL	: max-plus linear
PWA	: piecewise affine
QP	: quadratic programming
SQP	: sequential quadratic programming

II. Hybrid systems

A. General hybrid systems

Hybrid systems arise throughout business and industry in areas such as interactive distributed simulation, traffic control, plant process control, aircraft, robotics, manufacturing, and automotive applications. There are several possible definitions of hybrid systems. For some authors a hybrid system is a coupling of a continuous-time or analog system and a discrete-time or digital system (in practice often a continuous-time, analog plant and an asynchronous, digital controller). We shall use a somewhat different definition.

For us, hybrid systems arise from the interaction between continuous-variable systems[1] and discrete-event systems. In general we could say that a hybrid system can be in one of several modes of operation, whereby in each *mode* the behavior of the system can be described by a system of difference or differential equations, and that the system switches from one mode to another due to the occurrence of events (see Figure 1). The mode transitions may be caused by an external control signal, by an internal control signal (if the controller is already included in the system under consideration), or by the dynamics of the system itself, i.e., when a certain boundary in the state space is crossed. At a switching time instant there may be a reset of the state (i.e., a jump in the values of the state variables) and/or the dimension of the state may change.

There are many modeling and analysis techniques for hybrid systems. Typical modeling techniques are predicate calculus, real-time temporal logics, timed communicating

[1] Continuous-variable systems are systems the behavior of which can be described by a system of difference or differential equations.

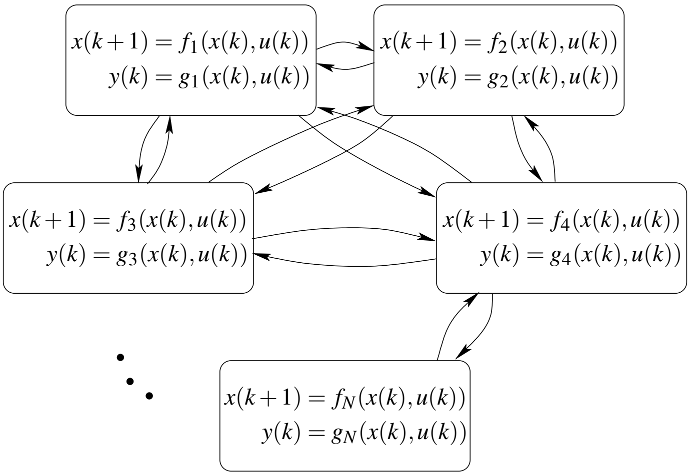

Fig. 1. Schematic representation of a hybrid system with N modes. In each mode the behavior of the hybrid system is described by a system of difference (or differential) equations. The system goes from one mode to another due to the occurrence of an event (this is indicated by the arrows).

sequential processes, hybrid automata, timed automata, timed Petri nets, and object-oriented modeling languages such as Modelica, SHIFT or Chi. Current analysis techniques for hybrid systems include formal verification, perturbation analysis, and computer simulation. Furthermore, special mathematical analysis techniques have been developed for specific subclasses of hybrid systems. We shall only discuss some of these methods. For more information on the other methods the interested reader is referred to [3], [4], [39], [44], [47], [57], [59] and the references cited therein.

An important trade-off in the context of modeling and analysis of hybrid systems is that of modeling power versus decision power: the more accurate the model is the less we can analytically say about its properties. Furthermore, many analysis and control problems lead to computationally hard problems for even the most elementary hybrid systems [9]. As tractable methods to analyze general hybrid systems are not available, several authors have focused on special subclasses of hybrid dynamical systems for which analysis and/or control design techniques are currently being developed. Some examples of such subclasses are:

- mixed logical dynamical (MLD) systems [5], [6],
- piecewise-affine (PWA) systems [56],
- linear complementarity systems [38], [58],
- extended linear complementarity systems [37],
- max-min-plus scaling (MMPS) systems [21],
- timed automata [2],
- timed Petri nets [48], [53].

In this paper we will consider PWA systems, MLD systems, and MMPS systems. Note that some of these classes (in particular MLD, PWA, (extended) linear complementarity and constrained MMPS systems) are equivalent [37], possibly under mild additional assumptions related to well-posedness and boundedness of input, state, output or auxiliary variables. Each subclass has its own advantages over the others. E.g., stability criteria were proposed for PWA systems [41], analysis and control techniques for PWA systems in [11], [29], [40], control and verification techniques for MLD hybrid models [5]–[7] and for MMPS systems [21], [24], [25], and conditions of existence and uniqueness of solution trajectories (well-posedness) for linear complementarity systems [38], [58]. So it really depends on the application which of these classes is best suited.

In the next subsections we will discuss some tractable classes of hybrid systems, for which MPC control design methods are available as we will see in Sections III and IV.

Note that — in contrast to the companion paper [26], which dealt with discrete-event systems, and where k was an event step counter k, — in this paper the counter k is either a sample step counter (in case of systems with a time-driven behavior) or an event counter (for event-driven systems). Furthermore, the inputs, outputs, and states are not necessarily time instants any longer as was the case in the companion paper [26].

B. PWA systems

PieceWise Affine (PWA) systems [56] are described by

$$\begin{aligned} x(k+1) &= A_i x(k) + B_i u(k) + f_i \\ y(k) &= C_i x(k) + D_i u(k) + g_i \end{aligned} \quad \text{for} \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} \in \Omega_i, \qquad (1)$$

for $i = 1, \ldots, N$ where $\Omega_1, \ldots, \Omega_N$ are convex polyhedra (i.e., given by a finite number of linear inequalities) in the input/state space. PWA systems have been studied by several authors (see [5], [6], [19], [40]–[43], [56] and the references therein) as they form the "simplest" extension of linear systems that can still model nonlinear and nonsmooth processes with arbitrary accuracy and that are capable of handling hybrid phenomena.

Example 2.1 As a very simple example of a PWA model we can consider an integrator with upper saturation:

$$x(k+1) = \begin{cases} x(k) + u(k) & \text{if } x(k) + u(k) \leqslant 1 \\ 1 & \text{if } x(k) + u(k) \geqslant 1 \end{cases} \qquad (2)$$

$$y(k) = x(k) \ . \qquad (3)$$

If we rewrite the model (2)–(3) as in (1), then we have

$$\begin{aligned} &\Omega_1 = \left\{ (x(k), u(k)) \in \mathbb{R}^2 \,\middle|\, x(k) + u(k) \leqslant 1 \right\} \\ &\Omega_2 = \left\{ (x(k), u(k)) \in \mathbb{R}^2 \,\middle|\, x(k) + u(k) \geqslant 1 \right\} \\ &A_1 = 1, \quad A_2 = 0, \quad B_1 = 1, \quad B_2 = 0 \\ &f_1 = 0, \quad f_2 = 1, \quad C_1 = C_2 = 1 \\ &D_1 = D_2 = 0, \quad g_1 = g_2 = 0 \ . \end{aligned}$$

□

C. MLD systems

This subsection is based on the seminal paper [6].

TABLE I
TRUTH TABLE.

X_1	X_2	$X_1 \wedge X_2$	$X_1 \vee X_2$	$\sim X_1$	$X_1 \Rightarrow X_2$	$X_1 \Leftrightarrow X_2$	X_1 **xor** X_2
T	T	T	T	F	T	T	F
T	F	F	T	F	F	F	T
F	T	F	T	T	T	F	T
F	F	F	F	T	T	T	F

1) Preliminaries: First, we will provide some tools to transform logical statements involving continuous variables into mixed-integer linear inequalities.

We consider logical statements that will be represented using the notation X_i, such as, e.g., X_1: "$x \leqslant 0$", or X_2: "temperature is cold". Then X_i is called a literal, and it has a truth value of either "T" (true) or "F" (false). We use the following notation for boolean connectives: "$\wedge$" (and), "$\vee$" (or), "$\sim$" (not), "$\Rightarrow$" (implies), "$\Leftrightarrow$" (if and only if), "**xor**" (exclusive or). These connectives are defined by means of the truth table given in Table I, and they satisfy several properties (see, e.g., [18]), such as:

$$X_1 \Rightarrow X_2 \quad \text{is the same as} \quad \sim X_1 \vee X_2 \tag{4}$$

$$X_1 \Rightarrow X_2 \quad \text{is the same as} \quad \sim X_2 \Rightarrow \sim X_1 \tag{5}$$

$$X_1 \Leftrightarrow X_2 \quad \text{is the same as} \quad (X_1 \Rightarrow X_2) \wedge (X_2 \Rightarrow X_1)\ . \tag{6}$$

One can associate with a literal X_i a logical variable $\delta_i \in \{0,1\}$ that has a value of either 1 if $X_i = \text{T}$, or 0 if $X_i = \text{F}$. A propositional logic problem, where a statement X must be proved to be true given a set of (compound) statements involving literals $X_1,\ldots,X_n$, can thus be solved by means of an integer linear program, by suitably translating the original compound statements into linear inequalities involving logical variables $\delta_1,\ldots,\delta_n$. In fact, the following propositions and linear constraints can easily be seen to be equivalent [60, p. 176]:

$$X_1 \wedge X_2 \quad \text{is equivalent to} \quad \delta_1 = \delta_2 = 1 \tag{7}$$

$$X_1 \vee X_2 \quad \text{is equivalent to} \quad \delta_1 + \delta_2 \geqslant 1 \tag{8}$$

$$\sim X_1 \quad \text{is equivalent to} \quad \delta_1 = 0 \tag{9}$$

$$X_1 \Rightarrow X_2 \quad \text{is equivalent to} \quad \delta_1 - \delta_2 \leqslant 0 \tag{10}$$

$$X_1 \Leftrightarrow X_2 \quad \text{is equivalent to} \quad \delta_1 - \delta_2 = 0 \tag{11}$$

$$X_1\, \mathbf{xor}\, X_2 \quad \text{is equivalent to} \quad \delta_1 + \delta_2 = 1\ . \tag{12}$$

We can use this computational inference technique to model logical parts of processes (on/off switches, discrete mechanisms, combinational and sequential networks), and heuristic knowledge about plant operation as integer linear inequalities. In this way we can construct models of hybrid systems.

As we are interested in systems which contain both logic and continuous dynamics, we wish to establish a link between the two worlds. As will be shown next, we end up with mixed-integer linear inequalities, i.e., linear inequalities involving both continuous variables $x \in \mathbb{R}^n$ and logical variables $\delta \in \{0,1\}^{n_\delta}$.

Consider the statement $X \overset{\text{def}}{=} [f(x) \leqslant 0]$, where $f : \mathbb{R}^n \to \mathbb{R}$. Assume that $x \in \mathscr{X}$, where $\mathscr{X}$ is a given bounded set, and define

$$M = \max_{x \in \mathscr{X}} f(x), \qquad m = \min_{x \in \mathscr{X}} f(x)\ . \tag{13}$$

Theoretically, an over-estimate (under-estimate) of M (m) suffices for our purpose. However, more realistic estimates provide computational benefits [60, p. 171]. Now it is easy to verify that

$$[f(x) \leqslant 0] \wedge [\delta = 1] \quad \text{is true if and only if} \quad f(x) - \delta \leqslant -1 + m(1-\delta) \tag{14}$$

$$[f(x) \leqslant 0] \vee [\delta = 1] \quad \text{is true if and only if} \quad f(x) \leqslant M\delta \tag{15}$$

$$\sim [f(x) \leqslant 0] \quad \text{is true if and only if} \quad f(x) \geqslant \varepsilon_{\text{tol}}\ , \tag{16}$$

where ε_{tol} is a small tolerance (typically the machine precision), beyond which the constraint is regarded as violated. By (4) and (15), it also follows that

$$[f(x) \leqslant 0] \Rightarrow [\delta = 1] \quad \text{is true if and only if} \quad f(x) \geqslant \varepsilon + (m - \varepsilon)\delta \tag{17}$$

$$[f(x) \leqslant 0] \Leftrightarrow [\delta = 1] \quad \text{is true if and only if} \quad \begin{cases} f(x) \leqslant M(1-\delta) \\ f(x) \geqslant \varepsilon + (m-\varepsilon)\delta\ . \end{cases} \tag{18}$$

Finally, we report procedures to transform products of logical variables, and of continuous and logical variables, in terms of linear inequalities by introducing some auxiliary variables [60, p. 178]. The product term $\delta_1\delta_2$ can be replaced by an auxiliary logical variable $\delta_3 = \delta_1\delta_2$. Then, $[\delta_3 = 1] \Leftrightarrow [\delta_1 = 1] \wedge [\delta_2 = 1]$, and therefore

$$\delta_3 = \delta_1\,\delta_2 \quad \text{is equivalent to} \quad \begin{cases} -\delta_1 + \delta_3 \leqslant 0 \\ -\delta_2 + \delta_3 \leqslant 0 \\ \delta_1 + \delta_2 - \delta_3 \leqslant 1\ . \end{cases}$$

Moreover, the term $\delta f(x)$, where $f : \mathbb{R}^n \to \mathbb{R}$ and $\delta \in \{0,1\}$, can be replaced by an auxiliary real variable $y = \delta f(x)$ that satisfies

$$[\delta = 0] \Rightarrow [y = 0], \quad [\delta = 1] \Rightarrow [y = f(x)]\ .$$

Therefore, by defining M and m as in (13), $y = \delta f(x)$ is equivalent to

$$y \leqslant M\delta \quad (19)$$
$$y \geqslant m\delta \quad (20)$$
$$y \leqslant f(x) - m(1-\delta) \quad (21)$$
$$y \geqslant f(x) - M(1-\delta) \ . \quad (22)$$

2) MLD systems: The results of the previous section will now be used now to express relations describing the evolution of systems where physical laws, logic rules, and operating constraints are interdependent. Before giving a general definition, we first consider an example.

Example 2.2 Consider the following PWA system:

$$x(k+1) = \begin{cases} 0.8x(k) + u(k) & \text{if } x(k) \geqslant 0 \\ -0.8x(k) + u(k) & \text{if } x(k) < 0 \end{cases} \quad (23)$$

where $x(k) \in [-10, 10]$, and $u(k) \in [-1, 1]$. The condition $x(k) \geqslant 0$ can be associated with a binary variable $\delta(k)$ such that

$$[\delta(k) = 1] \Leftrightarrow [x(k) \geqslant 0] \ .$$

By using the transformation (18), this equation can be expressed by the inequalities

$$-m\delta(k) \leqslant x(k) - m$$
$$-(M+\varepsilon)\delta \leqslant -x - \varepsilon \ ,$$

where $M = -m = 10$, and ε is a small positive scalar. Then (23) can be rewritten as

$$x(k+1) = 1.6\,\delta(k)x(k) - 0.8x(k) + u(k) \ .$$

By defining a new variable $z(k) = \delta(k)x(k)$, which, by (19)–(22), can be expressed as

$$z(k) \leqslant M\delta(k) \quad (24)$$
$$z(k) \geqslant m\delta(k) \quad (25)$$
$$z(k) \leqslant x(k) - m(1-\delta(k)) \quad (26)$$
$$z(k) \geqslant x(k) - M(1-\delta(k)) \ , \quad (27)$$

the evolution of system (23) is given by the linear state update equation

$$x(k+1) = 1.6z(k) - 0.8x(k) + u(k)$$

subject to the linear constraints (24)–(27). □

The example above can be generalized by describing systems through the following linear relations:

$$x(k+1) = Ax(k) + B_1u(k) + B_2\delta(k) + B_3z(k) \quad (28)$$
$$y(k) = Cx(k) + D_1u(k) + D_2\delta(k) + D_3z(k) \quad (29)$$
$$E_1x(k) + E_2u(k) + E_3\delta(k) + E_4z(k) \leqslant g_5, \quad (30)$$

where $x(k) = [\, x_\mathrm{r}^\mathrm{T}(k) \;\; x_\mathrm{b}^\mathrm{T}(k) \,]^\mathrm{T}$ with $x_\mathrm{r}(k) \in \mathbb{R}^{n_\mathrm{r}}$ and $x_\mathrm{b}(k) \in \{0,1\}^{n_\mathrm{b}}$ ($y(k)$ and $u(k)$ have a similar structure), and where $z(k) \in \mathbb{R}^{r_\mathrm{r}}$ and $\delta(k) \in \{0,1\}^{r_\mathrm{b}}$ are auxiliary variables. Systems of the form (28)–(30) are called *Mixed Logical Dynamical* (MLD) systems.

The MLD formalism allows specifying the evolution of the continuous variables (through linear dynamic equations), the evolution of the discrete variables (through propositional logic statements and automata), and also the mutual interaction between the two. As explained above the key idea of the approach consists of embedding the logic part in the state equations by transforming boolean variables into 0-1 integers, and by expressing the relations as mixed-integer linear inequalities. MLD systems are therefore capable of modeling a broad class of systems, such as PWA systems, linear hybrid systems, finite state machines, (bi)linear systems with discrete inputs, etc. [6].

Remark 2.3 It is assumed that for all $x(k)$ with $x_\mathrm{b}(k) \in \{0,1\}^{n_\mathrm{b}}$, all $u(k)$ with $u_\mathrm{b}(k) \in \{0,1\}^{m_\mathrm{b}}$, all $z(k) \in \mathbb{R}^{r_\mathrm{r}}$, and all $\delta(k) \in \{0,1\}^{r_\mathrm{b}}$ satisfying (30) it holds that $x(k+1)$ and $y(k)$ determined from (28)–(29) are such that $x_\mathrm{b}(k+1) \in \{0,1\}^{n_\mathrm{b}}$ and $y_\mathrm{b}(k) \in \{0,1\}^{l_\mathrm{b}}$. This is without loss of generality, as we can take binary components of states and outputs (if any) to be auxiliary variables as well (see the proof of Proposition 1 of [5]). □

D. MMPS systems

In [25] a class of discrete-event systems and hybrid systems has been introduced that can be modeled using the operations maximization, minimization, addition and scalar multiplication. Expressions that are built using these operations are called *Max-Min-Plus-Scaling* (MMPS) expressions.

Definition 2.4: An MMPS expression f of the variables $x_1, \ldots, x_n$ is defined by the grammar[2]

$$f := x_i | \alpha | \max(f_k, f_l) | \min(f_k, f_l) | f_k + f_l | \beta f_k \quad (31)$$

with $i \in \{1, 2, \ldots, n\}$, α, $\beta \in \mathbb{R}$, and where f_k, f_l are again MMPS expressions.
An example of an MMPS expression is, e.g., $3x_1 - 8x_2 + 9 + \max(\min(3x_1, -9x_2), -x_2 - 3x_3)$.

Consider now systems that can be described by

$$x(k+1) = \mathscr{M}_x(x(k), u(k), d(k)) \quad (32)$$
$$y(k) = \mathscr{M}_y(x(k), u(k), d(k)), \quad (33)$$

where $\mathscr{M}_x$, $\mathscr{M}_y$ are MMPS expressions in terms of the components of $x(k)$, $u(k)$ and the auxiliary variables $d(k)$, which are all real-valued. Such systems will be called MMPS systems. If in addition, we have a condition of the form

$$\mathscr{M}_\mathrm{c}(x(k), u(k), d(k)) \leqslant c(k) \ ,$$

with $\mathscr{M}_\mathrm{c}$ an MMPS expression, we speak about *constrained* MMPS systems. A typical example of an (unconstrained) MMPS is a traffic-signal controlled intersection [21].

[2]The symbol | stands for "or", and the definition is recursive.

Example 2.5 It is easy to verify that if we recast the PWA model (2)–(3) of Example 2.1 into an MMPS model we obtain

$$x(k+1) = \min(x(k)+u(k),1)$$
$$y(k) = x(k) \ . \qquad \square$$

E. Equivalence between **continuous** *PWA systems and MMPS systems*

In [37] it has been shown that PWA systems, MLD systems and constrained MMPS systems are equivalent under mild additional assumptions related to well-posedness and boundedness of input, state, output or auxiliary variables. Now we consider another equivalence between *continuous* PWA systems and (*unconstrained*) MMPS systems in more detail as it will be the basis for the MPC approach derived in Section IV below.

A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be a *continuous* PWA function if and only if the following conditions hold [19]:

1) The domain space $\mathbb{R}^n$ is divided into a finite number of polyhedral regions $R_{(1)},\dots,R_{(N)}$.
2) For each $i \in \{1,\dots,N\}$, f can be expressed as
$$f(x) = \alpha_{(i)}^{\mathrm{T}} x + \beta_{(i)} \tag{34}$$
for any $x \in R_{(i)}$ with $\alpha_{(i)} \in \mathbb{R}^n$ and $\beta_{(i)} \in \mathbb{R}$.
3) f is continuous on any boundary between two regions.

A continuous PWA system is a system of the form

$$x(k+1) = \mathscr{P}_x(x(k),u(k)) \tag{35}$$
$$y(k) = \mathscr{P}_y(x(k),u(k)) \ , \tag{36}$$

where $\mathscr{P}_x$ and $\mathscr{P}_y$ are vector-valued continuous PWA functions. Note that the main difference with the PWA systems introduced in Section II-B is that now we require the PWA functions to be continuous on the boundary between the regions that make up the partition of the domain.

Theorem 2.6 ([34], [51]*):* If f is a continuous PWA function of the form (34), then there exist index sets $I_1,\dots,I_\ell \subseteq \{1,\dots,N\}$ such that

$$f = \max_{j=1,\dots,\ell} \min_{i \in I_j} (\alpha_{(i)}^{\mathrm{T}} x + \beta_{(i)}) \ .$$

From the definition of MMPS functions it follows that (see also [34], [51]):

Lemma 2.7: Any MMPS function is also a continuous PWA function.

From Theorem 2.6 and Lemma 2.7 it follows that continuous PWA systems and (unconstrained) MMPS systems are equivalent, i.e., for a given continuous PWA model there exists an MMPS model (and vice versa) such that the input-output behavior of both models coincides.

Corollary 2.8: Continuous PWA models and (unconstrained) MMPS models are equivalent.

Note that this is an extension of the results of [37], which prove an equivalence between (not necessarily continuous) PWA models and MMPS models, but there some extra auxiliary variables and some additional algebraic MMPS constraints between the states, the inputs and the auxiliary variables were required to transform the PWA model into an MMPS model.

III. MPC FOR MLD SYSTEMS[3]

A. The MLD-MPC problem

An important control problem for MLD systems is to stabilize the system to an equilibrium state or to track a desired reference trajectory. In general finding a control law that attains these objectives for an MLD system is not an easy task, as in general MLD systems are neither linear[4] nor even smooth. MPC provides a successful tool to perform these task, as will be shown next. For the sake of brevity we will concentrate on the stabilization to an equilibrium state.

Consider the MLD system (28)–(30) and an equilibrium state/input/output triple $(x_{\mathrm{eq}},u_{\mathrm{eq}},y_{\mathrm{eq}})$, and let $(\delta_{\mathrm{eq}},z_{\mathrm{eq}})$ be the corresponding pair of auxiliary variables. Let $\hat{x}(k+j|k)$ denote the estimate of the state x at sample step $k+j$ based on the information available at sample step k. In a similar way we also define $\hat{y}(k+j|k)$, $\hat{\delta}(k+j|k)$, and $\hat{z}(k+j|k)$. Now we consider the MLD-MPC problem at sample step k with objective function

$$J(k) = \sum_{j=1}^{N_\mathrm{p}} \|\hat{x}(k+j|k)-x_{\mathrm{eq}}\|_{Q_x}^2 + \|u(k+j-1)-u_{\mathrm{eq}}\|_{Q_u}^2 + \|\hat{y}(k+j|k)-y_{\mathrm{eq}}\|_{Q_y}^2 + \|\hat{\delta}(k+j-1|k)-\delta_{\mathrm{eq}}\|_{Q_\delta}^2 + \|\hat{z}(k+j-1|k)-z_{\mathrm{eq}}\|_{Q_z}^2$$

where Q_u, Q_x are positive definite matrices, and Q_y, Q_δ, Q_z are nonnegative definite matrices. Furthermore, in addition to the MLD system equations we have the end-point condition

$$\hat{x}(k+N_\mathrm{p}|k) = x_{\mathrm{eq}} \ ,$$

and possibly also a control horizon constraint[5] of the form

$$u(k+j) = u(k+N_\mathrm{c}-1) \ \text{ for } j = N_\mathrm{c},\dots,N_\mathrm{p}-1.$$

Now we have:

Theorem 3.1 ([6]*):* Consider an MLD system (28)–(30) and an equilibrium state/input/output triple $(x_{\mathrm{eq}},u_{\mathrm{eq}},y_{\mathrm{eq}})$, and let $(\delta_{\mathrm{eq}},z_{\mathrm{eq}})$ be the corresponding pair of auxiliary variables. Assume that the initial state $x(0)$ is such that a feasible solution of the MLD-MPC problem exists for sample step 0. The input signal resulting from applying the optimal MLD-MPC input signal in a receding horizon approach stabilizes

[3] Also this section is based on [6].

[4] Due to the integer constraints $\delta_i \in \{0,1\}$, the linear inequality (30) results in a nonlinear relation between δ and x, u, and between z and x, u.

[5] While in other contexts introducing a control horizon constraint amounts to hugely down-sizing the optimization problem at the price of a reduced performance, for MLD systems the computational gain is only partial, since all the (auxiliary) variables $\hat{\delta}(k+\ell|k)$ and $\hat{z}(k+\ell|k)$ for $\ell = N_\mathrm{c},\dots,N_\mathrm{p}-1$ remain in the optimization.

the MLD system in the sense that

$$\lim_{k\to\infty} x(k) = x_{\mathrm{eq}}\,, \qquad \lim_{k\to\infty} \|y(k)-y_{\mathrm{eq}}\|_{Q_y} = 0\,,$$
$$\lim_{k\to\infty} u(k) = u_{\mathrm{eq}}\,, \qquad \lim_{k\to\infty} \|\delta(k)-\delta_{\mathrm{eq}}\|_{Q_\delta} = 0\,,$$
$$\lim_{k\to\infty} \|z(k)-z_{\mathrm{eq}}\|_{Q_z} = 0 \ .$$

B. Algorithms for the MLD-MPC optimization problem

Let us now show that the MLD-MPC problem can be recast as a mixed integer quadratic programming (MIQP) problem. For the MLD case using successive substitution for (28) results in the following prediction equation for the state:

$$\hat{x}(k+j|k) = A^j x(k) + \sum_{i=0}^{j-1} A^{j-i}\big(B_1 u(k+i) + B_2\delta(k+i) + B_3 z(k+i)\ \ .$$

For $\hat{y}(k+j|k)$ we have a similar expression. Now we define

$$\tilde{u}(k) = \big[\, u^{\mathrm{T}}(k)\ \ \ldots\ \ u^{\mathrm{T}}(k+N_{\mathrm{p}}-1)\,\big]^{\mathrm{T}}$$
$$\tilde{y}(k) = \big[\, \hat{y}^{\mathrm{T}}(k+1|k)\ \ \ldots\ \ \hat{y}^{\mathrm{T}}(k+N_{\mathrm{p}}|k)\,\big]^{\mathrm{T}}\ ,$$

and in similar way also $\tilde{\delta}(k)$ and $\tilde{z}(k)$. If we define

$$\tilde{V}(k) = \big[\tilde{u}^{\mathrm{T}}(k)\ \tilde{y}^{\mathrm{T}}(k)\ \tilde{\delta}^{\mathrm{T}}(k)\ \tilde{z}^{\mathrm{T}}(k)\big]^{\mathrm{T}}\ ,$$

we obtain the following equivalent formulation for the MLD-MPC problem:

$$\min_{\tilde{V}(k)} \tilde{V}^{\mathrm{T}}(k) S_1 \tilde{V}(k) + 2(S_2 + x^{\mathrm{T}}(k) S_3)\tilde{V}(k) \qquad (37)$$
$$\text{subject to } F_1\tilde{V}(k) \leqslant F_2 + F_3 x(k)\ , \qquad (38)$$

for appropriately defined matrices S_1, S_2, S_3, F_1, F_2, F_3. Note that $\tilde{V}(k)$ contains both real-valued and integer-valued components. As the objective function is quadratic, the problem (37)–(38) is an MIQP problem.

MIQP problems are classified as NP-hard [33], [55], which — loosely speaking — means that, in the worst case, the solution time grows exponentially with the problem size. Several algorithmic approach have been applied successfully to medium and large-size application problems [31], the four major ones being cutting plane methods, decomposition methods, logic-based methods, and branch-and-bound methods. In [6] the authors use a branch-and-bound method as several authors seems to agree on the fact that branch-and-bound methods are the most successful for mixed integer programming problems [30].

As described by [6], [30], the branch-and-bound algorithm for MIQP consists of solving and generating new quadratic programming (QP) problems in accordance with a tree search, where the nodes of the tree correspond to QP subproblems. The QP subproblems involve real-valued variables only, and are thus efficiently solvable using a modified simplex method or an interior point method [50], [52], [61].

For a worked example of the MLD-MPC approach we refer the interested reader to [6].

IV. MPC FOR CONTINUOUS PWA AND MMPS SYSTEMS

In the previous section we have already discussed how the MPC problem for MLD systems (and thus also PWA systems) can be recast as an MIQP problem. In this section we will define the MMPS-MPC problem and show that for (unconstrained) MMPS systems, and thus also for *continuous* PWA systems, the MPC problem can be transformed into solving a sequence of real (so not integer!) linear programming (LP) problems. The approach we propose is based on canonical forms for MMPS functions, which are introduced below, and it is similar to the cutting-plane algorithm for convex optimization problems.

A. Canonical forms of MMPS functions

Theorem 4.1: Any MMPS function $f : \mathbb{R}^n \to \mathbb{R}$ can be rewritten in the min-max canonical form

$$f = \min_{i=1,\ldots,K}\ \max_{j=1,\ldots,n_i} (\,\alpha^{\mathrm{T}}_{(i,j)} x + \beta_{(i,j)}) \qquad (39)$$

or in the max-min canonical form

$$f = \max_{i=1,\ldots,L}\ \min_{j=1,\ldots,m_i} (\,\gamma^{\mathrm{T}}_{(i,j)} x + \delta_{(i,j)}) \qquad (40)$$

for some integers K, L, $n_1,\ldots,n_K$, $m_1,\ldots,m_L$, vectors $\alpha_{(i,j)}, \gamma_{(i,j)}$, and real numbers $\beta_{(i,j)}, \delta_{(i,j)}$.

Proof: We will only sketch the proof of the theorem (see [28] for the full proof). Moreover, we only consider the min-max canonical form since the proof for the max-min canonical form is similar.

It is easy to verify that if f_k and f_l are affine functions, then the functions that result from applying the basic constructors of an MMPS function (max, min, $+$, and scaling — cf. (31)) are in min-max canonical form[6].

Now we use a recursive argument that consists in showing that if we apply the basic constructors of an MMPS function to two (or more) MMPS functions in min-max canonical form, then the result can again be transformed into min-max canonical form. Consider two MMPS functions f and g in min-max canonical form[7]: $f = \min(\max(f_1, f_2), \max(f_3, f_4))$ and $g = \min(\max(g_1, g_2), \max(g_3, g_4))$. Using the following properties (with $\alpha, \beta, \gamma, \delta \in \mathbb{R}$):

- minimization is distributive w.r.t. maximization, i.e., $\min\big(\alpha, \max(\beta,\gamma)\big) = \max\big(\min(\alpha,\beta), \min(\alpha,\gamma)\big)$. So
$$\min\big(\max(\alpha,\beta),\max(\gamma,\delta)\big) = \max\big(\min(\alpha,\gamma),\min(\alpha,\delta),\min(\beta,\gamma),\min(\beta,\delta)\big)\,;$$
- the max operation is distributive w.r.t. min. Hence,
$$\max\big(\min(\alpha,\beta),\min(\gamma,\delta)\big) =$$

[6] We allow "void" min or max statements of the form $\min(s)$ or $\max(s)$, which by definition are equal to s for any expression s. Alternatively, we can write $\min(s,s)$ or $\max(s,s)$.

[7] For the sake of simplicity we only consider two min-terms in f and g, each of which consists of the maximum of two affine functions. However, the proof also holds if more terms are considered.

$$\min\left(\max(\alpha,\gamma),\max(\alpha,\delta),\max(\beta,\gamma),\max(\beta,\delta)\right);$$

- $\min(\alpha,\beta)+\min(\gamma,\delta)=\min(\alpha+\gamma,\alpha+\delta,\beta+\gamma,\beta+\delta)$
  $\max(\alpha,\beta)+\max(\gamma,\delta)=\max(\alpha+\gamma,\alpha+\delta,\beta+\gamma,\beta+\delta)$
  $\max(\alpha,\beta)=-\min(-\alpha,-\beta)$;
- if $\rho\in\mathbb{R}$ is nonnegative, then

$$\rho\max(\alpha,\beta)=\max(\rho\alpha,\rho\beta)$$
$$\rho\min(\alpha,\beta)=\min(\rho\alpha,\rho\beta)\ ;$$

it can be shown that $\max(f,g)$, $\min(f,g)$, $f+g$ and βf can again be written in min-max canonical form. ■

B. The MMPS-MPC problem

We can use the deterministic model (32)–(33) either as a model of an MMPS system, as the equivalent model of a continuous PWA system, or as an approximation of a general smooth nonlinear system. Note that we do not include modeling errors or uncertainty in the model. However, since MPC uses a receding finite horizon approach, we can regularly update the model and the state estimate as new information and measurements become available.

We can make an estimate $\hat{y}(k+j|k)$ of the output of the system (32)–(33) at sample step $k+j$ based on the state $x(k)$ and the future input sequence $u(k),\ldots,u(k+j-1)$. Using successive substitution, we obtain an expression of the following form:

$$\hat{y}(k+j|k)=F_j(x(k),u(k),\ldots,u(k+j-1))$$

for $j=1,\ldots,N_{\rm p}$. Clearly, $\hat{y}(k+j|k)$ is an MMPS function of $x(k),u(k),\ldots,u(k+j-1)$.

In this section we consider the following output and input cost functions (see also Section IV-B of [26]):

$$J_{\rm out,1}(k)=\|\tilde{y}(k)-\tilde{r}(k)\|_1\ ,\quad J_{\rm in,1}(k)=\|\tilde{u}(k)\|_1\ ,\qquad(41)$$
$$J_{\rm out,\infty}(k)=\|\tilde{y}(k)-\tilde{r}(k)\|_\infty\ ,\quad J_{\rm in,\infty}(k)=\|\tilde{u}(k)\|_\infty\ .\qquad(42)$$

Since we have $|x|=\max(x,-x)$ for all $x\in\mathbb{R}$, it is easy to verify that these cost functions are also MMPS functions.

Just as in conventional MPC and in MPC for max-plus-linear discrete-event systems (cf. [26]), we can define (non)linear constraints

$$C_{\rm c}(k,\tilde{u}(k),\tilde{y}(k))\leqslant 0\ ,\qquad(43)$$

or

$$A_{\rm c}(k)\tilde{u}(k)+B_{\rm c}(k)\tilde{y}(k)\leqslant c_{\rm c}(k)\ ,\qquad(44)$$

and a control horizon constraint

$$u(k+j)=u(k+N_{\rm c}-1)\ \text{ for } j=N_{\rm c},\ldots,N_{\rm p}-1,\qquad(45)$$

or

$$\Delta^2 u(k+j)=0\quad\text{ for } j=N_{\rm c},\ldots,N_{\rm p}-1\ .\qquad(46)$$

This then results in the MMPS-MPC problem.

C. Algorithms for the MMPS-MPC optimization problem

1) Nonlinear or ELCP optimization: In general the MMPS-MPC optimization problem is a nonlinear, nonconvex optimization problem. Some of the methods discussed in Section V of the companion paper [26] can also be used to solve the MMPS-MPC optimization problem: we can use multi-start nonlinear optimization based on sequential quadratic programming (SQP) [10], [52], or we can use a method based on the extended linear complementarity problem (ELCP) [23]. However, both methods have their disadvantages.

If we use the SQP approach, then we usually have to consider a large number of initial starting points and perform several optimization runs to obtain (a good approximation of) the global minimum. In addition, the objective functions that appear in the MMPS-MPC optimization problem are non-differentiable and PWA (if we use the cost criteria given in (41)–(42)), which makes the SQP algorithm less suitable for them.

The main disadvantage of the ELCP approach is that the execution time of this algorithm increases exponentially as the size of the problem increases. This implies that this approach is not feasible if $N_{\rm c}$ or the number of inputs and outputs of the system are large.

An alternative option consists in transforming the MMPS system into an MLD system since (constrained) MMPS systems are equivalent to MLD systems [37]. The main difference between MLD-MPC and MMPS-MPC is that MLD-MPC requires the solution of *mixed integer-real* optimization problems. In general, these are also computationally hard optimization problems.

Now we will present another method to solve the MMPS-MPC optimization problem that is similar to the cutting-plane method used in convex optimization [12].

2) An LP-based algorithm: We assume that the cost criteria given in (41)–(42) are used[8]. Recall that these objective functions (and any linear combination of them) are MMPS functions. The same holds for the estimate of future output $\tilde{y}(k)$. So if we substitute $\tilde{y}(k)$ in the expression for $J(k)$, we finally obtain an MMPS function of $\tilde{u}(k)$ as objective function. From Theorem 4.1 it follows that this objective function can be written in min-max canonical form as follows (where — for the sake of simplicity of notation — we drop the index k):

$$J=\min_{i=1,\ldots,\ell}\max_{j=1,\ldots,n_i}(\alpha_{(i,j)}^{\rm T}\tilde{u}+\beta_{(i,j)})$$

for appropriately defined integers ℓ, $n_1,\ldots,n_\ell$, vectors $\alpha_{(i,j)}$ and integers $\beta_{(i,j)}$. Note that in general the expression obtained by straightforwardly applying the manipulations of the proof of Theorem 4.1 will contain a large number of

[8]The result below also holds for any other cost criterion that is an MMPS function of $\tilde{y}(k)$ and $\tilde{u}(k)$. So it follows from Theorem 2.6 that any continuous PWA norm function can also be used.

affine terms $\alpha_{(i,j)}^{\mathrm{T}}\tilde{u}+\beta_{(i,j)}$. However, many of these terms are redundant[9], and can thus be removed. This reduces the number of affine terms. Also note that the (computationally intensive) transformation into canonical form only has to be performed once — provided that we explicitly consider all terms that depend on k as additional variables when performing the transformation, — and that it can be done off-line.

The derivation below is similar to the cutting-plane algorithm for convex optimization (see, e.g., [12]). Hence, it requires constraints that are linear (or convex) in $\tilde{u}$. Note that the control horizon constraints (45) or (46) satisfy this condition. However, even if the original MPC constraint (43) is linear in $\tilde{u}(k)$ and $\tilde{y}(k)$, then in general this constraint is not linear any more after substitution of $\tilde{y}(k)$. Therefore, from now on we assume that there are only linear[10] constraints on the input $\tilde{u}(k)$:

$$P\tilde{u}+q \geqslant 0 \ . \tag{47}$$

In practice, constraints of the form (47) occur if we have to guarantee that the control signal $\tilde{u}(k)$ or the control signal rate $\Delta\tilde{u}(k)$ stay within certain bounds. Note that in general P and q may depend on $x(k)$ and k, but for the sake of simplicity of notation we do not explicitly indicate this dependence in this paper.

To obtain the optimal MMPS-MPC input signal at sample step k, we have to solve an optimization problem of the following form:

$$\begin{aligned}&\min_{\tilde{u}} \min_{i=1,\ldots,\ell} \max_{j=1,\ldots,n_i} (\alpha_{(i,j)}^{\mathrm{T}}\tilde{u}+\beta_{(i,j)})\\ &\text{subject to } P\tilde{u}+q \geqslant 0 \ ,\end{aligned}$$

or, equivalently,

$$\min_{i=1,\ldots,\ell} \min_{\tilde{u}} \max_{j=1,\ldots,n_i} (\alpha_{(i,j)}^{\mathrm{T}}\tilde{u}+\beta_{(i,j)}) \tag{48}$$

$$\text{subject to } P\tilde{u}+q \geqslant 0 \ . \tag{49}$$

Now let $i \in \{1,\ldots,\ell\}$ and consider

$$\begin{aligned}&\min_{\tilde{u}} \max_{j=1,\ldots,n_i} (\alpha_{(i,j)}^{\mathrm{T}}\tilde{u}+\beta_{(i,j)})\\ &\text{subject to } P\tilde{u}+q \geqslant 0 \ .\end{aligned}$$

It is easy to verify that this problem is equivalent to the following LP problem:

$$\min t \tag{50}$$

$$\text{subject to } \quad t \geqslant \alpha_{(i,j)}^{\mathrm{T}}\tilde{u}+\beta_{(i,j)} \quad \text{for } j=1,\ldots,n_i \tag{51}$$

$$P\tilde{u}+q \geqslant 0 \ . \tag{52}$$

[9] E.g., since they appear twice, or since there are other terms in the max (min) expression that are always larger (smaller) than the given term.

[10] The optimization algorithm used below, which is based on the cutting plane algorithm for convex optimization, can also deal with convex constraints. So we can also allow convex constraints instead of (47).

This LP problem can be solved efficiently using (variants of) the simplex method or an interior-point algorithm (see, e.g., [50], [61]).
To obtain the solution of (48)–(49), we solve (50)–(52) for $i=1,\ldots,\ell$ and afterward we select the solution $\tilde{u}_{(i)}^{\mathrm{opt}}$ for which $\max_{j=1,\ldots,n_i} (\alpha_{(i,j)}^{\mathrm{T}}\tilde{u}_{(i)}^{\mathrm{opt}}+\beta_{(i,j)})$ is the smallest[11]. This results in an algorithm to solve the MMPS-MPC problem that is more efficient than the SQP or the ELCP approach.

For a worked example of the MMPS-MPC approach we refer the interested reader to [27].

V. Related work in connection with MPC for hybrid systems

A similar approach as the one derived in Section V-C of the companion paper [26], where we showed that for monotonically nondecreasing objective functions and constraints the MPC optimization problem can be recast as convex or linear optimization problem, can also be applied to MPC for first-order hybrid systems with saturation [21], [22] (an example of these systems is a traffic-signal controlled intersection). The perturbed MMPS-MPC case is considered in [49].

Note that MPC is related to optimal control. In this context, optimal control of a classes of manufacturing systems is considered in [17]. Other methods for optimal control of hybrid systems are presented in [13]–[15], [35], [36], [45], [54].

VI. Conclusions

In this paper and its companion paper [26] we have presented an overview of some results in connection with MPC for some tractable classes of discrete-event systems and hybrid systems: max-plus-linear discrete-event systems, mixed logical dynamical systems, max-min-plus-scaling systems, and continuous piecewise-affine systems.

Extension of the MPC approach to other tractable classes of discrete-event systems and hybrid systems, improving the computational complexity, and searching for good and efficient approximations and solution methods are among the major current research topics in this field.

Acknowledgments
Many of the results described in this paper have been obtained by or in cooperation with A. Bemporad and W.M.P.H. Heemels.
Research partially funded by the Dutch Technology Foundation STW projects "Model predictive control for hybrid systems" (DMR.5675) and "Multi-agent control of large-scale hybrid systems" (DWV.6188), and by the European IST project "Modelling, Simulation and Control of Nonsmooth Dynamical Systems (SICONOS)" (IST-2001-37172).

[11] If we use a primal-dual simplex method or an interior-point method to solve the LP problems, we can improve the efficiency of the approach even further by stopping the optimization if we obtain a lower bound for the objective function of the current LP problem that is larger than the smallest final objective function of the LP problems that have already been solved.

VII. REFERENCES

[1] F. Allgöwer, T.A. Badgwell, J.S. Qin, J.B. Rawlings, and S.J. Wright, "Nonlinear predictive control and moving horizon estimation – An introductory overview," in *Advances in Control: Highlights of ECC '99* (P.M. Frank, ed.), pp. 391–449, London, UK: Springer, 1999.

[2] R. Alur and D. Dill, "A theory of timed automata," *Theoretical Computer Science*, vol. 126, pp. 183–235, 1994.

[3] P. Antsaklis, W. Kohn, M.D. Lemmon, A. Nerode, and S. Sastry, eds., *Hybrid Systems V*, vol. 1567 of *Lecture Notes in Computer Science*, Berlin, Germany: Springer-Verlag, 1999. (Proceedings of the 5th International Hybrid Systems Workshop, Notre Dame, Indiana, Sept. 1997).

[4] P.J. Antsaklis and A. Nerode, eds., "Special issue on hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, Apr. 1998.

[5] A. Bemporad, G. Ferrari-Trecate, and M. Morari, "Observability and controllability of piecewise affine and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 10, pp. 1864–1876, 2000.

[6] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, Mar. 1999.

[7] A. Bemporad, F.D. Torrisi, and M. Morari, "Optimization-based verification and stability characterization of piecewise affine and hybrid systems," in *Hybrid Systems: Computation and Control* (Proceedings of the 3rd International Workshop on Hybrid Systems: Computation and Control (HSCC 2000), Pittsburgh, Pennsylvania, Mar. 2000) (N. Lynch and B. Krogh, eds.), vol. 1790 of *Lecture Notes in Computer Science*, pp. 45–58, Berlin, Germany: Springer, 2000.

[8] L. Biegler, "Efficient solution of dynamic optimization and NMPC problems," in *Nonlinear Model Predictive Control* (F. Allgöwer and A. Zheng, eds.), vol. 26 of *Progress in Systems and Control Theory*, Basel, Switzerland: Birkhäuser Verlag, 2000.

[9] V.D. Blondel and J.N. Tsitsiklis, "Complexity of stability and controllability of elementary hybrid systems," *Automatica*, vol. 35, no. 3, pp. 479–489, Mar. 1999.

[10] P.T. Boggs and J.W. Tolle, "Sequential quadratic programming," *Acta Numerica*, vol. 4, pp. 1–51, 1995.

[11] F. Borrelli, *Constrained Optimal Control of Linear and Hybrid Systems*, vol. 290 of *Lecture Notes in Control and Information Sciences*. Berlin: Springer, 2003.

[12] S.P. Boyd and C.H. Barratt, *Linear Controller Design: Limits of Performance*. Englewood Cliffs, New Jersey: Prentice Hall, 1991.

[13] M.S. Branicky, "Analyzing and synthesizing hybrid control systems," in *Lectures on Embedded Systems* (G. Rozenberg and F. Vaandrager, eds.), vol. 1494 of *Lecture Notes in Computer Science*, pp. 74–113, Berlin: Springer, 1998.

[14] M.S. Branicky, V.S. Borkar, and S.K. Mitter, "A unified framework for hybrid control: Model and optimal control theory," *IEEE Transactions on Automatic Control*, vol. 43, no. 1, pp. 31–45, Jan. 1998.

[15] M.S. Branicky and G. Zhang, "Solving hybrid control problems: Level sets and behavioral programming," in *Proceedings of the 2000 American Control Conference*, Chicago, Illinois, pp. 1175–1180, June 2000.

[16] E.F. Camacho and C. Bordons, *Model Predictive Control in the Process Industry*. Berlin, Germany: Springer-Verlag, 1995.

[17] C.G. Cassandras, D.L. Pepyne, and Y. Wardi, "Optimal control of a class of hybrid systems," *IEEE Transactions on Automatic Control*, vol. 46, no. 3, pp. 398–415, Mar. 2001.

[18] D. Christiansen, *Electronics Engineers' Handbook*. New York: IEEE Press/McGraw Hill, 4th ed., 1997.

[19] L.O. Chua and A.C. Deng, "Canonical piecewise-linear representation," *IEEE Transactions on Circuits and Systems*, vol. 35, no. 1, pp. 101–111, Jan. 1988.

[20] D.W. Clarke, C. Mohtadi, and P.S. Tuffs, "Generalized predictive control – Part I. The basic algorithm," *Automatica*, vol. 23, no. 2, pp. 137–148, Mar. 1987.

[21] B. De Schutter, "Optimal control of a class of linear hybrid systems with saturation," *SIAM Journal on Control and Optimization*, vol. 39, no. 3, pp. 835–851, 2000.

[22] B. De Schutter, "Optimizing acyclic traffic signal switching sequences through an extended linear complementarity problem formulation," *European Journal of Operational Research*, vol. 139, no. 2, pp. 400–415, June 2002.

[23] B. De Schutter and B. De Moor, "The extended linear complementarity problem," *Mathematical Programming*, vol. 71, no. 3, pp. 289–325, Dec. 1995.

[24] B. De Schutter and T. van den Boom, "Model predictive control for max-min-plus systems," in *Discrete Event Systems: Analysis and Control* (Proceedings of the 5th International Workshop on Discrete Event Systems (WODES2000), Ghent, Belgium, Aug. 2000) (R. Boel and G. Stremersch, eds.), vol. 569 of *The Kluwer International Series in Engineering and Computer Science*, pp. 201–208, Boston, Massachusetts: Kluwer Academic Publishers, 2000.

[25] B. De Schutter and T.J.J. van den Boom, "Model predictive control for max-min-plus-scaling systems – Efficient implementation," in *Proceedings of the 6th International Workshop on Discrete Event Systems (WODES'02)* (M. Silva, A. Giua, and J.M. Colom, eds.), Zaragoza, Spain, pp. 343–348, Oct. 2002.

[26] B. De Schutter and T.J.J. van den Boom, "Model predictive control for discrete-event and hybrid systems – Part I: Discrete-event systems," in *Proceedings of the 16th International Symposium on Mathematical Theory of Networks and Systems (MTNS 2004)*, Leuven, Belgium, July 2004. Paper 312.

[27] B. De Schutter and T.J.J. van den Boom, "MPC for continuous piecewise-affine systems," *Systems & Control Letters*, vol. 52, no. 3–4, pp. 179–192, July 2004.

[28] B. De Schutter, T.J.J. van den Boom, and G.J. Benschop, "MPC for continuous piecewise-affine systems," in *Proceedings of the 15th IFAC World Congress (b'02)*, Barcelona, Spain, pp. 59–64, July 2002.

[29] G. Ferrari-Trecate, F.A. Cuzzola, D. Mignone, and M. Morari, "Analysis and control with performance of piecewise affine and hybrid systems," in *Proceedings of the 2001 American Control Conference*, Arlington, Virginia, pp. 200–205, June 2001.

[30] R. Fletcher and S. Leyffer, "Numerical experience with lower bounds for MIQP branch-and-bound," *SIAM Journal on Optimization*, vol. 8, no. 2, pp. 604–616, May 1998.

[31] C.A. Floudas, *Nonlinear and Mixed-Integer Optimization*. Oxford, UK: Oxford University Press, 1995.

[32] C.E. García, D.M. Prett, and M. Morari, "Model predictive control: Theory and practice – A survey," *Automatica*, vol. 25, no. 3, pp. 335–348, May 1989.

[33] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco: W.H. Freeman and Company, 1979.

[34] V.V. Gorokhovik and O.I. Zorko, "Piecewise affine functions and polyhedral sets," *Optimization*, vol. 31, pp. 209–221, 1994.

[35] S. Hedlund and A. Rantzer, "Hybrid control laws from convex dynamic programming," in *Proceedings of the 39th IEEE Conference on Decision and Control*, Sydney, Australia, pp. 472–477, Dec. 2000.
[36] S. Hedlund and A. Rantzer, "Convex dynamic programming for hybrid systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 9, pp. 1536–1540, Sept. 2002.
[37] W.P.M.H. Heemels, B. De Schutter, and A. Bemporad, "Equivalence of hybrid dynamical models," *Automatica*, vol. 37, no. 7, pp. 1085–1091, July 2001.
[38] W.P.M.H. Heemels, J.M. Schumacher, and S. Weiland, "Linear complementarity systems," *SIAM Journal on Applied Mathematics*, vol. 60, no. 4, pp. 1234–1269, 2000.
[39] T.A. Henzinger and S. Sastry, eds., *Hybrid Systems: Computation and Control*, vol. 1386 of *Lecture Notes in Computer Science*, Berlin, Germany: Springer-Verlag, 1998. (Proceedings of the First International Workshop on Hybrid Systems: Computation and Control (HSCC'98), Berkeley, California, Apr. 1998).
[40] M. Johansson, *Piecewise Linear Control Systems*, vol. 284 of *Lecture Notes in Control and Information Sciences*. Berlin, Germany: Springer, 2003.
[41] M. Johansson and A. Rantzer, "Computation of piecewise quadratic Lyapunov functions for hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 555–559, Apr. 1998.
[42] T.A.M. Kevenaar and D.M.W. Leenaerts, "A comparison of piecewise-linear model descriptions," *IEEE Transactions on Circuits and Systems – I: Fundamental Theory and Applications*, vol. 39, no. 12, pp. 996–1004, Dec. 1992.
[43] D.M.W. Leenaerts and W.M.G. van Bokhoven, *Piecewise Linear Modeling and Analysis*. Boston, Massachusetts: Kluwer Academic Publishers, 1998.
[44] D. Liberzon, *Switching in Systems and Control*. Systems & Control: Foundations and Applications, Boston, Massachusetts: Birkhäuser, 2003.
[45] B. Lincoln and A. Rantzer, "Optimizing linear system switching," in *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, Florida, pp. 2063–2068, Dec. 2001.
[46] J.M. Maciejowski, *Predictive Control with Constraints*. Harlow, UK: Prentice Hall, 2002.
[47] A.S. Morse, C.C. Pantelides, S. Sastry, and J.M. Schumacher, eds., "Special issue on hybrid systems," *Automatica*, vol. 35, no. 3, Mar. 1999.
[48] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, Apr. 1989.
[49] I. Necoara, B. De Schutter, T.J.J. van den Boom, and J. Hellendoorn, "Model predictive control for perturbed continuous piecewise affine systems with bounded disturbances," in *Proceedings of the 43rd IEEE Conference on Decision and Control*, Paradise Island, The Bahamas, pp. 1848–1853, Dec. 2004.
[50] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*. Philadelphia, Pennsylvania: SIAM, 1994.
[51] S. Ovchinnikov, "Max-min representation of piecewise linear functions," *Beiträge zur Algebra und Geometrie/Contributions to Algebra and Geometry*, vol. 43, no. 1, pp. 297–302, 2002.
[52] P.M. Pardalos and M.G.C. Resende, eds., *Handbook of Applied Optimization*. Oxford, UK: Oxford University Press, 2002.
[53] J.L. Peterson, *Petri Net Theory and the Modeling of Systems*. Englewood Cliffs, New Jersey: Prentice-Hall, 1981.
[54] A. Rantzer and M. Johansson, "Piecewise linear quadratic optimal control," *IEEE Transactions on Automatic Control*, vol. 45, no. 4, pp. 629–637, Apr. 2000.
[55] A. Schrijver, *Theory of Linear and Integer Programming*. Chichester, UK: John Wiley & Sons, 1986.
[56] E.D. Sontag, "Nonlinear regulation: The piecewise linear approach," *IEEE Transactions on Automatic Control*, vol. 26, no. 2, pp. 346–358, Apr. 1981.
[57] F.W. Vaandrager and J.H. van Schuppen, eds., *Hybrid systems: Computation and Control*, vol. 1569 of *Lecture Notes in Computer Science*, Berlin, Germany: Springer, 1999. (Proceedings of the Second International Workshop on Hybrid Systems: Computation and Control (HSCC'99), Berg en Dal, The Netherlands, Mar. 1999).
[58] A.J. van der Schaft and J.M. Schumacher, "Complementarity modeling of hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 483–490, Apr. 1998.
[59] A.J. van der Schaft and J.M. Schumacher, *An Introduction to Hybrid Dynamical Systems*, vol. 251 of *Lecture Notes in Control and Information Sciences*. London: Springer-Verlag, 2000.
[60] H.P. Williams, *Model Building in Mathematical Programming*. New York: Wiley, 3rd ed., 1993.
[61] S.J. Wright, *Primal-Dual Interior Point Methods*. Philadelphia, Pennsylvania: SIAM, 1997.