Delft University of Technology

Fac. of Information Technology and Systems **Control Systems Engineering**

Technical report CSE02-002

Model predictive control for max-min-plus-scaling systems – Efficient implementation*

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 max-min-plus-scaling systems – Efficient implementation," *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.

Control Systems Engineering
Faculty of Information Technology and Systems
Delft University of Technology
Delft, The Netherlands
Current URL: https://www.dcsc.tudelft.nl

Model Predictive Control for Max-Min-Plus-Scaling Systems – Efficient Implementation

B. De Schutter and T.J.J. van den Boom
Control Systems Engineering, Fac. ITS, Delft University of Technology
P.O. Box 5031, 2600 GA Delft, The Netherlands
E-mail: {b.deschutter,t.j.j.vandenboom}@its.tudelft.nl

Abstract

In previous work we have introduced model predictive control (MPC) for max-plus-linear and max-min-plus(-scaling) discrete-event systems. For max-plus-linear systems there are efficient algorithms to solve the corresponding MPC optimization problems. However, previously, for max-min-plus(-scaling) systems the only approach was to consider a limited subclass of decoupled max-min-plus systems or to use nonlinear nonconvex optimization algorithms, which are not efficient if the size of the system or the MPC optimization problem is large. In this paper, we present a more efficient approach that is based on canonical forms for max-min-plus-scaling functions and in which the MPC optimization problem is reduced to a set of linear programming problems.

1 Introduction

In [6, 8] we have extended the model predictive control (MPC) framework to max-min-plus and max-min-plus-scaling systems. These systems are extensions of max-plus-linear systems [1, 5], which can be used to model discrete-event systems with synchronization but no choice. The occurrence of choice can lead to the appearance of the minimum operation. This results in max-min-plus systems. A further extension is obtained by adding scalar multiplication. This yields max-min-plus-scaling (MMPS) systems, which are also equivalent to certain classes of hybrid systems, such as mixed logical dynamic systems [2] and piecewise-affine systems [11], as is shown in [10].

In MPC for max-plus-linear systems the resulting optimization problem can be rewritten as a convex or a linear optimization problem. However, such a nice property does not hold for MMPS systems, except for a limited class of systems (see [6, 8]). In general, the optimization problems that occur in MPC for MMPS systems are nonconvex nonlinear optimization problems, which in general are known to be hard to solve. The main result of this paper is the development of a new, more efficient method to solve the MPC optimization problem for MMPS systems. This method is based on canonical forms for MMPS functions and results in an algorithm that is similar to the cutting-plane algorithm for convex optimization [3]. The proposed algorithm consists in solving several linear programming problems and is more efficient than the algorithms used in [6, 8].

2 Max-min-plus-scaling systems

An MMPS function f of the variables $x_1,\ldots,x_n$ is defined by the recursive grammar[1]

$$f := x_i|\alpha|\max(f_k,f_l)|\min(f_k,f_l)|f_k+f_l|\beta f_k \quad , \qquad (1)$$

with $i \in \{1,\ldots,n\}$, $\alpha,\beta \in \mathbb{R}$, and where f_k and f_l are again MMPS functions.

Consider a system that can be described by state space equations of the following form[2]:

$$x(k) = \mathscr{M}_x(x(k-1),u(k),v(k)) \qquad (2)$$
$$y(k) = \mathscr{M}_y(x(k),u(k),v(k)) \quad , \qquad (3)$$

where $\mathscr{M}_x, \mathscr{M}_y$ are MMPS functions, and where $x(k)$ is the state vector, $y(k)$ the output vector, and $u(k)$ and $v(k)$ are the input vectors. More specifically, we discern two types of inputs: controllable inputs ($u(k)$) and uncontrollable inputs ($v(k)$). Systems the behavior of which can be described by a model of the form (2)–(3) will be called *(extended) MMPS systems*. Typical examples of MMPS systems in a discrete-event systems context are digital circuits, computer networks, telecommunication networks, and manufacturing plants. MMPS systems encompasses several other classes of discrete-event systems such as max-plus-linear systems,

[1] The symbol | stands for "or".

[2] The counter k in (2)–(3) can be either a sample step counter (for discrete-time systems), or an event counter (for discrete-event systems).

max-plus-bilinear systems, max-plus-polynomial systems, separated max-min-plus systems, and max-min systems [7]. So MMPS systems can be considered as a generalized framework for several classes of discrete-event systems. Moreover, recently a link between *constrained* MMPS systems and hybrid systems — among which piecewise-affine systems — has been established [10].

3 Canonical forms of MMPS functions

Let $\alpha, \beta, \gamma, \delta, \rho \in \mathbb{R}$ with $\rho \geqslant 0$. It is easy to verify that

$$\min(\max(\alpha,\beta),\max(\gamma,\delta)) = \max(\min(\alpha,\gamma),\min(\alpha,\delta),\min(\beta,\gamma),\min(\beta,\delta)) \quad (4)$$

$$\max(\min(\alpha,\beta),\min(\gamma,\delta)) = \min(\max(\alpha,\gamma),\max(\alpha,\delta),\max(\beta,\gamma),\max(\beta,\delta)) \quad (5)$$

$$\min(\alpha,\beta)+\min(\gamma,\delta) = \min(\alpha+\gamma,\alpha+\delta,\beta+\gamma,\beta+\delta) \quad (6)$$

$$\max(\alpha,\beta)+\max(\gamma,\delta) = \max(\alpha+\gamma,\alpha+\delta,\beta+\gamma,\beta+\delta) \quad (7)$$

$$\max(\alpha,\beta) = -\min(-\alpha,-\beta) \quad (8)$$

$$\rho\max(\alpha,\beta) = \max(\rho\alpha,\rho\beta) \quad (9)$$

$$\rho\min(\alpha,\beta) = \min(\rho\alpha,\rho\beta) \ . \quad (10)$$

Theorem 3.1 *Any MMPS function $f : \mathbb{R}^n \to \mathbb{R}$ can be rewritten in the min-max canonical form*

$$f = \min_{i=1,\ldots,K} \max_{j=1,\ldots,n_i} (\alpha_{(i,j)}^T x + \beta_{(i,j)}) \quad (11)$$

or in the max-min canonical form

$$f = \max_{i=1,\ldots,L} \min_{j=1,\ldots,m_i} (\gamma_{(i,j)}^T x + \delta_{(i,j)}) \quad (12)$$

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 prove the theorem for 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. (1)) are in min-max canonical form.
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[3]: $f = \min(\max(f_1,f_2),\max(f_3,f_4))$ and $g = \min(\max(g_1,g_2),\max(g_3,g_4))$. Now we show that $\max(f,g)$, $\min(f,g)$, $f+g$ and βf with $\beta \in \mathbb{R}$ can again be written in min-max canonical form:

$$\begin{aligned}
&\max(f,g)\\
&\quad = \max[\min(\max(f_1,f_2),\max(f_3,f_4)),\\
&\qquad\quad \min(\max(g_1,g_2),\max(g_3,g_4))]\\
&\quad = \max[\max(\min(f_1,f_3),\min(f_1,f_4),\min(f_2,f_3),\\
&\qquad\quad \min(f_2,f_4)),\max(\min(g_1,g_3),\\
&\qquad\quad \min(g_1,g_4),\min(g_2,g_3),\min(g_2,g_4))] \quad \text{(by (4))}\\
&\quad = \max(\min(f_1,f_3),\min(f_1,f_4),\min(f_2,f_3),\\
&\qquad\quad \min(f_2,f_4),\min(g_1,g_3),\\
&\qquad\quad \min(g_1,g_4),\min(g_2,g_3),\min(g_2,g_4))\\
&\quad = \min(\max(f_1,f_1,f_2,f_2,g_1,g_1,g_2,g_2),\\
&\qquad\quad \max(f_1,f_1,f_2,f_2,g_1,g_1,g_2,g_4),\ldots,\\
&\qquad\quad \max(f_3,f_4,f_3,f_4,g_3,g_4,g_3,g_4))\\
&\qquad\qquad \text{(since max is distributive w.r.t. min)}
\end{aligned}$$

$$\begin{aligned}
&\min(f,g)\\
&\quad = \min[\min(\max(f_1,f_2),\max(f_3,f_4)),\\
&\qquad\quad \min(\max(g_1,g_2),\max(g_3,g_4))]\\
&\quad = \min(\max(f_1,f_2),\max(f_3,f_4),\max(g_1,g_2),\max(g_3,g_4))
\end{aligned}$$

$$\begin{aligned}
&f+g\\
&\quad = \min(\max(f_1,f_2),\max(f_3,f_4))+\\
&\qquad\quad \min(\max(g_1,g_2),\max(g_3,g_4))\\
&\quad = \min(\max(f_1,f_2)+\max(g_1,g_2),\\
&\qquad \max(f_1,f_2)+\max(g_3,g_4),\max(f_3,f_4)+\max(g_1,g_2),\\
&\qquad\quad \max(f_3,f_4)+\max(g_3,g_4)) \quad \text{(by (6))}\\
&\quad = \min(\max(f_1+g_1,f_1+g_2,f_2+g_1,f_2+g_2),\\
&\qquad \max(f_1+g_3,f_1+g_4,f_2+g_3,f_2+g_4),\\
&\qquad \max(f_3+g_1,f_3+g_2,f_4+g_1,f_4+g_2),\\
&\qquad \max(f_3+g_3,f_3+g_4,f_4+g_3,f_4+g_4)) \quad \text{(by (7))}.
\end{aligned}$$

For $\beta \geqslant$ we have

$$\begin{aligned}
\beta f &= \beta\min(\max(f_1,f_2),\max(f_3,f_4))\\
&= \min(\max(\beta f_1,\beta f_2),\max(\beta f_3,\beta f_4)) \quad \text{(by (9),(10))}
\end{aligned}$$

and for $\beta < 0$ we have

$$\begin{aligned}
\beta f &= \beta\min(\max(f_1,f_2),\max(f_3,f_4))\\
&= -|\beta|\min(\max(f_1,f_2),\max(f_3,f_4))\\
&= -\min(\max(|\beta|f_1,|\beta|f_2),\max(|\beta|f_3,|\beta|f_4)) \ \text{(by (9),(10))}\\
&= \max(-\max(|\beta|f_1,|\beta|f_2),-\max(|\beta|f_3,|\beta|f_4)) \quad \text{(by (8))}\\
&= \max(\min(-|\beta|f_1,-|\beta|f_2),\min(-|\beta|f_3,-|\beta|f_4)) \ \text{(by (8))}\\
&= \max(\min(\beta f_1,\beta f_2),\min(\beta f_3,\beta f_4))\\
&= \min(\max(\beta f_1,\beta f_3),\max(\beta f_1,\beta f_4),\\
&\qquad \max(\beta f_2,\beta f_3),\max(\beta f_2,\beta f_4)) \quad \text{(by (5))}. \quad \square
\end{aligned}$$

[3]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.

4 MPC for MMPS systems

In this section we give a short overview of the main results of [6, 8] in which we have extended the MPC framework to MMPS systems. We apply these results to the model (2)–(3), which — in contrast to [6, 8] — also has uncontrollable inputs. Related results can be found in [2]. More extensive information on conventional MPC for (linear and nonlinear) discrete-time systems can be found in [4, 12] and the references therein.

Consider the deterministic model (2)–(3). Note that this model does not include modeling errors or uncertainty. However, since MPC uses a receding finite horizon approach, we can regularly update the model and the state estimate as new measurements become available.

In MPC we compute at each step k an optimal control input that minimizes a cost criterion over the period $[k, k+N_\mathrm{p}-1]$ where N_p is the prediction horizon. For the system (2)–(3) we can make an estimate $\hat{y}(k+j|k)$ of the output at step $k+j$ based on the state[4] $x(k-1)$, the future controllable inputs $u(k+i)$, $i=0,\ldots,j$, and (estimates) of the future uncontrollable inputs $v(k+i)$, $i=0,\ldots,j$. Using successive substitution, we obtain an expression of the form $\hat{y}(k+j|k) = F_j(x(k-1), u(k), \ldots, u(k+j), v(k), \ldots, v(k+j))$ for $j=0,\ldots,N_\mathrm{p}-1$. Clearly, $\hat{y}(k+j|k)$ is an MMPS function of $x(k-1), u(k), \ldots, u(k+j), v(k), \ldots, v(k+j)$.

The cost criterion $J(k) = J_\mathrm{out}(k) + \lambda J_\mathrm{in}(k)$ used in MMPS-MPC reflects the reference tracking error (J_out) and the control effort (J_in) where λ is a nonnegative weight parameter. Let r denote the reference or due date signal. Define the vectors

$$\begin{aligned}
\tilde{u}(k) &= \begin{bmatrix} u^T(k) & \ldots & u^T(k+N_\mathrm{p}-1) \end{bmatrix}^T \\
\tilde{v}(k) &= \begin{bmatrix} v^T(k) & \ldots & v^T(k+N_\mathrm{p}-1) \end{bmatrix}^T \\
\tilde{y}(k) &= \begin{bmatrix} \hat{y}^T(k|k) & \ldots & \hat{y}^T(k+N_\mathrm{p}-1|k) \end{bmatrix}^T \\
\tilde{r}(k) &= \begin{bmatrix} r^T(k) & \ldots & r^T(k+N_\mathrm{p}-1) \end{bmatrix}^T \ .
\end{aligned}$$

We consider the following output and input cost functions[5]:

$$J_{\mathrm{out},1}(k) = \|\tilde{y}(k) - \tilde{r}(k)\|_1, \quad J_{\mathrm{in},1}(k) = \|\tilde{u}(k)\|_1, \tag{13}$$

$$J_{\mathrm{out},\infty}(k) = \|\tilde{y}(k) - \tilde{r}(k)\|_\infty, \quad J_{\mathrm{in},\infty}(k) = \|\tilde{u}(k)\|_\infty \ . \tag{14}$$

Note that these cost functions are also MMPS functions (recall that we have $|x| = \max(x, -x)$ for all $x \in \mathbb{R}$). In fact, the results presented below hold for any cost criterion that is an MMPS function of $\tilde{y}(k)$ and $\tilde{u}(k)$.

In practical situations, there will be constraints on the input and output signals (caused by limited capacity of buffers, limited transportation rates, saturation, etc.) In general, this is reflected in a nonlinear constraint of the form $C_\mathrm{c}(k, \tilde{u}(k), \tilde{v}(k), \tilde{y}(k)) \geqslant 0$.

The MMPS-MPC problem at step k consists in minimizing $J(k)$ over all possible future (controllable) input sequences subject to the constraints. To reduce the complexity of the optimization problem a control horizon N_c is introduced in MPC: for discrete-event systems we could, e.g., take the rate of change of controllable input to be constant beyond event step $k+N_\mathrm{c}$ (see [8]):

$$\Delta u(k+j) = \Delta u(k+N_\mathrm{c}-1) \ \text{ for } j = N_\mathrm{c}, \ldots, N_\mathrm{p}-1, \tag{15}$$

where $\Delta u(k) = u(k) - u(k-1)$. For a manufacturing system, this implies that the feeding rate of the raw material is constant after $k+N_\mathrm{c}$. Alternatively, we can set the controllable input constant as is done in "conventional" MPC:

$$u(k+j) = u(k+N_\mathrm{c}-1) \ \text{ for } j = N_\mathrm{c}, \ldots, N_\mathrm{p}-1. \tag{16}$$

In addition to a decrease in the number of optimization parameters and thus also the computational burden, a smaller control horizon N_c also gives a smoother control signal, which is often desired in practical situations.

MPC uses a receding horizon principle. This means that after computation of the optimal control sequence $u(k), u(k+1), \ldots, u(k+N_\mathrm{c}-1)$, only the first control sample $u(k)$ will be implemented, subsequently the horizon is shifted one step, next the model and the state are updated using new information from the measurements, and a new MPC optimization is performed for step $k+1$.

5 Algorithms for the MMPS-MPC optimization problem

5.1 Nonlinear optimization

In general the MMPS-MPC optimization problem is a nonlinear, nonconvex optimization problem. In [6, 8] we have discussed some algorithms to solve the MMPS-MPC optimization problem: we can use multi-start nonlinear optimization based on sequential quadratic programming (SQP), or we can use a method based on the extended linear complementarity problem (ELCP). 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 to) the global minimum. In addition, the objective functions that appear in the MMPS-MPC optimization problem are nondifferentiable and piecewise-affine (if we use the cost criteria given in (13)–(14) or in [9]), which makes the SQP approach less suitable for them. The main disadvantage of the ELCP approach is that the execution time of the algorithm increases exponentially as

[4] We assume that at step k the current state can be measured, estimated or predicted using previous measurements.

[5] In conventional MPC usually quadratic cost functions of the form $J_\mathrm{out}(k) = \|\tilde{y}(k) - \tilde{r}(k)\|_2^2$ and $J_\mathrm{in}(k) = \|\tilde{u}(k)\|_2^2$ are used. In a discrete-event context, however, other choices are more appropriate (see [8, 9]).

the size of the problem increases. This implies that this approach is not feasible if N_c or the number of inputs and outputs of the system are large.

An alternative option consists in transforming the MMPS system into a mixed-logic (MLD) system [2] since MMPS systems are equivalent to MLD systems [10]. 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.

5.2 A new algorithm

We assume that the cost criterion is an MMPS function of $\tilde{y}(k)$ and $\tilde{u}(k)$ (e.g., one of the cost criteria given in (13)–(14) or a linear combination of them). Note that the estimate of the future output $\tilde{y}(k)$ is also an MMPS function of $x(k-1)$, $\tilde{v}(k)$ and $\tilde{u}(k)$. So if we substitute $\tilde{y}(k)$ in the expression for $J(k)$, we finally obtain an MMPS function as objective function. From Theorem 3.1 it follows that this objective function can be written in min-max canonical form:

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

for appropriately defined integers $\ell, n_1, \ldots, n_\ell$, vectors $\alpha_{(i,j)}$ and integers $\beta_{(i,j)}$. Note that $\alpha_{(i,j)}$ and $\beta_{(i,j)}$ depend on k via $x(k-1)$, $\tilde{v}(k)$ or $\tilde{r}(k)$; however, for ease of notation, we drop the index k from $\alpha_{(i,j)}$ and $\beta_{(i,j)}$. Although in general the expression obtained by straightforwardly applying the manipulations of the proof of Theorem 3.1 may contain a large number of affine arguments $\alpha_{(i,j)}^T \tilde{u}(k) + \beta_{(i,j)}$, often many of these terms are redundant and can thus be removed. This reduces the number of affine arguments.

The derivation below is similar to the cutting-plane algorithm for convex optimization (see, e.g., [3]). Hence, it requires constraints that are linear (or convex) in $\tilde{u}(k)$. Note that the control horizon constraints (15) and (16) satisfy this condition. However, even if the original MPC constraint $C_c(k, \tilde{u}(k), \tilde{v}(k), \tilde{y}(k)) \geqslant 0$ 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 constraints on the input $\tilde{u}(k)$:

$$P\tilde{u}(k) + q \geqslant 0 \ . \tag{17}$$

Note that in practice such constraints occur if we have to guarantee that the control signal $\tilde{u}$ or the control signal rate $\Delta\tilde{u}$ stay within certain bounds.

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

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

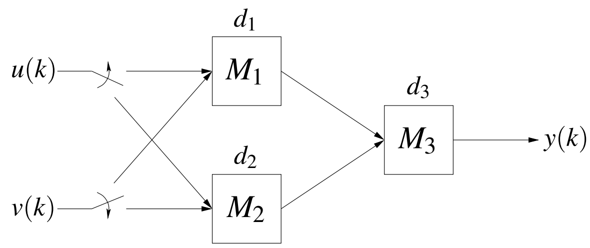

Figure 1. A simple manufacturing system.

$$\text{subject to } P\tilde{u}(k) + q \geqslant 0 \, .$$

or equivalently

$$\min_{i=1,\ldots,\ell} \min_{\tilde{u}(k)} \max_{j=1,\ldots,n_i} (\alpha_{(i,j)}^T \tilde{u}(k) + \beta_{(i,j)}) \tag{18}$$

$$\text{subject to } P\tilde{u}(k) + q \geqslant 0 \ . \tag{19}$$

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

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

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

$$\min t \tag{20}$$

$$\text{subject to} \begin{cases} t \geqslant \alpha_{(i,j)}^T \tilde{u}(k) + \beta_{(i,j)} & \text{for } j = 1, \ldots, n_i \\ P\tilde{u}(k) + q \geqslant 0 \ . \end{cases} \tag{21}$$

This problem can be solved efficiently using a simplex method or an interior-point algorithm. To obtain the solution of (18)–(19), we solve (20)–(21) for $i = 1, \ldots, \ell$ and afterward we select the solution $\tilde{u}_{(i)}^{\text{opt}}(k)$ for which $\max_{j=1,\ldots,n_i} (\alpha_{(i,j)}^T \tilde{u}_{(i)}^{\text{opt}}(k) + \beta_{(i,j)})$ is the smallest. This results in an algorithm to solve the MMPS-MPC problem that is much more efficient than the SQP or ELCP approach.

6 Worked example

Consider the manufacturing system of Figure 1, which consists of 3 processing units M_1, M_2, and M_3 with processing times d_1, d_2 and d_3. Raw material is coming from two sources: from an external provider, over which we have no control, and from a source for which we can completely control the release times (e.g., a storage unit with a large capacity so that its stock level never runs down to zero). Following the convention of Section 2 the time instants at which the kth batch of raw material from the controllable source and the external source arrives at the system are denoted by $u(k)$ and $v(k)$ respectively. The raw material from both sources can processed by either M_1 or M_2, which perform similar tasks. However, M_2 is slower than M_1. Therefore, the first part of the raw material for the kth product is processed on M_2 and the second part on M_1[6]. This implies that the kth batch of raw material destined for M_1 arrives at the production unit at time instant $t = \max(u(k), v(k)$, and that the kth batch destined for M_2 arrives at time instant $t = \min(u(k), v(k)$. The intermediate components generated by M_1 and M_2 are sent to M_3 where assembly takes places.

We assume that the transportation times in the manufacturing system are negligible, and that in between the production units there are storage buffer with a sufficiently large capacity, so that no buffer overflows occur. The time instant at which processing unit M_i starts processing the kth batch is denoted by $x_i(k)$, and $y(k)$ is the time instant at which the kth finished product leaves the system. Assume that each production unit starts working for the kth time as soon as the raw material is available and as soon as the production unit has finished processing the previous part. Hence,

$$\begin{aligned}
x_1(k) &= \max\big(x_1(k-1)+d_1, \max(u(k),v(k))\big)\\
&= \max\big(x_1(k-1)+d_1, u(k), v(k)\big)\\
x_2(k) &= \max\big(x_2(k-1)+d_2, \min(u(k),v(k))\big)\\
x_3(k) &= \max\big(x_3(k-1)+d_3, x_1(k)+d_1, x_2(k)+d_2\big)\\
y(k) &= x_3(k)+d_3 \quad .
\end{aligned}$$

Now assume for the sake of simplicity that $d_3 \ll d_1, d_2$ and that M_3 never is a bottleneck (i.e., we always have $x_3(k-1)+d_3 \leqslant x_1(k)+d_1$ and $x_3(k-1)+d_3 \leqslant x_2(k)+d_2$). Then the model description reduces to:

$$\begin{aligned}
x_1(k) &= \max\big(x_1(k-1)+d_1, u(k), v(k)\big)\\
x_2(k) &= \max\big(x_2(k-1)+d_2, \min(u(k),v(k))\big)\\
y(k) &= \max\big(x_1(k)+d_1+d_3, x_2(k)+d_2+d_3\big) \quad .
\end{aligned}$$

We will apply MPC to this system. Let[7] $N_\mathrm{c} = N_\mathrm{p} = 2$, and assume that the MPC objective function $J(k)$ is given by

$$\begin{aligned}
J(k) &= J_{\mathrm{out},\infty}(k) - \lambda J_{\mathrm{in},1}(k)\\
&= \max(\max(y(k)-r(k),0),\ \max(y(k+1)-r(k+1),0))\\
&\qquad -\lambda(u(k)+u(k+1))\\
&= \max(y(k)-r(k), y(k+1)-r(k+1), 0)\\
&\qquad -\lambda u(k) - \lambda u(k+1)\\
&= \max(y(k)-r(k)-\lambda u(k)-\lambda u(k+1), y(k+1)-r(k+1)\\
&\qquad -\lambda u(k)-\lambda u(k+1), -\lambda u(k)-\lambda u(k+1)) \quad ,
\end{aligned}$$

where $\lambda > 0$ is a weighting parameter and $r(k)$ the due date signal. Furthermore, we have assumed that the first release time for the controllable input source occurs after time instant $t = 0$ (Hence, $u(k) \geqslant 0$ for all k). Note that the objective function $J(k) = J_{\mathrm{out},\infty}(k) - \lambda J_{\mathrm{in},1}(k)$ corresponds to a trade-off between minimizing the maximal tardiness and maximizing the (controllable) input time instants.

Now it is easy to verify that $y(k)$ and $y(k+1)$ can be expressed as the following functions of the current state $x(k-1)$ and the future inputs $u(k)$, $v(k)$, $u(k+1)$, $v(k+1)$:

$$\begin{aligned}
y(k) = \max\big(&x_1(k-1)+2d_1+d_3, u(k)+d_1+d_3,\\
&v(k)+d_1+d_3, x_2(k-1)+2d_2+d_3,\\
&\min(u(k)+d_2+d_3, v(k)+d_2+d_3)\big)\\
y(k+1) = \max\big(&x_1(k-1)+3d_1+d_3, u(k)+2d_1+d_3,\\
&v(k)+2d_1+d_3, u(k+1)+d_1+d_3,\\
&v(k+1)+d_1+d_3, x_2(k-1)+3d_2+d_3,\\
&\min(u(k)+2d_2+d_3, v(k)+2d_2+d_3),\\
&\min(u(k+1)+d_2+d_3, v(k+1)+d_2+d_3)\big) \quad .
\end{aligned}$$

Hence,

$$\begin{aligned}
J(k) = \max\Big(&x_1(k-1)+2d_1+d_3-r(k)-\lambda u(k)-\lambda u(k+1),\\
&d_1+d_3-r(k)+(1-\lambda)u(k)-\lambda u(k+1),\\
&v(k)+d_1+d_3-r(k)-\lambda u(k)-\lambda u(k+1),\\
&x_2(k-1)+2d_2+d_3-r(k)-\lambda u(k)-\lambda u(k+1),\\
&x_1(k-1)+3d_1+d_3-r(k+1)-\lambda u(k)-\lambda u(k+1),\\
&2d_1+d_3-r(k+1)+(1-\lambda)u(k)-\lambda u(k+1),\\
&v(k)+2d_1+d_3-r(k+1)-\lambda u(k)-\lambda u(k+1),\\
&d_1+d_3-r(k+1)-\lambda u(k)+(1-\lambda)u(k+1),\\
&v(k+1)+d_1+d_3-r(k+1)-\lambda u(k)-\lambda u(k+1),\\
&x_2(k-1)+3d_2+d_3-r(k+1)-\lambda u(k)-\lambda u(k+1),\\
&-\lambda u(k)-\lambda u(k+1),\\
&\min\big(d_2+d_3-r(k)+(1-\lambda)u(k)-\lambda u(k+1),\\
&\quad v(k)+d_2+d_3-r(k)-\lambda u(k)-\lambda u(k+1)\big),\\
&\min\big(2d_2+d_3-r(k+1)+(1-\lambda)u(k)-\lambda u(k+1),\\
&\quad v(k)+2d_2+d_3-r(k+1)-\lambda u(k)-\lambda u(k+1)\big),\\
&\min\big(d_2+d_3-r(k+1)-\lambda u(k)+(1-\lambda)u(k+1),\\
&\quad v(k+1)+d_2+d_3-r(k+1)-\lambda u(k)-\lambda u(k+1)\big)\Big). \qquad (22)
\end{aligned}$$

Note that this is an MMPS expression in max-min canonical form. In order to be able to apply the method of Section 5.2 we have to rewrite $J(k)$ into min-max canonical form. Equation (22) can be written compactly as

$$J(k) = \max(t_1, t_2, \ldots, t_{11}, \min(t_{12}, t_{13}), \min(t_{14}, t_{15}), \min(t_{16}, t_{17}))$$

where $t_1, \ldots, t_{17}$ are appropriately defined affine functions of $x_1(k-1)$, $x_2(k-1)$, $r(k)$, $u(k)$, $v(k)$, $u(k+1)$, $v(k+1)$. The min-max canonical form of $J(k)$ is then given by

$$J(k) = \min\big(\max(t_1, t_2, \ldots, t_{11}, t_{12}, t_{14}, t_{16}),$$

[6] So if $v(k) < u(k)$, then the kth batch of raw material coming from the external (uncontrollable) source is processed on M_2 and the raw material coming from the controllable source on M_1; if $u(k) < v(k)$, the reverse holds.

[7] We take such low values for N_c and N_p so that the size of the analytical expression for the MPC objective function $J(k)$ is still small so that it can be listed explicitly. In practice, larger values would be more appropriate.

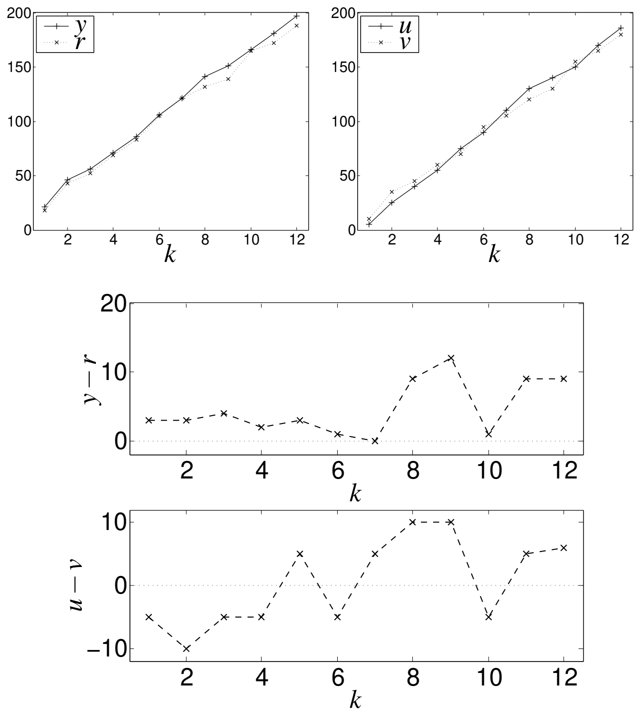

Figure 2. The closed-loop MPC output signal y, the due date signal r, the inputs signals u and v, and the difference signals $y-r$ and $u-v$.

$$
\begin{aligned}
&\max(t_1,t_2,\ldots,t_{11},t_{12},t_{14},t_{17}), \max(t_1,t_2,\ldots,t_{11},t_{12},t_{15},t_{16}),\\
&\max(t_1,t_2,\ldots,t_{11},t_{12},t_{15},t_{17}), \max(t_1,t_2,\ldots,t_{11},t_{13},t_{14},t_{16}),\\
&\max(t_1,t_2,\ldots,t_{11},t_{13},t_{14},t_{17}), \max(t_1,t_2,\ldots,t_{11},t_{13},t_{15},t_{16}),\\
&\max(t_1,t_2,\ldots,t_{11},t_{13},t_{15},t_{17})\big) \enspace .
\end{aligned}
$$

Hence, the optimal MPC strategy for step k can be computed by solving 8 linear programming problems, and by selecting the overall optimum.

Let us now compute the closed-loop MPC input signal over a simulation period $[1,12]$ with $\lambda = 0.1$, $d_1 = 10$, $d_2 = 15$, $d_3 = 1$, $x_1(0) = x_2(0) = -20$ (this corresponds to all production units being idle at the beginning of step $k = 1$ since $u(k), v(k) \geqslant 0$ for all k), with

$$
\begin{aligned}
\{r(k)\}_{k=1}^{12} &= 18,\ 43,\ 52,\ 69,\ 83,\ 105,\ 121,\ 132,\ 139,\\
&\quad\ 165,\ 172,\ 188\\
\{v(k)\}_{k=1}^{12} &= 10,\ 35,\ 45,\ 60,\ 70,\ 95,\ 105,\ 120,\ 130,\\
&\quad\ 155,\ 165,\ 180,
\end{aligned}
$$

and with the constraints $u(1) \geqslant 0$ and $5 \leqslant \Delta u(k) \leqslant 20$ for $k = 1,\ldots,12$. In Figure 2 we have plotted the closed-loop MPC input signal u, the uncontrollable input signal from the external source, the output signal y, the due date signal r, and the difference signals $y-r$ and $u-v$.

7 Conclusions

We have considered model predictive control (MPC) for max-min-plus-scaling (MMPS) systems. In general, this leads to nonlinear nonconvex optimization problems. We have presented a method based on canonical forms for MMPS functions and similar to the cutting-plane convex optimization algorithm to solve these optimization problems. More specifically, the approach consists in solving several linear programming problems and afterward selecting the solution that yields the smallest objective function. This yields a method that is more efficient than just applying nonlinear optimization as was done in previous research.

Acknowledgment

Research partially funded by the Dutch Technology Foundation STW project "Model predictive control for hybrid systems" (DMR.5675).

References

[1] F. Baccelli, G. Cohen, G. Olsder, and J. Quadrat. *Synchronization and Linearity*. Wiley, 1992.

[2] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, Mar. 1999.

[3] S. Boyd and C. Barratt. *Linear Controller Design: Limits of Performance*. Prentice Hall, 1991.

[4] E. Camacho and C. Bordons. *Model Predictive Control in the Process Industry*. Springer, 1995.

[5] R. Cuninghame-Green. *Minimax Algebra*. Springer, 1979.

[6] B. De Schutter and T. van den Boom. Model predictive control for max-min-plus systems. In R. Boel and G. Stremersch, editors, *Discrete Event Systems: Analysis and Control*, pages 201–208. Kluwer, 2000.

[7] B. De Schutter and T. van den Boom. On model predictive control for max-min-plus-scaling discrete event systems. Tech. Rep., Control Systems Eng., Fac. ITS, Delft Univ. of Techn., The Netherlands, June 2000.

[8] B. De Schutter and T. van den Boom. Model predictive control for max-min-plus-scaling systems. In *Proc. 2001 American Control Conf.*, pages 319–324, Arlington, June 2001.

[9] B. De Schutter and T. van den Boom. Model predictive control for max-plus-linear discrete event systems. *Automatica*, 37(7):1049–1056, July 2001.

[10] W. Heemels, B. De Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *Automatica*, 37(7):1085–1091, July 2001.

[11] D. Leenaerts and W. van Bokhoven. *Piecewise Linear Modeling and Analysis*. Kluwer, 1998.

[12] J. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, 2002.