Conditional Reachability Of Uncertain Max Plus Linear Systems ⋆

Renato Markele Ferreira Cândido [a,b], Laurent Hardouin [b], Mehdi Lhommeau [b],
Rafael Santos Mendes [a,b],

[a] *Department of Computer Engineering and Industrial Automation - DCA, School of Electrical and Computer Engineering, University of Campinas - UNICAMP - 13083-970 Campinas, SP, Brazil.*

[b] *Laboratoire Angevin de Recherche en Ingénierie des Systèmes - ISTIA - Université d'Angers, 62 av. Notre Dame du Lac, 49100 Angers, France.*

Abstract

The reachability analysis problem of Max Plus Linear (MPL) systems has been properly solved using the Difference-Bound Matrices approach. In this work, the same approach is considered in order to solve the reachability analysis problem of MPL systems subjected to bounded noise, disturbances and/or modeling errors, called uncertain MPL (uMPL) systems. Moreover, using the previous results on uMPL reachability analysis, we solve the *conditional reachability problem*, herein defined as the support calculation of the probability density function involved in the stochastic filtering problem.

Key words: Reachability, Conditional Reachability, Max-Plus Linear Systems, Piece-wise affine Systems, Difference-Bound Matrices, Interval Analysis.

1 Introduction

Discrete event systems subject to only synchronization and time delay phenomena are a class of dynamic systems which can be described in a linear way in the max-plus algebra. The max-plus algebra is an idempotent semiring, an algebraic structure also called dioid [8], in which the operations of sum ($\oplus$) and product ($\otimes$) are defined as the maximization and addition, respectively. Synchronization phenomena are modeled thanks to maximization: the start of a task waits for the completion of the preceding tasks, while the delay phenomena are depicted thanks to the classical sum: the completion time of a task is equal to the starting time plus the task duration.

The Max Plus Linear (MPL) equations are used to model manufacturing systems, telecommunication networks, railway networks, and parallel computing [8,11]. The linearity property has advantaged the emergence of a specific theory for the performance analysis [26] and the control of these systems, e.g., optimal open loop control [15,29] and optimal state-feedback control. Among closed-loop strategies we can cite the model matching problem [30] and the control strategies allowing the state to stay in a specific state subspace or semimodule [7,21,28,33,36].

The MPL systems may be subjected to noise and disturbances, which should be taken into account in order to avoid tracking error or closed loop instability [39]. In general, these perturbations are max-plus-multiplicative and appear as uncertainties in the max plus model parameters. As a result the system matrices are uncertain. The Stochastic Max Plus Linear (SMPL) systems are defined as MPL systems where the matrices entries are characterized by random variables [17,24,27,37,39]. In this work, although the probabilistic aspects of the uncertainties are not considered, we are interested in systems where the uncertain parameters can vary over a known interval. Formally, we define the uncertain Max-Plus Linear (uMPL) systems as *nondeterministic* MPL systems where, at each event step, the entries of the sys-

---

⋆ This work was supported by the Brazilian Agencies CAPES (99999.002340/2015-01), CNPq (164765/2013-1) and FAPESP (2015/19384-9).

Email addresses: `rmarkele@dca.fee.unicamp.br` (Renato Markele Ferreira Cândido), `laurent.hardouin@univ-angers.fr` (Laurent Hardouin), `mehdi.lhommeau@univ-angers.fr` (Mehdi Lhommeau), `rafael@dca.fee.unicamp.br` (Rafael Santos Mendes).

tem matrices can, independently, take an arbitrary value within a real interval [13,14,35], as detailed in section 2.1.

To assess whether the system reaches a certain state from a set of initial conditions is of great interest in many applications and concerns the reachability analysis. In [20] residuation is used to determine if a state is reachable from a single initial condition. In [19], it is shown that if the initial set is a rational semimodule the reachable set is also a rational semimodule. These authors mention that this set has a "simple shape" and suggest that an efficient numerical method remains to be designed. In [32] reachability analysis of timed automata is tackled by considering max-plus polyhedra, a more general class of set than semimodules. For a more exhaustive presentation on max-plus polyhedra, see [6].

In [3] the forward reachability problem is addressed by considering as initial set, the union of regions depicted as difference bound matrices (DBM) [16]. In [2], backward reachability analysis of autonomous MPL systems has also been studied by considering a final set depicted as union of DBM. In [4], these results have been extended to nonautonomous MPL systems.

As shown in [3], to describe an MPL system by means of DBM it is necessary to express it as a Piece-Wise Affine System (PWA). This is always possible [25] and it is done by partitioning the state space into regions in which the system can be modeled by affine equations (in classical algebra). The PWA system is simply the union of these affine systems and the key point is that each affine system and its corresponding active state space region can be independently represented by one DBM (this is detailed in section 2.3). The main advantage of this representation is the existence of many efficient algorithms for DBM manipulation and its drawback is the upsizing of the representation of an MPL system from one compact state equation to multiple DBM.

It should be remarked that, on one hand, [4] have proved that any region described as a max-plus polyhedron can also be described by a union of DBM. On the other hand, the complexity of the algorithms involving max-plus polyhedra are in general polynomial, while the complexity of the DBM approach critically depends on the number of PWA subsystems, which grows exponentially with the dimension of the system. Due to the exponential complexity, the DBM approach comfortably handles reachability computations for MPL models with *up to* twenty state variables, see [4, Sec. 5]. Approaches based on max-plus polyhedra seems to be a promising way to reduce the complexity of reachability computations for MPL systems and, therefore, to extend the dimension of the addressable problem. However, to the best of the authors' knowledge, there are no approaches based on max-plus polyhedra for solving the forward and the backward reachability problem for general MPL systems and such methods remain to be designed.

In this work, we aim to extend the DBM approach in order to analyze uMPL systems. It is shown that uMPL systems can be partitioned into components that can be fully represented by DBM and that it is efficient for reachability analysis of uMPL systems. Then, for the forward reachability analysis, given a set of initial conditions represented by a union of finitely many DBM, the sets of states that *may be* reached at each event step are computed. Similarly, for the backward reachability analysis, given a set of final conditions represented by a union of finitely many DBM, the sets of all states that *may lead* to the set of final conditions in a fixed number of steps can be computed.

Bayesian methods provide a rigorous general framework for dynamic state estimation problems [22]. The objective of the Bayesian state estimation is to construct the posterior probability density function (PDF) of the states based on all information available. It should be noted that the computation of the states PDF is quite difficult. Although these problems are very closely related, this paper only concerns the reachability problem and therefore the purpose is limited to the calculation of the support of the prior and the posterior state estimation, which does not require the use of probability measures (section 5). We define the *conditional reachability problem* as the support calculation of the posterior PDF of the uMPL system states. We assume that a sequence of measurements related to the state through an uMPL equation is given and then we show that this problem can be solved by using the previous results on reachability analysis of uMPL systems.

The paper is organized as follows: Section 2 recalls the MPL systems and their decompositions as PWA systems, as well as the DBM representation of PWA systems generated by MPL systems. Section 3 extends the PWA systems to uMPL systems. Section 4 presents reachability analysis for uMPL systems. Section 5 deals with the conditional reachability problem. Section 6 applies the results of the paper in order to solve the conditional reachability problem for a given uMPL system. Finally, Section 7 concludes the work.

2 Preliminaries

2.1 Max Plus Linear Systems

A set S, endowed with two internal operations: *sum*($\oplus$) and *product*($\otimes$) is a **dioid** or **idempotent semiring** if the sum is associative, commutative and idempotent (i.e. $a \oplus a = a$) and the product is associative and left and right distributive with respect to the sum [1]. The null (or

---

[1] the product is not necessarily commutative

zero) element, denoted by ε, is such that $\forall a \in S, a \oplus \varepsilon = a$, and the identity element, denoted by e, is such that $\forall a \in S,\ a \otimes e = e \otimes a = a$. Besides, the zero element is absorbing for the $\otimes$ operation (i.e. $\forall a \in S,\ a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$) [8, Def. 4.1]. In this algebraic structure, a partial order relation is defined by:

$$a \succeq b \Leftrightarrow a = a \oplus b. \tag{1}$$

Given these conditions, it appears that the set $\mathbb{R} \cup \{-\infty\} \cup \{\infty\}$ and the operations: $\alpha \oplus \beta \equiv \max\{\alpha, \beta\}$ and $\alpha \otimes \beta \equiv \alpha + \beta$, with $\varepsilon = -\infty$, $e = 0$, and with the convention that $\infty \otimes \varepsilon = \varepsilon$, is a dioid. Moreover, it can be stated that this is a **complete dioid** since it is closed for infinite sums and the left and right distributivity of the product extends to infinite sums [2]. This set is called **Max-Plus** semiring and noted by $\overline{\mathbb{R}}_{max}$. Note that $\overline{\mathbb{R}}_{max}$ is linearly ordered with respect to $\oplus$ and the order $\succeq$ in $\overline{\mathbb{R}}_{max}$ coincides with the usual linear order $\geq$.

The $\oplus$ and $\otimes$ operations can be extended to matrices as follows. If $A, B \in \overline{\mathbb{R}}_{max}^{n\times p}$ and $C \in \overline{\mathbb{R}}_{max}^{p\times q}$, then: $[A \oplus B]_{ij} = a_{ij} \oplus b_{ij}$ and $[A \otimes C]_{ij} = \bigoplus_{k=1}^{p} a_{ik} \otimes c_{kj}$.

The autonomous model of an MPL system is given by:

$$\mathbf{x}(k) = A \otimes \mathbf{x}(k-1), \tag{2}$$

where the entries of matrix $A \in \overline{\mathbb{R}}_{max}^{n\times n}$ are the parameters of the model, a_{ij} represents the minimal delay between two events. The variable $k \in \mathbb{N}$ is an event-number and the state vector $\mathbf{x} \in \overline{\mathbb{R}}_{max}^{n}$ is a **dater**, i.e, $\mathbf{x}(k)$ contains the k-th date of occurrence of each event of the system.

The nonautonomous model of an MPL system is defined as:

$$\mathbf{x}(k) = A \otimes \mathbf{x}(k-1) \oplus B \otimes \mathbf{u}(k), \tag{3}$$

where $\mathbf{u}$ is an external input and $B \in \overline{\mathbb{R}}_{max}^{n\times m}$.

Any nonautonomous MPL system can be transformed into an augmented autonomous MPL model by considering $F = (A\ B) \in \overline{\mathbb{R}}_{max}^{n\times(n+m)}$ and $\mathbf{y}(k-1) = \big(\mathbf{x}(k-1)^T\ \mathbf{u}(k)^T\big)^T$ [8, Sec. 2.5.4].

$$\mathbf{x}(k) = F \otimes \mathbf{y}(k-1). \tag{4}$$

To model uncertain systems, the entries of matrix A are assumed to be nondeterministic. This is consistent with the assumption that the entries of A are associated to the system delays, that are subject to variations due to disturbances. Formally, it is assumed that at each event step k the entries a_{ij} can, independently, take an arbitrary value within the real interval $[\underline{a}_{ij},\ \overline{a}_{ij}]$. The autonomous model of an uncertain MPL (uMPL) system is given by:

$$\mathbf{x}(k) = A(k) \otimes \mathbf{x}(k-1), \tag{5}$$

where $A(k) \in \overline{\mathbb{R}}_{max}^{n\times n}$ is an uncertain matrix whose entries are in intervals $[\underline{a}_{ij},\ \overline{a}_{ij}]$, with $\underline{a}_{ij},\ \overline{a}_{ij} \in \mathbb{R} \cup \{-\infty\}$.

Remark 1 *To assure FIFO (first in, first out) rule, the matrix* $A(k)$ *must satisfy* $A(k) \geq I_n$, *where* I_n *is the identity matrix in* $\overline{\mathbb{R}}_{max}^{n\times n}$.

Remark 2 *On real physical systems the uncertainty of the system matrix entries may be coupled. Indeed, the equation* $\mathbf{x}(k) = \mathcal{A}_0(k) \otimes \mathbf{x}(k) \oplus \mathcal{A}_1(k) \otimes \mathbf{x}(k-1)$ *frequently represents real systems and is equivalent to equation* (5) *with* $A(k) = \mathcal{A}_0^*(k) \oplus \mathcal{A}_1(k)$. *In this case, even if the independence of the entries of* $\mathcal{A}_0(k)$ *and* $\mathcal{A}_1(k)$ *is assumed, in general, the entries of* $A(k)$ *will be coupled. Therefore, assuming the independence of the entries of* $A(k)$ *will lead to conservative results. This work will be focused in systems modeled by equation* (5). *However the results can be extended to systems of the type* $\mathbf{x}(k) = \mathcal{A}_0(k) \otimes \mathbf{x}(k) \oplus \mathcal{A}_1(k) \otimes \mathbf{x}(k-1)$ *(see end of section 3).*

Remark 3 *The approach presented in this work compute the* exact *set of states that can be reached from a initial set via the uMPL system given by* (5), *in a given number of event steps. Thus if the real system can be modeled as the uMPL system given by* (5) *the approach leads to the exact reach sets. On the other hand, if the uMPL system is a conservative representation for the real system, the approach compute an* over approximation *for the reach sets.*

2.2 Piece-wise Affine Systems

This section presents a procedure to express an MPL system as a Piece-wise Affine (PWA) system [4, Sec. 2.2]. Classical PWA systems are described in [10], [25] and [38]. The PWA systems are characterized by a partition of the state space, defined by finitely many linear inequalities, and by affine equations that are active within each component of the partition.

Every MPL model can be expressed as a PWA system in the event domain [25]. Consider a *generic* MPL system given by:

$$\mathbf{z}(k) = A \otimes \mathbf{x}(k-1), \tag{6}$$

where $A \in \overline{\mathbb{R}}_{max}^{n\times p}$ and $\mathbf{z}$ and $\mathbf{x}$ are vectors of appropriate dimensions.

---

[2] For complete dioids, the order relation (1) can be written as: $a \succeq b \Leftrightarrow a = a \oplus b \Leftrightarrow b = a \wedge b$, where $a \wedge b$ is the greatest lower bound of a and b.

Remark 4 *Equation* (6) *is generic in the sense that it can represent either an autonomous MPL system* ($p = n$) *or an nonautonomous MPL system* ($p = n + m$)*.*

The PWA system representing (6) can be constructed from the coefficients $\mathbf{g} = (g_1, \cdots, g_n) \in \{1, \cdots, p\}^n$ [4, Sec. 2.2]. Each $\mathbf{g}$ is associated with a dynamics and a region $R_{\mathbf{g}}$ such that, for all $\mathbf{x} \in R_{\mathbf{g}}$, the element g_i corresponds to the index of the maximum term of the i-th system equation of (6), which can be expressed as $z_i(k) = \bigoplus_{j=1}^{p}\{a_{ij} \otimes x_j(k-1)\}$, i.e.,

$$a_{ig_i} \otimes x_{g_i}(k-1) = \bigoplus_{j=1}^{p}\{a_{ij} \otimes x_j(k-1)\}. \qquad (7)$$

From (1), equation (7) can be expressed as:

$$a_{ij} + x_j(k-1) \leq a_{ig_i} + x_{g_i}(k-1) \quad \forall j \in \{1, \cdots, p\}. \quad (8)$$

Therefore, the region $R_{\mathbf{g}}$ which represents the set of all $\mathbf{x} \in \overline{\mathbb{R}}^p_{max}$ that satisfies (8), is given by:

$$R_{\mathbf{g}} = \bigcap_{i=1}^{n} \bigcap_{\substack{j=1 \\ j \neq g_i}}^{p} \left\{ \mathbf{x} \in \overline{\mathbb{R}}^p_{max} : x_j - x_{g_i} \leq a_{ig_i} - a_{ij} \right\}. \quad (9)$$

From (7) and (8), the affine dynamics that is active in $R_{\mathbf{g}}$ is given by:

$$z_i(k) = x_{g_i}(k-1) + a_{ig_i}, \qquad 1 \leq i \leq n. \qquad (10)$$

Example 1 *Consider the following MPL system:*

$$\mathbf{x}(k) = \begin{pmatrix} 3 & 7 \\ 2 & 4 \end{pmatrix} \otimes \mathbf{x}(k-1).$$

According to (9) *we have the following regions* $R_{\mathbf{g}} \in \overline{\mathbb{R}}^2_{max}$*, for* $\mathbf{g} \in \{1,2\}^2 = \{(1,1), (1,2), (2,1), (2,2)\}$*.*

$$\begin{aligned}
R_{(1,1)} &= \{x_2 - x_1 \leq -4\} \cap \{x_2 - x_1 \leq -2\} = \{x_2 - x_1 \leq -4\}, \\
R_{(1,2)} &= \{x_2 - x_1 \leq -4\} \cap \{x_1 - x_2 \leq 2\} = \emptyset, \\
R_{(2,1)} &= \{x_1 - x_2 \leq 4\} \cap \{x_2 - x_1 \leq -2\} = \{-4 \leq x_2 - x_1 \leq -2\}, \\
R_{(2,2)} &= \{x_1 - x_2 \leq 4\} \cap \{x_1 - x_2 \leq 2\} = \{x_1 - x_2 \leq 2\}.
\end{aligned}$$

Thus, according to (10)*, the corresponding PWA system is given by:*

$$\mathbf{x}(k) = \begin{cases} \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix} \mathbf{x}(k-1) + \begin{pmatrix} 3 \\ 2 \end{pmatrix} & \textit{if} \quad \mathbf{x}(k-1) \in R_{(1,1)}, \\ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \mathbf{x}(k-1) + \begin{pmatrix} 7 \\ 2 \end{pmatrix} & \textit{if} \quad \mathbf{x}(k-1) \in R_{(2,1)}, \\ \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} \mathbf{x}(k-1) + \begin{pmatrix} 7 \\ 4 \end{pmatrix} & \textit{if} \quad \mathbf{x}(k-1) \in R_{(2,2)}, \end{cases}$$

In [5, Algorithm 1] the algorithm describes a general procedure to construct a PWA system corresponding to an MPL system. The worst-case complexity of the algorithm is $\mathcal{O}(p^n(np + p^3))$ and the bottleneck resides in the worst-case cardinality of the collection of regions $R_{\mathbf{g}}$, given by p^n [4, Sec. 2.3].

Remark 5 *The worst-case cardinality of the number of regions can often be tightened: practically, each row i of an $n \times p$ matrix has $p'_i \leq p$ finite elements, thus the worst-case cardinality is $\prod_{i=1}^{n} p'_i \leq p^n$. Besides, many regions are empty then the number of regions is drastically smaller than the worst-case bound. In (Adzkiya et al. 2015a., Sec. 5.1), some experiments were carried out in order to test the efficiency of the approach: for any given n it was generated an $n \times n$ matrix A with 2 finite elements (generated between 1 and 100) that were randomly placed in each row. Table 1, presents the average number of regions and the average time to generate the PWA system over 10 experiments, for* $n \in \{10, 13, 16, 19\}$*. The experiments were run in a 12-core Intel Xeon 3.47 GHz PC with 24 GB of memory.*

Table 1
Computation of PWA systems (average over 10 experiments)

n	number of regions	generating time
10	7.01×10^2	4.73(s)
13	5.06×10^3	46.70 (s)
16	4.91×10^4	7.90 (min)
19	3.48×10^5	67.07 (min)

2.3 Difference Bound Matrices

The Difference Bound Matrices (DBM) [16] are one of the most efficient data structures for handling regions. Particularly, each component of a PWA system generated by an MPL system can be represented by a DBM. Following, we present a formal definition of DBM.

Definition 1 ***(Difference-Bound Matrix [16])*** *A DBM is a square matrix that represents the intersection of finitely many sets defined by* $x_i - x_j \bowtie_{i,j} \alpha_{i,j}$*, where*

$\bowtie_{i,j} \in \{<, \leq\}$[3] *and $i \neq j$ with $i, j \in \{0, ..., n\}$, $\alpha_{i,j} \in \mathbb{R} \cup \{+\infty\}$ is the upper bound. An artificial value x_0 is considered. It is assumed equal to 0 and it is used to represent bounds over a single variable, e.g., $x_i \leq \alpha_{i,0} \Leftrightarrow x_i - x_0 \leq \alpha_{i,0}$ or $x_i \geq -\alpha_{0,i} \Leftrightarrow x_0 - x_i \leq \alpha_{0,i}$.*

Therefore, a DBM is a matrix D in which $d_{i+1,j+1} = (\alpha_{i,j}, \bowtie_{i,j})$, represents the upper bound and the strictness of the sign of $x_i - x_j$ for $i, j \in \{0, ..., n\}$. From Definition 1, a DBM is a representation for a system of linear inequalities:

$$\begin{cases} x_i - x_j \bowtie_{ij} \alpha_{ij} \\ x_0 = 0 \end{cases} \quad i \neq j \text{ and } i, j \in \{0, ..., n\} \,. \tag{11}$$

Example 2 *The set $X = \{x_1 \in \mathbb{R} : 1 < x_1 \leq 4\}$ can be represented by the following DBM:*

$$D^{(X)} = \begin{array}{cc} \begin{array}{cc} x_0 & x_1 \end{array} & \\ \begin{pmatrix} (0, \leq) & (-1, <) \\ (4, \leq) & (0, \leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1 \end{array} \end{array}$$

Note that $d^{(X)}_{12} = (-1, <)$ represents $x_0 - x_1 < -1$ and $d^{(X)}_{21} = (4, \leq)$ represents $x_1 - x_0 \leq 4$.

The solution set of (11) is the **region** of the DBM D, noted by $\mathcal{R}(D)$. In general, the same region can be represented by different DBM. However, each DBM admits an equivalent and unique representation in canonical form [16, Th. 2]. By definition, if D^c is the canonical form of D, then d^c_{ij} is the cost of the least-cost path in D from i to j [16, Sec. 4.1]. The Floyd-Warshall algorithm, presented in [18], can be used to obtain the canonical-form representation of a DBM with a complexity that is cubic w.r.t. its dimension.

Remark 6 *The canonical DBM that represents the set $\{\mathbf{x} \in \mathbb{R}^n\}$ is an $(n+1) \times (n+1)$ matrix, noted by $D^{(\mathbb{R}^n)}$, which has entries equal to $(0, \leq)$ in the diagonal and $(\infty, <)$ elsewhere.*

Checking the emptiness of a DBM corresponds to check, over the potential graph, for circuits with a strictly negative weight, which corresponds to an unfeasible constraint in (11), i.e., a constraint with $\alpha_{i,i} < 0$ or $\alpha_{i,i} = 0$ and $\bowtie_{i,i} = \{<\}$. This can be achieved by the Bellman-Ford algorithm [9, Sec. 5], which is cubic w.r.t. the dimension of its input. However, if a DBM is in the canonical form, the complexity of checking its emptiness reduces to linear w.r.t. its dimension [5, Sec. 2.3.3]. Besides, if the DBM is in the canonical form, its projection onto a subset of its variables can be find by deleting the rows and columns corresponding to the complementary variables [16, Sec. 4.1].

In many application it is necessary to compute the intersection of DBM, which is again a DBM. Given two $n \times n$ DBM, $D^{(X_1)}$ and $D^{(X_2)}$, with entries $d^{(X_1)}_{ij} = (a_{ij}, \bowtie^{(X_1)}_{ij})$ and $d^{(X_2)}_{ij} = (b_{ij}, \bowtie^{(X_2)}_{ij})$, respectively, the intersection $D^{(X_1 \cap X_2)} = D^{(X_1)} \cap D^{(X_2)}$ is defined by:

$$d^{(X_1 \cap X_2)}_{ij} = \begin{cases} (a_{ij}, \bowtie^{(X_1)}_{ij}) & \text{if } a_{ij} < b_{ij} \\ & \text{or } (a_{ij} = b_{ij},\ \bowtie^{(X_1)}_{ij} \leq \bowtie^{(X_2)}_{ij}), \\ (b_{ij}, \bowtie^{(X_2)}_{ij}) & \text{otherwise.} \end{cases} \tag{12}$$

Another important operation is the Cartesian product of DBM. Given two DBM $D^{(X_1)}$ and $D^{(X_2)}$ with dimensions $(p+1) \times (p+1)$ and $(n+1) \times (n+1)$, respectively, the Cartesian product of its regions is given by $\mathcal{R}(D^{(X_1)}) \times \mathcal{R}(D^{(X_2)}) = \{(\mathbf{x}'^T, \mathbf{x}^T)^T \in \mathbb{R}^{p+n} : \mathbf{x}' \in \mathcal{R}(D^{(X_1)}),\ \mathbf{x} \in \mathcal{R}(D^{(X_2)})\}$. From the DBM point of view, the Cartesian product can be represented by an augmented DBM $D^{(X_1 \times X_2)} = D^{(X_1)} \times D^{(X_2)}$ with dimension $(p+n+1) \times (p+n+1)$, such that $\mathcal{R}(D^{(X_1 \times X_2)}) = \mathcal{R}(D^{(X_1)}) \times \mathcal{R}(D^{(X_2)})$. Note that, in general, $D^{(X_1)} \times D^{(X_2)} \neq D^{(X_2)} \times D^{(X_1)}$.

Each region and the corresponding affine dynamics of a PWA system generated by an MPL system can be characterized by a DBM [5, Sec. 2.3.5]. From Definition 1 each region of the PWA system, presented in (9), can be represented by a $(p+1) \times (p+1)$ DBM. Furthermore, the affine dynamics (10) can be expressed as an intersection of sets:

$$\bigcap_{i=1}^{p} \{z_i(k) - x_{g_i}(k-1) \leq a_{ig_i}\} \cap \bigcap_{i=1}^{p} \{x_{g_i}(k-1) - z_i(k) \leq -a_{ig_i}\} \,. \tag{13}$$

Therefore, the affine dynamics can be represented by an $(n+p+1) \times (n+p+1)$ DBM, which constrain the variables $\mathbf{z}(k) = (z_1(k) \ \cdots \ z_n(k))^T$ and $\mathbf{x}(k-1) = (x_1(k-1) \ \cdots \ x_p(k-1))^T$ and their differences.

Example 3 *Each component of the PWA system of example 1 can be represented by the following DBM: (Notation: $\mathbf{x}' \equiv \mathbf{x}(k)$ and $\mathbf{x} \equiv \mathbf{x}(k-1)$)*

$$D^{(1,1)} = \begin{array}{cc} \begin{array}{ccccc} x_0 & x'_1 & x'_2 & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0, \leq) & (\infty, <) & (\infty, <) & (\infty, <) & (\infty, <) \\ (\infty, <) & (0, \leq) & (\infty, <) & (3, \leq) & (\infty, <) \\ (\infty, <) & (\infty, <) & (0, \leq) & (2, \leq) & (\infty, <) \\ (\infty, <) & (-3, \leq) & (-2, \leq) & (0, \leq) & (\infty, <) \\ (\infty, <) & (\infty, <) & (\infty, <) & (-4, \leq) & (0, \leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x'_1 \\ x'_2 \\ x_1 \\ x_2 \end{array} \end{array}$$

$$D^{(2,1)} = \begin{array}{cc} \begin{array}{ccccc} x_0 & x'_1 & x'_2 & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0, \leq) & (\infty, <) & (\infty, <) & (\infty, <) & (\infty, <) \\ (\infty, <) & (0, \leq) & (\infty, <) & (\infty, <) & (7, \leq) \\ (\infty, <) & (\infty, <) & (0, \leq) & (2, \leq) & (\infty, <) \\ (\infty, <) & (\infty, <) & (-2, \leq) & (0, \leq) & (4, \leq) \\ (\infty, <) & (-7, \leq) & (\infty, <) & (-2, \leq) & (0, \leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x'_1 \\ x'_2 \\ x_1 \\ x_2 \end{array} \end{array}$$

---

[3] The symbols $<$ and $\leq$ are assumed to be totally ordered with $<$ strictly less than $\leq$.

$$D^{(2,2)} = \begin{array}{c} \\ \\ \end{array} \begin{array}{ccccc|c} x_0 & x'_1 & x'_2 & x_1 & x_2 & \\ (0,\leq) & (\infty,<) & (\infty,<) & (\infty,<) & (\infty,<) & x_0 \\ (\infty,<) & (0,\leq) & (\infty,<) & (\infty,<) & (7,\leq) & x'_1 \\ (\infty,<) & (\infty,<) & (0,\leq) & (\infty,<) & (4,\leq) & x'_2 \\ (\infty,<) & (\infty,<) & (\infty,<) & (0,\leq) & (2,\leq) & x_1 \\ (\infty,<) & (-7,\leq) & (-4,\leq) & (\infty,<) & (0,\leq) & x_2 \end{array}$$

In the following, are presented some important results for calculating the image and the inverse image of a DBM w.r.t a PWA system generated by an MPL system.

Proposition 1 ***[1, Th. 1]*** *The image and the inverse image of a DBM w.r.t. a subsystem of a PWA system generated by an MPL system is a DBM.*

Given an MPL system (possibly nonautonomous) characterized by a matrix $A \in \overline{\mathbb{R}}_{max}^{n\times p}$, the general procedure for calculating the image of a DBM D, with region $\mathcal{R}(D) \in \mathbb{R}^p$, w.r.t. each subsystem of a PWA system generated by A is: 1) compute the Cartesian product of $D^{(\mathbb{R}^n)}$ and D (see remark 6); 2) intersect the obtained DBM with the DBM generated by region (9) and dynamics (13); 3) compute the canonical form of the intersection; and 4) project the canonical-form representation over the variables at event k (i.e. $z_1(k), ..., z_n(k)$) (cf. Proposition 1).

Similarly, the general procedure for calculating the inverse image of a DBM D, with region $\mathcal{R}(D) \in \mathbb{R}^n$, w.r.t. each subsystem of a PWA system generated by A is: 1) compute the Cartesian product of D and $D^{(\mathbb{R}^p)}$; 2) intersect the obtained DBM with the DBM generated by region (9) and dynamics (13); 3) compute the canonical form of the intersection; and 4) project the canonical-form representation over the variables at event $k-1$ (i.e. $x_1(k-1), ..., x_p(k-1)$) (cf. Proposition 1).

The worst-case complexity of calculating the image or the inverse image critically depends on computing the canonical-form representation. The complexity of calculating the canonical-form representation of a DBM is cubic w.r.t. its dimension (see section 2.3). Thus, the worst-case complexity is $\mathcal{O}((n+p)^3)$ [4].

2.4 Interval Analysis

Interval arithmetic is presented in [34] and extended to max-plus algebra in [12,23,29,31]. An interval is defined as:

$$[\mathbf{x}] = [\underline{x},\ \overline{x}] = \{x \in \overline{\mathbb{R}}_{max} : \underline{x} \leq x \leq \overline{x}\}. \qquad (14)$$

The intersection of two intervals $[\mathbf{x}]$ and $[\mathbf{y}]$ is empty or an interval, defined by:

$$[\mathbf{x}] \cap [\mathbf{y}] = [\max\{\underline{x}, \underline{y}\},\ \min\{\overline{x}, \overline{y}\}]. \qquad (15)$$

If the intervals have nonempty intersection, the union is an interval defined by:

$$[\mathbf{x}] \cup [\mathbf{y}] = [\min\{\underline{x}, \underline{y}\},\ \max\{\overline{x}, \overline{y}\}]. \qquad (16)$$

The Max-Plus operations can be extended to intervals as follows:

$$[\mathbf{x}] \oplus [\mathbf{y}] = \{x \oplus y : x \in [\mathbf{x}]\,,\ y \in [\mathbf{y}]\} = [\underline{x} \oplus \underline{y},\ \overline{x} \oplus \overline{y}], \quad (17)$$
$$[\mathbf{x}] \otimes [\mathbf{y}] = \{x \otimes y : x \in [\mathbf{x}]\,,\ y \in [\mathbf{y}]\} = [\underline{x} \otimes \underline{y},\ \overline{x} \otimes \overline{y}]. \quad (18)$$

According to (1) and (17) a partial order for intervals in $\overline{\mathbb{R}}_{max}$ can be defined as:

$$[\mathbf{x}] \succeq [\mathbf{y}] \Leftrightarrow \underline{x} \succeq \underline{y} \text{ and } \overline{x} \succeq \overline{y}. \qquad (19)$$

In particular,

$$[\mathbf{x}] = [\mathbf{y}] \Leftrightarrow \underline{x} = \underline{y} \text{ and } \overline{x} = \overline{y}. \qquad (20)$$

Moreover, the Max-Plus sum can be extended to a finite number of intervals:

$$\bigoplus_{i=1}^{n} [\mathbf{x}]_i = \left\{ \bigoplus_{i=1}^{n} x_i : x_i \in [\mathbf{x}]_i \right\} = \left[\bigoplus_{i=1}^{n} \underline{x}_i,\ \bigoplus_{i=1}^{n} \overline{x}_i \right]. \quad (21)$$

The uMPL systems (Sec. 2.1) can be viewed as MPL systems whose each matrix entry a_{ij} is in the interval $[\mathbf{a_{ij}}] = [\underline{a}_{ij},\ \overline{a}_{ij}]$. Defining $[\mathbf{A}]$ as a *matrix of intervals* such that $A(k) \in [\mathbf{A}] = ([\mathbf{a_{ij}}])_{1\leq i \leq n,\ 1 \leq j \leq p}$, a generic model for an uMPL system is given by:

$$\mathbf{z}(k) = A(k) \otimes \mathbf{x}(k-1),\ \ A(k) \in \overline{\mathbb{R}}_{max}^{n\times p},\ \ A(k) \in [\mathbf{A}]\,. \quad (22)$$

The i-th system equation of (22) can be rewritten as:

$$z_i(k) = \bigoplus_{j=1}^{p} \{a_{ij} \otimes x_j(k-1)\},\ \ a_{ij} \in [\mathbf{a_{ij}}]\,. \qquad (23)$$

Then, given $\mathbf{x}(k-1)$, and by using (21), $z_i(k)$ is in the interval defined by:

$$[\mathbf{z_i}]\,(k) = \left[\bigoplus_{j=1}^{p} \{\underline{a}_{ij} \otimes x_j(k-1)\},\ \bigoplus_{j=1}^{p} \{\overline{a}_{ij} \otimes x_j(k-1)\} \right]. \quad (24)$$

In this work, we aim to represent the intervals $[\mathbf{z_i}]\,(k)$ as DBM. As we discuss in the next section, the intervals $[\mathbf{z_i}]\,(k)$ cannot be represented by a single DBM. Therefore, we propose a partition of the state space in which interval (24) can be expressed as a DBM suitable form.

3 Partitioned Uncertain Max Plus Systems

This section presents the main contribution. We aim to use the DBM data structure for the reachability analysis of uMPL systems. In Section 2.2 we have seen that every MPL system can be expressed as a PWA system and Section 2.3 shows how DBM representation of PWA systems is efficient for reachability analysis. Seeking for generality, we observe that the reachability analysis of an MPL system through the DBM approach is possible because each partition $R_{\mathbf{g}}$ (9) and corresponding dynamics (13) are DBM. In the following, we propose a partition of the state space of uMPL systems that satisfies this property. On this purpose let us rewrite interval (24) as:

$$\bigoplus_{j=1}^{p}\{\underline{a}_{ij}\otimes x_j(k-1)\} \preceq z_i(k) \preceq \bigoplus_{j=1}^{p}\{\overline{a}_{ij}\otimes x_j(k-1)\} \quad (25)$$

According to Definition 1, the DBM can easily handle restrictions of the type $\bigoplus_{j=1}^{p}\{\underline{a}_{ij}\otimes x_j(k-1)\} \preceq z_i(k)$, since they can be rewritten as $\bigcap_{j=1}^{p}\{x_j(k-1) - z_i(k) \leq -\underline{a}_{ij}\}$. On the other hand, the DBM are not suitable to represent the union of sets. Therefore, restrictions of the type $z_i(k) \preceq \bigoplus_{j=1}^{p}\{\overline{a}_{ij}\otimes x_j(k-1)\} \equiv \bigcup_{j=1}^{p}\{z_i(k) - x_j(k-1) \leq \overline{a}_{ij}\}$ cannot be represented by a single DBM. The main result of this work is to propose a partition of the state space in which the dynamics (25) can be expressed as a DBM suitable form.

Let us consider the problem of finding the region where, for all i, $[\mathbf{z_i}](k)$ can be expressed as:

$$[\mathbf{z_i}](k) = \left[\bigoplus_{j=1}^{p}\{\underline{a}_{ij}\otimes x_j(k-1)\}, \quad \overline{a}_{ig_i}\otimes x_{g_i}(k-1)\right], \quad (26)$$

where $\mathbf{g} = (g_1, \cdots, g_n)$ with $g_i \in \{1, \cdots, p\}$ has the same interpretation as in (9).

This problem corresponds to find a region where the following equality holds for all $i \in \{1, ..., n\}$:

$$\left[\bigoplus_{j=1}^{p}\{\underline{a}_{ij}\otimes x_j\},\ \overline{a}_{ig_i}\otimes x_{g_i}\right] = \left[\bigoplus_{j=1}^{p}\{\underline{a}_{ij}\otimes x_j\},\ \bigoplus_{j=1}^{p}\{\overline{a}_{ij}\otimes x_j\}\right]. \quad (27)$$

From (20), the equality holds if:

$$\overline{a}_{ig_i}\otimes x_{g_i} = \bigoplus_{j=1}^{p}\{\overline{a}_{ij}\otimes x_j\} \quad \forall i. \quad (28)$$

According to (1), for all i, j, equation (28) is equivalent to:

$$\overline{a}_{ig_i}\otimes x_{g_i} \succeq \overline{a}_{ij}\otimes x_j \Leftrightarrow x_j - x_{g_i} \leq \overline{a}_{ig_i} - \overline{a}_{ij}. \quad (29)$$

The region corresponding to (29) is given by:

$$R^u_{\mathbf{g}} = \bigcap_{i=1}^{n}\bigcap_{\substack{j=1\\ j\neq g_i}}^{p}\left\{\mathbf{x}\in\overline{\mathbb{R}}^p_{max} : x_j - x_{g_i} \leq \overline{a}_{ig_i} - \overline{a}_{ij}\right\}. \quad (30)$$

Region (30) defines a partition for uMPL systems. Moreover, if $\mathbf{x}\in R^u_{\mathbf{g}}$ then $z_i(k)\in[\mathbf{z_i}](k)$, as defined in (26).

Example 4 *Consider the following uMPL system:*

$$\mathbf{x}(k) = A(k)\otimes\mathbf{x}(k-1),\ \textit{where}\ A(k)\in\begin{pmatrix} 2 & [5,\ 6] \\ [3,\ 4] & 3 \end{pmatrix}.$$

According to (30) *and* (26)*, the corresponding partitioned uMPL system is: (Notation:* $\mathbf{x}'\equiv\mathbf{x}(k)$ *and* $\mathbf{x}\equiv\mathbf{x}(k-1)$*)*

$$\mathbf{x}'\in\begin{cases}\begin{pmatrix}[(2\otimes x_1)\oplus(5\otimes x_2),\ 2\otimes x_1]\\ [(3\otimes x_1)\oplus(3\otimes x_2),\ 4\otimes x_1]\end{pmatrix} & \textit{if}\quad \mathbf{x}\in R^u_{(1,1)},\\ \begin{pmatrix}[(2\otimes x_1)\oplus(5\otimes x_2),\ 6\otimes x_2]\\ [(3\otimes x_1)\oplus(3\otimes x_2),\ 4\otimes x_1]\end{pmatrix} & \textit{if}\quad \mathbf{x}\in R^u_{(2,1)},\\ \begin{pmatrix}[(2\otimes x_1)\oplus(5\otimes x_2),\ 6\otimes x_2]\\ [(3\otimes x_1)\oplus(3\otimes x_2),\ 3\otimes x_2]\end{pmatrix} & \textit{if}\quad \mathbf{x}\in R^u_{(2,2)},\end{cases}$$

where: $R^u_{(1,1)} = \{\mathbf{x}\in\overline{\mathbb{R}}^2_{max} : x_2 - x_1 \leq -4\}$, $R^u_{(2,1)} = \{\mathbf{x}\in\overline{\mathbb{R}}^2_{max} : -4 \leq x_2 - x_1 \leq 1\}$ *and* $R^u_{(2,2)} = \{\mathbf{x}\in\overline{\mathbb{R}}^2_{max} : x_2 - x_1 \geq 1\}$*. The region* $R^u_{(1,2)}$ *is empty. The regions* $R^u_{(1,1)}$*,* $R^u_{(2,1)}$ *and* $R^u_{(2,2)}$ *are shown in figure 1.*

From Definition 1, each region (30) can be represented by a $(p+1)\times(p+1)$ DBM. From (26), for all $i\in\{1, ..., n\}$, $z_i(k)$ is in the set defined by the following inequalities:

$$z_i(k) \preceq \overline{a}_{ig_i}\otimes x_{g_i}(k-1), \quad (31)$$

$$z_i(k) \succeq \bigoplus_{j=1}^{p}\{\underline{a}_{ij}\otimes x_j(k-1)\} \Leftrightarrow \begin{cases} z_i(k) \succeq \underline{a}_{i1}\otimes x_1(k-1),\\ \vdots\\ z_i(k) \succeq \underline{a}_{ip}\otimes x_p(k-1).\end{cases} \quad (32)$$

From this set, the following region can be defined:

$$\bigcap_{i=1}^{n}\{z_i(k) - x_{g_i}(k-1) \leq \overline{a}_{ig_i}\}\cap\bigcap_{i=1}^{n}\bigcap_{j=1}^{p}\left\{x_j(k-1) - z_i(k) \leq -\underline{a}_{ij}\right\} \quad (33)$$

According to Definition 1, it is straightforward to see that the dynamics of a partitioned uMPL system can be represented by a $(n+p+1)\times(n+p+1)$ DBM.

Remark 7 *Each component of a partitioned uMPL system (region plus corresponding dynamics) can be fully characterized by the intersection of* (30) *and* (33). *This intersection can be represented by a* $(n + p + 1) \times (n + p + 1)$ *DBM which constrains the variables* $[z_1, \ldots, z_n, x_1, \ldots, x_p]$ *and their differences.*

Given $[\mathbf{A}] = [\underline{A},\ \overline{A}]$, where $\underline{A}$, and $\overline{A} \in \overline{\mathbb{R}}^{n\times p}_{max}$, Algorithm 1 describes a procedure to generate a partitioned uMPL system represented by a collection of DBM $\mathtt{D}$.

Algorithm 1 Generating a partitioned uMPL system represented by a collection of DBM.

```
input: $[\mathbf{A}] = [\underline{A},\ \overline{A}]$, where $\underline{A},\ \overline{A} \in \overline{\mathbb{R}}^{n\times p}_{max}$
output: $\mathtt{D}$

1: $\mathtt{D} \leftarrow \emptyset$;
2: for each $\mathbf{g} \in \{1, ..., p\}^n$ do
3:     Compute the region $R^u_{\mathbf{g}}$ according to (30);
4:     if $R^u_{\mathbf{g}}$ is not empty then
5:         Compute the dynamics active in $R^u_{\mathbf{g}}$ according to (33);
6:         Generate the DBM $D^{\mathbf{g}}$ that represents $R^u_{\mathbf{g}}$ and the corresponding active dynamics;
7:         Save $D^{\mathbf{g}}$ into the output variable: $\mathtt{D} \leftarrow \mathtt{D} \cup \{D^{\mathbf{g}}\}$;
8:     end if
9: end for each
```

The worst-case complexity of Algorithm 1 is calculated as follows. The maximum number of iterations in steps 2, 3 and 4 is p^n, np and p^3. Thus, the worst-case complexity is $\mathcal{O}(p^n(np+p^3))$. Note that the algorithm for generating a PWA from an MPL systems has the same worst-case complexity (see section 2.2).

Remark 8 *We shall remark that the complexity of representing an uMPL systems as a collection of DBM critically depends on the number of regions of the partitioned systems. The regions of the partioned uMPL system only depends on the upper bound of the uMPL system matrix (see equation 30), which can be represented by a deterministic matrix. Therefore, the results of the experiment mentioned in remark 5 (for deterministic MPL systems) holds for uMPL systems as well.*

Example 5 *In this example, the uMPL system of example 4 is alternatively represented as a collection of DBM. For each* $\mathbf{g} \in \{1, 2\}^2 = \{(1, 1),\ (1, 2),\ (2, 1),\ (2, 2)\}$, *we compute a DBM* $D^{\mathbf{g}}$ *which represents the region* $R^u_{\mathbf{g}}$ *and the corresponding dynamics. The DBM* $D^{(1,1)}$ *is constructed as follows. From* (30), *we have that:* $R^u_{(1,1)} = \{\mathbf{x} \in \overline{\mathbb{R}}^2_{max} : \underbrace{x_2 - x_1 \leq -4}_{d^{(1,1)}_{54}}\}$. *From* (33), *the dynamics active in* $R^u_{(1,1)}$ *is given by:*

$$\underbrace{\{x'_1 - x_1 \leq 2\}}_{d^{(1,1)}_{24}} \cap \underbrace{\{x'_2 - x_1 \leq 4\}}_{d^{(1,1)}_{34}} \cap \underbrace{\{x_1 - x'_1 \leq -2\}}_{d^{(1,1)}_{42}} \cap \underbrace{\{x_2 - x'_1 \leq -5\}}_{d^{(1,1)}_{52}} \cap \underbrace{\{x_1 - x'_2 \leq -3\}}_{d^{(1,1)}_{43}} \cap \underbrace{\{x_2 - x'_2 \leq -3\}}_{d^{(1,1)}_{53}}.$$

Thus, $D^{(1,1)}$ *is given by:*

$$D^{(1,1)} = \begin{array}{c} \\ \end{array}\begin{array}{ccccc|c} x_0 & x'_1 & x'_2 & x_1 & x_2 & \\ (0, \leq) & (\infty, <) & (\infty, <) & (\infty, <) & (\infty, <) & x_0 \\ (\infty, <) & (0, \leq) & (\infty, <) & (2, \leq) & (\infty, <) & x'_1 \\ (\infty, <) & (\infty, <) & (0, \leq) & (4, \leq) & (\infty, <) & x'_2 \\ (\infty, <) & (-2, \leq) & (-3, \leq) & (0, \leq) & (\infty, <) & x_1 \\ (\infty, <) & (-5, \leq) & (-3, \leq) & (-4, \leq) & (0, \leq) & x_2 \end{array}$$

The same procedure is used to compute $D^{(2,1)}$ *and* $D^{(2,2)}$.

$$D^{(2,1)} = \begin{array}{ccccc|c} x_0 & x'_1 & x'_2 & x_1 & x_2 & \\ (0, \leq) & (\infty, <) & (\infty, <) & (\infty, <) & (\infty, <) & x_0 \\ (\infty, <) & (0, \leq) & (\infty, <) & (\infty, <) & (6, \leq) & x'_1 \\ (\infty, <) & (\infty, <) & (0, \leq) & (4, \leq) & (\infty, <) & x'_2 \\ (\infty, <) & (-2, \leq) & (-3, \leq) & (0, \leq) & (4, \leq) & x_1 \\ (\infty, <) & (-5, \leq) & (-3, \leq) & (1, \leq) & (0, \leq) & x_2 \end{array}$$

$$D^{(2,2)} = \begin{array}{ccccc|c} x_0 & x'_1 & x'_2 & x_1 & x_2 & \\ (0, \leq) & (\infty, <) & (\infty, <) & (\infty, <) & (\infty, <) & x_0 \\ (\infty, <) & (0, \leq) & (\infty, <) & (\infty, <) & (6, \leq) & x'_1 \\ (\infty, <) & (\infty, <) & (0, \leq) & (\infty, <) & (3, \leq) & x'_2 \\ (\infty, <) & (-2, \leq) & (-3, \leq) & (0, \leq) & (-1, \leq) & x_1 \\ (\infty, <) & (-5, \leq) & (-3, \leq) & (\infty, <) & (0, \leq) & x_2 \end{array}$$

Remark 9 *If* $\underline{a}_{ij} = \overline{a}_{ij}\ \forall i, j$ *(deterministic case), region* $R^u_{\mathbf{g}}$, *given by* (30), *equals region* $R_{\mathbf{g}}$, *given by* (9). *In this case, for all* $\mathbf{x} \in R^u_{\mathbf{g}}$, *inequality* (32) *can be expressed as* $z_i(k) \succeq \bigoplus_{j=1}^{p}\{\underline{a}_{ij} \otimes x_j(k-1)\} = \bigoplus_{j=1}^{p}\{\overline{a}_{ij} \otimes x_j(k-1)\} = \overline{a}_{ig_i} \otimes x_{g_i}(k-1)$. *Therefore, it is straightforward to see that the set* (33) *equals the set* (13).

As proved below, the image and the inverse image of a DBM through each partition of an uMPL system is a DBM. Therefore, the DBM approach is useful for reachability analysis of uMPL systems. Proposition 2 is an extension of Proposition 1 to uMPL systems.

Proposition 2 *The image and the inverse image of a DBM w.r.t. a subsystem of a partitioned uMPL system is a DBM.*

PROOF. Given a partitioned uMPL system defined by the regions (30) and corresponding dynamics (26), the general procedure for calculating the image of a DBM D, with region $\mathcal{R}(D) \in \mathbb{R}^p$, w.r.t. each subsystem of the partitioned uMPL system can be decomposed in the following steps: 1) compute the Cartesian product of $D^{(\mathbb{R}^n)}$ and D (see remark 6); 2) intersect the obtained DBM with the DBM generated by region (30) and dynamics (33); 3) compute the canonical form of the intersection; and 4) project the canonical-form representation over the variables at event k (i.e. $z_1(k), ..., z_n(k)$). Similarly, the general procedure for calculating the inverse image of a DBM D, with region $\mathcal{R}(D) \in \mathbb{R}^n$, w.r.t. each subsystem of a partitioned uMPL system generated by an uMPL system can be decomposed in the following steps: 1) compute the Cartesian product of D and $D^{(\mathbb{R}^p)}$; 2) intersect the obtained DBM with the DBM generated by region (30) and dynamics (26); 3) compute the canonical form of the intersection; and 4) project the canonical-form representation over the variables at event $k-1$ (i.e. $x_1(k-1), ..., x_p(k-1)$). As stated in the proof of [1, Th. 1], the Cartesian product of DBM is an augmented DBM, the intersection of DBM is a DBM, the canonical form of a DBM is a DBM and the orthogonal projection of a DBM is a DBM. Then, the image and the inverse of a DBM w.r.t. each subsystem of a partitioned uMPL system is a DBM. As in the computation of the image and the inverse image of a DBM w.r.t a subsystem of a PWA system generated by an MPL system, the worst-case complexity is $\mathcal{O}((n+p)^3)$ and critically depends on computing the canonical-form representation, whose complexity is cubic w.r.t its dimension (see section 2.3).

Corollary 1 *The image and the inverse image of a union of finitely many DBM w.r.t. a partitioned uMPL system is a union of finitely many DBM.*

Computing the image of a union of q DBM w.r.t. a partitioned uMPL system can be done by computing the image of each DBM w.r.t each component of the partitioned uMPL system. Similarly, computing the inverse image of a union of q DBM w.r.t. a partitioned uMPL system can be done by computing the inverse image of each DBM w.r.t each component of the partitioned uMPL system. Thus the worst-case complexity of computing the image or the inverse image depends on the number of DBM (considered to be q), on the worst-case cardinality of the collection of subsystems (given by p^n) and on the worst-case complexity of computing the image of each DBM w.r.t each component of the partitioned uMPL system (given by $\mathcal{O}((n+p)^3)$). Therefore, the worst-case complexity is $\mathcal{O}(qp^n(n+p)^3)$.

Recalling remark 2, in many practical situations the uMPL systems are modeled by:

$$\mathbf{x}(k) = \mathcal{A}_0(k) \otimes \mathbf{x}(k) \oplus \mathcal{A}_1(k) \otimes \mathbf{x}(k-1) \oplus B(k) \otimes \mathbf{u}(k), \tag{34}$$

where $\mathcal{A}_0(k) \in [\mathcal{A}_\mathbf{0}]$, $\mathcal{A}_1(k) \in [\mathcal{A}_\mathbf{1}]$, $B(k) \in [\mathbf{B}]$, $\mathbf{x} \in \overline{\mathbb{R}}^n_{max}$ and $\mathbf{u} \in \overline{\mathbb{R}}^m_{max}$.

In the following, it is shown how the results presented in this section can be extended to systems in this form. The system (34) can be expressed as:

$$\mathbf{x}(k) = H(k) \otimes \mathbf{r}(k), \tag{35}$$

where $H(k) \in ([\mathcal{A}_\mathbf{0}] \ \ [\mathcal{A}_\mathbf{1}] \ \ [\mathbf{B}])$ and $\mathbf{r}(k) = (\mathbf{x}^T(k)\,\mathbf{x}^T(k-1)\ \mathbf{u}^T(k))^T$. Then, according to (30), the region corresponding to each component of the partition is given by:

$$R^u_\mathbf{g} = \bigcap_{i=1}^{n} \bigcap_{\substack{j=1 \\ j \neq g_i}}^{n+n+m} \left\{ \mathbf{r} \in \overline{\mathbb{R}}^{n+n+m}_{max} : r_j - r_{g_i} \leq \overline{h}_{ig_i} - \overline{h}_{ij} \right\}, \tag{36}$$

and, according to (33), the corresponding dynamics is given by (note that $r_i(k) = x_i(k)$ for $i \in \{1, ..., n\}$.):

$$\bigcap_{i=1}^{n} \{ r_i - r_{g_i} \leq \overline{h}_{ig_i} \} \cap \bigcap_{i=1}^{n} \bigcap_{j=1}^{n+n+m} \left\{ r_j - r_i \leq -\underline{h}_{ij} \right\}. \tag{37}$$

Therefore, the uMPL system (34) can be represented by a collection of $(n+n+m+1) \times (n+n+m+1)$ DBM. As a result, proposition 2 holds for this model of uMPL systems.

4 Reachability Analysis of Uncertain MPL systems

This section presents an extension for uMPL systems of some of the results on reachability analysis introduced in [2], [3] and [4]. In the following, the definition of *reach set* is recalled [3, Def. 3]. Moreover, in order to be compatible with the uncertainty context in which the uMPL systems are defined, a modification in the definition of *backward reach set* [2, Def. 7] is introduced.

Definition 2 (reach set) *Given an uMPL system and a nonempty set of initial conditions $X_0 \subseteq \mathbb{R}^n$, the* ***reach set*** *$X_N$ at the event step $N > 0$ is the set of all states $\{\mathbf{x}(N) : \mathbf{x}(0) \in X_0\}$ that* can be reached *via the uMPL dynamics, possibly by application of controls.*

Definition 3 (backward reach set) *Given an uMPL system and a nonempty set of final positions $X_0 \subseteq \mathbb{R}^n$, the* ***backward reach set*** *X_{-N} is the set of all states $\mathbf{x}(-N)$ that* may lead *to X_0 in N steps of the uMPL dynamics, possibly by application of controls.*

4.1 Forward Reachability Analysis

For *autonomous* uMPL systems, given a nonempty set of initial conditions X_0, the reach set X_k at the event step k can be recursively calculated as the image of the reach set X_{k-1} w.r.t the uMPL dynamics:

$$X_k = \mathcal{I}_{[\mathbf{A}]}\{X_{k-1}\} = \{A \otimes \mathbf{x} : \mathbf{x} \in X_{k-1},\, A \in [\mathbf{A}]\}. \tag{38}$$

From Corollary 1, if X_{k-1} is a union of q_{k-1} DBM, then $X_k = \mathcal{I}_{[\mathbf{A}]}\{X_{k-1}\}$ is a union of q_k DBM. Thus, by induction, it can be concluded that if X_0 is a union of q_0 DBM, then X_k is a union of q_k DBM, for each $k \in \mathbb{N}$. Given the set of initial conditions X_0, computing X_N at the event step N can be done as follows: first, construct the partitioned uMPL system generated by $A(k)$; then, for each $k \in \{1, ..., N\}$, compute the image of X_{k-1} w.r.t. the partitioned uMPL system. The worst-case complexity to compute $\mathcal{I}_{[\mathbf{A}]}\{X_{k-1}\}$, for each $k \in \{1, ..., N\}$ is $\mathcal{O}(q_{k-1}n^{n+3})$. Thus, the overall complexity is $\mathcal{O}(n^{n+3}\sum_{k=1}^{N} q_{k-1})$.

Remark 10 *Given the cardinality q_{k-1} of the DBM union set at event step $k-1$, the worst-case cardinality q_k is $q_{k-1}n^n$, which corresponds to the maximum possible number of nonempty DBM representing the image of the intersection of each DBM at $k-1$ and each region of the partitioned system. Practically many regions are empty, and even for nonempty regions, many intersections of DBM and regions are also empty, then the cardinality q_k is drastically smaller than its worst-case bound. However, in general, it is not possible to quantify the exact cardinality q_k a priori [4, Sec. 5].*

For *nonautonomous* uMPL systems, given a nonempty set of initial conditions X_0 and the set of inputs U_k for each $k \in \mathbb{N}$, the reach set X_k at the event step k can be recursively calculated as:

$$X_k = \mathcal{I}_{[\mathbf{F}]}\{X_{k-1} \times U_k\} = \{F \otimes \mathbf{y} : \mathbf{y} \in X_{k-1} \times U_k,\ F \in [\mathbf{F}]\}. \tag{39}$$

where $F = (A\ B)$ and $\mathbf{y} = \left(\mathbf{x}^T\ \mathbf{u}^T\right)^T$.

If X_{k-1} is a union of q_{k-1} DBM and U_k is a union of r_k DBM then $X_{k-1} \times U_k$ is a union of $\bar{q}_{k-1} = q_{k-1}r_k$ DBM. Thus, from Corollary 1, $X_k = \mathcal{I}_{[\mathbf{F}]}\{X_{k-1} \times U_k\}$ is a union of q_k DBM. By induction, it can be concluded that if X_0 is a union of q_0 DBM and U_k is a union of r_k DBM for each $k \in \mathbb{N}$, then X_k is a union of q_k DBM, for each $k \in \mathbb{N}$. Given a nonautonomous uMPL system, the set of initial conditions X_0 and set of inputs U_k for each $k \in \{1, ..., N\}$, computing X_N can be done as follows: first, construct the partitioned uMPL system generated by $[\mathbf{F}] = ([\mathbf{A}]\ \ [\mathbf{B}])$; then, for each $k \in \{1, ..., N\}$, compute the image of $X_{k-1} \times U_k$ w.r.t. the partitioned uMPL system. The worst-case complexity to compute $\mathcal{I}_{[\mathbf{F}]}\{X_{k-1} \times U_k\}$, for each $k \in \mathbb{N}$ is $\mathcal{O}(\bar{q}_{k-1}(n+m)^{n+3})$. Thus, the overall complexity is $\mathcal{O}((n+m)^{n+3}\sum_{k=1}^{N} \bar{q}_{k-1})$.

4.2 Backward Reachability Analysis

For *autonomous* uMPL systems, given a set of final positions X_0, the backward reach set X_{-k} can be recursively calculated as the inverse image of the reach set X_{-k+1} w.r.t the uMPL dynamics:

$$X_{-k} = \mathcal{I}^{-1}_{[\mathbf{A}]}\{X_{-k+1}\} = \{\mathbf{x} \in \mathbb{R}^n : \exists A \in [\mathbf{A}] : A \otimes \mathbf{x} \in X_{-k+1}\}. \tag{40}$$

From Corollary 1 it can be shown that if X_0 is a union of q_0 DBM, then X_{-k} is a union of q_{-k} DBM, for each $k \in \mathbb{N}$. Given the set of final conditions X_0, computing X_{-N} can be done as follows: first, construct the partitioned uMPL system generated by $A(k)$; then, for each $k \in \{1, ..., N\}$, compute the inverse image of X_{k-1} w.r.t. the partitioned uMPL system. The worst-case complexity to compute $\mathcal{I}^{-1}_{[\mathbf{A}]}\{X_{-k+1}\}$, for each $k \in \mathbb{N}$ is $\mathcal{O}(q_{-k+1}n^{n+3})$. Thus, the overall complexity is $\mathcal{O}(n^{n+3}\sum_{k=1}^{N} q_{-k+1})$.

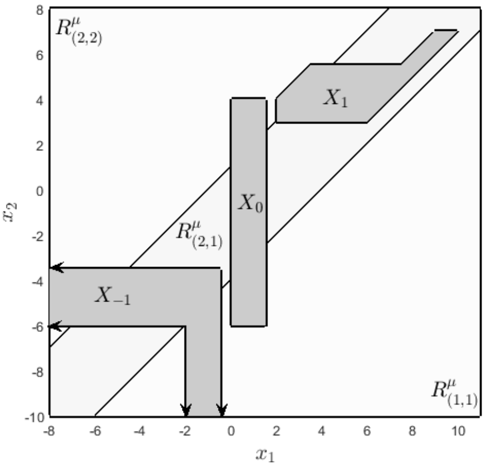

Fig. 1. Reachability Analysis for uMPL systems.

For *nonautonomous* uMPL systems, given a set of final conditions X_0 and the set of inputs U_{-k} for each $k \in \mathbb{N}$, the backward reach set X_{-k} can be recursively calculated as the inverse image of X_{-k+1}:

$$\begin{aligned} X_{-k} &= \mathcal{I}^{-1}_{[\mathbf{F}]}\{X_{-k+1}\} \\ &= \{\mathbf{x} \in \mathbb{R}^n : \exists \mathbf{u} \in U_{-k+1},\ \exists F \in [\mathbf{F}] : F \otimes \mathbf{y} \in X_{-k+1}\}. \end{aligned} \tag{41}$$

Given an nonautonomous uMPL system, characterized by the matrices $[\mathbf{A}]$ and $[\mathbf{B}]$, the set of final positions X_0 and set of inputs U_{-k+1} for each $k \in \{1, ..., N\}$, computing X_{-N} can be done as follows: first, construct the partitioned uMPL system generated by $[\mathbf{F}] = ([\mathbf{A}]\ \ [\mathbf{B}])$; then, for each $k \in \{1, ..., N\}$, compute the inverse image of X_{k-1} w.r.t. the partitioned uMPL system; next, intersect the inverse image with $\mathbb{R}^n \times U_{-k+1}$; and finally, project the intersection over the state variables. From Corollary 1, it can be shown that X_{-k} is a union of finitely many DBM, for $k \in \mathbb{N}$. The worst-case complexity to compute $\mathcal{I}^{-1}_{[\mathbf{F}]}\{X_{-k+1}\}$ is $\mathcal{O}(\bar{q}_{-k+1}(n+m)^{n+3})$, where: $\bar{q}_{-k+1} = q_{-k+1}r_{-k+1}$ and q_{-k+1} and r_{-k+1} are, respectively, the cardinality of the DBM union set representing X_{-k+1} and U_{-k+1}. Thus, the overall complexity to compute X_{-N} is $\mathcal{O}((n+m)^{n+3}\sum_{k=1}^{N} \bar{q}_{-k+1})$.

Example 6 *Considering the partitioned uMPL system obtained in Example 5 and the set of initial conditions $X_0 = \{\mathbf{x} \in \overline{\mathbb{R}}^2_{max} : 0 \le x_1 \le 1.5,\ -6 \le x_2 \le 4\}$, the* reach set X_1 *is calculated as follows. The set X_0 can be represented by the following DBM:*

$$D^{(X_0)} = \begin{array}{cc} \begin{array}{ccc} x_0 & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0,\leq) & (0,\leq) & (6,\leq) \\ (1.5,\leq) & (0,\leq) & (\infty,<) \\ (4,\leq) & (\infty,<) & (0,\leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1 \\ x_2 \end{array} \end{array}$$

The cross product $D^{(\mathbb{R}^2\times X_0)} = D^{(\mathbb{R}^2)} \times D^{(X_0)}$ is given by:

$$D^{(\mathbb{R}^2\times X_0)} = \begin{array}{cc} \begin{array}{ccccc} x_0 & x_1' & x_2' & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0,\leq) & (\infty,<) & (\infty,<) & (0,\leq) & (6,\leq) \\ (\infty,<) & (0,\leq) & (\infty,<) & (\infty,<) & (\infty,<) \\ (\infty,<) & (\infty,<) & (0,\leq) & (\infty,<) & (\infty,<) \\ (1.5,\leq) & (\infty,<) & (\infty,<) & (0,\leq) & (\infty,<) \\ (4,\leq) & (\infty,<) & (\infty,<) & (\infty,<) & (0,\leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1' \\ x_2' \\ x_1 \\ x_2 \end{array} \end{array}$$

The intersection of $D^{(\mathbb{R}^2\times X_0)}$ and $D^{(1,1)}$ is given by: $D^{(\mathbb{R}^2\times X_0)} \cap D^{(1,1)} =$

$$\begin{array}{cc} \begin{array}{ccccc} x_0 & x_1' & x_2' & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0,\leq) & (\infty,<) & (\infty,<) & (0,\leq) & (6,\leq) \\ (\infty,<) & (0,\leq) & (\infty,<) & (2,\leq) & (\infty,<) \\ (\infty,<) & (\infty,<) & (0,\leq) & (4,\leq) & (\infty,<) \\ (1.5,\leq) & (-2,\leq) & (-3,\leq) & (0,\leq) & (\infty,<) \\ (4,\leq) & (-5,\leq) & (-3,\leq) & (-4,\leq) & (0,\leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1' \\ x_2' \\ x_1 \\ x_2 \end{array} \end{array}$$

The canonical form of the intersection is given by: $cf(D^{(\mathbb{R}^2\times X_0)} \cap D^{(1,1)}) =$

$$\begin{array}{cc} \begin{array}{ccccc} x_0 & x_1' & x_2' & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0,\leq) & (-2,\leq) & (-3,\leq) & (0,\leq) & (6,\leq) \\ (3.5,\leq) & (0,\leq) & (-1,\leq) & (2,\leq) & (9.5,\leq) \\ (5.5,\leq) & (2,\leq) & (0,\leq) & (4,\leq) & (11.5,\leq) \\ (1.5,\leq) & (-2,\leq) & (-3,\leq) & (0,\leq) & (7.5,\leq) \\ (-2.5,\leq) & (-6,\leq) & (-7,\leq) & (-4,\leq) & (0,\leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1' \\ x_2' \\ x_1 \\ x_2 \end{array} \end{array}$$

The image of X_0 w.r.t. the component $\mathbf{g} = (1,1)$, noted by $X_1^{(1,1)}$, is obtained by computing the orthogonal projection of $cf(D^{(\mathbb{R}^2\times X_0)} \cap D^{(1,1)})$ over $\mathbf{x}'$, which corresponds to delete the rows and columns corresponding to $\mathbf{x}$:

$$D^{(X_1^{(1,1)})} = \begin{array}{cc} \begin{array}{ccc} x_0 & x_1' & x_2' \end{array} & \\ \begin{pmatrix} (0,\leq) & (-2,\leq) & (-3,\leq) \\ (3.5,\leq) & (0,\leq) & (-1,\leq) \\ (5.5,\leq) & (2,\leq) & (0,\leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1' \\ x_2' \end{array} \end{array}$$

Therefore $X_1^{(1,1)} = \mathcal{R}(D^{(X_1^{(1,1)})}) = \{\mathbf{x}' \in \mathbb{R}^2 : \ 2 \leq x_1' \leq 3.5, \ 3 \leq x_2' \leq 5.5, \ 1 \leq x_2' - x_1' \leq 2\}$. Applying the same procedure for the other components, we obtain $X_1^{(2,1)} = \{\mathbf{x}' \in \mathbb{R}^2 : \ 2 \leq x_1' \leq 8.5, \ 3 \leq x_2' \leq 5.5, \ -3 \leq x_2' - x_1' \leq 2\}$ and $X_1^{(2,2)} = \{\mathbf{x}' \in \mathbb{R}^2 : \ 6 \leq x_1' \leq 10, \ 4 \leq x_2' \leq 7, \ -3 \leq x_2' - x_1' \leq -2\}$. Finally, the reach set X_1 is the union of the images of X_0 w.r.t. each component of the partitioned uMPL system. Thus, $X_1 = X_1^{(1,1)} \cup X_1^{(2,1)} \cup X_1^{(2,2)} = \{\mathbf{x}' \in \mathbb{R}^2 : \ 2 \leq x_1' \leq 8.5, \ 3 \leq x_2' \leq 5.5, \ -3 \leq x_2' - x_1' \leq 2\} \cup \{\mathbf{x}' \in \mathbb{R}^2 : \ 6 \leq x_1' \leq 10, \ 4 \leq x_2' \leq 7, \ -3 \leq x_2' - x_1' \leq -2\}$.

The backward reach set X_{-1} is calculated as follows. First, we compute $D^{(X_0\times\mathbb{R}^2)} = D^{(X_0)} \times D^{(\mathbb{R}^2)}$:

$$D^{(X_0\times\mathbb{R}^2)} = \begin{array}{cc} \begin{array}{ccccc} x_0 & x_1' & x_2' & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0,\leq) & (0,\leq) & (6,\leq) & (\infty,<) & (\infty,<) \\ (1.5,\leq) & (0,\leq) & (\infty,<) & (\infty,<) & (\infty,<) \\ (4,\leq) & (\infty,<) & (0,\leq) & (\infty,<) & (\infty,<) \\ (\infty,<) & (\infty,<) & (\infty,<) & (0,\leq) & (\infty,<) \\ (\infty,<) & (\infty,<) & (\infty,<) & (\infty,<) & (0,\leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1' \\ x_2' \\ x_1 \\ x_2 \end{array} \end{array}$$

Next, we compute $D^{(X_0\times\mathbb{R}^2)} \cap D^{(1,1)}$ and its canonical form: $cf(D^{(X_0\times\mathbb{R}^2)} \cap D^{(1,1)}) =$

$$\begin{array}{cc} \begin{array}{ccccc} x_0 & x_1' & x_2' & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0,\leq) & (0,\leq) & (-1,\leq) & (2,\leq) & (\infty,<) \\ (1.5,\leq) & (0,\leq) & (-1,\leq) & (2,\leq) & (\infty,<) \\ (3.5,\leq) & (2,\leq) & (0,\leq) & (4,\leq) & (\infty,<) \\ (-0.5,\leq) & (-2,\leq) & (-3,\leq) & (0,\leq) & (\infty,<) \\ (-4.5,\leq) & (-6,\leq) & (-7,\leq) & (-4,\leq) & (0,\leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1' \\ x_2' \\ x_1 \\ x_2 \end{array} \end{array}$$

Then, the inverse image of X_0 w.r.t. the component $\mathbf{g} = (1,1)$, noted by $X_{-1}^{(1,1)}$, is obtained by computing the orthogonal projection of $cf(D^{(X_0\times\mathbb{R}^2)} \cap D^{(1,1)})$ over $\mathbf{x}$:

$$D^{(X_{-1}^{(1,1)})} = \begin{array}{cc} \begin{array}{ccc} x_0 & x_1 & x_2 \end{array} & \\ \begin{pmatrix} (0,\leq) & (2,\leq) & (\infty,<) \\ (-0.5,\leq) & (0,\leq) & (\infty,<) \\ (-4.5,\leq) & (-4,\leq) & (0,\leq) \end{pmatrix} & \begin{array}{c} x_0 \\ x_1 \\ x_2 \end{array} \end{array}$$

Therefore $X_{-1}^{(1,1)} = \mathcal{R}(D^{(X_{-1}^{(1,1)})}) = \{\mathbf{x} \in \mathbb{R}^2 : \ -2 \leq x_1 \leq -0.5, \ x_2 \leq -4.5, \ x_2 - x_1 \leq -4\}$. Applying the same procedure for the other components, we obtain $X_{-1}^{(2,1)} = \{\mathbf{x} \in \mathbb{R}^2 : \ -7 \leq x_1 \leq -0.5, \ -6 \leq x_2 \leq -3.5, \ -4 \leq x_2 - x_1 \leq 1\}$ and $X_{-1}^{(2,2)} = \{\mathbf{x} \in \mathbb{R}^2 : x_1 \leq -4.5, \ -6 \leq x_2 \leq -3.5, \ x_2 - x_1 \geq 1\}$. Finally, the backward reach set X_{-1} is the union of the inverse images of X_0 w.r.t. each component of the partitioned uMPL system. Thus, $X_{-1} = X_{-1}^{(1,1)} \cup X_{-1}^{(2,1)} \cup X_{-1}^{(2,2)} = \{\mathbf{x} \in \mathbb{R}^2 : \ -2 \leq x_1 \leq -0.5, \ x_2' \leq -4.5, \ x_2 - x_1 \leq -4\} \cup \{\mathbf{x} \in \mathbb{R}^2 : \ x_1 \leq -0.5, \ -6 \leq x_2 \leq -3.5, \ -x_2 - x_1 \geq -4\}$.

The set of initial/final conditions X_0, the reach set X_1 and the backward reach set X_{-1} are shown in Figure 1.

5 Conditional Reachability Analysis

Bayesian methods provide a rigorous general framework for dynamic state estimation problems [22]. Consider the following system:

$$\mathbf{x}(k) = f_{k-1}\left(\mathbf{x}(k-1), \mathbf{w}(k)\right), \tag{42}$$

$$\mathbf{z}(k) = h_k\left(\mathbf{x}(k), \mathbf{v}(k)\right). \tag{43}$$

Where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{z} \in \mathbb{R}^l$ are, respectively, the state and measurement vectors; $\mathbf{w} \in \mathbb{R}^m$ and $\mathbf{v} \in \mathbb{R}^r$ are independent identically distributed (iid) process noise sequence; $f_{k-1} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is in general a nonlinear transition function and $h_k : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^l$ is the measurement function.

In the Bayesian approach, one aims to construct the posterior PDF $p(\mathbf{x}_k|\mathbf{z}_1, ..., \mathbf{z}_k)$, which is the PDF of the states $\mathbf{x}(k)$ given all the available information $\mathbf{z}(1), ..., \mathbf{z}(k)$ at the event step k. The posterior PDF may be obtained recursively in two stages: prediction and update [22]. In the prediction stage it is assumed that the required PDF $p(\mathbf{x}_{k-1}|\mathbf{z}_1, ..., \mathbf{z}_{k-1})$ is available at the event step $k-1$. Therefore, using the system model and the *Chapman-Kolmogorov* equation it is possible to obtain the *prior* PDF $p(\mathbf{x}_k|\mathbf{z}_1,...,\mathbf{z}_{k-1})$, based on all information available at the event step $k-1$. In the update stage, the required PDF $p(\mathbf{x}_k|\mathbf{z}_1,...,\mathbf{z}_k)$ is obtained by updating the prior PDF, via the Bayes rule, based on the new available information $\mathbf{z}_k$ and on the measurement model.

In this work, the functions $f_{k-1}(.)$ and $h_k(.)$ are assumed to be uMPL systems as defined in Section 2.1, i.e:

$$\mathbf{x}(k) = A(k) \otimes \mathbf{x}(k-1),\ A(k) \in [\mathbf{A}]\,, \tag{44}$$

$$\mathbf{z}(k) = C(k) \otimes \mathbf{x}(k),\ C(k) \in [\mathbf{C}]\,. \tag{45}$$

The elements of matrices $A(k) \in \mathbb{R}^{n\times n}$ and $C(k) \in \mathbb{R}^{l\times n}$ are stochastic processes with supports in real intervals(see section 2.4). No further assumptions are made on these processes.

We define the calculation of the support of $p(\mathbf{x}_k|\mathbf{z}_1,...,\mathbf{z}_k)$, denoted by $X_{k|k}$, as the *conditional reachability problem*. Assuming that $X_{k-1|k-1}$ is known at the event step $k-1$, the support of the *prior* PDF $p(\mathbf{x}_k|\mathbf{z}_1,...,\mathbf{z}_{k-1})$, noted by $X_{k|k-1}$, can be calculated via (38) for autonomous uMPL sytems:

$$X_{k|k-1} = \mathcal{I}_{[\mathbf{A}]}\{X_{k-1|k-1}\} = \{A \otimes \mathbf{x} : \mathbf{x} \in X_{k-1|k-1},\ A \in [\mathbf{A}]\}, \tag{46}$$

and via (39) for *nonautonomous* uMPL systems:

$$\begin{aligned} X_{k|k-1} &= \mathcal{I}_{[\mathbf{F}]}\{X_{k-1|k-1} \times U_k\} \\ &= \{F \otimes \mathbf{y} : \mathbf{y} \in X_{k-1|k-1} \times U_k,\ F \in [\mathbf{F}]\}. \end{aligned} \tag{47}$$

Remark 11 *The set $X_{k-1|k-1}$ is assumed to be a given union of $q_{k-1|k-1}$ DBM, then $X_{k|k-1}$ is a union of $q_{k|k-1}$ DBM. Thus, the worst-case complexity to compute $X_{k|k-1}$ is $\mathcal{O}(q_{k-1|k-1}n^{n+3})$ for autonomous systems and $\mathcal{O}(\bar{q}_{k-1|k-1}(n+m)^{n+3})$ for nonautonomous systems (see section 4.1).*

In the update stage, the new information $\mathbf{z}_k$ can be used to update $X_{k|k-1}$. By using (40) it is possible to obtain the set of all states $\tilde{X}_{k|k}$ that may lead to $\mathbf{z}_k$ via the measurement model in one event step.

$$\tilde{X}_{k|k} = \mathcal{I}^{-1}_{[\mathbf{C}]}\{\mathbf{z}_k\} = \{\mathbf{x} \in \mathbb{R}^n : \exists C \in [\mathbf{C}] : C \otimes \mathbf{x} \in Z_k\}. \tag{48}$$

Remark 12 *The measurement $\mathbf{z}_k$ can be represented by a single DBM $D^{(z_k)}$, then $\tilde{X}_{k|k}$ is a union of $\tilde{q}_{k|k}$ DBM. Therefore, the worst-case complexity to compute $\tilde{X}_{k|k}$ is $\mathcal{O}(n^l(l+n)^3)$.*

Then, the support of $p(\mathbf{x}_k|\mathbf{z}_1,...,\mathbf{z}_k)$, noted by $X_{k|k}$, is obtained by updating $X_{k|k-1}$ as follows:

$$X_{k|k} = X_{k|k-1} \cap \tilde{X}_{k|k}. \tag{49}$$

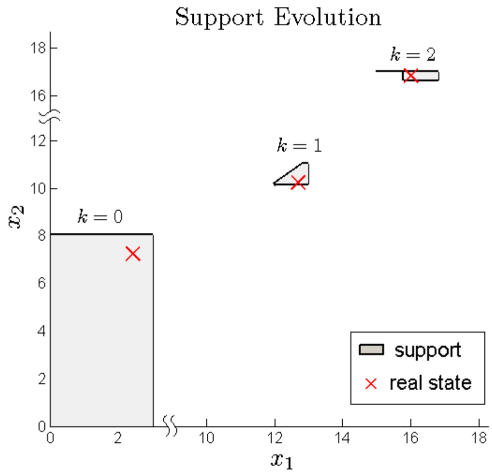

Fig. 2. Support evolution of $p(\mathbf{x}_k|\mathbf{z}_1,...,\mathbf{z}_k)$.

Remark 13 *The intersection of two sets represented by the union of finitely many DBM is again a union of finitely many DBM. Therefore, $X_{k|k}$ is a union of $q_{k|k}$ DBM. The worst-case cardinality $q_{k|k}$ of the DBM union set $X_{k|k}$ is $q_{k|k} = q_{k|k-1}\tilde{q}_{k|k}$, and the worst-case complexity to compute $X_{k|k}$, given $X_{k|k-1}$ and $\tilde{X}_{k|k}$, is $\mathcal{O}(q_{k|k-1}\tilde{q}_{k|k}n^2)$.*

6 Results

In this section, we use the results from Section 5 to calculate the support of the posterior PDF of an uMPL system. The system considered is characterized by:

$$A(k) \in \begin{pmatrix} 2 & [5,\ 6] \\ [3,\ 4] & 3 \end{pmatrix}, \quad C(k) \in \begin{pmatrix} [1,\ 2] & 1 \\ 0 & [1,\ 3] \end{pmatrix}.$$

The sequence of observation obtained via simulation[4] is given by: $\mathbf{z}(1) = [14.16,\ 13.18]^T$ and $\mathbf{z}(2) = [18.00,\ 19.68]^T$. Using this sequence and the initial condition $X_0 = \left\{\mathbf{x} \in \mathbb{R}^2 : 0 \le x_1 \le 3,\ 0 \le x_2 \le 8\right\}$, the sets $X_{k|k}$ for $k = 1, 2$, were calculated according to (49). Figure 2 depicts the sets $X_{k|k}$ and the set of initial conditions X_0.

7 Conclusions

In this work we have presented a procedure to partition the state space of an uMPL system into components that can be completely characterized by DBM. This has lead us to be able to present a procedure for computing the

---

[4] For the simulation, it was considered that the matrices entries are uniformly distributed in the corresponding intervals, e.g., the element a_{12} of $A(k)$ is uniformly distributed between 5 and 6.

image and the inverse image of a DBM w.r.t. each component of the partitioned uMPL system which is similar to the procedure of computing the image and the inverse image of a DBM w.r.t. each component of a PWA system generated by an (deterministic) MPL system. Consequently, most of the previous results on reachability analysis of MPL systems could be extended to uMPL systems. The complexity of the proposed algorithms has the same worst-case bound comparing with the algorithms proposed in [4], with the advantage of handling a broader class of MPL systems. In section 5, we have presented an application of the approach were we use the forward and backward reachability analysis to compute the support of the posterior PDF of the states of an uMPL system.

References

[1] D. Adzkiya, B. De Schutter, and A. Abate. Finite abstractions of max-plus-linear systems. *Automatic Control, IEEE Transactions on*, 58(12):3039–3053, Dec 2013.

[2] D. Adzkiya, B. De Schutter, and A. Abate. Backward reachability of autonomous max-plus-linear systems. In *Proceedings of the 12th IFAC/IEEE Workshop on Discrete Event Systems*, pages 117–122, Cachan, France, May 2014.

[3] D. Adzkiya, B. De Schutter, and A. Abate. Forward reachability computation for autonomous max-plus-linear systems. In *Proceedings of the 20th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2014)*, pages 248–262, Grenoble, France, April 2014.

[4] D. Adzkiya, B. De Schutter, and A. Abate. Computational techniques for reachability analysis of max-plus-linear systems. *Automatica*, 53(3):293–302, 2015.

[5] D. Adzkiya, Y. Zhang, and A. Abate. Verisimpl 2: An open-source software for the verification of max-plus-linear systems. *Discrete Events Dynamical Systems*, 2015.

[6] Xavier Allamigeon, Stéphane Gaubert, and Éric Goubault. Inferring min and max invariants using max-plus polyhedra. In *Proceedings of the 15th International Symposium on Static Analysis*, SAS'08, pages 189–204, Berlin, Heidelberg, 2008. Springer-Verlag.

[7] S. Amari, I. Demongodin, J. J. Loiseau, and C. Martinez. Max-plus control design for temporal constraints meeting in timed event graphs. *IEEE Transactions on Automatic Control*, 57(2):462–467, Feb 2012.

[8] F. Baccelli, G. Cohen, G.J. Olsder, and J.P. Quadrat. *Synchronization and Linearity : An Algebra for Discrete Event Systems*. Wiley and Sons, 1992.

[9] Richard Bellman. On a Routing Problem. *Quarterly of Applied Mathematics*, 16:87–90, 1958.

[10] Alberto Bemporad, Giancarlo Ferrari-trecate, and Manfred Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE Transactions on Automatic Control*, 45:1864–1876, 1999.

[11] T. Brunsch, J. Raisch, and L. Hardouin. Modeling and control of high-throughput screening systems. *Control Engineering Practice*, 20:1:14–23, January 2012. doi:10.1016/j.conengprac.2010.12.006.

[12] Thomas Brunsch, Laurent Hardouin, Carlos Andrey Maia, and Joerg Raisch. Duality and interval analysis over idempotent semirings. *Linear Algebra and its Applications*, 437:2436–2454, 2012.

[13] Katarna Cechlrov. Eigenvectors of interval matrices over maxplus algebra. *Discrete Applied Mathematics*, 150(1):2 – 15, 2005.

[14] Katarna Cechlrov and R.A. Cuninghame-Green. Interval systems of max-separable linear equations. *Linear Algebra and its Applications*, 340(1):215 – 224, 2002.

[15] Guy Cohen, Stéphane Gaubert, and Jean-Pierre Quadrat. Max-plus algebra and system theory: Where we are and where to go now. *Annual Reviews in Control*, 23:207 – 219, 1999.

[16] DavidL. Dill. Timing assumptions and verification of finite-state concurrent systems. In Joseph Sifakis, editor, *Automatic Verification Methods for Finite State Systems*, volume 407 of *Lecture Notes in Computer Science*, pages 197–212. Springer Berlin Heidelberg, 1990.

[17] M. DiLoreto, S. Gaubert, R. Katz, and J-J Loiseau. Duality between invariant spaces for max-plus linear discrete event systems. *SIAM Journal on Control and Optimization*, 48(8):5606–5628, December 2010.

[18] Robert W. Floyd. Algorithm 97: Shortest path. *Communications of the ACM*, 5(6):345, 1962.

[19] Stephane Gaubert and Ricardo Katz. Reachability and invariance problems in max-plus algebra. In *Positive Systems, Proceedings of the First Multidisciplinary International Symposium on Positive Systems: Theory and Applications (POSTA 2003), Rome, Italy*, pages 15–22, 2003.

[20] Gazarik, J. Michael, Kamen, and W. Edward. Reachability and observability of linear systems over max-plus. *Kybernetika*, 35(1):[2]–12, 1999.

[21] V. M. Gonçalves, C. A. Maia, and L. Hardouin. On the steady-state control of timed event graphs with firing date constraints. *IEEE Transactions on Automatic Control*, 61(8):2187–2202, Aug 2016.

[22] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, 140(2):107 –113, apr 1993.

[23] L. Hardouin, B. Cottenceau, M. Lhommeau, and E. Le Corronc. Interval systems over idempotent semiring. *Linear Algebra and its Applications*, 431(5-7):855–862, August 2009. `doi:10.1016/j.LAA.2009.03.039`.

[24] L. Hardouin, C. A. Maia, B. Cottenceau, and M. Lhommeau. Max-plus Linear Observer: Application to manufacturing Systems. In *, WODES'10*, pages 171–176, September 2010. `istia.univ-angers.fr/~hardouin/Observer.html`.

[25] W. P. M. H. Heemels, B. De Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *AUTOMATICA*, 37:1085–1091, 2001.

[26] B. Heidergott, G.J. Olsder, and J.W. van der Woude. *Max Plus at Work: Modeling and Analysis of Synchronized Systems : a Course on Max-Plus Algebra and Its Applications*. Number v. 13 in Max Plus at work: modeling and analysis of synchronized systems : a course on Max-Plus algebra and its applications. Princeton University Press, 2006.

[27] Bernd F. Heidergott. *Max-Plus Linear Stochastic Systems and Perturbation Analysis (The International Series on Discrete Event Dynamic Systems)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[28] R. D. Katz. Max-plus (a,b)-invariant spaces and control of timed discrete-event systems. *IEEE Transactions on Automatic Control*, 52(2):229–241, Feb 2007.

[29] M. Lhommeau, L. Hardouin, J.-L. Ferrier, and I. Ouerghi. Interval Analysis in Dioid : Application to Robust Open Loop Control for Timed Event Graphs. In *and*, pages 7744–7749, seville, dec 2005. `doi:10.1109/CDC.2005.1583413`.

[30] Mehdi Lhommeau, Laurent Hardouin, and Bertrand Cottenceau. Optimal control for (max, +)-linear systems in the presence of disturbances. In *Positive Systems, Proceedings of the First Multidisciplinary International Symposium on Positive Systems: Theory and Applications (POSTA 2003), Rome, Italy, August 28-30, 2003*, pages 47–54, 2003.

[31] Grigori L. Litvinov and Andrei N. Sobolevskiī. Idempotent interval analysis and optimization problems. *Reliable Computing*, 7(5):353–377, 2001.

[32] Qi Lu, Michael Madsen, Martin Milata, Sren Ravn, Uli Fahrenberg, and Kim G. Larsen. Reachability analysis for timed automata using max-plus algebra. *The Journal of Logic and Algebraic Programming*, 81(3):298 – 313, 2012.

[33] C. A. Maia, L. Hardouin, R. Santos-Mendes, and J. J. Loiseau. A super-eigenvector approach to control constrained max-plus linear systems. In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 1136–1141, Dec 2011.

[34] Ramon E. Moore and Fritz Bierbaum. *Methods and Applications of Interval Analysis (SIAM Studies in Applied and Numerical Mathematics) (Siam Studies in Applied Mathematics, 2.)*. Soc for Industrial & Applied Math, 1979.

[35] Helena Mykov. Interval systems of max-separable linear equations. *Linear Algebra and its Applications*, 403:263 – 272, 2005.

[36] I. Necoara, B. De Schutter, T.J.J. van den Boom, and H. Hellendoorn. Robust control of constrained max-plus-linear systems. *Int. J. of Robust and Nonlinear Control*, 19(2):218–242, January 2009.

[37] G. J. Olsder, J. A. C. Resing, R. E. De Vries, M. S. Keane, and G. Hooghiemstra. Discrete event systems with stochastic processing times. *IEEE Transactions on Automatic Control*, 35(3):299–302, Mar 1990.

[38] E. D. Sontag. Nonlinear regulation, the piecewise linear approach. In *Proc.Princeton Conf.on Information Sciences and Systems, Princeton, March 1980*, 1980.

[39] T.J.J. van den Boom and B. De Schutter. Model predictive control for perturbed max-plus-linear systems. *Systems & Control Letters*, 45(1):21–33, January 2002.