(Max,+)-automata with partial observations

J. Komenda * **S. Lahaye** ** **J.L. Boimond** **

* *Institute of Mathematics, Czech Academy of Sciences, Brno Branch, Czech Republic*
** *LARIS, Angers University, Angers, France*

Abstract: (Max,+)-automata are weighted automata over the (max,+) semiring. In this paper we investigate simulation like equivalences between (max,+)-automata. Since (max,+)-automata are nondeterministic (weighted) automata, there exist extensions of bisimilarity properties that are weaker than equality of their weighted languages (formal power series). The main advantage of bisimulation like properties is that they can be checked in polynomial time, while equality (as well as inequality) of formal power series is undecidable. We show that a form of weak simulation can be used as a sufficient condition for comparing the formal power series.

Keywords: (Max,+)-automata, partial observations, weak bisimulation, residuation theory.

1. INTRODUCTION

An important class of timed discrete-event systems Cassandras and Lafortune (2008) can be modeled by (max,+)-automata, which are weighted automata with weights (multiplicities) in the $(\mathbb{R}\cup\{-\infty\},\max,+)$ semiring. They have been introduced in Gaubert (1995) as a model of Timed Discrete Event (dynamical) Systems (TDES) exhibiting both synchronization of tasks and resource sharing (choice) phenomena and moreover can be nondeterministic.

It has been shown in Gaubert and Mairesse (1999) that (max,+)-automata have a strong expressive power in terms of timed Petri nets: every 1-safe timed Petri net can be represented by special (max,+)-automata, called heap models. A compositional approach to the modeling of timed Petri nets using (max,+)-automata has been presented in Lahaye et al. (2015).

So far very limited effort has been devoted to the investigation of partially observed (max,+)-automata. Similarly as for logical automata, partial observations arise when some events can not be monitored because of high cost of sensors or due to technical reasons. Moreover, partial observations are also useful as abstractions in hierarchical control of finite automata, where natural projections consisting in abstracting some low level events are less complex if observer property is satisfied. It is to be expected that hierarchical control will also be useful for timed systems.

The goal of this paper is twofold. Firstly, we will present some results about partially observed weighted automata from the computer science literature in the standard (max,+) algebraic framework and extend them. Secondly, we will show that partial observations are useful for approximated solutions of fundamental undecidable problems in (max,+)-automata, namely equalities and comparisons of their behaviors, i.e. rational (max,+)-formal power series. It is well known that equalities as well as inequalities of (max,+) formal power series are undecidable in general Krob (1994). However, bisimulations and simulations as stronger properties (implying respectively equalities and inequalities of behaviors) are decidable in polynomial time. For timed discrete-event systems, the latter properties are however too strong. We show that after rendering different subsets of events unobservable, proposed variants of the underlying weak bisimulations and weak simulations are weaker conditions than bisimulations and simulations, while they still imply the equalities and inequalities of behaviors, i.e. (max,+) formal power series.

2. ALGEBRAIC PRELIMINAIRES

In this section results and notations about (max,+) algebra, (max,+)-automata, and Petri nets are introduced. The reader is invited to consult the references Baccelli et al. (1992), Gaubert (1992), and Gaubert (1995) for more complete treatement.

The semiring $\mathbb{R}_{\max} = (\mathbb{R}\cup\{-\infty\},\max,+)$ is commonly known as (max,+) algebra (with a wrong use of the term algebra). This semiring is called idempotent, because it has idempotent addition, denoted by $\oplus$: $a\oplus b = \max(a,b)$. The conventional addition plays the role of multiplication, denoted by $a\otimes b$ (or ab when unambiguous). The zero element is denoted by ε $(=-\infty)$ and the identity element is denoted by e $(=0)$. Idempotent semirings are usually called dioids. If we add $T=+\infty$ to $\mathbb{R}_{\max}$, the resulting dioid is complete (as an ordered set) and denoted $\overline{\mathbb{R}}_{\max}$.

The set of square $n\times n$ matrices with coefficients in the (max,+) algebra $\mathbb{R}_{\max}$, endowed with usually defined matrix addition and multiplication, also denoted by $\oplus$ and $\otimes$, is an idempotent semiring, denoted $\mathbb{R}_{\max}^{n\times n}$. The zero element is the matrix denoted ε_n and exclusively composed of ε $(=-\infty)$. The identity element I_n of $\mathbb{R}_{\max}^{n\times n}$ is the matrix with e $(=0)$ on the diagonal and ε $(=-\infty)$ elsewhere. We will also work with non square $m\times n$ matrices, where the multiplication is defined for matrices of compatible sizes in the same way as in the classical linear algebra. The transposed matrix of A is denoted by A^T, where $A^T_{ij} = A_{ji}$.

We need a concept of Boolean residuation (also called relative residuation in Damljanovic et al. (2014)) that is not usual in the literature on (max,+)-linear systems. Residuation theory allows defining 'pseudo-inverses' of some isotone maps (f is isotone if $a \preceq b \Rightarrow f(a) \preceq f(b)$) defined on ordered sets, such as dioids (see Baccelli et al. (1992), 4.4.4).

Definition 1. An isotone map $f : \mathscr{D} \to \mathscr{C}$, where $\mathscr{D}$ and $\mathscr{C}$ are complete dioids, is said to be residuated if there exists an isotone map $h : \mathscr{C} \to \mathscr{D}$ such that

$$f \circ h \preceq Id_{\mathscr{C}} \text{ and } h \circ f \succeq Id_{\mathscr{D}}, \quad (1)$$

where $Id_{\mathscr{C}}$ and $Id_{\mathscr{D}}$ are identity maps of $\mathscr{C}$ and $\mathscr{D}$ respectively. Recall that h is unique, it is denoted $f^\sharp$ and called residual of f.

Residuals of isotone maps correspond to extremal solutions of inequalities. If f is residuated then $\forall y \in \mathscr{C}$, the least upper bound of subset $\{x \in \mathscr{D} | f(x) \preceq y\}$ exists and belongs to this subset. This greatest solution is equal to $f^\sharp(y)$.

Theorem 2. In a complete dioid $\mathscr{D}$ the isotone map $R_a : x \mapsto x \otimes a$ is residuated. The greatest solution of $x \otimes a \preceq b$ exists and is equal to $R_a{}^\sharp(b)$, also denoted $b \phi a$. This 'quotient' satisfies the following formulæ

$$(x \phi a) \otimes a \preceq x, \qquad (f.1) \qquad (x \otimes a) \phi a \succeq x. \qquad (f.2)$$

We will need residuation of matrix multiplication.

The residuated mapping of the left matrix multiplication, *i.e.* the greatest solution X to the inequality $A \otimes X \leq B$ is denoted $A \backslash B$. Recall from Baccelli et al. (1992) that for matrices $A \in \mathscr{D}^{m \times n}$ and $B \in \mathscr{D}^{m \times p}$ over a complete dioid $\mathscr{D}$, where the infimum is denoted by $\wedge$ we have

$$X = A \backslash B \in \mathscr{D}^{n \times p} : \ (A \backslash B)_{ij} = \wedge_{k=1}^{m} A_{ki} \backslash B_{kj}.$$

Similarly, the residuated mapping of the right multiplication, *i.e.* the greatest solution Y to the inequality $Y \otimes C \leq F$ for given matrices $C \in \mathscr{D}^{m \times n}$ and $F \in \mathscr{D}^{p \times n}$ is denoted $F \phi C$. We recall from Baccelli et al. (1992) that

$$Y = F \phi C \in \mathscr{D}^{p \times m} : \ (F \phi C)_{ij} = \wedge_{k=1}^{n} F_{ik} \phi C_{jk}.$$

Now we are ready to recall relative (Boolean) right residuals from Damljanovic et al. (2014).

Definition 3. (Boolean right residuation)
Given matrices $A \in \overline{\mathbb{R}}_{\max}^{m \times n}$ and $B \in \overline{\mathbb{R}}_{\max}^{m \times p}$, the Boolean right residual, $A \backslash B \in \mathbb{R}_{\max}^{n \times p}$, is defined as

$$(A \backslash^{\mathscr{B}} B)_{ij} = \begin{cases} e = 0, & \text{if } \forall k = 1, \ldots, m : \ A_{ki} \leq B_{kj}, \\ \varepsilon = -\infty, & \text{otherwise.} \end{cases}$$

Note that Boolean right residuation can be defined in an equivalent way using residuation of the canonical injection of the Boolean semiring $\mathscr{B} = (\varepsilon, e, \oplus, \otimes)$, where $\oplus$ is logical "or" and $\otimes$ is logical "and ", to $\overline{\mathbb{R}}_{\max}$ denoted by $I : \mathscr{B} \to \overline{\mathbb{R}}_{\max}$. Then $\mathscr{B}$ is a complete subdioid of the dioid $\overline{\mathbb{R}}_{\max}$ and the residuated mapping $I^\sharp : \mathbb{R}_{\max} \to \mathscr{B}$ exists and is defined for $r \in \mathbb{R}_{\max}$ by

$$I^\sharp(r) = \begin{cases} e = 0, & \text{if } r \geq 0, \\ \varepsilon = -\infty, & \text{otherwise.} \end{cases}$$

Now it can be observed that $A \backslash^{\mathscr{B}} B = I^\sharp(A \backslash B)$. Similarly, Boolean left residual can be defined by $A \phi^{\mathscr{B}} B = I^\sharp(A \phi B)$. Otherwise stated, Boolean residuation is nothing else, but the residuation of the following composed mapping: the canonical injection I composed with matrix multiplication.

(Max,+)-automata are recalled below.

Definition 4. ((Max,+)-automaton)
A *(max,+)-automaton* G is a quintuple $(Q, A, \alpha, \mu, \beta)$ with

- Q and A are resp. finite sets of states and of events;

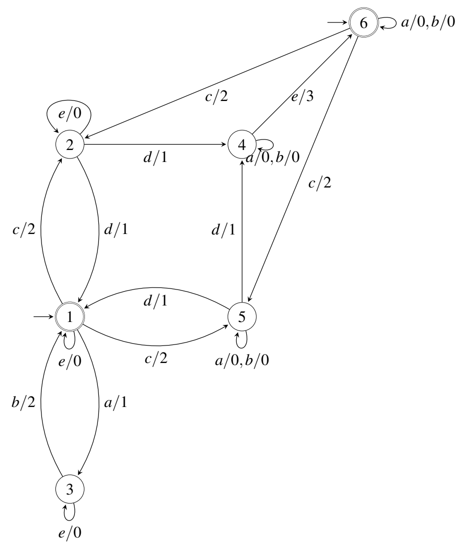

Fig. 1. A (max,+)-automaton.

- $\alpha \in \mathbb{R}_{\max}^{1 \times |Q|}$ is a vector of initial delays: $\alpha_q \neq \varepsilon$ means the state $q \in Q$ is initial with the intial delay given by α_q, while $\alpha_q = \varepsilon$ means that q is not an initial state;
- $\mu : A^* \to \mathbb{R}_{\max}^{|Q| \times |Q|}$ is a (monoid) morphism given by the family of matrices $\mu(a) \in \mathbb{R}_{\max}^{|Q| \times |Q|}$, $a \in A$: for a sequence $w = a_1 a_2 \ldots a_n \in A^*$, we have
$$\mu(w) = \mu(a_1 a_2 \ldots a_n) = \mu(a_1)\mu(a_2)\ldots\mu(a_n);$$
- $\beta \in \mathbb{R}_{\max}^{|Q| \times 1}$ is a vector of final delays: $\beta_q \neq \varepsilon$ means the state $q \in Q$ is final with the final delay given by β_q, while $\beta_q = \varepsilon$ means that q is not a final state;

We recall that $\mu : A^* \to \mathbb{R}_{\max}^{|Q| \times |Q|}$ is a (monoid) morphism means that for all $a, b \in A$ we have $\mu(ab) = \mu(a) \otimes \mu(b)$, which can be naturally extended to a finite number of factors and from events to sequences (words) as well. An example of a (max,+)-automaton is displayed on figure 1. There are two states which are both initial and final: 1 and 6 and no other states are neither initial nor final. The final states are displayed with double circles. We use a convention that initial and final delays are not added if they are equal to $e = 0$.

Note that morphism matrices correspond to transitions in the following way: $[\mu(a)]_{qq'} \neq \varepsilon$ means that there exists a transition from state q to state q' labeled by event a and $[\mu(a)]_{qq'}$ is then the duration of this transition. We call π a *path* from q_0 to q_m. We denote $\sigma(\pi)$ the product $\otimes$ of the weights along the path π, that is

$$\sigma(\pi) = \bigotimes_{i=1,\ldots,m} t(q_{i-1}, a_i, q_i) = \bigotimes_{i=1,\ldots,m} [\mu(a_i)]_{q_{i-1}, q_i}.$$

Let $p, q \in Q$ and $w \in A^*$. We denote by $p \overset{w}{\rightsquigarrow} q$ the set of paths from p to q which are labeled by w. It can be shown that

$$[\mu(a_1 a_2 \ldots a_m)]_{q_0 q_m} = \bigoplus_{\pi \in q_0 \overset{a_1 \ldots a_m}{\rightsquigarrow} q_m} \sigma(\pi) \ . \qquad (2)$$

A (max,+)-automaton is said to be *deterministic* if there is a unique initial state and for all $a \in A$ there is at most one transition with label a from each state. This means that each line of $\mu(a)$ contains at most one nonzero element (i.e. not equal to $\varepsilon = -\infty$).

The state of a (max,+)-automaton G is a vector $x(w) \in \mathbb{R}_{\max}^{1\times|Q|}$ that is defined for $w \in A^*$ as follows:

$$x(w) = \alpha \otimes \mu(w) \ . \tag{3}$$

The component of the state vector $[x(w)]_q$ is interpreted as the date at which state $q \in Q$ is reached after execution of sequence of transitions w starting from an initial state (with the convention that $[x(w)]_q = \varepsilon$ if state q is not reached from an initial state using the input sequence w).

The elements of x are known as *generalized daters*, and they are governed by recurrent equations

$$\begin{aligned} x(\varepsilon) &= \alpha, \\ x(wa) &= x(w)\mu(a). \end{aligned} \tag{4}$$

The behavior of a (max,+)-automaton is then defined as the formal power series $l(G) : A^* \to \overline{\mathbb{R}}_{\max}$ defined by

$$l(G)(w) = x(w) \otimes \beta = \alpha \otimes \mu(w) \otimes \beta. \tag{5}$$

The language of a (max,+)-automaton (i.e. the support of its formal power series) contains the words $w \in A^*$ such that there exists a final state q with $[x(w)]_q \neq \varepsilon$.

3. PARTIALLY OBSERVED TDES AND THEIR BEHAVIOURS

In this section, we investigate fundamental properties of natural projections for (max,+)-automata. We start with defining natural projections. As usual in partially observed systems we assume that the event set A is decomposed into observable event subset $A_o \subseteq A$ and unobservable event subset $A_{uo} = A \setminus A_o$. We recall that *natural projection* $P : A^* \to A_o^*$ is a homomorphism (of free monoids) defined as $P(a) = \varepsilon$ (ε here denotes the empty string), for $a \in A_{uo}$, and $P(a) = a$, for $a \in A_o$. The *inverse image* of P is defined for $w \in A_o^*$ as $P^{-1}(w) = \{s \in A^* \mid P(s) = w\}$, i.e. it outputs a language (set of words).

A major problem with partially observed quantitative (e.g. timed or probabilistic) DES such as (max,+)-automata is that unobservable transitions still may have effects on the observable quantitative behaviour of the system. Namely, we assume that unobservable events can not be monitored, but still have an observed duration. Therefore, given a (max,+) formal power series we propose the following definition of natural projection.

Definition 5. (Projection of (max,+)-series)
For $s : A^* \to \mathbb{R}_{\max}$ we define $Ps : A_o^* \to \mathbb{R}_{\max}$ by setting

$$Ps(w) = \bigoplus_{v \in P^{-1}(w)} s(v).$$

For instance, if $s = 1a \oplus 3a\tau \oplus 4a\tau a$ with $A_o = \{a\}$ then $P(s) = 3a \oplus 4aa$, because by definition $v \in P^{-1}(a) = \{a, a\tau\}$ and $P(s)(a) = s(a) \oplus s(a\tau) = 1 \oplus 3 = 3$.

Unlike logical automata, we define natural projection of (max,+)-automata as nondeterministic automata. This is natural, because firstly (max,+)-automata as weighted automata are nondeterministic (which distinguish them from Mealy automata with outputs in a semiring), and secondly, it is well known that not all (max,+)-automata can be determinized. The definition below consists simply in removing unobservable events and the duration of unobservable events is putting forward: it is added to the duration of (all) the next observable events. We denote by $\mu(\tau)^* = (\bigoplus_{a \in A_{uo}} \mu(a))^*$. Note that we have due to the morphism property that $\mu(\tau)^* = \bigoplus_{v \in A_{uo}^*} \mu(v)$.

The advantage of projected automata defined as a nondeterministic (max,+)-automata is that the projected automata can be defined over the same state space as the original (max,+)-automaton, which simplifies the algebraic manipulations with them. We use the following definition.

Definition 6. (Projected (max,+)-automaton)
Let $G = (Q, A, \alpha, \mu, \beta)$ be a (max,+)-automaton and $A_o \subseteq A$ denotes the subset of observable events with associated natural projection $P : A^* \to A_o^*$. The projected (max,+)-automaton is defined as $PG = (Q, A_o, \alpha_P, \mu_P, \beta_P)$, where

- $\alpha_P = \alpha$;
- $\mu_P : A_o^* \to \mathbb{R}_{\max}^{|Q|\times|Q|}$ is a morphism with generators $\mu_P(a) = \mu(\tau)^* \otimes \mu(a)$ for $a \in A_o$;
- $\beta_P = \mu(\tau)^* \otimes \beta$;

Remark 7. (1) The above definition simply means that we do not observe the occurrence of unobservable events, but we keep track of the time that has elapsed when unobservable events occur: we add the duration of (a sequence of) unobservable events to the next observable event if it exists or to the final delay of the previous observable event if there is no observable event after this sequence of unobservable events. The latter case typically occur in the case of non live (finite) support languages.

(2) We recall at this point that a similar definition, associating an automaton $\mathscr{A}^*$ to an automaton $\mathscr{A}$ has already appeared in Buchholz and Kemper (2003). The definition therein has a serious drawback. The transition matrix is defined as $\mu_P(a) = \mu(\tau)^* \otimes \mu(a) \otimes \mu(\tau)^*$. The definition of $\mathscr{A}^*$ is based on the transition function rather than morphism matrix and the matrix mapping corresponding to the transition function of $\mathscr{A}^*$ is not a morphism from a free monoid of observable words to the multiplicative monoid of (max,+)-matrices. This is because the authors define the transition function with the empty string (no observation yet) as a transition function based on transitive closure of unobservable transitions, also known as unobservable reach. However, the morphism property requires that the empty word is mapped into the identity matrix, which is not true in general. Hence, we prefer definition above that enables to preserve the morphism property of μ_P. Namely, we have that $\mu_P(\varepsilon) = I_{\|Q\|}$, the (max,+)-identity matrix. It should be clear that due to $\mu(\tau)^* \otimes \mu(\tau)^* = \mu(\tau)^*$ both definitions lead to the same projected automata, because in (max,+)-semiring we have $a^* \otimes a^* = a^*, \forall a$, due to idempotency. The definition from Buchholz and Kemper (2003) is given in the more general setting of weighted automata, which include among others probabilistic automata that are defined over a non idempotent semiring.

(3) This construction based on $\mu(\tau)^*$ naturally requires an assumption ensuring that $\mu(\tau)^*$ can be computed, e.g. that there is no cycle of unobservable events.

It is easy to see that the behavior of a projected (max,+)-automaton corresponds to the natural projection of its behavior (associated (max,+)-formal power series). Namely, we have the following property, which simply means that similarly as for logical automata, the above definitions of natural projections for max,+)-automata and for formal power series are compatible with each other.

Proposition 8. For any $w \in A_o^*$ we have $l(PG)(w) = P(l(G))(w)$.

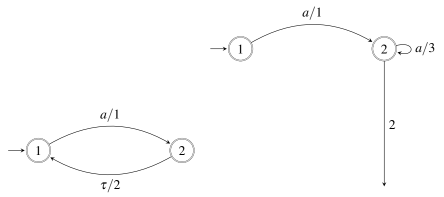

Fig. 2. G (left) and its projected automaton PG (right).

An example of a (max,+)-automaton $G = (Q, A, \alpha, \mu, \beta)$ together with its projected (max,+)-automaton

$$PG = (Q, A_o, \alpha_P, \mu_P, \beta_P),$$

is given on figure 2. In this example we have

$$\alpha_P = \alpha = [0 \ \ \varepsilon],$$

$$\mu_P(a) = \mu(\tau)^* \otimes \mu(a) = \begin{bmatrix} 0 & \varepsilon \\ 2 & 0 \end{bmatrix} \otimes \begin{bmatrix} \varepsilon & 1 \\ \varepsilon & \varepsilon \end{bmatrix} = \begin{bmatrix} \varepsilon & 1 \\ \varepsilon & 3 \end{bmatrix}, \text{ and}$$

$$\beta_P = \mu(\tau)^* \otimes \beta = \begin{bmatrix} 0 & \varepsilon \\ 2 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

3.1 Relations between (max,+)-automata

It is well known that most common relations between (max,+)-automata consisting in comparing their behaviors in terms of equality or inequality of their formal power series cannot be used in practice, because equality as well as inequality of rational formal power series are undecidable in general. On the other hand, stronger behavioural equivalence, called simulation relations between (max,+)-automata are decidable for weighted automata as shown in Damljanovic et al. (2014).

Any relation $R \subseteq Q \times Q'$ can be viewed as a Boolean matrix $B \in \{\varepsilon, e\}^{|Q| \times |Q'|}$, where $(q, q') \in R$ iff $B_{q,q'} = e$. We recall an algebraic approach to simulations from Damljanovic et al. (2014).

Definition 9. (Simulation of (max,+)-automaton)
A Boolean matrix $B \in \{\varepsilon, e\}^{|Q| \times |Q'|}$ is called a (forward) simulation between (max,+)-automata $G = (Q, A, \alpha, \mu, \beta)$ and $G' = (Q', A, \alpha', \mu', \beta')$ if the following inequalities hold true:

- $\alpha \leq \alpha' \otimes B$,
- $B \otimes \mu(a) \leq \mu'(a) \otimes B$ and
- $B \otimes \beta \leq \beta'$.

As usual, we say that G' is simulated by G if there exists a simulation between G and G'. Similarly, G and G' are called bisimilar if there exist two simulations: B between G and G' and $\tilde{B}$ between G' and G. Note that it is not necessary that $\tilde{B}$ is a transposed matrix (relation) of B. Hence, a bisimulation is not simply defined by replacing the three inequalities in the definition of a simulation by equalities, because it would be just a special case of bisimulation, where $\tilde{B} = B^T$. However, if there exists a Boolean matrix $B \in \{\varepsilon, e\}^{|Q| \times |Q'|}$such that the three inequalities in Definition hold as equalities, then B is called a bisimulation between G and G'.

The existence of a simulation relation between G and G' requires that for all paths $(q_0, a_1, q_1) \ldots (q_{n-1}, a_n, q_n)$ in G there exists a path $(q'_0, a_1, q'_1) \ldots (q'_{n-1}, a_n, q'_n)$ in G' such that $\mu(a_i)_{q_{i-1} q_i} \leq \mu(a_i)_{q'_{i-1} q'_i}$ for all $i = 1, \ldots, n$. It should be clear that if B exists then it comes that $l(G) \leq l(G')$, i.e. for all $w \in A^*$ we have that $l(G)(w) \leq l(G')(w)$.

In particular, if both G and G' share the same set of states, a diagonal relation (corresponding to the identity matrix) is a simulation if $\alpha \leq \alpha'$, $\beta \leq \beta'$, and $\mu(a) \leq \mu'(a)$. It should be clear that simulation is too strong for (max,+)-automata if one is rather interested in comparing the formal power series, because it requires not only that all logical transitions possible in one automaton are mimicked in the second one, but also that the durations of all transitions in the first automaton are smaller or equal to the durations of corresponding transitions in the second automaton. This is clearly too strong, because in manufacturing systems and other application areas it is much more relevant to compare the execution times of completed sequences (corresponding to words from underlying marked languages) rather than the durations of individual events. In fact, this is too restrictive to require that all events from a sequence in one automaton have a duration smaller or equal to the duration of matching event in the other automaton in order to guarantee an inequality of two formal power series.

A natural possibility is then to replace simulation by weak simulation. The following definition has been introduced in Buchholz and Kemper (2003).

Definition 10. (Weak simulation between (max,+)-automata)
A relation $R \subseteq Q \times Q'$ is called a (forward) weak simulation between (max,+)-automata $G = (Q, A, \alpha, \mu, \beta)$ and $G' = (Q', A', \alpha', \mu', \beta')$ if R is a simulation between PG and PG'.

Note that this definition is possible because of the same state sets for the original and the projected (max,+)-automata. Since simulations are stronger than inequalities of behaviors, it immediately follows that existence of a weak simulation between (max,+)-automata G and G' implies that $l(PG) \leq l(PG')$, where $l(PG), l(PG') : A_o^* \to \mathbb{R}_{\max}$ are behaviors of projected automata.

The following algebraic characterization of weak simulation between (max,+)-automata is straightforward from definition 4 by combining algebraic characterization of simulations in definition 3.1 and definition 6 of projected automata.

Proposition 11. $B \in \{\varepsilon, e\}^{|Q| \times |Q'|}$ is a (forward) weak simulation between (max,+)-automata $G = (Q, A, \alpha, \mu, \beta)$ and $G' = (Q', A', \alpha', \mu', \beta')$ if, and only if, the following inequalities hold true:

- $\alpha \leq \alpha' \otimes B$,
- $B \otimes \mu(\tau)^* \otimes \mu(a) \leq \mu'(\tau)^* \otimes \mu'(a) \otimes B$ and
- $B \otimes \mu(\tau)^* \otimes \beta \leq \mu(\tau)^* \otimes \beta'$

A fix-point algorithm for computing the largest simulations (and bisimulations) has been presented in Damljanovic et al. (2014). This algorithm can be adapted for computing the largest weak simulations. Now we recast the algorithm from Damljanovic et al. (2014) for checking the existence of a simulation between weighted automata G_1 and G_2. This algorithm consists in fixpoint computation of the largest simulation. It is based on the concept of Boolean residuation of matrices defined in Definition 3. Boolean matrices B_i, $i = 0, 1, 2, \ldots$ below are computed in a recursive way.

$$B_0 = \beta \backslash^{\mathscr{B}} \beta',$$

$$B_{i+1} = B_i \wedge \bigwedge_{a \in A} \left[(\mu'(a) \otimes B_i^T) /^{\mathscr{B}} \mu(a) \right]^T$$

Algorithm from Damljanovic et al. (2014) starts with computation of the Boolean matrix $B_0 = \beta \backslash^{\mathscr{B}} \beta'$. Then the sequence of

Boolean matrices is constructed recursively and at each step it is checked whether $B_{i+1} = B_i$. If this is the case then this fix-point construction outputs matrix B_k for $k = i$ as the largest Boolean matrix, which satisfies the second and the third axioms (ii) and (iii) of (forward) simulation, cf. Definition 3.1. It then suffices to check if B_k satisfies the axiom (i) of Definition 3.1. If this is the case, then B_k is the greatest forward simulation between G and G'. In the opposite case there does not exist any forward simulation between G and G'. Note that since the sequence $\{B_i\}_{i=0,1,...}$ is descending and the set of Boolean matrices from is clearly finite, the algorithm converges in at most $O(|Q| \times |Q'|)$ steps. The above fix-point computation of the largest bisimulation is based on matrix/vector residuation, which is well known to be equivalent to the dual (i.e. maximum replaced by minimum) multiplication with the so called conjugate matrix (i.e. transposed matrix with inversed coefficients) of Butkovic (2010). It is well known that matrix multiplication requires at most $O(mn^2)$ for non square matrices in $\mathbb{R}_{\max}^{m \times n}$ which include Boolean matrices as well, and that this upper bound can be improved. Altogether, it can be stated that the above algorithm is of polynomial time complexity in the number of states of the (max,+)-automata (i.e. the dimensions of the corresponding matrices), because it only uses polynomial time operations on Boolean and (max,+) matrices.

4. AN APPROXIMATE APPROACH TO COMPARING BEHAVIORS

Some researchers have been interested in approximate solutions to algorithmically unfeasible problems such as determinization of (max,+)-automata. In this paper we are interested in semi-decision procedures for checking inequalities between (max,+) formal power series, a well known undecidable problem.

In this paper our approach goes against the traditional application of weak simulations, which is a variant of simulation for partially observed systems, where some internal (unobservable) actions (denoted by τ in process algebra community) occur in the system. We rather use weak simulations as a means to weaken simulation as defined previously and which appears as a too strong condition for a comparison purpose. However, definition 4 stated above is not suitable for our purposes. The main reason is that it does not distinguish between unobservable events but considers only a single type of unobservable transition, denoted by τ and called internal action. This is actually common to the computer science literature on process algebras, where instead of different unobservable events there is a single unobservable (internal) event denoted by τ.

A natural question is how to choose a subset of observable events. Obviously, we want to render as much as possible events unobservable in order to cope with the main issue related to (strong) simulation, namely the requirement that all related transitions must satisfy a given inequality, while for inequality of formal power series it is sufficient that the total weight of a path satisfy the inequality. However, due to the term $\mu(\tau)^*$ used in (max,+)-automata with partial observations we cannot choose too many events as unobservable, because we exclude cycles labeled solely by unobservable events in order to guarantee convergence in the computation of $\mu(\tau)^*$. We can then consider different maximal sets of unobservable events consisting in choosing one observable event per cycle or equivalently, rendering unobservable all but one event in every cycle. It should be stated that this approach naturally can not give an exact answer to the undecidable problem of checking inequality of formal power series, but only a partial answer. This is also related to the fact that cycles of unobservable events that should be excluded.

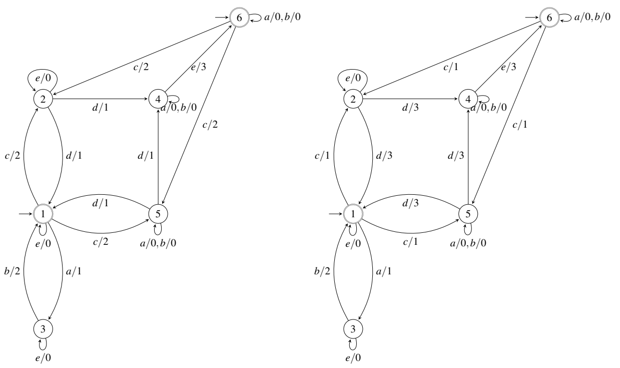

Fig. 3. (Max,+)-automata G_1 (left) and G_2 (right)

We have already mentioned that the definition of weak simulation is not suitable in our situation, because we do need to distinguish between different unobservable events in order to guarantee that existence of a weak simulation is sufficient for comparing behaviors (formal power series). This is best seen from the following example.

Let us consider two deterministic (max,+)-automata corresponding to formal power series $s_1 = 1\sigma \oplus 2\sigma'$ and $s_2 = 1\sigma' \oplus 2\sigma$, where both events $\sigma', \sigma \in A$ are unobservable. We have a weak simulation (even weak bisimulation), because in $\mu(\tau)^*$ we take the maximum of $\mu(\sigma')$ and $\mu(\sigma)$, hence

$$\mu_1(\tau)^* = \mu_2(\tau)^* = \begin{bmatrix} 0 & 2 \\ \varepsilon & 0 \end{bmatrix}.$$

However, it is not true that $s_1 \leq s_2$, because $s_1(\sigma') = 2 > 1 = s_2(\sigma')$. Therefore, we present a stronger version of weak simulation, called label dependent weak bisimulation where $\mu(\tau)^*$ is replaced by $\mu(w)$ for all $w \in A_{uo}^*$. Note that since we exclude unobservable loops, the number of such unobservable words is finite. Moreover, we will use this version of weak simulation only as a sufficient condition for comparing behaviors of (max,+)-automata and in many situations there will be natural candidates for events that will be made unobservable for this purpose.

Definition 12. (label dependent weak simulation)
$B \in \{\varepsilon, e\}^{|Q| \times |Q'|}$ is a label dependent weak simulation between (max,+)-automata $G = (Q, A, \alpha, \mu, \beta)$ and $G' = (Q', A', \alpha', \mu', \beta')$ if and only if for all $w \in A_{uo}^*$ the following inequalities hold true:

- $\alpha \otimes \mu(w) \leq \alpha' \otimes \mu'(w) \otimes B$,
- $B \otimes \mu(a) \otimes \mu(w) \leq \mu'(a) \otimes \mu'(w) \otimes B$ and
- $B \otimes \beta \leq \beta'$.

Theorem 13. (Main result) Let there exist a label dependent weak simulation between (max,+)-automata G and G'. Then $l(G) \leq l(G')$.

5. EXAMPLE

Our approach is illustrated via the following example. Consider two (max,+)-automata given in Fig. 3. These (max,+)-automata correspond to timed Petri net models of manufacturing systems. The timed Petri net corresponds to T-time Petri net (with timing associated to transitions) of figure 4, where the duration

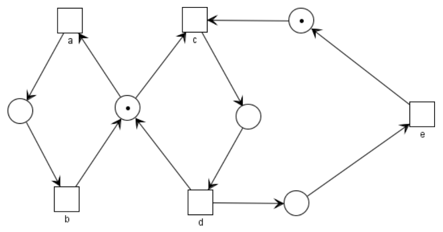

Fig. 4. Petri net $\mathscr{G}$

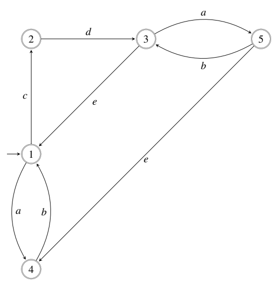

Fig. 5. Language (marking automaton $\mathscr{M}$) of $\mathscr{G}$

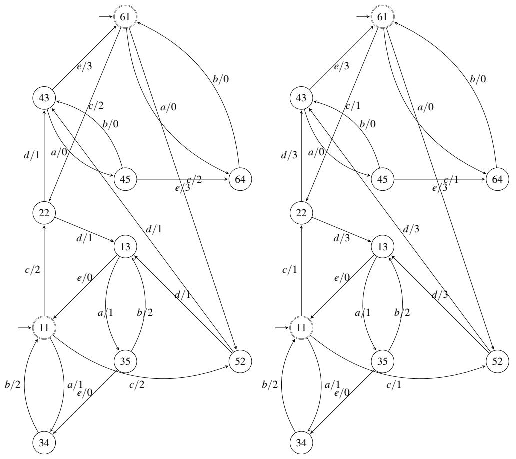

Fig. 6. $\mathscr{A}_1 = G_1 \| \mathscr{M}$ (left) and $\mathscr{A}_2 = G_2 \| \mathscr{M}$ (right)

of transitions a, b, c, d, c are $1, 2, 2, 1, 3$ and $1, 2, 1, 3, 3$, respectively. This Petri net can be viewed as a synchronization of two timed state graphs. The corresponding (max,+)-automata given in Fig. 3 are then obtained using the synchronous product technique described in Lahaye et al. (2015). These (max,+)-automata are nondeterministic, because they have two initial states and moreover there are two different transitions labeled by c at state 1 and there are also two different transitions labeled by d at states 2 and 5. Unfortunately, it is known from Lahaye et al. (2015) that the (max,+)-automata have languages different (larger than) from the languages of the original Petri nets. Therefore, in order to have a meaningful example we need to compute the synchronous product with the language of these Petri nets, which is given by the marking automaton $\mathscr{M}$ on figure 5.

The resulting (max,+)-automata $\mathscr{A}_1 = G_1 \| \mathscr{M}$ and $\mathscr{A}_2 = G_2 \| \mathscr{M}$ that have the same behaviour as the original Petri nets $\mathscr{G}_1$ and $\mathscr{G}_2$ are displayed on figure 6. The final states are displayed with double circles. We see that there are two states which are both initial and final: $(1, 1)$ and $(6, 1)$ and no other states are neither initial nor final. We repeat the convention that initial and final delays are not added if equal to $e = 0$.

It is easy to see that there is no simulation relation between $\mathscr{A}_1$ and $\mathscr{A}_2$. Indeed, the problem is that the transition labeled by c has value 2 in $\mathscr{A}_1$, while it has value 1 in $\mathscr{A}_2$. Intuitively, this is compensated by smaller duration of the event d in $\mathscr{A}_1$, namely 1, while the duration of d is $\mathscr{A}_2$ is equal to 3. Unfortunately, there does not exist a formal algorithm to check that $l(\mathscr{A}_1) \le l(\mathscr{A}_2)$, because these automata are nondeterministic due to non unique initial states and nondeterministic transitions labeled by c and d. However, since not all states are marked (final), the total duration of the string cd from the initial state 11 is equal to 3 in $\mathscr{A}_1$, while it equals 4 in $\mathscr{A}_2$, hence the required inequality is satisfied for this marked string. Similarly the inequality holds for larger strings. This can be made formal by using weak bisimulation (considering event c as unobservable) and Theorem 13 to conclude the claimed inequality. We compute projected automata $P(\mathscr{A}_1)$ and $P(\mathscr{A}_2)$ according to definition 6, where $\mu(\tau)^*$ is replaced by $\mu(c)^*$. and verify that initial and finite delays are equal and all the morphism matrices are equal as well, except for $\mu_{P,1}(d) = \mu_1(c)^* \otimes \mu_1(d)$ and $\mu_{P,2}(d) = \mu_2(c)^* \otimes \mu_2(d)$, because $\mu_{P,1}(d) \le \mu_{P,2}(d)$. The conclusion is there exists a simulation relation (namely the diagonal relation given by the identity matrix I) from $P(\mathscr{A}_1)$ to $P(\mathscr{A}_2)$.

6. CONCLUSION

In this paper simulation like equivalences between (max,+)-automata have been studied. We have shown how a variant of weak simulation can be used as a partial solution to the undecidability of inequalities between (max,+)-formal power series.

We plan to investigate modular and hierarchical control of (max,+) automata based on the concepts presented in this paper. For specifications that do not share the modular structure with the system it will be interesting to generalize the theory of decomposability and conditional decomposability to (max,+)-formal power series.

REFERENCES

Baccelli, F., Cohen, G., Olsder, G., and Quadrat, J. (1992). *Synchronization and Linearity*. John Wiley and Sons.

Buchholz, P. and Kemper, P. (2003). Weak bisimulation for (max/+) automata and related models. *J. Autom. Lang. Comb.*, 8(2), 187–218.

Butkovic, P. (2010). *Max-linear systems : theory and algorithms*. Springer.

Cassandras, C.G. and Lafortune, S. (2008). *Introduction to discrete event systems*. Springer, second edition.

Damljanovic, N., Ciric, M., and Ignjatovic, J. (2014). Bisimulations for weighted automata over an additively idempotent semiring. *Theor. Comput. Sci.*, 534, 86–100.

Gaubert, S. (1992). *Théorie des systèmes linéaires dans les dioïdes*.

Gaubert, S. (1995). Performance Evaluation of (max,+) Automata. *IEEE TAC*, 40(12), 2014–2025.

Gaubert, S. and Mairesse, J. (1999). Modeling and analysis of timed petri nets using heaps of pieces. *IEEE Transaction on Automatic Control*, 44(4), 683–698.

Krob, D. (1994). The equality problem for rational series with multiplicities in the tropical semiring is undecidable. *IJAC*, 4(3), 405–426.

Lahaye, S., Komenda, J., and Boimond, J. (2015). Compositions of (max, +) automata. *Discrete Event Dynamic Systems*, 25(1-2), 323–344.