Compromise Approach for Predictive Control of Timed Event Graphs with Specifications Defined by P-time Event Graphs

Philippe Declerck

I. ABSTRACT

In this paper, the aim is to make the predictive control of a plant described by a Timed Event Graph which follows the specifications defined by a P-time Event Graph. We propose a compromise approach between the ideal optimality of the solution and the on-line application of the computed solution when the relevant optimal control cannot be applied for a given computer. The technique is based on a reduction of the number of iterations of the fixed point algorithm such that the computed control remains causal. The analysis of the partial satisfaction of the specifications at each iteration of the algorithm defined in the (max, +) algebra shows that a subset of constraints is guaranteed by the control computed at each iteration while another one is possibly satisfied.

P. Declerck is with LISA/LARIS EA7315, University of Angers, 62 avenue Notre-Dame du Lac, 49000 Angers, France. Tel. +33 244 687568 - Fax. +33 244 687561. e-mail. philippe.declerck@univ-angers.fr

keywords: Timed Event Graphs, P-time Petri nets, predictive control, causality, fixed point, consistency.

II. Introduction

A classical problem is the predictive control of a Timed Event Graph where some events are stated as controllable, meaning that the corresponding transitions (input) may be delayed from firing until some arbitrary time provided by a supervisor. In this paper, the specifications are defined by a P-time Event Graph which describes the desired behavior of the interconnections of all internal transitions. Using the (max, +) algebra, we wish to determine an input in order to obtain the desired behavior defined by the specifications.

Since last two decades, this subject or its variant has already been considered in many papers giving advanced results. However, the consideration of the causality phenomenon in the control procedure still poses a problem. Indeed, Predictive Control approaches calculate a future control which must be applied on-line: The application of the control must be made after the past dates of the initial state which is the known initial starting point of the problem. The application of the first calculated control must be made after an availability time, which is the sum of the last past date of the known state and the computer time (The execution time of the on-line procedure) [10]. In this paper, this constraint is named 'causality constraint' and the control is said to be 'causal' when the causality constraint is satisfied. Leading to unsolved control problems, this difficulty arises if we consider too imperative desired output and/or the practical control of large scale systems such as transportation systems, real-time systems,...

Therefore, the aim of this paper is to propose an approach for predictive control by improving the satisfaction of the causality constraint. A possible solution is to reduce the availability time by having a lower computer time. A technique given in [9] is based on a restriction of the state space leading to a convergence of the pseudo-polynomial algorithm at the first iteration which provides a strongly polynomial algorithm. However, the considered problem must satisfy a space condition. In this paper, we propose a compromise approach defined as follows. We can reduce the CPU time of the fixed point algorithm by limiting the number of iterations such that the computed control remains causal. In other words, the algorithm is stopped at a given iteration without waiting for the convergence and the occurrence of a causality problem. The causal control generated by this unusual approach is greater than the optimal control as the convergence based on a minimization is not waited. Only a subset of the constraints is satisfied but the solution meets the deadlines: These two aspects (which respectively correspond to points b) and a) defined in Section IV) define the compromise considered in this paper aiming at satisfying the causality constraint. Note that another possible compromise is to take another point of view: All the constraints are satisfied but the desired output is not satisfied as the objective of this possible technique (beyond the scope of this paper) is to postpone the desired output (and increase the control) such that the causality constraint is satisfied. As in the previous approach, the causal solution is not optimal with respect to the ideal control problem as a constraint (point a) defined below) is not satisfied.

This approach can be sufficient if the important constraints are guaranteed by a preliminary analysis. Clearly, the satisfaction of safety regulations for a grade crossing is obligatory contrary to the satisfaction of the following non-crucial constraint taken in the food industry: In good bakery practice, the dough stays in the fermentation room from three to five hours, the time depending on room temperature and flour or gluten quality; if these times are too short or too long, the quality of the product will slightly be damaged (bad inner structure and grain in the finished loaf). Therefore, the resolution of this problem implies that we focus on the partial validity of the different constraints at each iteration which allows the application of a control to the process without waiting the complete convergence of the algorithm. This objective needs the generalization of [9] by introducing new theoretical results based on residuation theory and the fundamental Theorem 3.1.1 in [5] analyzed by different authors (K. Zimmermann (1976), P. Butkovic, H. Goto and S. Masuda [11]). At the best of our knowledge, the analysis of the partial consistency of the constraints is an original topic in the max-plus algebra.

Let us briefly put our contribution into the general context of control and give some related works. A first class of approaches works on the behavior of the state model by handling its characteristics with frequently the building of closed-loop blocs connected to the inputs of the system [15] [14] [1] [13]. A current support is the transfer function on a dioid of series with the assumption of canonical initial condition. An objective is the development of controllers in order to keep trajectories inside a space deduced from a given specification. Note that approaches based on a feedback defined by a Petri net present a causality condition as the duration and the initial marking of each added place must be non-negative. Traditionally applied to linear discrete-time models, the framework of Predictive Control can also be adapted to discrete event systems. Considering the state equations, an usual step of the approach developed in [18] [17] is to transform the (max, +) problem in a linear programming problem in the conventional algebra which allows the application of classical algorithms. The principal advantage of this technique is the consideration of general models. However, as shown above, model predictive control is an on-line approach which needs *efficient algorithms*: The application of *generic* algorithms of linear programming (see the algorithms cited in [18]) leads to the limitation of the size of the considered systems. When stochastic max-plus systems are considered, an approach based on a technique called variability expansion can reduce the computational complexity of this optimization problem [3].

The crucial point is that the structures of the matrices describing the models in standard algebra present *specific characteristics* [12]. So, an objective is to make the most of these specific structures of the systems and to deduce an approach having a reduced CPU time. An answer is to use the (max, +) algebra which allows the application of efficient algorithms of path theory which are often *strongly* polynomial [8] [9] [10]. As discussed in [7], the algorithms specific to path algebra, surpass the best generic algorithms of linear programming when they are applied to the relevant specific problems. A second advantage is the possibility to write formal expressions contrary to linear programming which focus on the numerical results. So, we can define an off-line preparation which can avoid the repetition of the same calculations at each step of the procedure and reduce the CPU time. In fact, the calculation of some matrices can be made once as it does not depend on the on-line aspect, but only depends on the models and the size of the given horizon (Chapter 5 in [8]).

In this paper, we assume that the control problem has an admissible solution satisfying the different criteria of the problem (defined below in Assumption 1) but can present a non-causal control. The consistency of the models and the problem (Chapters 2 and 4 in [8]) is beyond the scope of this paper. The value of the maximum number of iterations such that the control remains causal is assumed to be known and is non-null (Assumption 2): It has been approximated in a preparatory phase of tests based on off-line simulations close to the real conditions of the predictive control. We consider that each transition is observable: The event date of each transition firing is assumed to be available. No hypothesis is made on the structure of the Event Graphs which does not need to be strongly connected. The initial marking should only satisfy the classical liveness condition and the usual hypothesis of First In First Out (FIFO) places is used. The presentation of the model of the P-time Event Graph is omitted and can be found in Chapter 2 in [8]. The principle of the model predictive control can be found in [18].

The broad outline of the paper is as follows. Mathematical tools are remembered in Section III. We then describe the control problem and the relevant fixed point algorithm which calculates the control and the state trajectories in Section IV [9]. The main contribution of this paper is the analysis of the partial consistency of the constraints in Section V where a variation of the classical procedure of predictive control is proposed. This paper is an improved version of [10] containing new material, such as: A new section on causality constraint; an original Corollary analyzing the partial consistency of $Ax = b$; a generalized Theorem 5 considering the case where some equalities are never satisfied and a new Theorem 6 based on on-line data; the proofs of the different propositions; the pedagogical Examples 1, 2 and 4; Illustrating the main points, the pedagogical example 3 is a more complex version of the example given in [9].

III. Preliminary remarks

A monoid is a pair $(\$, \oplus)$ where $\$$ is a nonempty set, the operation $\oplus$ is associative and presents a neutral element ε. A semiring $\$$ is a triple $(\$, \oplus, \otimes)$ where $(\$, \oplus)$ and $(\$, \otimes)$ are monoids, $\oplus$ is commutative, $\otimes$ is distributive in relation to $\oplus$ and the zero element ε of $\oplus$ is the absorbing element of $\otimes$ ($\varepsilon \otimes a = a \otimes \varepsilon = \varepsilon$). A dioid $\mathcal{D}$ is an idempotent semi-ring (the operation $\oplus$ is idempotent, that is $a \oplus a = a$). The set $\mathbb{R} \cup \{-\infty\}$, provided with the maximum operation denoted by $\oplus$ and the addition denoted by $\otimes$ is an example of dioid denoted by $\mathbb{R}_{\max} = (\mathbb{R} \cup \{-\infty\}, \oplus, \otimes)$: So, $x \oplus y = max(x, y)$ and $x \otimes y = x + y$. The neutral elements of $\oplus$ and $\otimes$ are represented by $\varepsilon = -\infty$ and $e = 0$, respectively. The absorbing element of $\otimes$ is ε. The minimum operation is denoted by $\wedge$. The partial order denoted by $\leqslant$ is defined in $\mathbb{R}^n$ as follows: $x \leqslant y \iff x \oplus y = y \iff x \wedge y = x \iff x_i \leqslant y_i$, for i from 1 to n. The notation

$x < y$ means that $x \leqslant y$ and $x \neq y$. A dioid $\mathcal{D}$ is complete if it is closed for infinite sums, and the distributivity of the multiplication with respect to the addition applies to infinite sums. $(\forall c \in \mathcal{D}\ ) \ (\forall\ A \subseteq \mathcal{D})\ c \otimes (\bigoplus_{x \in A} x) = \bigoplus_{x \in A} c \otimes x$. For example, $\overline{\mathbb{R}}_{\max} = (\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}, \oplus, \otimes)$ is complete. The operations $\oplus$ and $\otimes$ are extended to matrices as follows:

If $\alpha \in \mathcal{D}$ and if P, $Q \in \mathcal{D}^{m\mathrm{x}n}$ then

$$(\alpha \otimes P)_{i,j} = \alpha \otimes P_{i,j} \text{ and } (P \oplus Q)_{i,j} = P_{i,j} \oplus Q_{i,j} \text{ for all } i, j;$$

If $P \in \mathcal{D}^{m\mathrm{x}p}$ and $Q \in \mathcal{D}^{p\mathrm{x}n}$ then

$$(P \otimes Q)_{i,j} = \bigoplus_{k=1}^{p} P_{i,k} \otimes Q_{k,j} \text{ for all } i, j.$$

The identity matrix is denoted by $\mathcal{I}$: $\mathcal{I}_{i,j} = e$ if $i = j$ and $\mathcal{I}_{i,j} = \varepsilon$ if $i \neq j$. The zero matrix is only composed of the entries ε and is denoted by ε. The dimensions of the matrices $\mathcal{I}$ and ε can easily be deduced from the context. The set of nxn matrices with entries in the complete dioid $\mathcal{D}$ including the two operations $\oplus$ and $\otimes$ is a complete dioid, which is denoted by $\mathcal{D}^{n\mathrm{x}n}$. We can deal with non-square matrices if we complete them with rows or columns provided the entries equal ε.

A mapping f is monotone or isotone if $x \leq y$ implies $f(x) \leq f(y)$. Let f: *E*→ *F* be an isotone mapping, where (*E*, $\leq$) and (*F*,$\leq$) are ordered sets. The mapping f is said to be residuated if for all $y \in \mathcal{D}$, the least upper bound of subset $\{x \in \mathcal{D} \mid f(x) \leq y\}$ exists and lies in this subset. The isotone mapping $x \in (\overline{\mathbb{R}}_{\max})^n \mapsto A \otimes x$, defined over $\overline{\mathbb{R}}_{\max}$ is residuated (see [2]) and the left $\otimes-$residuation of B by A is denoted by

$$A \backslash B = \max\{x \in (\overline{\mathbb{R}}_{\max})^n \text{ such that } A \otimes x \leqslant B\}.$$

This greatest element (also called maximum) of the last set is also denoted by x^+. Moreover,

$$(x^+)_i = \bigwedge_{j=1}^{m} A_{ji} \backslash b_j$$

where A is an $m \times n$ matrix and residuation $\backslash$ has priority over minimization $\wedge$.

Example 1. Let $A = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix}$ and $b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$ where all the entries $a_{i,j}$,b_i are defined over $\mathbb{R}$. System $A \otimes x \leqslant B$ is equivalent to $\max(a_{i,1} + x_1, a_{i,2} + x_2) \leqslant b_i$ $(\forall i \in \{1, 2\})$ in the standard algebra, or equivalently, the set of inequalities $a_{i,j} + x_j \leqslant b_i$ $(\forall i, j \in \{1, 2\})$. So, we obtain $x_j \leqslant b_i - a_{i,j}$ $(\forall i, j \in \{1, 2\})$ or the developed form

$$\begin{cases} x_1 \leqslant \min(b_1 - a_{1,1}, b_2 - a_{2,1}) \\ x_2 \leqslant \min(b_1 - a_{1,2}, b_2 - a_{2,2}) \end{cases}.$$

Therefore, we directly obtain the greatest solution

$$(x^+)_i = \min(b_1 - a_{1,i}, b_2 - a_{2,i})(\forall i \in \{1, 2\})$$

which considers finite entries. This expression is generalized by formula $(x^+)_i = \bigwedge_{j=1}^{m} A_{ji} \backslash b_j$ containing the residuated form $A_{ji} \backslash b_j$ which avoids the ambiguities when infinite entries are considered. ■

The notation $card(X)$ stands for the cardinality of the set X. The Kleene star is defined by:

$$A^* = \bigoplus_{i=0}^{+\infty} A^{\otimes i}$$

where $A^{\otimes i}$ represents the (max, +) product of i matrices A: $A^{\otimes i} = A \otimes A \otimes ... \otimes A$. Denoted by Im$A$, the image of A is $\{A \otimes x \mid x \in \mathbb{R}^n_{\max}\}$ which is the set of all linear combinations of columns of A in the max-plus meaning. A matrix A is called row (column) $\mathbb{R}-$astic when it has no null row (column): $A_{i,.} \neq \varepsilon$ for any row i ($A_{.,j.} \neq \varepsilon$ for any column j). A matrix A is called doubly $\mathbb{R}-$astic if it is both row and column $\mathbb{R}-$astic.

Let us consider the initial control problem of this paper defined over $\mathbb{R}_{\max}$. Below, the variable $x_i(k)$ is the date of the k^{th} firing of the transition x_i and n is the dimension of $x(k)$.

IV. Control Problem

In this paper, we consider a classical predictive control based on the infinite repetition of a control step on a finite sliding horizon. Generalizing the Backward Approach (The basic problem is explained in Chapter 5.6 'Backward Equations' in [2]), this control step is the resolution of the following control problem where the objective is the determination of the greatest control u (with respect to the componentwise order) on an arbitrary horizon $[k_s+1, k_f]$ such that its application to the Timed Event Graph defined by

$$\begin{cases} x(k+1) = A \otimes x(k) \oplus B \otimes u(k+1) \\ y(k) = C \otimes x(k) \end{cases} \tag{1}$$

for $k \geq k_s$, satisfies the following conditions:

a) $y \leq \underline{z}$ knowing the trajectory of the desired output $\underline{z}$;

b) The state trajectory follows the model of the P-time Event Graph algebraically defined by (and described below)

$$\begin{pmatrix} x(k) \\ x(k+1) \end{pmatrix} \geq \begin{pmatrix} A^{=} & A^{+} \\ A^{-} & A^{=} \end{pmatrix} \otimes \begin{pmatrix} x(k) \\ x(k+1) \end{pmatrix}; \tag{2}$$

c) The initial value of the state trajectory $x(k)$ for $k \geq k_s$ is finite and is a known vector denoted by $\underline{x}(k_s)$. This " non-canonical " initial condition is the result of a past evolution of the process.

Underlined symbols like $\underline{x}(k_s)$ correspond to known data of the problem and, state $x(k)$ and output $y(k)$ are estimated in the following resolutions.

The system (2) can always be obtained and corresponds to a P-time Event Graph where the initial marking of each place is equal to at most one. When we consider the places having a unitary (respectively, null) initial marking, the lower bound T_1^- of the temporization of place

p_1 linking its input transition x_j to its output transition x_i generates the entry $A^-_{i,j} = T^-_1 \geq 0$ (respectively, $A^=_{i,j} = T^-_1 \geq 0$) as we have $x_i(k+1) \geq x_j(k) + T^-_1$ (respectively, $x_i(k) \geq x_j(k) + T^-_1$). Similarly, the upper bound T^+_1 of the temporization of this place generates the entry $A^+_{j,i} = -T^+_1 \leq 0$ (respectively, $A^=_{i,j} = -T^+_1 \leq 0$) as we have $x_j(k) \geq x_i(k+1) - T^+_1$ (respectively, $x_j(k) \geq x_i(k) - T^+_1$).

Example 2.

Let us consider an elementary P-time Event Graph having a unique place p_1 associated with a time interval $[T^-_1, T^+_1]$ and connecting an input transition denoted x_1 to an output transition denoted x_2. If the initial marking is null, we have $A^- = 0$, $A^+ = 0$ and $A^= = \begin{pmatrix} \varepsilon & -T^+_1 \\ T^-_1 & \varepsilon \end{pmatrix}$.

If now the initial marking of p_1 is unitary, we obtain $A^= = 0$, $A^- = \begin{pmatrix} \varepsilon & \varepsilon \\ T^-_1 & \varepsilon \end{pmatrix}$ and $A^+ = \begin{pmatrix} \varepsilon & -T^+_1 \\ \varepsilon & \varepsilon \end{pmatrix}$. Other examples can be found in [9] and, Chapters 2 and 3 of [8]. ■

A. *Relations on horizon* $[k_s, k_f]$

The relations of the Timed Event Graph can be rewritten under the following classical form on horizon $[k_s, k_f]$.

$$X = \Omega_h \otimes x(k_s) \oplus \Psi_h \otimes U \tag{3}$$

where $h = k_f - k_s$, $X = \begin{pmatrix} x(k_s+1)^t & x(k_s+2)^t & \cdots & x(k_f-1)^t & x(k_f)^t \end{pmatrix}^t$ (t: transposed), $U = \begin{pmatrix} u(k_s+1)^t & u(k_s+2)^t & \cdots & u(k_f-1)^t & u(k_f)^t \end{pmatrix}^t$, Ω_h is a column of h blocks $(\Omega_h)_i = A^{\otimes i}$ for $i = 1$ to h and Ψ_h is a h x h matrix of blocks $(\Psi_h)_{i,j}$ for $i, j \in \{1, 2, \ldots, h\}$ where $(\Psi_h)_{i,j} = A^{\otimes(i-j)} \otimes B$ for $i \geq j$ and ε otherwise.

Below we consider the additional constraints (2) for $k \geq k_s$ and an autonomous Timed Event Graph defined by the inequality $x(k) \geq A \otimes x(k-1)$ which is the relaxation of the earliest firing rule, starting from $x(k_s) = \underline{x}(k_s)$.

$$\begin{cases} \begin{pmatrix} x(k_s) \\ X \end{pmatrix} \geq D_h \otimes \begin{pmatrix} x(k_s) \\ X \end{pmatrix} \text{ and} & (4) \\ x(k_s) = \underline{x}(k_s) & (5) \end{cases}$$

where D_h is a tridiagonal matrix of blocks $(D_h)_{i,j}$ for $i, j \in \{1, 2, \ldots, h+1\}$: This square matrix is composed of a main diagonal ($(D_h)_{i,i} = A^=$ for $i \in \{1, \ldots, h+1\}$), an upper diagonal ($(D_h)_{i,i+1} = A^+$ for $i \in \{1, \ldots, h\}$), a lower diagonal ($(D_h)_{j+1,j} = A \oplus A^-$ for $j \in \{1, \ldots, h\}$); all other blocks are zero matrices (square submatrix ε). The matrix D_h is a $n.(h+1)$ x $n.(h+1)$ matrix.

B. Fixed point algorithm

We introduce the following extended state vector $\overline{x} = \begin{pmatrix} (x(k_s))^t & (X)^t \end{pmatrix}^t$ which expresses the complete state trajectory. Let $(\overline{x})^+$ be the greatest estimate of the state trajectory and $F = \begin{pmatrix} \underline{x}(k_s)^t & (C \backslash \underline{z}(k_s+1))^t & \cdots & (C \backslash \underline{z}(k_f))^t \end{pmatrix}^t$. The following theorem shows that the problem can be rewritten in a form of a fixed point inequality which is solved by Algorithm 1 below.

***Theorem** 1:* [9] The greatest state and control trajectory of the control problem is the greatest solution of the following fixed point inequality system

$$\begin{cases} \overline{x} \leq D_h \backslash \overline{x} \wedge F \\ U \leq \Psi_h \backslash X \\ X \leq \Omega_h \otimes x(k_s) \oplus \Psi_h \otimes U \end{cases} \tag{6}$$

with condition $\underline{x}(k_s) \leq x^+(k_s)$. ■

Therefore, the problem is rewritten under a general, fixed-point formulation $x \leq f(x)$ which allows the control problem to be resolved (Chapter 4 in [8], [9]). Function f is a (min, max, +) function which can be defined by the following grammar: $f = b, x_1, x_2, \ldots, x_n \mid f \otimes a \mid f \wedge f \mid f \oplus f$ where a, b are arbitrary real numbers ($a, b \in \mathbb{R}$). The existence of the greatest solution on complete lattices can be proven by using the famous fixed point theorem of Knaster-Tarski which can be deduced from the fixed point theorem of Amann (1977) whose proof uses the fixed point theorem of Bourbaki (1940) and Kneser (1950) [21]. The conditions of the Knaster-Tarski theorem are satisfied: The general form of the problem is such that $x \leq f(x)$ where f is an isotone function defined on a complete lattice $\overline{\mathbb{R}}_{max}$=($\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}, \leq$) and where x corresponds to $(x(k_s)^t, X^t, U^t)^t$.

Algorithm 1 below is a fixed point algorithm calculating the greatest state and control. Since it follows the algorithm of McMillan and Dill [16] [20], Algorithm 1 is also pseudo-polynomial. Remember that in this classical algorithm, the greatest solution to $x \leq f(x)$ is given by the iterations of $x_{\langle i \rangle} \leftarrow x_{\langle i-1 \rangle} \wedge f(x_{\langle i-1 \rangle})$ if the finite starting point $x_{\langle 0 \rangle}$ is greater than the final solution. Here, number $\langle i \rangle$ represents the number of iterations and not the number of components of vector x. Also note that Algorithm 1 is close to the Alternating Method given in [6] which solves the equality $A \otimes x = B \otimes y$. Starting from $x_{\langle 0 \rangle} = F$, the trajectory $\overline{x}$ is minimized in each iteration of the following algorithm where each iteration $\langle i \rangle$ with $i > 0$ is composed of the three steps 1, 2 and 3. Algorithm 1 proposes an initial state $x(k_s)$ satisfying $x(k_s) \leq \underline{x}(k_s)$ and generates a trajectory starting from $x(k_s)$ given by the expression $\Omega_h \otimes x(k_s) \oplus \Psi_h \otimes U$. The control problem is solved under the condition $\underline{x}(k_s) \leq x(k_s)$ which implies $x(k_s) = \underline{x}(k_s)$ (Point c)).

Algorithm 1 [9]

Step 0 (initialization): $\langle i\rangle \leftarrow \langle 0\rangle$; $(\overline{x})^2 \leftarrow F$

Repeat

- $\langle i\rangle \leftarrow \langle i+1\rangle$ (numbering of the iteration)
- Step 1: $(\overline{x})^1 \leftarrow D_h^* \backslash (\overline{x})^2$
- Step 2: $U \leftarrow \Psi_h \backslash X^1$
- Step 3: $(\overline{x})^2 \leftarrow (\overline{x})^1 \wedge \begin{pmatrix} +\infty \\ \Omega_h \otimes x^1(k_s) \oplus \Psi_h \otimes U \end{pmatrix}$

until $X^1 = X^2$. ■

Vectors $(\overline{x})^1 = \begin{pmatrix} (x^1(k_s))^t & (X^1)^t \end{pmatrix}^t$ and $(\overline{x})^2 = \begin{pmatrix} (x^2(k_s))^t & (X^2)^t \end{pmatrix}^t$ present the same dimensions as $\overline{x} = \begin{pmatrix} (x(k_s))^t & (X)^t \end{pmatrix}^t$ and correspond to useful intermediate values leading to the obtention of the optimal vector $\overline{x}$. Vectors $(\overline{x})^1$ and $(\overline{x})^2$ are respectively computed in Steps 1 and 3 which are respectively backward and forward calculations. Step 1 is deduced from the resolution of $\overline{x} \leq D_h \backslash \overline{x} \wedge (\overline{x})^2$ and the application of Theorem 4.73 in ([2]). The obtained solution $(\overline{x})^1$naturally satisfies $(\overline{x})^1 \leq D_h \backslash (\overline{x})^1$ which corresponds to (4). The rest of the algorithm checks that this calculated solution, also satisfies $X^1 = \Omega_h \otimes x^1(k_s) \oplus \Psi_h \otimes U$ with $U = \Psi_h \backslash X^1$.

C. Example 3

This example is a more complex version of the example given in [9]. In this paper, we will show that the new matrix B leads to a different convergence: It needs two iterations while the example in [9] converges in three iterations (Or one iteration when a specific subspace is considered).

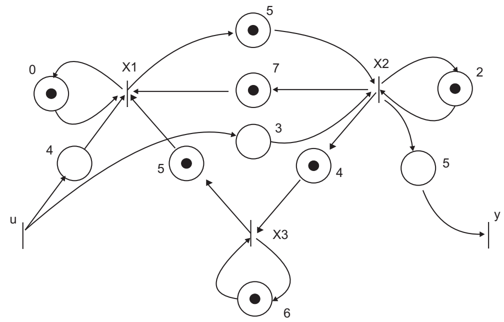

Fig. 1. Plant in Example 3: Timed Event Graph (example [9] modified)

Timed Event Graph (Fig. 1): $A = \begin{pmatrix} 0 & 7 & 5 \\ 5 & 2 & \varepsilon \\ \varepsilon & 4 & 6 \end{pmatrix}$, $B = \begin{pmatrix} 4 \\ 3 \\ \varepsilon \end{pmatrix}$ and $C = \begin{pmatrix} \varepsilon & 5 & \varepsilon \end{pmatrix}$.

P-time Event Graph (Fig. 2): $A^{=} = \begin{pmatrix} \varepsilon & \varepsilon & -11 \\ \varepsilon & \varepsilon & -11 \\ 1 & 1 & \varepsilon \end{pmatrix}$, $A^{-} = \begin{pmatrix} \varepsilon & 0 & 1 \\ 3 & \varepsilon & 4 \\ 1 & 2 & \varepsilon \end{pmatrix}$ and $A^{+} =$

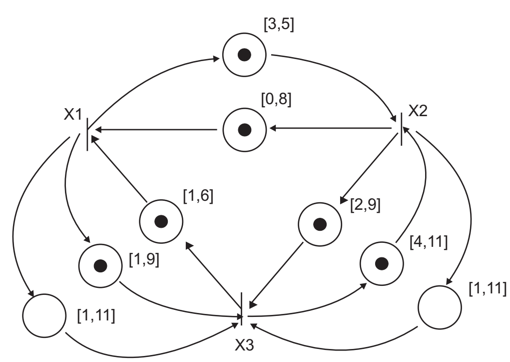

Fig. 2. Specifications in Example 3: P-Time Event Graph [9]

$$\begin{pmatrix} \varepsilon & -5 & -9 \\ -8 & \varepsilon & -9 \\ -6 & -11 & \varepsilon \end{pmatrix}.$$

Let $h = 3$. The desired output $z(k)$ and the initial condition $\underline{x}(k_s)$ are as follows: $z(k) = 25,\ 25,\ 28$ for $k_s + 1 \leq k \leq k_s + 3$ and $\underline{x}(k_s) = \begin{pmatrix} 2 & 0 & 3 \end{pmatrix}^t$. Needing two iterations, Algorithm 1 gives the following results: $u(k) = 4,\ 10,\ 16$ for $k_s + 1 \leq k \leq k_s + 3$, $x(k_s) = \begin{pmatrix} 2 & 0 & 3 \end{pmatrix}^t$, $x(k_s + 1) = \begin{pmatrix} 8 & 7 & 9 \end{pmatrix}^t$, $x(k_s + 2) = \begin{pmatrix} 14 & 13 & 15 \end{pmatrix}^t$, $x(k_s + 3) = \begin{pmatrix} 20 & 19 & 21 \end{pmatrix}^t$ and $y(k) = 12,\ 18,\ 24$ for $k_s + 1 \leq k \leq k_s + 3$. Matrices $(D_h)^*$ and Ψ_h are given below.

$$(\Psi_h)^t = \begin{pmatrix} 4 & 3 & \varepsilon & 10 & 9 & 7 & 16 & 15 & 13 \\ \varepsilon & \varepsilon & \varepsilon & 4 & 3 & \varepsilon & 10 & 9 & 7 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 4 & 3 & \varepsilon \end{pmatrix}$$

Note that matrix Ψ_h is different from matrix Ψ_h given in [9]. The possibility that the third row of Ψ_h is null will be examined in the following sections.

$(D_h)^* =$

$$\begin{pmatrix}
0 & 0 & -1 & -7 & -5 & -8 & -13 & -12 & -14 & -20 & -18 & -21 \\
-2 & 0 & -3 & -8 & -7 & -9 & -15 & -13 & -16 & -21 & -20 & -22 \\
1 & 1 & 0 & -6 & -4 & -7 & -12 & -11 & -13 & -19 & -17 & -20 \\
6 & 7 & 5 & 0 & 1 & -1 & -7 & -5 & -8 & -13 & -12 & -14 \\
5 & 5 & 4 & -2 & 0 & -3 & -8 & -7 & -9 & -15 & -13 & -16 \\
7 & 8 & 6 & 1 & 2 & +0 & -6 & -4 & -7 & -12 & -11 & -13 \\
12 & 13 & 11 & 6 & 7 & 5 & 0 & 1 & -1 & -7 & -5 & -8 \\
11 & 12 & 10 & 5 & 6 & 4 & -2 & 0 & -3 & -8 & -7 & -9 \\
13 & 14 & 12 & 7 & 8 & 6 & 1 & 2 & 0 & -6 & -4 & -7 \\
18 & 19 & 17 & 12 & 13 & 11 & 6 & 7 & 5 & 0 & 1 & -2 \\
17 & 18 & 16 & 11 & 12 & 10 & 5 & 6 & 4 & -2 & 0 & -3 \\
19 & 20 & 18 & 13 & 14 & 12 & 7 & 8 & 6 & 1 & 2 & 0
\end{pmatrix}$$

. ■

As we focus on the satisfaction of the constraints (conditions b)) during the convergence, we assume that Algorithm 1 gives a solution to the initial problem. The following assumption clarifies the context of this paper.

Assumption 1: Algorithm 1 converges to a finite solution satisfying conditions a), b) and c) at the end of its complete execution. ■

This assumption implies that the models are consistent and that the desired output z is taken sufficiently large with respect to the initial condition $\underline{x}(k_s)$. Unfortunately, this obtained solution can be non-causal. The causality constraint is now described and defined algebraically.

D. Causality constraint

Following the classical procedure of the Predictive Control, computed control $u(k_s+1)$ must on-line be applied after the dates of $\underline{x}(k_s)$ which are the data of the problem. Each component $(u(k_s+1))_i$ must be greater than the date of its possible application which is the sum (in the standard algebra) of the maximum of the components of $\underline{x}(k_s)$ and the computer time T_{comp} which is the time taken from the start of the algorithm until the end as measured by an ordinary clock (Let us remember that the CPU time is the amount of time for which a central processing unit was used for processing instructions of a computer program contrary to the computer time T_{comp} which includes the CPU time and also the variable time spent by the computer in executing Kernel routines.). The relevant control satisfying this causal constraint is called "causal" for the given control problem. More formally, we have

$$\bigoplus_{i\in\{1,\ldots,n\}} \underline{x}_i(k_s)\otimes T_{comp} \leq \bigwedge_{i\in\{1,\ldots,card(u)\}} u_i(k_s+1) \tag{7}$$

where $\bigoplus_{i\in\{1,\ldots,n\}} \underline{x}_i(k_s)\otimes T_{comp}$ is the availability date of the calculated control. We can also rewrite this causality condition under the form of a (max, +) inequality

$$G_u\otimes\underline{x}(k_s)\leq u(k_s+1) \tag{8}$$

where G_u is the $\otimes-$product of T_{comp} and a full matrix of zeros E ($E_{i,j}=e=0$) with appropriate dimensions. The previous expression is equivalent to

$$T_{comp}\leq T^{maj}_{comp} \tag{9}$$

with $T_{comp}^{maj} = (E \otimes \underline{x}(k_s)) \backslash u(k_s + 1)$. The value of T_{comp}^{maj} is computed by the relevant step $T_{comp}^{maj} \leftarrow (E \otimes \underline{x}(k_s)) \backslash u(k_s+1)$ which can be executed at the end of Algorithm 1 (denoted T_{comp}^{maj}) or at the end of each iteration $\langle i \rangle$ (denoted $T_{comp,\langle i \rangle}^{maj}$). Variable T_{comp}^{maj} expresses a majorant of the necessary resource which must be allocated for the program execution time. Contrary to T_{comp}, the values T_{comp}^{maj} and $T_{comp,\langle i \rangle}^{maj}$ do not depend on the computer performance and the instructions of Algorithm 1, but depend on the used data, that is, the parameters of the control problem and the last calculated control. As its value depends on the successive minimizations of the control, we have the relevant minimization: $T_{comp,\langle i' \rangle}^{maj} \leq T_{comp,\langle i \rangle}^{maj}$ for $i' \geq i$ and $T_{comp}^{maj} \leq T_{comp,\langle i \rangle}^{maj}$ for any iteration $\langle i \rangle$.

Therefore, $T_{comp,\langle i \rangle}^{maj}$ is a majorant of the computer time at iteration $\langle i \rangle$ denoted $T_{comp,\langle i \rangle}$. As condition $T_{comp,\langle i \rangle} \leq T_{comp,\langle i \rangle}^{maj}$ is necessary, we can conclude that the control is not always causal if there is an iteration $\langle i \rangle$ such that $T_{comp,\langle i \rangle} \nleq T_{comp,\langle i \rangle}^{maj}$ and that the control problem for the current data and computer presents a causality problem. The same remark holds for the last iteration and, T_{comp} and T_{comp}^{maj}. Note that the result $T_{comp,\langle i \rangle}^{maj} < 0$ (respectively, $T_{comp}^{maj} < 0$) shows a serious causality problem as naturally we also have the condition $0 < T_{comp,\langle i \rangle}$ (respectively, $0 < T_{comp}$).

Example 3 (continued).

Table I shows the control for the two iterations. As the control trajectories are close, we obtain equality $T_{comp,\langle 1 \rangle}^{maj} = T_{comp,\langle 2 \rangle}^{maj}$. So, $T_{comp,\langle 1 \rangle}^{maj} = T_{comp,\langle 2 \rangle}^{maj} = T_{comp}^{maj} = 1$. ■

Let us analyze the control problem for a given computer. Knowing an estimation of the computer time of each iteration ΔT_{comp}, we can approximate $T_{comp,\langle i \rangle}$ with $i.\Delta T_{comp}$. So, $T_{comp,\langle i \rangle} \backsimeq i.\Delta T_{comp} \leq T_{comp,\langle i \rangle}^{maj}$ which also gives $i \leq i_{maj} = \lfloor \frac{T_{comp}^{maj}}{\Delta T_{comp}} \rfloor$ in the standard algebra where $\lfloor x \rfloor$ is the largest integer not greater than x. So, i_{maj} is an estimate of the maximum number of

TABLE I

COMPUTED CONTROL U

Control	$u(k_s+1)$	$u(k_s+2)$	$u(k_s+3)$
Iteration $\langle 1 \rangle$	4	10	17
Iteration $\langle 2 \rangle$	4	10	16

iterations where the problem is causal which can be used in the on-line procedure. Different cases appears for $T^{maj}_{comp,\langle i \rangle} = 1$.

- Inequality $T_{comp,\langle i \rangle} \leq T^{maj}_{comp,\langle i \rangle}$ is satisfied for $\Delta T_{comp} = 0.4$ and iterations $\langle 1 \rangle$ and $\langle 2 \rangle$: the control does not present a causality problem and the optimal approach can be applied directly. Also, $i_{maj} = 2$.
- The inequality is never satisfied for $\Delta T_{comp} = 1.2$ and the optimal approach and the proposed strategy cannot be applied. Also, $i_{maj} = 0$.
- The inequality is satisfied for $\Delta T_{comp} = 0.6$ and iteration $\langle 1 \rangle$ but not $\langle 2 \rangle$. This case where the optimal control cannot be applied is the framework of this paper. As $i_{maj} = 1$, the computed control is causal if the algorithm is stopped at the first iteration. ■

V. CONSTRAINT CONSISTENCY

The expression of the causal constraint shows that its satisfaction can be made with two possible approaches as $\underline{x}(k_s)$ is a datum of the problem:

- An increase of the control $u(k_s+1)$ which can be generated by an increase of the desired output z. As said in the Section Introduction, this technique is beyond the scope of this paper.
- A decrease of G_u produced by a diminution of T_{comp}. A possible technique which is analyzed

in this paper is to consider a limited number of iterations of Algorithm 1 leading to a reduction of the CPU time and consequently, of the computer time T_{comp}. Clearly, stopping Algorithm 1 after the first iteration gives naturally the largest chance for causality to hold and is the most secure choice. Other choices leading to a better minimization of the control are possible and an estimate of the maximum number of iterations such that the control remains causal can be approximated in a preparatory phase of tests based on off-line simulations close to the real conditions of the predictive control. So, we take the following assumption.

Assumption 2: The value of the maximum number of iterations such that the control remains causal is assumed to be known and is non-null. ■

As the objective of Algorithm 1 is naturally to fulfill the requirements of the control problem, below we focus on the satisfaction of the different constraints at each iteration which can be complete or partial:

- In Section V-A, the convergence of Algorithm 1 needs only one iteration where each constraint is satisfied. The main result of [9] is given below.
- In Sections V-B and V-C, we consider an unfinished convergence of Algorithm 1 where only a subset of the constraints is satisfied.

A. Complete validity of the constraints

Theorem 2 below highlights an important case where the convergence of Algorithm 1 is efficient as it presents only one iteration: The satisfaction of the control problem is guaranteed by the computed control if it satisfies the following condition. Remember that (3) is relevant to the state equation (1) of the Timed Event Graph while relation (4) corresponds to the additional constraints of the P-time Event Graph (2) mainly. A simplified form of Theorem 3 in [9] is as follows.

Theorem 2: The trajectory $(\overline{x})^2$ satisfies the system composed of (3) and (4) when $\Psi_h \otimes U = X^1$. Moreover, $(\overline{x})^2 = (\overline{x})^1$.$\blacksquare$

Algorithm 1 is strongly polynomial since the resolution needs a unique iteration composed of the simple application of elementary operations $\oplus$, $\otimes$, $\wedge$ and $\backslash$. Below, the addition of equality $x(k_s) = x(k_s)$ facilitates the rewriting of the condition of Theorem 2 with a simpler notation: The problem is now to check the existence of a solution of $\overline{u} \in \mathbb{R}^{\overline{q}}$ in the equality

$$\overline{B} \otimes \overline{u} = \overline{x} \text{ for any } \overline{x} \in \mathbb{R}^{\overline{n}} \text{ satisfying } \overline{x} \geq \overline{A} \otimes \overline{x} \tag{10}$$

with the following notation: $\overline{B} = \begin{pmatrix} \mathcal{I} & \varepsilon \\ \varepsilon & \Psi_h \end{pmatrix}$, $\overline{u} = \begin{pmatrix} x(k_s) \\ U \end{pmatrix}$, $\overline{x} = \begin{pmatrix} x(k_s) \\ X \end{pmatrix}$, $\overline{A} = D_h$, $\overline{n}$=card($\overline{x}$) and $\overline{q}$=$card(\overline{u})$. *In this subsection*, we assume in (10) that matrix $\overline{B}$ has no null rows as $\overline{x}$ is finite, otherwise, equality $\overline{B}\otimes\overline{u} = \overline{x}$ has no solution. Without a loss of generality, we assume that matrix $\overline{B}$ has no null columns so that $\overline{u}$=$\overline{B}\backslash\overline{x}$ is finite. This assumption only expresses that each control transition is connected to the system. So, matrix $\overline{B}$ is called doubly $\mathbb{R}-$astic. We naturally assume that the associated graph of $\overline{A}$ does not contain circuits with strictly positive weight so that $\overline{A}^* \in \mathbb{R}_{\max}^{\overline{n}x\overline{n}}$. Theorem 3 below analyzes the existence of a solution $\overline{u}$ in (10) and provides conditions which lead to a convergence in one iteration.

Theorem 3: [9] The greatest vector $\overline{u}$=$\overline{B}\backslash\overline{x}$ satisfies the system (10) if and only if

$$B \otimes (\overline{B}\backslash\overline{A}^*) = \overline{A}^*. \blacksquare$$

So, Theorem 3 gives practical tests which use only the entries of $\overline{B}$ and $\overline{A}^*$ without calculating the state and the control.

When the equality $\overline{B} \otimes (\overline{B} \backslash \overline{A}^*) = \overline{A}^*$ is not satisfied for all columns of $\overline{A}^*$ but for some columns denoted $(\overline{A}^*)^{=}_{.,k}$ (So, $\overline{B} \otimes (\overline{B} \backslash (\overline{A}^*)^{=}_{.,k}) = (\overline{A}^*)^{=}_{.,k}$), a predictive control using a space controller compensates for the non-satisfaction of the condition $\overline{B} \otimes (\overline{B} \backslash \overline{A}^*) = \overline{A}^*$ by reducing the state space to the subspace $\mathrm{Im}(\overline{A}^*)^{=}$ under the condition $\mathrm{Im}(\overline{A}^*)^{=} \neq \emptyset$ [9].

Let us show that Example 3 does not satisfy the conditions of Theorem 3 and the previous approach.

Example 3 (continued).

The direct calculation in Scilab shows that $\overline{B} \otimes (\overline{B} \backslash \overline{A}^*) \neq \overline{A}^*$. Let us give more details. Matrix $\overline{B}$ is not row $\mathbb{R}-$astic as the third row of Ψ_h is null. The relevant equality $\overline{B}_{i,.} \otimes \overline{u} = \overline{x}_i$ is never satisfied for $i = 6$ since $\overline{B}_{6,.} = \varepsilon$ and $\overline{x}_6$ is finite. Similarly, $\overline{B}_{i,.} \otimes (\overline{B} \backslash (\overline{A}^*)_{.,k}) = (\overline{A}^*)_{i,k}$ is never satisfied for $i = 6$ and any $k \in \{1, \ldots, \overline{n}\}$ since row $(\overline{A}^*)_{6,.}$ is finite (Remember that $\overline{A}^* = (D_h)^*$). Therefore, it implies $\overline{B} \otimes (\overline{B} \backslash \overline{A}^*) \neq \overline{A}^*$. Another possible approach is the predictive control based on a space controller but we cannot apply it as there is no column $k \in \{1, \ldots, \overline{n}\}$ such that $\overline{B} \otimes (\overline{B} \backslash (\overline{A}^*)_{.,k}) = (\overline{A}^*)_{.,k}$: So, $\mathrm{Im}(\overline{A}^*)^{=} = \emptyset$. ■

B. Partial validity of the constraints

Therefore, the aim of this part is to generalize Theorems 2 and 3 by analyzing the consistency of each row of system (10). The validity of the constraints at each iteration of Algorithm 1 can now be partial.

This approach uses the following fundamental theorem which considers the finite solutions to equality

$$A \otimes x = b$$

where $A \in \mathbb{R}_{\max}^{mxn}$, $b \in \mathbb{R}^m$. The relevant set of solutions over $\mathbb{R}$ is denoted by S. System $A \otimes x = b$

is said to be inconsistent if $S = \emptyset$, that is, it has no finite solution.

We denote the set of indexes for the rows $I = \{1, .., m\}$ and for the columns $J = \{1, .., n\}$ as A is a $(m\text{x}n)$ matrix. Remember that x^+ is the greatest solution to $A \otimes x \leq b$. In the following definition, we consider the finite entries of A which can imply the equality $A_{i,j} \otimes x_j^+ = b_i$ where $A_{i,j}$, $x_j^+$and $b_i \in \mathbb{R}$:

$$\text{For } j \in J,\ V_j = \{i \in I \text{ such that } A_{i,j} \text{ is finite and } x_j^+ = A_{i,j} \backslash b_i\}.$$

***Theorem** 4:* (Theorem 3.1.1(c) in [5] page 54, K. Zimmermann (1976)) Let $A \in \mathbb{R}_{\max}^{mxn}$ be doubly $\mathbb{R}-$astic and $b \in \mathbb{R}^m$. Then, $x \in S$ if and only if

$$x \leq x^+ \text{and} \bigcup_{j \in J |\ x_j = x_j^+} V_j = I. \ \blacksquare$$

***Corollary** 1:* (Corollary 3.1.2 in [5],) Let $A \in \mathbb{R}_{\max}^{mxn}$ be doubly $\mathbb{R}-$astic and $b \in \mathbb{R}^m$. Then, the following three statements are equivalent:

1) $\text{card}(S) \neq 0$

2) $x^+ \in S$ $\quad\blacksquare$

3) $\bigcup_{j \in J} V_j = I$.

If a row $A_{i,.}$ is null (A is not row $\mathbb{R}-$astic), we have $\bigcup_{j \in J} V_j \neq I$ and we can conclude that there is no finite solution (moreover, the unique row $A_{i,.} \otimes x = b_i$ with $A_{i,.} = \varepsilon$ has no infinite solution ($-\infty$ or $+\infty$) as ε is absorbing and $b_j \in \mathbb{R}$): So, the system is inconsistent. Now, if a column $A_{.,j}$ is null (A is not column $\mathbb{R}-$astic), V_j is empty and there is no effect on the equality. Note that this theorem and its corollary are a slight generalization of Theorem 2.1 and Corollary 2.1 in [4] where A is defined over $\mathbb{R}$.

Example 4. Consider $A \otimes x = b$ with $A = \begin{pmatrix} \varepsilon & \varepsilon & 3 \\ 5 & 7 & 6 \\ 0 & 6 & 0 \end{pmatrix}$ and the right-hand side vector

$b = \begin{pmatrix} 5 & 8 & 7 \end{pmatrix}^t$. Let matrix M be defined by $M_{i,j} = A_{i,j} \backslash b_j$. Below, the minimal elements $M_{i,j}$ of each column j are written in bold.

so, $M = \begin{pmatrix} +\infty & +\infty & \mathbf{2} \\ \mathbf{3} & \mathbf{1} & \mathbf{2} \\ 7 & \mathbf{1} & 7 \end{pmatrix}$ and $x^+ = \begin{pmatrix} 3 & 1 & 2 \end{pmatrix}^t$. For columns 1, 2 and 3, we obtain $V_1 = \{2\}$, $V_2 = \{2, 3\}$, $V_3 = \{1, 2\}$ respectively. As $\bigcup\limits_{j \in J} V_j = \{1, 2, 3\} = I$, there is at least a solution: card$(S) \neq 0$ and $x^+ = \begin{pmatrix} 3 & 1 & 2 \end{pmatrix}^t \in S$. Moreover, $x = \begin{pmatrix} \theta & 1 & 2 \end{pmatrix}^t$ with $\theta < 3$ is also a solution as $\bigcup\limits_{j \in J |\ x_j = x_j^+} V_j = V_2 \cup V_3 = \{1, 2, 3\} = I$. ■

We now introduce the following variation of the previous results which is used in the rest of this paper: It gives the rows i where the equality $A_{i,.} \otimes x^+ = b_i$ holds. Contrary to the previous results, the following corollary does not take the assumption that matrix A is row $\mathbb{R}-$astic. At the best of our knowledge, this result is original in the max-plus algebra as it considers the case $\bigcup\limits_{j \in J} V_j \neq I$ showing a partial consistency.

***Corollary** 2:* Let $A \in \mathbb{R}_{\max}^{mxn}$ be column $\mathbb{R}-$astic and $b \in \mathbb{R}^m$. Then,

$$A_{i,.} \otimes x^+ = b_i \text{ is satisfied, if and only if, } i \in \bigcup_{j \in J} V_j.$$

Proof. The proof is given in the appendix. ■

Example 4 modified. The right-hand side vector is now $b = \begin{pmatrix} 5 & 8 & 9 \end{pmatrix}^t$. We obtain: $M = \begin{pmatrix} +\infty & +\infty & \mathbf{2} \\ \mathbf{3} & \mathbf{1} & \mathbf{2} \\ 9 & 3 & 9 \end{pmatrix}$ and $x^+ = \begin{pmatrix} 3 & 1 & 2 \end{pmatrix}^t$. For columns 1, 2 and 3, we obtain $V_1 = \{2\}$, $V_2 = \{2\}$ and $V_3 = \{1, 2\}$ respectively. As $\bigcup\limits_{j \in J} V_j \neq I = \{1, 2, 3\}$, there is no solution: $S = \varnothing$. Set $\bigcup\limits_{j \in J} V_j = \{1, 2\}$ gives the rows where the equality is satisfied: $A_{1,.} \otimes x^+ = b_1 = 5$ and $A_{2,.} \otimes x^+ = b_2 = 8$ but $A_{3,.} \otimes x^+ < b_3 = 9$. ■

Corresponding to the condition of Theorem 3, the following equality

$$\overline{B} \otimes \overline{v} = \overline{A}^* \ , \tag{11}$$

where $\overline{v}$ is a ($\overline{q}$x$\overline{n}$) matrix will be useful. In the rest of the paper, the assumption that matrix $\overline{B}$ is row $\mathbb{R}-$astic is not taken.

The following property is an application of Corollary 2 which allows an analysis of each row of system (11) by inspection of the sets $V_{j,k}$ defined as follows. The greatest solution is denoted by $\overline{v}^+$. We denote the set of indexes for the rows $I = \{1,..,\overline{n}\}$ and for the columns $J = \{1,..,\overline{q}\}$ as $\overline{B}$ is a ($\overline{n}$x$\overline{q}$) matrix. Let $K = \{1,..,\overline{n}\}$ be the set of indexes of columns of $\overline{A}^*$. Corresponding to column $j \in J$ of $\overline{B}$ and column $k \in K$ of $\overline{A}\ ^*$, $V_{j,k}$ is defined by

$$V_{j,k} = \{i \in I \text{ such that } \overline{B}_{i,j} \text{ is finite and } \overline{v}^+_{j,k} = \overline{B}_{i,j} \backslash (\overline{A}^*)_{i,k}\}.$$

As the set V_j exploited in Corollary 2, the set $V_{j,k}$ expresses a consistency of each row $(\overline{B})_{i,.} \otimes \overline{v}^+_{.,k} = (\overline{A}^*)_{i,k}$ for $i \in V_{j,k}$ where this equality considers a column $k \in K$ of $\overline{A}\ ^*$. Now we introduce the following notations which will be useful in the rest of the paper. Let us denote

$$I_{g,k} = \bigcup_{j \in J} V_{j,k}$$

the set of guaranteed rows for a given $k \in K$ (Matrix Δ defined in Example 3 below is deduced from $I_{g,k}$),

$$I_g = \bigcap_{k \in K} I_{g,k}$$

the set of guaranteed rows for any $k \in K$,

$$I_p = \bigcup_{k \in K} I_{g,k} \supset I_g$$

the set of possibly satisfied rows, and

$$I_{ns} = \{i \in I \mid i \notin I_p\}$$

the set of non satisfied rows. So, $I = I_p \cup I_{ns}$.

***Property** 1:* Matrix $\overline{v}$ is a solution to system (11) if and only if

$$\overline{v} \leq \overline{v}^+ \text{ and} \bigcap_{k \in K} \bigcup_{j \in J| \ \mathbf{v}_{j,k} = \mathbf{v}^+_{j,k}} V_{j,k} = I.$$

The set I_g gives the rows of (11) where the equality holds for $\overline{v}$=$\overline{v}^+$.

Proof. The proof is given in the appendix. ■

We now make the connection between the partial consistency of $\overline{B} \otimes \overline{v} = \overline{A}^*$ and the consistency of each row of system (10) composed of $\overline{B}_{\ i,.} \otimes \overline{u} = \overline{x}_i$ and $(\overline{x})_i \geq (\overline{A})_{i,.} \otimes \overline{x}$. Giving a more complete version of Property 4 in [10], the following result generalizes Theorem 3 (Section V-A) by considering the consistency of each row.

***Theorem** 5:* For the greatest vector $\overline{u}$=$\overline{B} \setminus \overline{x}$ with $\overline{x}$ $\in$Im$\overline{A}^*$,

- equality $\overline{B}_{\ i,.} \otimes \overline{u} = \overline{x}_i$ for $i \in I_g$ is always satisfied.
- equality $\overline{B}_{\ i,.} \otimes \overline{u} = \overline{x}_i$ for $i \in I_{g,k}$ is always satisfied when $\overline{x}$ $\in$Im$(\overline{A}^*)_{.,k}$ for a given $k \in K$.
- equality $\overline{B}_{\ i,.} \otimes \overline{u} = \overline{x}_i$ for $i \in I_p$ is possibly satisfied.

Moreover, equality $\overline{B}_{\ i,.} \otimes \overline{u} = \overline{x}_i$ for $i \in I_{ns}$ is never satisfied for any $\overline{u}$ solution to $\overline{B} \otimes \overline{u} \leq \overline{x}$ when $\overline{x}$ $\in$Im$\overline{A}^*$.

Proof. The proof is given in the appendix. ■

The first three points of Theorem 5 consider the greatest vector $\overline{u}$=$\overline{B} \setminus \overline{x}$. In the first point, the set I_g guarantees the consistency of a subset of constraints in (11) for any state trajectory $\overline{x}$ $\in$Im$\overline{A}^{\ *}$. The same remark holds for the set $I_{g,k}$ but the state trajectory $\overline{x}$ follows a unique direction $(\overline{A}^*)_{.,k}$ with $k \in K : \overline{x} = \lambda_k \otimes (\overline{A}^*)_{.,k}$. We directly deduce that, depending on the state evolution inside the space Im$\overline{A}^{\ *}$, the set I_p gives the rows of $\overline{B} \otimes \overline{u} = \overline{x}$ where the equality is possibly satisfied. Finally, I_{ns} completes I_p as it is the set of rows where the equality is never satisfied even if we consider, not the greatest vector $\overline{u}$=$\overline{B} \setminus \overline{x}$ but, any solution $\overline{u}$ solution to $\overline{B} \otimes \overline{u} \leq \overline{x}$ when $\overline{x}$ $\in$Im$\overline{A}^{*}$.

C. Partial consistency and iterations of Algorithm 1

The following result will be useful in this subsection. Vector X' is a co-state also considered in the well-known "backward approach" (see part 5.6.2 in [2]).

Property *2:* [9] $X' \leq X^1$ and $X^2 = X'$ where $X' = \Omega_h \otimes x^1(k_s) \oplus \Psi_h \otimes U$.

Analyzing an iteration of Algorithm 1, Property 2 shows the minimization of the state trajectory produced by steps 2 and 3, and the fact that the state equation is satisfied at the end of each iteration. Moreover, we can also say that the proposed method satisfies point a): Each iteration of Algorithm 1 proposes a control which generates an output satisfying the point a) relevant to the desired output (expressed by vector F). Indeed, step 1 also makes a minimization and Algorithm 1 starts from F. The objective is now the analysis of the constraints and we will show that a subset of constraints is always guaranteed at each iteration.

Considering system (4) at each iteration of Algorithm 1, the following theorem completes Theorem 2 (Section V-A). Generalizing Theorem 3 in [10], it is based on Property 2 and Theorem 5 mainly.

Theorem *6:* Each inequality

$$\overline{x}_i \geq (D_h)_i \otimes \overline{x} \text{ of } (4)$$

is satisfied at the end of each iteration of Algorithm 1 when the control computed at step 2 of Algorithm 1 satisfies equality

$$X^1_{i-n} = (\Psi_h)_{i-n,.} \otimes U$$

for $i \in I_{p'} = I_p \cap \{n+1, ..., \overline{n}\}$. The same result holds for $(\overline{x})_i^1 \geq (D_h)_i \otimes (\overline{x})^1$where $(\overline{x})^1$ is calculated at step 1.

Proof. The proof is given in the appendix. ■

Contrary to Theorem 6, the following corollary does not depend on the on-line computation of X_{i-n}^1 and U .

Corollary *3:* Each inequality

$$\overline{x}_i \geq (D_h)_i \otimes \overline{x} \text{ of } (4)$$

for $i \in I_g$ is satisfied at the end of each iteration of Algorithm 1 for the control calculated at step 2 of Algorithm 1.

Proof. We can apply the first case of Theorem 5: The equality $X_{i-n}^1 = (\Psi_h)_{i-n,.} \otimes U$ holds when $i \in I_{g'} = I_g \cap \{n+1, ..., \overline{n}\}$. ■

If the control problem presents a causality problem, Algorithm 1 can be stopped without waiting for its convergence: Without computing the state and the control, the analysis guarantees the satisfaction of a subset of the constraints at the end of each iteration. Knowing the maximum number of iterations denoted i_{maj}, we can choose the first iteration $\langle 1 \rangle$ which is the most secure choice or, an iteration $\langle i \rangle$ with $i \leq i_{maj}$ where the choice $i = i_{maj}$ gives the minimum solution with respect to the other possible choices. In all cases, the desired output is met (Point a)).

Remark *1:* As in the complete convergence of Algorithm 1, the last iteration proposes an initial state $x^1(k_s)$ (we have $x^2(k_s) = x^1(k_s)$ by construction) satisfying $x^1(k_s) \leq \underline{x}(k_s)$ which is the starting point of the state trajectory (under condition $x^1(k_s) = \underline{x}(k_s)$).

D. Example 3 (continued)

We now consider the partial consistency of $\overline{B} \otimes \overline{v} = \overline{A}^*$ and deduce the consistency of each row of system (10) composed of $\overline{B}_{i,.} \otimes \overline{u} = \overline{x}_i$ and $(\overline{x})_i \geq (\overline{A})_{i,.} \otimes \overline{x}$. Generated by the execution of Algorithm 1, Tables I in Section IV-D and II show the evolution of the control U and the state trajectory X^2 obtained at the end of the iterations $\langle 1 \rangle$ and $\langle 2 \rangle$ and, the validity of the relevant different relations in $\overline{x} \geq \overline{A} \otimes \overline{x}$ (except obvious relations corresponding to $x(k_s)$) which mainly express the additional constraints given by the P-time Event Graph. Notation: p for possible; g for guaranteed; s for satisfied; ns for non satisfied.

TABLE II

CONSISTENCY DEDUCED FROM THE ROWS OF $\overline{B} \otimes \overline{v} = \overline{A}^*$ AND COMPUTED STATE X^2 | ON-LINE SATISFACTION (S, NS) OF THE CORRESPONDING CONSTRAINTS IN $\overline{x} \geq \overline{A} \otimes \overline{x}$

Row i of $\overline{B} \otimes \overline{v} = \overline{A}^*$	4	5	6	7	8	9	10	11	12
Consistency	p	**g**	ns	p	**g**	ns	p	**g**	ns
State X^2	$x_1(k_s+1)$	$\mathbf{x_2(k_s+1)}$	$x_3(k_s+1)$	$x_1(k_s+2)$	$\mathbf{x_2(k_s+2)}$	$x_3(k_s+2)$	$x_1(k_s+3)$	$\mathbf{x_2(k_s+3)}$	$x_3(k_s+3)$
Iteration $\langle 1 \rangle$	8 | s	7 | s	9 | s	14 | ns	13 | s	15 | s	21 | s	20 | s	21 | ns
Iteration $\langle 2 \rangle$	8 | s	7 | s	9 | s	14 | s	13 | s	15 | s	20 | s	19 | s	21 | s

We have $\overline{n} = (h+1).n = 12$ and $\overline{q} = n+h.card(u)= 6$ as $n = 3$ and $h = 3$. So, $I = \{1, ..., 12\}$, $J = \{1, ..., 6\}$ and $K = \{1, ..., 12\}$. Remember that $\overline{A}^* = (D_h)^*$ and $\overline{B} = \begin{pmatrix} \mathcal{I} & \varepsilon \\ \varepsilon & \Psi_h \end{pmatrix}$. Each entry $\Delta_{i,k}$ of the following $\overline{n}$ x $\overline{n}$ symbol matrix gives the row index $i \in I_{g,k} = \bigcup_{j \in J} V_{j,k}$ for each column $(\overline{A}^*)_{.,k}$ where symbol $=$ expresses that the relevant equality $\overline{B}_{i,.} \otimes \overline{v}_{.,k} = (\overline{A}^*)_{i,k}$ is satisfied while symbol $<$ shows that $\overline{B}_{i,.} \otimes \overline{v}_{.,k} < (\overline{A}^*)_{i,k}$ is obtained.

$$\Delta = \begin{pmatrix} = & = & = & = & = & = & = & = & = & = & = & = \\ = & = & = & = & = & = & = & = & = & = & = & = \\ = & = & = & = & = & = & = & = & = & = & = & = \\ = & < & = & < & = & < & = & < & = & < & = & < \\ = & = & = & = & = & = & = & = & = & = & = & = \\ < & < & < & < & < & < & < & < & < & < & < & < \\ = & = & = & = & = & = & < & = & < & = & < & = \\ = & = & = & = & = & = & = & = & = & = & = & = \\ < & < & < & < & < & < & < & < & < & < & < & < \\ = & = & = & = & = & = & = & = & = & < & = & = \\ = & = & = & = & = & = & = & = & = & = & = & = \\ < & < & < & < & < & < & < & < & < & < & < & < \end{pmatrix} \qquad (12)$$

Let us consider the columns $k \in K$ of matrix Δ. As each column contains the symbol $<$, the equality $\overline{B} \otimes \overline{u} = (\overline{A}^*)_{.,k}$ does not hold: we obtain $I_{g,k} \neq I$ for any $k \in K$ and $\mathrm{Im}(\overline{A}^*)^{=} = \emptyset$. As said at the end of Section V-A, the approach of [9] cannot be applied.

Let us now analyze the rows $i \in I$ of matrix Δ. The set I_g is directly obtained by the intersection of the sets $I_{g,k}$ expressed by the columns $\Delta_{.,k}$ (Each row contains the symbol $=$ only) while I_p is the union of these sets (Each row contains the symbol $=$ at least once). The set I_{ns} is given by the remaining rows (The rows do not contain the symbol $=$). So, $I_g = \{1, 2, 3, 5, 8, 11\}$, $I_p = \{1, 2, 3, 4, 5, 7, 8, 10, 11\} \supset I_g$ and $I_{ns} = \{6, 9, 12\}$ with $I = I_p \cup I_{ns}$.

Guaranteed rows (I_g)

The system composed of $\overline{B}_{i,.} \otimes \overline{u} = \overline{x}_i$ and $(\overline{x})_i \geq (\overline{A})_{i,.} \otimes \overline{x}$ is guaranteed for $i \in I_g$. The rows $i = 5$, 8 and 11 correspond to X_{i-3} for $i \in \{5, 8, 11\}$ or, $x_2(k_s+1)$, $x_2(k_s+2)$ and $x_2(k_s+3)$, respectively. We have

$$\begin{cases} X_2 = x_2(k_s+1) = 7 = (\Psi_h)_{2,.} \otimes U = \begin{pmatrix} 3 & \varepsilon & \varepsilon \end{pmatrix} \otimes U = 7, \\ X_5 = x_2(k_s+2) = 13 = (\Psi_h)_{5,.} \otimes U = \begin{pmatrix} 9 & 3 & \varepsilon \end{pmatrix} \otimes U = 13 \text{ and} \\ X_8 = x_2(k_s+3) = 20 = (\Psi_h)_{8,.} \otimes U = \begin{pmatrix} 15 & 9 & 3 \end{pmatrix} \otimes U = 20. \end{cases}$$

at iteration $\langle 1 \rangle$. The results are similar for iteration $\langle 2 \rangle$. Moreover, the relevant constraints $(\overline{x})_i \geq (\overline{A})_{i,.} \otimes \overline{x}$ are always satisfied which is coherent with Corollary 3: $x_2(k) \geq (A \oplus A^-)_{2,.} \otimes x(k-1) \oplus A^{=}_{2,.} \otimes x(k) \oplus A^{+}_{2,.} \otimes x(k+1)$ for $k = k_s+1$ and k_s+2 and $x_2(k) \geq (A \oplus A^-)_{2,.} \otimes x(k-1) \oplus A^{=}_{2,.} \otimes x(k)$

for $k = k_s + 3$. Table II is coherent with these results.

Possibly satisfied rows but not guaranteed ($i \in I_p$ with $i \notin I_g$)

Each equality $\overline{B}_{\ i,.} \otimes \overline{u} = \overline{x}_i$ for $i \in \{i \in I_p \ \mid i \notin I_g\} = \{4, 7, 10\}$ is possibly satisfied when the relevant row $(\overline{x})_i \geq (\overline{A})_{i,.} \otimes \overline{x}$ is satisfied. Let us show this possibility. After steps 2 and 3 of iteration $\langle 2 \rangle$, all the constraints and $\overline{B}_{\ i,.} \otimes \overline{u} = \overline{x}_i$ for $i \in \{4, 7, 10\}$ are satisfied that is, $(\Psi_h)_{i-3,.} \otimes U = X_{i-3}$ for $i \in \{4, 7, 10\}$: We have respectively

$$\begin{cases} X_1^1 = X_1^2 = x_1(k_s + 1) = 8 = (\Psi_h)_{1,.} \otimes U = \begin{pmatrix} 4 & \varepsilon & \varepsilon \end{pmatrix} \otimes U, \\ X_4^1 = X_4^2 = x_1(k_s + 2) = 14 = (\Psi_h)_{4,.} \otimes U = \begin{pmatrix} 10 & 4 & \varepsilon \end{pmatrix} \otimes U \text{ and,} \\ X_7^1 = X_7^2 = x_1(k_s + 3) = 20 = (\Psi_h)_{7,.} \otimes U = \begin{pmatrix} 16 & 10 & 4i \end{pmatrix} \otimes U. \end{cases}$$

The relevant inequalities of $\overline{x} \geq \overline{A} \otimes \overline{x}$ are satisfied at the end of steps 1 and 3 of iteration $\langle 2 \rangle$.

Non satisfied rows (I_{ns})

The system composed of $\overline{B}_{\ i,.} \otimes \overline{u} = \overline{x}_i$ and $(\overline{x})_i \geq (\overline{A})_{i,.} \otimes \overline{x}$ is not satisfied for $i \in I_{ns} = \{6, 9, 12\}$. At the end of iteration $\langle 2 \rangle$, the following data illustrate this point:

$$\begin{cases} X_3 = x_3(k_s + 1) = 9 > (\Psi_h)_{3,.} \otimes U = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \end{pmatrix} \otimes U = \varepsilon, \\ X_6 = x_3(k_s + 2) = 15 > (\Psi_h)_{6,.} \otimes U = \begin{pmatrix} 7 & \varepsilon & \varepsilon \end{pmatrix} \otimes U = 11 \text{ and} \\ X_9 = x_3(k_s + 3) = 21 > (\Psi_h)_{9,.} \otimes U = \begin{pmatrix} 13 & 7 & \varepsilon \end{pmatrix} \otimes U = 17. \end{cases}$$

The non-consistency can also come from the additional constraints: $X_3 = x_3(k_s + 3) = 21 \not\geq A^{=}_{3,1} \otimes x_1(k_s + 3) = 1 \otimes 21$ at the end of iteration $\langle 1 \rangle$. ■

VI. Conclusion

In this paper, we consider the situation where the causality phenomenon prevents the convergence of Algorithm 1 and the determination of the optimal control for a given computer. We focus on the reduction of the CPU time of the predictive control leading to the satisfaction of the causality constraint. An approach is to use a space controller leading to a convergence in one iteration with the satisfaction of all the additional constraints expressed by the P-time Event Graph [9]. As this technique needs the satisfaction of a space condition and a modification of Algorithm 1, we propose a generalization of this approach which can be applied to any system when only a subset of crucial additional constraints must be satisfied. Considering less restrictive conditions, this second approach needs a minor modification of Algorithm 1 that is to stop Algorithm 1 at a given iteration. The analysis has shown that, at the end of each iteration: For a subset of additional constraints, the satisfaction of the relevant constraints is guaranteed by the computed control; for another subset, the control can generate the satisfaction of the relevant constraints but for a state evolution into a specific space; Finally, the proposed approach shows the possibility of reduction of the CPU time and enlarges the class of problems where the predictive control can be applied. Among the different perspectives, an open problem is to determine the best approximation of control, state and output in relation to a (max, +) distance when the initial control problem has no solution.

REFERENCES

[1] Amari S., I. Demongodin, J.J. Loiseau, C. Martinez (2012) Max-Plus Control Design for Temporal Constraints Meeting in Timed Event Graphs. IEEE Transactions on Automatic Control, Institute of Electrical and Electronics Engineers, 2012, 57 (2), pp. 462 - 467.

[2] Baccelli F., Cohen G., Olsder G.J. and Quadrat J.P. (1992) Synchronization and Linearity. An Algebra for Discrete Event Systems. Available from http://maxplus.org, New York, Wiley.

[3] van den Boom T., B. Heidergott, and B. De Schutter (2007) Complexity reduction in MPC for stochastic max-plus-linear discrete event systems by variability expansion. Automatica, vol. 43, no. 6, pp. 1058-1063, June 2007.

[4] Butkovic P. (2003) Max-algebra: the linear algebra of combinatorics? Linear Algebra and its Applications 367, pp. 313-335.

[5] Butkovic P. (2010) Max-algebra: theory and algorithms. Springer-Verlag London, Monographs in mathematics.

[6] Cuninghame-Green R.A. and P. Butkovic (2003) The equation Ax=By over (max, +). Theoretical Computer Science 293, pp. 3-12.

[7] Declerck P. (2010) Trajectory Tracking Control of a Timed Event Graph with Specifications Defined by a P-time Event Graph: On-line control and Off-line preparation. Wodes 2010, Berlin, Germany, August 30-September 1

[8] Declerck P. (2013) Discrete Event Systems in Dioid Algebra and Conventional Algebra. Focus Series in Automation & Control, ISTE Ltd and John Wiley.

[9] Declerck P. and Guezzi A. (2014) Predictive Control of Timed Event Graphs with Specifications Defined by P-time Event Graphs. Available from http://perso-laris.univ-angers.fr/˜declerck/ , Journal of Discrete Event Dynamic Systems, September 2014, Volume 24, Issue 3, pp. 261-273.

[10] Declerck P. (2014) Causality Phenomenon and Compromise Technique for Predictive Control of Timed Event Graphs with Specifications Defined by P-Time Event Graphs. Available from http://perso-laris.univ-angers.fr/˜declerck/ , WODES'14 (12th IFAC International Workshop on Discrete Event Systems), May 14-16, 2014, pp. 99-104.

[11] Goto H. and S. Masuda (2004) On the properties of the greatest subsolution for linear equations in the max-plus algebra. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E87-A, No.2, pp. 424-432.

[12] Guezzi A., P. Declerck and J.-L. Boimond (2008) From monotone inequalities to Model Predictive Control. Available from http://perso-laris.univ-angers.fr/˜declerck/, ETFA 2008, Hamburg, Germany 15-18 September 2008.

[13] David-Henriet X., Brunsch T., Raisch J., Hardouin L. (2012) Stock Reduction for Timed Event Graphs Based on Output Feedback. In : WODES 2012. Guadalajara. Proceedings of the 11th edition of Workshop On Discrete Event System, 2012.

[14] Katz R. D. (2007) Max-Plus (A,B)-Invariant Spaces and Control of Timed Discrete-Event Systems. IEEE Transactions on Automatic Control, Vol.52, No. 2, pp. 229-241.

[15] Maia C. A., Andrade C. R., Hardouin L. (2011) On the control of max-plus linear system subject to state restriction. Automatica. 2011. Vol. 47 n°5 p. 988 - 992.

[16] McMillan K. and Dill D. (1992) Algorithms for interface timing verification. Proceedings of the IEEE, International Conference on Computer Design: VLSI in Computers and Processors, pp. 48-51.

[17] Necoara I., T.J.J. van den Boom, B. De Schutter, and H. Hellendoorn (2008) Stabilization of max-plus-linear systems using model predictive control: The unconstrained case. Automatica, vol. 44, no. 4, pp. 971-981, April 2008.

[18] De Schutter B. and van den Boom T. (2001) Model predictive control for max-plus-linear discrete event systems. Automatica, Vol. 37(7), pp. 1049-1056.

[19] De Schutter B. and T.J.J. van den Boom (2003) MPC for discrete-event systems with soft and hard synchronization

constraints. International Journal of Control, 76(1):82-94, January 2003.

[20] Walkup E. (1995) Optimization of linear max-plus systems with application to timing analysis. University of Washington, Ph.D. thesis.

[21] Zeidler E. (1986) Nonlinear Functional Analysis and its Applications I. Fixed-Point Theorems. Springer-Verlag New York, Inc.

VII. Appendix

Proof of Corollary 2.

Firstly, the greatest solution x^+ always satisfies $A \otimes x \leq b$ and a subset of rows satisfies $A_{i,.} \otimes x^+ = b_i$. In other words, a subsystem satisfies equality $A_1 \otimes x^+ = b_1$ and another one $A_2 \otimes x^+ \leq b_2$ with $(A_2)_{i,.} \otimes x^+ \neq (b_2)_{i,.}$ after permutation of the rows in two parts. Note that the case $(A_2)_{i,.} = \varepsilon$ is possible as it implies $A_2 \otimes x^+ \leq b_2$ with b_2 over $\mathbb{R}$. Corollary 1 can be applied to $A_1 \otimes x = b_1$. Firstly, this system is row $\mathbb{R}-$astic otherwise there is no solution. Secondly, considering $S_1 = \{x \in \mathbb{R}^n \mid A_1 \otimes x = b_1\}$, point 2) is satisfied: We have $x^+ \in S_1$ as $A_1 \otimes x^+ = b_1$. It implies point 3) $\bigcup\limits_{j \in J} V_j = I_1$ where I_1 is the number of rows of A_1.

Conversely, let us assume that $i \in \bigcup\limits_{j \in J} V_j$. So, $i \in V_j$ for some $j \in J$. By definition of V_j, each index i satisfies: $i \in I$, $A_{i,j}$ is finite and $x_j^+ = A_{i,j} \backslash b_i$. So, we have equality $A_{i,j} \otimes x_j^+ = A_{i,j} \otimes (A_{i,j} \backslash b_i) = b_i$ over $\mathbb{R}$ which implies equality $A_{i,.} \otimes x^+ = b_i$ as only one equality $A_{i,j} \otimes x_j^+ = b_i$ for a given j is sufficient. ■

Proof of Property 1.

The first point is a direct application of Theorem 4 to system (11) for a given $k \in K$: The system $\overline{B} \otimes \overline{v}_{.,k} = (\overline{A}^*)_{.,k}$ has a solution $\overline{v}_{.,k}$ if and only if $\overline{v}_{.,k} \leq \overline{v}^+_{.,k}$ and $\bigcup\limits_{j \in J \mid \mathbf{v}_{j,k} = \mathbf{v}^+_{j,k}} V_{j,k} = I$. In the second point, the consideration of the specific solution $\overline{v}^+$ simplifies the writing of the sets: Indeed, applying Corollary 2, each index of the set $I_{g,k} = \bigcup\limits_{j \in J} V_{j,k}$ leads to an equality in the relevant row $\overline{B}_{i,.}\otimes \overline{v}_{.,k} = (\overline{A}^{*})_{i,k}$ for $k \in K$ and finally, the intersection of the sets $I_g = \bigcap_{k\in K} \bigcup_{j\in J} V_{j,k}$ gives the rows $i \in I$ of $\overline{B} \otimes \overline{v} = (\overline{A}^{*})$ where the equality holds for $\overline{v}{=}\overline{v}^{+}$, that is, $card(K)$ equalities $\overline{B}_{i,.}\otimes \overline{v}_{.,k} = (\overline{A}^{*})_{i,k}$ are satisfied. ■

Proof of Theorem 5.

Let us consider the first case. Taking $i \in I_g$ ($I_g = \bigcap_{k\in K} I_{g,k}$ with $I_{g,k} = \bigcup_{j\in J} V_{j,k}$), we must prove that $\overline{B}_{\ i,.} \otimes (\overline{B} \setminus \overline{x}) = \overline{x}_i$ for any $\overline{x}$ $\in$Im $\overline{A}^{*}$, that is, $\overline{x} = \overline{A}^{*} \otimes \lambda$ where $\lambda \in \mathbb{R}^{K}$. Firstly, $\overline{B}_{i,.}$ $\otimes(\overline{B} \setminus\overline{x}) = \overline{B}_{i,.} \otimes [\overline{B} \setminus(\overline{A}^{*} \otimes \lambda)] = \overline{B}_{i,.} \otimes[\overline{B} \setminus(\bigoplus_{k\in K} \lambda_k \otimes (\overline{A}^{*})_{.,k})] \geq$

$\overline{B}_{i,.} \otimes[\bigoplus_{k\in K} \overline{B} \setminus(\lambda_k \otimes (\overline{A}^{*})_{.,k})]$ (Property f2 page 180 in [2]) $=$

$\overline{B}_{i,.} \otimes[\bigoplus_{k\in K} \lambda_k \otimes \overline{B} \setminus(\overline{A}^{*})_{.,k}] =$

$\bigoplus_{k\in K} \lambda_k \otimes \overline{B}_{i,.} \otimes[\overline{B} \setminus(\overline{A}^{*})_{.,k}]$.

Secondly, Property 1 says that the set I_g gives the rows of (11) where the equality holds, that is $\overline{B}_{\ i,.}\otimes \overline{v}_{.,k} = (\overline{A}^{*})_{i,k}$ where $\overline{v}_{.,k} = \overline{B} \setminus(\overline{A}^{*})_{.,k}$ for any $k \in K$. It implies $\bigoplus_{k\in K} \lambda_k \otimes \overline{B}_{\ i,.} \otimes [\overline{B} \setminus(\overline{A}^{*})_{.,k}] = \bigoplus_{k\in K} \lambda_k \otimes (\overline{A}^{*})_{i,k} = (\overline{A}^{*})_{i,.} \otimes \lambda = \overline{x_i}$ for $i \in I_g$ and for any λ.

Finally, we obtain $\overline{B}_{i,.} \otimes(\overline{B} \setminus\overline{x}) \geq \overline{x_i}$ and the equality $\overline{B}_{i,.} \otimes(\overline{B} \setminus\overline{x}) = \overline{x_i}$ holds as $\overline{B} \otimes[\overline{B} \setminus\overline{x}] \leq \overline{x}$ by definition of the residuation.

The second case considers less restrictive sets $I_{g,k}$ and says that each equality $i \in I_{g,k}$ is also satisfied. Indeed, we have Im$\overline{A}^{*} = \{\overline{A}^{*} \otimes \lambda$ such that $\lambda \in \mathbb{R}^{\overline{n}}\}$ by definition of the image but the state trajectory $\overline{x}$ can also be a linear combination of a subset of the columns of $\overline{A}^{*}$. So, in the case where the state trajectory $\overline{x}$ follows a unique direction $(\overline{A}^{*})_{.,k}$ with $k \in K$, the equalities $\overline{B}_{\ i,.} \otimes\overline{u} = \overline{x}_i$ for $i \in I_{g,k}$ are satisfied. Indeed, we can take the relations of the previous case with the following modifications: Set I_g is replaced by $I_{g,k}$; we have equality $\overline{B}_{i,.} \otimes(\overline{B} \setminus\overline{x}) = \lambda_k \otimes \overline{B}_{i,.}$ $\otimes[\overline{B} \setminus(\overline{A}^{*})_{.,k}]$ for an only k as the application of Property f2 page 180 in [2] which produces an inequality is not necessary; Property 1 is replaced by Corollary 2 which considers set $I_{g,k}$

and $(\overline{A}^*)_{.,k}$ for a given k.

The third case is immediate: As the state evolution inside the space Im$\overline{A}^*$ can coincide with the previous case, that is, a direction $(\overline{A}^*)_{.,k}$ with $k \in K$, the set I_p gives the rows of $\overline{B} \otimes \overline{u} = \overline{x}$ where the equality is possibly satisfied.

In the fourth case, we consider $i \in I_{ns}$ and any solution $\overline{u}$ solution to $\overline{B} \otimes \overline{u} \leq \overline{x}$ when $\overline{x} \in$Im$\overline{A}^*$. Considering the resolution of $\overline{B} \otimes \overline{v}_{.,k} = (\overline{A}^*)_{.,k}$ for a given k , we cannot find a solution $\overline{v}_{.,k}$ satisfying equality $\overline{B}_{i,.} \otimes \overline{v}_{.,k} = (\overline{A}^*)_{i,k}$ for $i \notin I_p$ as $i \notin I_{g,k} \subset I_p$. It implies that there is no solution $\overline{u}$ to $\overline{B} \otimes \overline{u} \leq \lambda_k \otimes (\overline{A}^*)_{.,k}$ also satisfying equality $\overline{B}_{i,.} \otimes \overline{u} = \lambda_k \otimes (\overline{A}^*)_{i,k}$ for any $\lambda_k \neq \varepsilon$ since this equality leads to $\overline{B}_{i,.} \otimes (-\lambda_k) \otimes \overline{u} = (\overline{A}^*)_{i,k}$. A fortiori, the same conclusion can be said for a variable $\overline{u}$ common to $\overline{B} \otimes \overline{u} \leq \lambda_k \otimes (\overline{A}^*)_{.,k}$ for any k which satisfy all the relevant equalities $\overline{B}_{i,.} \otimes \overline{u} = \lambda_k \otimes (\overline{A}^*)_{i,k}$ with $\lambda_k \neq \varepsilon$. Finally, we can conclude that the equality $\overline{B}_{i,.} \otimes \overline{u} = \bigoplus_{k \in K} \lambda_k \otimes (\overline{A}^*)_{i,k} = \overline{x}$ cannot be satisfied for any $\overline{u}$ solution to $\overline{B} \otimes \overline{u} \leq \overline{x}$. ■

Proof of Theorem 6.

Firstly, all the additional constraints are satisfied in Step 1. Indeed, Step 1 at each iteration of Algorithm 1 calculates a solution $(\overline{x})^1$ which satisfies $(\overline{x})^1 \leq D_h \backslash (\overline{x})^1$ (Theorem 4.73 in [2]) which is equivalent to $(\overline{x})^1 \geq D_h \otimes (\overline{x})^1$ (Lemma 4.77 in [2]) After remembering that $D_h = \overline{A}$, note that $\{ \overline{x} \in \mathbb{R}^{\overline{n}}$ such that $\overline{x} \geq \overline{A} \otimes \overline{x}\}$=Im$\overline{A}^*$ (Lemma 4.77 page 191 in [2], [14]). Moreover, the equality $\overline{x} = \overline{B} \otimes \overline{u}$ leads to $X^1_{i-n} = (\Psi_h)_{i-n,.} \otimes U$ where $i \in \{n+1, ..., \overline{n}\}$ in the notation of Algorithm 1. These two connections show that we can apply the third case of Theorem 5: By assumption, the equality $X^1_{i-n} = (\Psi_h)_{i-n,.} \otimes U$ holds when $i \in I_{p'} = I_p \cap \{n+1, ..., \overline{n}\}$.

Secondly, let us now prove that $X^1_{i-n} = X'_{i-n}$ for $i \in I_{g'}$ where $X' = \Omega_h \otimes x^1(k_s) \oplus \Psi_h \otimes U$. Indeed, Property 2 shows that $X^1 \geq X' = \Omega_h \otimes x^1(k_s) \oplus \Psi_h \otimes U$ and we have $X^1_{i-n} =$

$(\Psi_h)_{i-n,.} \otimes U$ by assumption. These two points imply the desired result $X^1_{i-n} = X'_{i-n}$. Moreover, Property 2 says that $X^2 = X'$ and we finally obtain $X^2_{i-n} = X^1_{i-n} = X'_{i-n}$ for $i \in I_{g'}$. So, we have $(\overline{x})^2_i = (\overline{x})^1_i$ for $i \in I_g$ as $x^2(k_s) = x^1(k_s)$ by construction. Now consider $(\overline{x})^1 \geq D_h \otimes (\overline{x})^1$. As Step 3 by construction implies $(\overline{x})^1 \geq (\overline{x})^2$, we obtain $(\overline{x})^2_i \geq (D_h)_i \otimes (\overline{x})^2$ for $i \in I_g$. Indeed, $(\overline{x})^2_i = (\overline{x})^1_i \geq (D_h)_i \otimes (\overline{x})^1 \geq (D_h)_i \otimes (\overline{x})^2$. We conclude that the corresponding additional constraints are satisfied at the end of each iteration $\langle i \rangle$ of Algorithm 1. ■