From Linear Programming to Graph Theory: Standardization of the Algebraic Model of Timed Event Graphs

Abdelhak Guezzi[1], **Philippe Declerck**[1]

[1]LISA EA4014, University of Angers, 62 avenue Notre-Dame du Lac, 49000 Angers, France
(abdelhak.guezzi@etud.univ-angers.fr, philippe.declerck@istia.univ-angers.fr)

ABSTRACT

The aim of this paper is the standardization of the algebraic model of Timed Event Graphs defined in the conventional algebra. The result of the proposed technique is an auto-regressive model similar to the sampled state equations of the automatic control. Based on standard incidence matrices, this new model is a set of inequalities defined on a reduced horizon which allows an efficient calculation of the state trajectory knowing the initial state and the control. The starting point of the approach is the algebraic model deduced from the initial Timed Event Graph [6]. Using linear programming, we give two dual techniques which allow the building of the standardized model and the corresponding Timed Event Graph. This study improves our understanding of the connections between linear programming and graph theory.
Keywords: Discrete Event Systems, Timed Event Graphs, Algebraic Model, Linear Programming, Control Systems.

1. Introduction

In the literature, different papers [3] [4] [6] propose the modeling of Timed Event Graphs and P-time Event Graphs in the conventional algebra. The result is a dynamical model defined by a polyhedron $A.x \leq b$. The application of a classical algorithm of linear programming allows the determination of the earliest trajectories and this model can also be used in model predictive control [6]. Another application is the calculation of the production rate [3] [4]. Like the (max, +) algebra, the main characteristic of this approach is that the concepts of lattice theory can be used.

Deduced from the initial Timed Event Graph [6], this dynamical model uses special incidence matrices. However, they are different from the incidence matrices of the fundamental relation of marking. The aim of this paper is to improve this dynamical model: we desire to deduce a new form of the model defined on a reduced horizon with the following characteristics:

- The matrices of this new model are well-known ingoing/outgoing incidence matrices used in the fundamental relation of marking; A Petri net can directly be deduced from this model.
- Therefore, this model allows an efficient calculation of the state knowing the past state and the control: it avoids the repetition of the same calculations in the iterative calculation of the state.

This objective is well-known in (max, +) algebra. The objective is that the calculation time is similar to the determination time of the state using the following state equation [1] in (max, +) algebra

$$x(k) = A \otimes x(k-1) \oplus B \otimes u(k) \qquad (1)$$

where maximization and addition operations are respectively denoted $\oplus$ and $\otimes$. Depending on the size of matrices, the calculation time of a state trajectory x is polynomial and small sizes can be considered without computer. Each variable $x_i(k)$ represents the date of the k^{th} firing of internal transition x_i. Vector u corresponds to input transitions denoted u (a supervisory or an external process can fix the events of transition u) while vector y corresponds to output transitions denoted y (the process defined by the considered event graph determines the relevant events). Let us note that the above equality (1) cannot immediately be deduced from the event graph and needs an intermediate step.

In the context of discrete event systems, a more general aim is also the development of a graph theory but completely defined in the standard algebra. Even if the determination of the maximal paths is well-known since the sixties, we propose a different point of view which is algebraic. The main advantage is to bring connections between automatic control of continuous systems and control of discrete event systems. A consequence is a better understanding of these theories. The proposed technique can be used by non-specialist of graph theory which can only learn a limited number of concepts. Another advantage is the possible application of polyvalent algorithms of linear programming like the simplex: they can be applied in classical automatic control but also in discrete event systems. A perspective is the analysis of the calculation time of the different algorithms. Let us recall that, although some artificial examples show exponential running time, the simplex is very efficient in practice and *on the average* [7].

In the following part, we define the problem by describing the initial algebraic model and the desired model of the Timed Event Graph. We then propose two dual approaches which solve the problem in linear programming. A graphical interpretation of the second approach is also given. Finally, we discuss the duality of the two approaches. In part 3.1, we also generalize the technique of Roy (see [5]) to the modeling of an Event Graph: recall that, contrary to the PERT approach where each task is associated with an arc, this technique considers the initial dates of the events and the following events.

2009-CIE39-FR

2. Problem of standardization

2.1. Initial model

The following general model composed of (2), (3) and (4) can be deduced from any Timed Event Graph. The procedure is given in Appendix 1. Let us firstly consider the inequations such as the values of control $u(k)$ and past state $x(k-1)$ can modify current state $x(k)$. The internal inequalities are :

$$\begin{pmatrix} A_{.,1} & A_{.,0} \end{pmatrix} \begin{pmatrix} x(k-1) \\ x(k) \end{pmatrix} \leq -T^A \qquad (2)$$

where notation $A_{.,1}$ (respectively, $A_{.,0}$) corresponds to the columns relevant to the components of $x(k-1)$ (respectively, $x(k)$) without restriction on the rows, the input inequalities are :

$$\begin{pmatrix} B_1 & B_0 \end{pmatrix} \begin{pmatrix} u(k) \\ x(k) \end{pmatrix} \leq -T^B \qquad (3)$$

and the output inequalities which determine the output are :

$$\begin{pmatrix} C_1 & C_0 \end{pmatrix} \begin{pmatrix} x(k) \\ y(k) \end{pmatrix} \leq -T^C \qquad (4)$$

Each row of matrices $A = \begin{pmatrix} A_{.,1} & A_{.,0} \end{pmatrix}$, $B = \begin{pmatrix} B_1 & B_0 \end{pmatrix}$ and $C = \begin{pmatrix} C_1 & C_0 \end{pmatrix}$, is null except two coefficients 1 and -1.

Without reduction of generality, the following assumptions are made. We suppose that inequations corresponding to internal transitions and places with null initial marking are only present in (2). In the contrary case, the possible relations of this type which are in system (3) or (4) can be moved in system (2): they correspond to rows of (3) such as $(B_1)_{\mathtt{i},.} = 0$ for instance. With the aim of simplify the presentation, we also assume that there is no direct relation between the inputs and the outputs, that is, inequality $\begin{pmatrix} D_1 & D_0 \end{pmatrix} \begin{pmatrix} u(k) \\ y(k) \end{pmatrix} \leq -T^D$

As this case corresponds to the existence of a place between an input transition and an output transition, a simple introduction of an internal transition yields the desired Event Graph and so, the above inequality can be rewritten in forms (3) and (4). Let us note that the writing of relations (3) and (4) assumes that the set of input and output places presents a null initial marking.

Below, we file relations of (2) in the numeric order of the initial marking of places. Upper rows corresponds to null initial marking. The first index in $A_{M_0,.}$ corresponds to the initial marking M_0 while the second one in $A_{.,\Delta}$ corresponds to the shift of numbering Δ relevant to $x(k-\Delta)$.

$$\begin{pmatrix} A_{0,1} & A_{0,0} \\ A_{1,1} & A_{1,0} \end{pmatrix} . \begin{pmatrix} x(k-1) \\ x(k) \end{pmatrix} \leq \begin{pmatrix} -T_0^A \\ -T_1^A \end{pmatrix} \qquad (5)$$

with $A_{0,1} = 0$

2.2. Final model using incidence matrices

The objective is now to obtain the following relation between $x(k-1)$ and $x(k)$ whose form is :

$$\begin{pmatrix} W^+_{x\to x} & -W^-_{x\to x} \end{pmatrix} \begin{pmatrix} x(k-1) \\ x(k) \end{pmatrix} \leq -T_{x\to x} \qquad (6)$$

, relation between $u(k)$ and $x(k)$ whose form is

$$\begin{pmatrix} W^+_{u\to x} & -W^-_{u\to x} \end{pmatrix} \begin{pmatrix} u(k) \\ x(k) \end{pmatrix} \leq -T_{u\to x} \qquad (7)$$

and relation between $x(k)$ and $y(k)$ whose form is

$$\begin{pmatrix} W^+_{x\to y} & -W^-_{x\to y} \end{pmatrix} \begin{pmatrix} x(k) \\ y(k) \end{pmatrix} \leq -T_{x\to y} \qquad (8)$$

Inequation system (6) corresponds to initial marking equal to one and, systems (7) and (8) to null initial marking. So, the variables are connected by input and output incidence matrices which allows a simple calculation. Finally, a corresponding Timed Event Graph defined as follows can be built. The incidence matrices of fundamental equation of marking are :

$W^+ = \begin{pmatrix} W^+_{u\to x} & 0 & 0 \\ 0 & W^+_{x\to x} & 0 \\ 0 & W^+_{x\to y} & 0 \end{pmatrix}$ and $W^- = \begin{pmatrix} 0 & W^-_{u\to x} & 0 \\ 0 & W^-_{x\to x} & 0 \\ 0 & 0 & W^-_{x\to y} \end{pmatrix}$, for a vector of transitions $\begin{pmatrix} u^t & x^t & y^t \end{pmatrix}^t$. The temporisations are $T_{u\to x}$, $T_{x\to x}$ and $T_{x\to y}$ and each internal place (respectively, input/output place) presents an initial marking equal to one (respectively, equal to zero).

As the initial marking of the input/output place is null, the determination of system (8) is immediate. As system (4) expresses the places linking the internal transitions to the output transitions, matrices C_1 and C_0 have a characteristic structure of incidence matrix which allows the simple calculation of $y(k)$ because each row of C_1 and C_0 has a unique non-null coefficient. We have $C_1 = W^+_{x\to y}$, $C_0 = -W^-_{x\to y}$ and $T^C = T_{x\to y}$.

The establishment of system (6) is more difficult as the initial marking of each internal place of system (5) can be equal to zero or one. Particularly, the iterative calculation of $x(k)$ must be made without considering the following relations connected the entries of state vector $x(k)$ for given k

$$A_{0,0}.x(k) \leq -T_0^A \qquad (9)$$

Therefore, the algebraic determination of the incidence matrices and the associated temporisations of system (6) is the goal of the following part.

3. Technique 1 using linear programming

Let us consider transition x_i and systems (3) and (5). Let us analyze the possible effects on date $x_i(k)$ produced by:

- the firing dates of a control transition $u_j(k)$ (case a) and also, produced by
- the firing dates of the upstream transitions of places whose initial marking is one $x_j(k-1)$ (case b). We focus on the earliest behavior and so we want to calculate the minimal effect.

- Case a) The minimal effect is the minimal difference $x_i(k) - u_j(k)$ or $min(c'x)$ where c' is a null row-vector except $c'_i = 1$ et $c'_j = -1$ (i and j are respectively the index of internal transition x_i and input transition u_j) for the following constraints

$$\begin{pmatrix} 0 & A_{0,0} \\ B_1 & B_0 \end{pmatrix} \begin{pmatrix} u(k) \\ x(k) \end{pmatrix} \leq \begin{pmatrix} -T_0^A \\ -T^B \end{pmatrix} \quad (10)$$

  which is classical problem of linear programming. For each pair (x_i, u_j), the resolution of this problem gives the minimal difference ΔT (precisely, $\Delta T_{x_i(k),u_j(k)}$) and we can write relation $x_i(k) - u_j(k) \geq \Delta T$ or the more usual expression $x_i(k) \geq u_j(k) + \Delta T$. In the new graph, it corresponds to a place between input transition u_j and outgoing transition x_i with temporisation ΔT and null initial marking.
- Case b) The minimal effect is the minimal difference $x_i(k) - x_j(k-1)$ or $min(c'x)$ where c' is a null row-vector except $c'_i = 1$ and $c'_j = -1$ (i and j are the indexes of outgoing transition x_i and ingoing transition x_j) for the following constraints

$$\begin{pmatrix} 0 & A_{0,0} \\ A_{1,1} & A_{1,0} \end{pmatrix} \cdot \begin{pmatrix} x(k-1) \\ x(k) \end{pmatrix} \leq \begin{pmatrix} -T_0^A \\ -T_1^A \end{pmatrix} \quad (11)$$

  which is classical problem of linear programming. For each pair (x_i, x_j), the resolution of this problem gives the minimal difference ΔT (precisely, $\Delta T_{x_i(k),x_j(k-1)}$) and we can write relation $x_i(k) - x_j(k-1) \geq \Delta T$ or the more usual expression $x_i(k) \geq x_j(k-1) + \Delta T$. In the new graph, it corresponds to a place between ingoing transition x_j and outgoing transition x_i with temporisation ΔT and initial marking equal to one.

The two procedures allow the consideration of all effects on each transition i produced by control $u(k)$ and the past evolution expressed by $x(k-1)$. The existence of an effect can be graphically explained by the existence of a path between transitions. If the two procedures are repeated for each transition x_i, relations (2) and (3) can be replaced by the new system (6) and (7). In particular, system (9) $A_{0,0}.x(k) \leq -T_0^A$ can now be disregarded as the relevant possible effects on the firing dates of x_i are considered in (6) and (7).
To summarize, the approach generates a new Timed Event Graph which does not contain internal places whose initial marking is null. It contains a set of places whose number is lower than $card(x)(card(x) + card(u))$ as the procedure considers each pair of transitions defined above and their possible connections.

3.1. Example

This example describes the processing manufacture of sheets. Its processing can be described as follows: A sheet is printed, folded and packed. The PERT approach starts from the following table of tasks.

Label	Description of the tasks	Duration	Previous tasks
A	printing	10	-
B	folding	8	A
C	packaging	3	B

Another description based on the transitions is chosen in the approach of Roy [5]. The firing of each transition corresponds to the starting of a task. The relevant table is as follows.

Label	Description of the events	Duration of the started task	Following events
u	Introduction of a raw part in the process	0	x_1
x_1	Beginning of A	10	$x_1(k+1), x_2$
x_2	Beginning of B	8	$x_2(k+1), x_3$
x_3	Beginning of C	3	$x_3(k+1), y$
y	Made part	-	-

From the previous table describing the transition events, we can generate the initial Timed Event Graph.

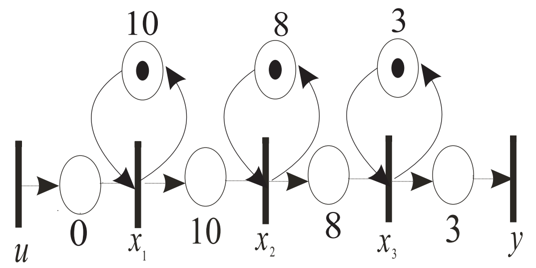

Fig. 1: Initial Timed Event Graph

Let us note that the deduced Event Graph is smaller than the event Graph deduced from the first table of tasks. The relevant matrices are as follows: $A_{0,1} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$, $A_{0,0} = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix}$, $A_{1,1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, $A_{1,0} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$, $B_1 = 1$, $B_0 = \begin{pmatrix} -1 & 0 & 0 \end{pmatrix}$, $C_1 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}$ and $C_0 = -1$. $T_0^A = \begin{pmatrix} 10 \\ 8 \end{pmatrix}$, $T_1^A = \begin{pmatrix} 10 \\ 8 \\ 3 \end{pmatrix}$, $T^B = 0$ and $T^C = 3$.
Let us apply the approach.

a) The minimal difference $x_i(k) - u_j(k)$ is $\begin{pmatrix} 0 \\ 10 \\ 18 \end{pmatrix}$. So,

$W^+_{u\to x} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$, $-W^-_{u\to x} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$ and $-T_{u\to x} = \begin{pmatrix} 0 \\ -10 \\ -18 \end{pmatrix}$

b) The minimal difference $x_i(k) - x_j(k-1)$ is $\begin{pmatrix} 10 & -\infty & -\infty \\ 20 & 8 & -\infty \\ 28 & 16 & 3 \end{pmatrix}$. So, $W^+_{x\to x} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

$-W^-_{x\to x} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & -1 \end{pmatrix}$ and $-T_{x\to x} = \begin{pmatrix} -10 \\ -20 \\ -28 \\ -8 \\ -16 \\ -3 \end{pmatrix}$

As a consequence, input and internal inequalities can be replaced by the deduced relations. The output inequalities are kept. Therefore, the new Timed Event Graph is completely characterized.

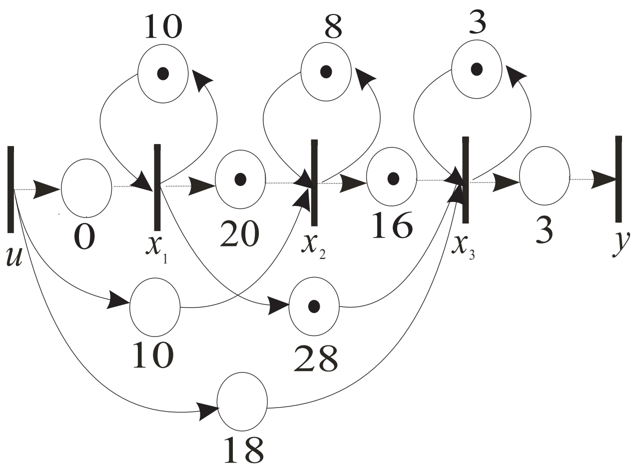

Fig. 2: Final Timed Event Graph

Initial and final models give the same earliest state trajectory.

k	0	1	2	3
x	0	10	20	35
	0	20	30	45
	0	28	38	53
u	-	0	0	35
y	3	31	41	56

4. Dual technique 2

In this part, we propose a second algebraic resolution of the problem based on duality. Before, let us recall the dual problems connected by Theorem of duality.
Primal problem (P): $\min\ y.b$ with $y \in R_+^n$, $y.A = c$ and y real positive.
and
Dual problem (D): $\max c.x$ with $x \in R^m$, $A.x \leq b$ and x real. ■

- In case b) of the previous part, we search the minimal difference $x_i(k) - x_j(k-1)$ or more formally, we solve the problem defined as follows: $min(c'x)$ where c' is a null row-vector except $c'_i = 1$ and $c'_j = -1$ with the constraints
$$\begin{pmatrix} 0 & A_{0,0} \\ A_{1,1} & A_{1,0} \end{pmatrix} . \begin{pmatrix} x(k-1) \\ x(k) \end{pmatrix} \leq \begin{pmatrix} -T_0^A \\ -T_1^A \end{pmatrix} \tag{12}$$
As the minimisation of $c'.x$ corresponds to the maximisation of $-c'.x$, this problem corresponds to (D) where $A = \begin{pmatrix} 0 & A_{0,0} \\ A_{1,1} & A_{1,0} \end{pmatrix}$, $b = \begin{pmatrix} -T_0^A \\ -T_1^A \end{pmatrix}$ et $c = -c'$.
Therefore, the primal problem is as follows.
$\min \quad y. \begin{pmatrix} -T_0^A \\ -T_1^A \end{pmatrix}$ with $y \in R_+^n$, $y. \begin{pmatrix} 0 & A_{0,0} \\ A_{1,1} & A_{1,0} \end{pmatrix} = -c'$ and y real positive.
So, this problem is the maximisation of the non-negative product of y by a vector of temporisations $\begin{pmatrix} T_0^A \\ T_1^A \end{pmatrix}$ under constraints
$$y. \begin{pmatrix} 0 & A_{0,0} \\ A_{1,1} & A_{1,0} \end{pmatrix} = -c' \tag{13}$$
and $y \geq 0$ where c' is a null row-vector except $c'_i = 1$ and $c'_j = -1$ and, i and j are respectively the indexes of outgoing transition x_i and ingoing transition x_j.
- Symmetrically, the primal problem of case a) is as follows.
$\max y. \begin{pmatrix} T_0^A \\ T^B \end{pmatrix}$ with $y \in R_+^m$ and constraints
$$y. \begin{pmatrix} 0 & A_{0,0} \\ B_1 & B_0 \end{pmatrix} = -c' \tag{14}$$
and $y \geq 0$ where c' is a null row-vector except $c'_i = 1$ et $c'_j = -1$ (i and j are respectively the index of internal transition x_i and input transition u_j)

Analysis of the solution The following theorem on incidence matrix allows the analysis of the optimal solution y_{opt}.

Theorem (Chapter 1 in [5])

An incidence matrix is totally unimodular, that is every square sub-matrix extracted from this matrix has a determinant equal to -1, 0 or 1.

Proof
This theorem can directly be deduced from Theorem of Heller-Tompkins 1956 (Annex 2 in [5]). ■

Corollary Incidence matrix $\begin{pmatrix} 0 & A_{0,0} \\ A_{1,1} & A_{1,0} \end{pmatrix}$ is totally unimodular. ■ We can now analyse solution y_{opt}.

Theorem of Hoffman-Kruskal 1956 (Annex 2 in [5])

The optimal solution of the linear programming (P) is integer for any vector c with integer entries and any cost vector b if and only if matrix A is totally unimodular. ■

Corollary

Optimal solution y_{opt} of the linear programming (P) is integer. ■

Moreover, let us recall that the coefficients of y_{opt} are non-negative: $y_{opt} \geq 0$.

Connection with graph theory We now show that approach 2 corresponds to the determination of the greatest paths in graph theory. We have $y.AW = -c'$ with $AW = \begin{pmatrix} 0 & A_{0,0} \\ A_{1,1} & A_{1,0} \end{pmatrix}$

where c'is a null row-vector except $c_e = 1$ (e: end) and $c_s = -1$ (s: start) and, e and s are respectively the indexes of outgoing transition x_e and ingoing transition x_s.

We now consider the existence of a path going from s to e. For each product $y.AW$, each $y_i \neq 0$ selects a row i of AW and the relevant place. We now know that y is integer. Let us consider column of x_s. Product $y.AW_{.,s} = -c_s = 1$ shows that there is place i_1 such that $AW_{i_1,s} = 1$: it corresponds to a positive weighting ($y_{i_1} \geq 1$ as y is an integer) in y. Graphically, it corresponds to place i_1 (with upstream (entering) transition x_s) which is the first place of the path.

Let consider the outgoing transition j_1 of place i_1 and the relevant column x_{j_1}. This transition exists as the graph is a Timed Event Graph. So, $AW_{i_1,j_1} = -1$. If product $y.AW_{.,j_1} = 0$, then there is at least a positive coefficient in column j_1 and a downstream (outgoing) place i_2 of transition j_1. In this step, a pair of coefficients negative and positive of this column are selected. Graphically, this step corresponds to transition j_1 with upstream place i_1 and a downstream (outgoing) place i_2 . As $y_{i1} \geq 1$, there is $y_{i_2} \geq 1$.

As above, place i_2 has an outgoing transition j_2 and we can consider the relevant column and repeat the procedure which generates a new place if product $y.AW_{.,j_2}$ is null. This last one stops when product $y.W_{.,e} = -1$ which correspond to the last transition x_e. So, the coefficients of y are positive or null, and describe a path going from s to e.

We now show that the path is unique. Vector y can choose a set of places which includes the places of the path. However, other places can be chosen. Let us consider a column of transition x_1 where $y.AW_{.,1} = 0$. Suppose that y selects a downstream place of transition x_1 denoted by k_1 and different from $i_1$which does not belong to the path described above. Condition $y.AW_{.,1} = 0$ cannot be satisfied as $y_{i_1} \geq 1$ and $y_{k_1} \geq 1$. The only possibility is the existence of an upstream place of transition x_1 . The structure of Timed Event Graphs entails that an ingoing transition is necessary. As the relevant product must be null, new places are necessary and the procedure can only create a circuit but it is not possible as the primal problem search the maximal weight.

Consequently, integer vector y can only choose a unique path from transition x_s to transition x_e and its coefficients are zero or one. To summarize, the algorithm of linear programming chooses a greatest path among the possible paths from s to e.

5. Primal/Dual connection

Let us introduce notation $X^{ad} = \{x \in R^m | A.x \leq b\}$ which expresses the set of admissible x in the dual problem. Also, $Y^{ad} = \{y \in R^n_+ | y.A = c\}$

Theorem (Chapter 4 in [2]) :

1. If $y \in Y^{ad}$ et $x \in X^{ad}$, then $y.b \geq c.x$
2. If $\bar{y} \in Y^{ad}$, $\bar{x} \in X^{ad}$ and $y.b = c.x$ then $\bar{y}$ and $\bar{x}$ are respectively optimal for (P) and (D);
3. If (P) or (D) has a finite optimal solution, then the same conclusion holds for the other problem and the associated optimal values are equal;
4. If (P) or (D) has an infinite optimal solution, then the other one has no solution. ■

Let us note that approach 1 is based on the dates while approach 2 is based on the weights of the paths between the considered vertices. Using points 1 and 2, the following theorem makes the connections between the two approaches.

Theorem

For problems 1 and 2, we have

$y.T \leq x_i(k) - x_j(k-1)$ and $y_{opt}.T = (x_i(k) - x_j(k-1))_{opt}$ where $T = \begin{pmatrix} T_0^A \\ T_1^A \end{pmatrix}$.

Proof

Point 1 of theorem of duality says that $y.b \geq c.x$ or $y.(-T) \geq -c'.x$. Therefore, $y.T \leq x_i(k) - x_j(k-1)$. Moreover, the associated optimal values are equal (point 2) if the two problems have a finite optimal solution. ■

In other words, the maximal weight of the paths between vertices j and i is equal to the minimal time difference between the same vertices.

6. Appendix 1: Matrix expression of a Timed Event Graph

A Petri net is a pair (GR, M_0), where $GR = (R, V)$ is a bipartite graph with a finite number of nodes (the set V) which are partitioned into the disjoint sets of places P and transitions TR (transitions are denoted t while temporisations are denoted T); R consists of pairs of the form (p_i,t_i) and (t_i,p_i) with $p_i \in P$ and $t_i \in TR$. Initial marking M_0 is a vector of dimension $\mid P \mid$ for which each element $(M_0\ )_i$ is the number of initial tokens in the corresponding place $p_i \in P$. Set ${}^\bullet p$ is the set of input transitions of p and $p^\bullet$ is the set of output transitions of place $p \in P$. Set ${}^\bullet t_i$ (respectively, $t_i^\bullet$) is the set of the input (respectively, output) places of transition $t_i \in TR$.

For a Petri net with $\mid P \mid$ places and $\mid TR \mid$ transitions, the incidence matrix $W = [W_{ij}]$ is an $\mid P \mid \times \mid TR \mid$ matrix of integers and its entry is given by $W_{ij} = W^+_{ij} - W^-_{ij}$ where W^+_{ij} is the weight of the arc from transition j to an output place i and W^-_{ij} is the weight of the arc to transition j from an input place i.

A Petri net is called an Event Graph if each place has exactly one upstream and one downstream transition. Timed Petri nets allow the modelling of discrete event systems with sojourn time constraints of the tokens inside the places. Consistent with dioid $\overline{\mathbb{R}}_{max}$ (see [1]), a temporization defined in R^+ is associated with each place. Each place $p_l \in P$ is associated with a temporization T_l, and, an initial marking denoted m_l.

Well-known in the (max, +) algebra, the "dater" type is considered: each variable $x_i(k)$ represents the date of the k^{th} firing of transition t_i. With a misuse of language, the transition associated with variable $x_i(k)$ will be denoted x_i. If the places follow a FIFO functioning which guarantees that the tokens do not overtake one another, a correct numbering of the events can be carried out. The evolution can be described by the following inequalities expressing relations between the firing dates of transitions. Let us recall that an Event Graph can be considered as a set of subgraphs made up of a place $p_l \in P$ linked with one upstream transition $\{t_j\} = ^\bullet p_l$ and one downstream transition $\{t_i\} = p_l^\bullet$.

Using temporization T_l , the following inequality for each place p_l where $(j, i) = (^\bullet p, p^\bullet)$ can be written:

$T_l + x_j(k - (M_0\ )_l) \leq x_i(k)$ or equivalently, $x_j(k - (M_0\ )_l) - x_i(k) \leq -T_l$.

In the above inequality, weight 1 of $x_j(k - (M_0\ )_l)$ (respectively, -1 of $x_i(k)$) is the weight of the arc going from t_j to place p_l (respectively, the arc going from place p_l to transition t_i) which is equal to W_{lj}^+ (respectively, $-W_{li}^-$).

Let Δ be the maximum number of initial tokens $(M_0\ )_l$ for $p_l \in P$. The set of the previous inequalities which describes a Timed Event Graph, can be expressed with the following form: Column-vector $-T$ is a vector of temporisations where T_l is the temporization of place p_l.

$$\left(\ G \ \right) \times \begin{pmatrix} x(k-\Delta) \\ x(k-\Delta+1) \\ .... \\ x(k-1) \\ x(k) \end{pmatrix} \leq \left(\ -T \ \right) \tag{15}$$

where matrix $G = [G_\Delta G_{\Delta-1} G_{\Delta-2} .... \ G_1 G_0]$ has an order of $(\mid P \mid \times (\Delta + 1). \mid TR \mid)$.

Each place corresponds to a row of G which contains the weights of its entering and outgoing arcs. Matrix G_i for $i \in [1, \Delta]$ contains the weights of the arcs entering the places with i tokens and matrix G_0 contains:

1. the weights of arcs entering the places with no token;
2. the weights of the arc outgoing from each place with negative sign (usually expressed by $-W^-$).

From the above description on the weight of the arcs, the following relation between matrices G_i and incidence matrix W is deduced:

$$W = \sum_{i=0}^{\Delta} G_i$$

. As in classical automatic control, system inequalities (15) can be rewritten on a reduced horizon by increasing the size of the state vector. Such a form will simplify the calculations. Roughly speaking, as a place contains a maximum number of Δ tokens, the general idea is to split each place containing $i \geq 2$ tokens into i places, where each place contains only one token. A systematic procedure is detailed in [6]. Model (15) with $\Delta = 1$ is considered in this paper.

7. Appendix 2: Technique of standardization in (max, +) algebra

In the (max, +) algebra, it is well-known that the earliest trajectory of a Timed Event Graph can be completely described with the following state equation [1]

$$x(k) = A \otimes x(k-1) \oplus B \otimes u(k) \tag{16}$$

where maximization and addition operations are respectively denoted $\oplus$ and $\otimes$. Each variable $x_i(k)$ represents the date of the k^{th} firing of transition x_i. So, the knowledge of the control and the state at the previous step allows the determination of the state at the next step with a simple calculation. The technique of building is as follows. The following model is first deduced from the Timed Event Graph after a possible increase of the state vector

$$x(k) \geq A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B_0 \otimes u(k) \tag{17}$$

and the application of Theorem 4.75 part 1 in [1] gives a simpler form

$$x(k) \geq (A_0)^* \otimes A_1 \otimes x(k-1) \oplus (A_0)^* \otimes B_0 \otimes u(k) \tag{18}$$

where the Kleene star of A is defined by: $A^* = \bigoplus_{i=0}^{+\infty} A^i$. Therefore, the state equation of the automatic control is rediscovered but in (max, +) algebra

$$x(k) \geq A \otimes x(k-1) \oplus B \otimes u(k) \tag{19}$$

with $A = (A_0)^* \otimes A_1$ and $B = (A_0)^* \otimes B_0$. Let us recall that the Kleene star expresses the greatest paths of the associated graph. If we only focus on a model expressing the earliest trajectory, we can consider the corresponding equality. Let us recall that a new Timed Event Graph can be deduced from inequality (19) such that: Each relation $x_i(k) \geq A_{i,j} \otimes x_j(k-1)$ corresponds to an internal place whose initial marking is exactly equal to one (upstream transition is x_j , downstream transition is x_i and temporisation value is $A_{i,j}$); Each relation $x_i(k) \geq B_{i,j} \otimes u_j(k)$ corresponds to an external place whose initial marking is null (upstream transition is u_j , downstream transition is x_i and temporisation value is $B_{i,j}$).

REFERENCES

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

[2] J.C. Culioli. Introduction à l'optimisation Ellipses 1994.

[3] P. Declerck and A. Guezzi and J.-L. Boimond, *Cycle Time of P-time Event Graphs*, 4th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2007), Angers, France, available from http://www.istia.univ-angers.fr/∼declerck, 09-12 may 2007.

[4] Ph. Declerck and A. Guezzi and Cécile Gros. Temps de cycle des Graphes d'Evénements Temporisés et P-temporels. La 5éme Conférence Internationale Francophone d'Automatique CIFA'2008 Bucarest (Roumanie).

[5] M. Gondran and M. Minoux, *Graphs and Algorithms*, John Wiley and Sons, New York, 1984.

[6] A. Guezzi, Ph. Declerck, JL. Boimond, *From monotone inequalities to Model Predictive Control*, Emerging Technologies and Factory Automation, ETFA 2008, IEEE International Conference, 15-18 Sept. 2008.

[7] A. Schrijver, *Theory of linear and integer programming*, John Wiley and Sons, 1987.