# A Sufficient Condition for the Liveness of Weighted Event Graphs[*]

Olivier Marchetti and Alix Munier-Kordon
Laboratoire d'Informatique de Paris 6 - Pôle ASIM
Université Pierre et Marie Curie
4 Place Jussieu, PARIS 75252 Cedex 05, FRANCE
olivier.marchetti@lip6.fr, alix.munier@lip6.fr

2nd November 2004

**Abstract**

The aim of this paper is to develop a sufficient condition of liveness of a weighted event graph (in short WEG) computable in polynomial time. Many industrial problems may be modelled using WEG and a fast polynomial algorithm to decide if a system is live may be interesting in an optimization context. We prove that any unitary WEG may be transformed into a normalized WEG such that the values of the arcs adjacent to any transition depend on the transition. A simple sufficient condition of liveness can be expressed on this new WEG and polynomially computed. We also proved that this condition is necessary for a circuit with two transitions.

**Keywords :** Petri-nets, Liveness, Sheduling, Manufacturing.

## 1   Introduction

Petri nets are a well known formalism for modelling complex systems [1]. Weighted Petri nets, in which every arcs are valued by a number of tokens, have been few studied. Most of the authors [2, 3, 4] are interested in general classes of Petri nets (*i.e.* with no particular assumptions on the structure of the graph). Weighted event graphs are a very simple class of Petri nets since there is no conflict between the firings of the transitions and general tools usually give poor results.

In this paper, we are interested in a central decision problem : is a weighted event graph $G$ is live, which means that every transition may be fired infinitely often ? The main practical interest of this work can be found in the design of industrial or embedded systems.

Indeed, in the context of industrial systems, event graph and weighted event graph may be used for modelling cyclic versions of classical scheduling problems. The model was first introduced by Chrétienne [5]. The authors usually suppose that the graph is live and

---

the problem consists in maximizing the throughput of the systems subject to ressource constraints [6, 7].

However, another classical cyclic scheduling problem consists in minimizing the number of work-in-process. For event graph, this corresponds to minimizing the number of tokens in some circuits of the system. Tokens can be viewed as products simultaneously treated in the production chain, and due to the storage cost, this number must be minimum. Notice that, in this case, the graph is live iff there is at least one token in every circuit. So, the minimization problem is in $\mathcal{NP}$, which allows many authors to develop efficient optimization algorithms to solve it [8, 9, 10].

We prove in this paper that for an important class of weighted event graph called unitary graph, we can polynomially transform the graph in a such way that the number of tokens remains constant in every circuits. It means that the minimization of the work-in-process may have a practical meaning even for this class of nets.

In the context of the design of embedded systems, the main model considered is Synchronous Data-flow introduced by Lee and Messerschmitt [11, 12]. A set of processes communicates data using FIFO files (buffers). In fact, the constraints between the processes can be modelled using a weighted event graph : transitions correspond to the processes and the places to the buffers. Nowadays, the cost of the memories is very important for the design of these systems, so the designers try to minimize the size of the buffers [13]. We observe that this problem is equivalent to minimize the overall number of tokens of a weighted event graph. The decision problem associated is, also in this case, the liveness of the graph for a given initial marking.

For both classes of applications, authors solve the liveness problem of a unitary graph using a pseudo-polynomial algorithm [14, 15, 16] . Indeed, it is proved in [17] that any unitary weighted event graph is equivalent to an event graph of pseudo-polynomial size. The liveness is checked on this graph, which yields to a pseudo-polynomial algorithm. So, if the values on the arcs are too important, these algorithms cannot be used.

In this paper, our purpose is to develop a polynomial algorithm to decide if $G$ is live. This algorithm must be fast (not pseudo-polynomial) to be used frequently in heuristics or Branch and Bound methods. The complexity of the liveness of a unitary WEG is still open, and we were not able to answer to this interesting fundamental question. However we developed an original (to our knowledge) sufficient condition of liveness based on an original property of unitary graphs which can be computed polynomially.

This paper is organized as follow : section 2 is devoted to the description of the problem. In section 3, we prove some basic preliminary results on weighted event graphs. Section 4 deals with the normalization of a weighted event graph : we prove that any unitary graph may be transformed polynomially in such a way that all the values of the arcs adjacent to a given transition are equal. Even if this transformation seems to be independent from the expansion, we show that there is a strong relation between the number of duplicates and the values on the arcs. In section 5, we prove a sufficient condition of liveness and we show that it is necessary for any circuit with two transitions. Finaly, we conclude with some perspectives in section 6.

## 2   Basic definitions and notations

Let us consider a weighted event graph $G(P,T)$ (WEG in short) given by a set of transitions $T = \{t_1, ..., t_n\}$ and a set of places $P = \{p_1, ..., p_m\}$. Every place $p \in P$ is defined between two transitions $t_i$ and $t_j$, so we note $p = (t_i, t_j)$ (see figure 1). For any transition $t \in T$, we set :

$$\mathcal{P}^+(t) = \{p = (t, t') \in P, t' \in T\}$$

$$\mathcal{P}^-(t) = \{p = (t', t) \in P, t' \in T\}$$

The arcs $(t_i, p)$ and $(p, t_j)$ are valued by two strictly positive integers denoted respectively by $w(p)$ and $v(p)$ and called the marking functions. We also denote by $M_0(p)$ the initial marking of the place $p$.
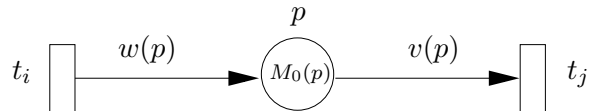


Figure 1: A place $p$ with two transitions $t_i$ and $t_j$

For any firing of the transition $t_i$ (resp. $t_j$), $w(p)$ (resp. $v(p)$) tokens are placed to (resp. removed from) the place $p$. So, if $\nu_i$ (resp. $\nu_j$) is the number of firings of the transition $t_i$ (resp. $t_j$), the number of tokens in the place $p$ is :

$$M(p) = M_0(p) + \nu_i w(p) - \nu_j v(p)$$

For any $\nu \in \mathbb{N}^\star$ and for any $t \in T$, $< t, \nu >$ denotes the $\nu$th firing of $t$.

If $w(p) = v(p) = 1$ for any place $p \in P$, then $G$ is an event graph. For a sake of simplicity, we set for any place $p \in P$ :

1. $gcd(w(p), v(p)) = gcd_p$ the great common divisor of the integers $w(p)$ and $v(p)$;

2. $lcm(w(p), v(p)) = lcm_p$ the least common multiplier of the integers $w(p)$ and $v(p)$.

Let us consider a path $\mu$ of $G$ defined as a sequence of $n$ places such that $\mu = \{p_1 = (t_1, t_2), p_2 = (t_2, t_3), \ldots, p_n = (t_{n-1}, t_n)\}$. The weight of $\nu$, denoted by $W(\mu)$, is defined as :

$$W(\mu) = \prod_{p \in P \cap \mu} \frac{w(p)}{v(p)}$$

Several authors [17, 2] noticed that a necessary condition of liveness of a WEG $G$ is that every circuit has a weight equal to or superior than 1. This condition is trivially not sufficient : for instance, if we consider an event graph (*i.e.* $w(p) = v(p)$ for every place $p \in P$), then this condition is always fulfilled and is not sufficient to decide about the liveness.

In this paper, we reduce our study to strongly connected WEG such that every circuit has a unitary weight. This class of WEG, called unitary graphs, was first introduced in [17] and is usually sufficiently large to model the real-life problems. Indeed, any not unitary

3

strongly connected graph $G$ will have unbounded places. Since tokens model products (in an assembly line) or data (in a computer system), their number is necessary bounded in practical applications. It means that if the graph obtained is not unitary, there was certainly a problem in the design of the line or in the design of the system.

# 3 Preliminaries

We prove in this section some basic technical properties on the markings of the places.

Let $p = (t_i, t_j) \in P$. We say that $p$ induces a precedence constraint between $< t_i, \nu_i >$ and $< t_j, \nu_j >$ iff :

**Condition 1** $< t_j, \nu_j >$ can be done after $< t_i, \nu_i >$;

**Condition 2** $< t_j, \nu_j - 1 >$ can be done before $< t_i, \nu_i >$ but not $< t_j, \nu_j >$.

**Lemma 3.1.** *A place $p = (t_i, t_j) \in P$ models a precedence constraint between the $\nu_i$th firing of $t_i$ and the $\nu_j$th firing of $t_j$ iff :*

$$w(p) > M_0(p) + w(p)\nu_i - v(p)\nu_j \geq \max(w(p) - v(p), 0)$$

*Proof.* A place $p = (t_i, t_j) \in P$ models a precedence constraint between $< t_i, \nu_i >$ and the $< t_j, \nu_j >$ iff Conditions 1 and 2 hold.

1. Condition 1 is equivalent to

$$M_0(p) + w(p)\nu_i - v(p)\nu_j \geq 0$$

2. Condition 2 is equivalent to

$$v(p) > M_0(p) + w(p)(\nu_i - 1) - v(p)(\nu_j - 1) \geq 0$$

Combining these two inequalities, we get the inequality required. $\square$

One can observe that the deadlock freeness of a strongly connected weighted event graph is equivalent to the liveness : indeed, the liveness is equivalent to the non existence of a circuit in the developed graph (*i.e* the infinite graph modelling all the precedence constraints on the firings of the transitions). These circuits correspond exactly to deadlocks.

## 3.1 Useful tokens

**Lemma 3.2.** *The initial marking $M_0(p)$ of any place $p = (t_i, t_j)$ may be replaced by $M_0^*(p) = \left\lfloor \frac{M_0(p)}{gcd_p} \right\rfloor .gcd_p$ without any influence on the precedence constraints induced by $p$.*

*Proof.* Let $p = (t_i, t_j)$ be a place of $G$. We set $A$ (resp. $B$) to the set of precedence constraints induced by an initial marking of $M_0(p)$ (resp. $M_0^*(p)$). We prove that $A = B$.

Using the Euclidian division of $M_0(p)$ by $gcd_p$, we get :

$$M_0(p) = M_0^*(p) + R_{gcd}(M_0(p))$$
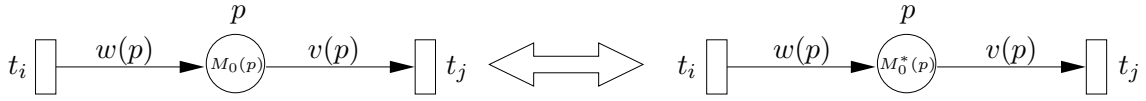
where $R_{gcd}(M_0(p)) \in \{0, ..., gcd_p - 1\}$.

4

Figure 2: Notion of useful tokens

$A \subset B$ Let us consider a precedence constraint of $A$ between $< t_i, \nu_i >$ and $< t_j, \nu_j >$. By lemma 3.1,

$$w(p) > M_0(p) + w(p)\nu_i - v(p)\nu_j \geq \max(w(p) - v(p), 0)$$

So, we get :

$$w(p) > M_0^*(p) + R_{gcd}(M_0(p)) + w(p)\nu_i - v(p)\nu_j \geq \max(w(p) - v(p), 0)$$

Clearly,
$$w(p) > M_0^*(p) + w(p)\nu_i - v(p)\nu_j$$

Now, since $M_0^*(p) + w(p)\nu_i - v(p)\nu_j \equiv 0(gcd_p)$, $\max(w(p) - v(p), 0) \equiv 0(gcd_p)$ and $R_{gcd}(M_0(p)) \in \{0, ..., gcd_p - 1\}$, we get :

$$M_0^*(p) + w(p)\nu_i - v(p)\nu_j \geq \max(w(p) - v(p), 0)$$

and the precedence constraint between $< t_i, \nu_i >$ and $< t_j, \nu_j >$ belongs also to $B$.

$B \subset A$ Let us consider now a precedence constraint of $B$ between $< t_i, \nu_i >$ and $< t_j, \nu_j >$. By lemma 3.1,

$$w(p) > M_0^*(p) + w(p)\nu_i - v(p)\nu_j \geq \max(w(p) - v(p), 0)$$

Clearly,

$$M_0^*(p) + R_{gcd}(M_0(p)) + w(p)\nu_i - v(p)\nu_j \geq \max(w(p) - v(p), 0)$$

Now, since $M_0^*(p) + \nu_i.w(p) - \nu_j.v(p) \equiv 0(gcd_p)$, we get

$$w(p) - gcd_p \geq M_0^*(p) + w(p)\nu_i - v(p)\nu_j$$

As $R_{gcd}(M_0(p)) < gcd_p$,

$$w(p) > M_0^*(p) + R_{gcd}(M_0(p)) + w(p)\nu_i - v(p)\nu_j \geq \max(w(p) - v(p), 0)$$

and the precedence constraint between $< t_i, \nu_i >$ and $< t_j, \nu_j >$ belongs also to $A$.

$\square$

In the rest of the paper, we assume that the initial marking of any place $p$ is a multiple of $gcd_p$.

## 3.2 Equivalent places

Two places $p_1 = (t_i, t_j)$ and $p_2 = (t_i, t_j)$ are said equivalent if they induce the same precedence constraints between the firings of their adjacent transitions.

**Lemma 3.3.** *Two places $p_1 = (t_i, t_j)$ and $p_2 = (t_i, t_j)$ such that for $\frac{w(p_2)}{w(p_1)} = \frac{v(p_2)}{v(p_1)} = \frac{M_0(p_2)}{M_0(p_1)} = \Delta \in \mathbb{N}^\star$ are equivalent.*

*Proof.* Let us denote by $A_1$ (resp. $A_2$) the set of precedence relations between the firings of $t_i$ and $t_j$ induced by $p_1$ (resp. $p_2$). Let us consider a precedence relation $a \in A_1$ defined between $< t_i, \nu_i >$ and $< t_j, \nu_j >$. Then, by lemma 3.1, we get :

$$w(p_1) > M_0(p_1) + w(p_1).\nu_i - v(p_1).\nu_j \geqslant \max(w(p_1) - v(p_1), 0)$$

$$\Updownarrow \times \Delta$$

$$\Delta.w(p_1) > \Delta.[M_0(p_1) + w(p_1).\nu_i - v(p_1).\nu_j] \geqslant \Delta.\max(w(p_1) - v(p_1), 0)$$

$$\Updownarrow$$

$$w(p_2) > M_0(p_2) + w(p_2).\nu_i - v(p_2).\nu_j \geqslant \max(w(p_2) - v(p_2), 0)$$

These latter inqualities are equivalent to $a \in B$, which completes the proof. $\qquad\square$

# 4 Normalization of a weigthed event graph

We prove in this section that any unitary WEG $G$ may be transformed into an equivalent normalized graph, *i.e.* a WEG such that, for every transition $t_i$, the marking functions of its adjacent arcs are equal. Firstly, we define formally the notion of normalized graph and we show that the computation of the values of an equivalent normalized graph may be solved using a shortest path algorithm of a simple valued graph $\mathcal{G}$. Then, we prove that the value of any circuit of $\mathcal{G}$ is null, so that every unitary graph may be normalized. Finaly, we study the relation between expansion and normalization of a unitary graph.

**Definition 4.1.** *A transition $t_i$ is called normalized iff there exists $Z_i \in \mathbb{N}^\star$ such that :*

$$
\begin{array}{ccccccc}
\forall (p_1, p_2) & \in & \mathcal{P}^+(t_i) & \times & \mathcal{P}^-(t_i), & w(p_1) = v(p_2) = Z_i \\
\forall (p_1, p_2) & \in & \mathcal{P}^+(t_i) & \times & \mathcal{P}^+(t_i) & w(p_1) = w(p_2) = Z_i \\
\forall (p_1, p_2) & \in & \mathcal{P}^-(t_i) & \times & \mathcal{P}^-(t_i) & v(p_1) = v(p_2) = Z_i
\end{array}
$$

*A graph $G$ is said normalized iff all its transitions are normalized.*

By lemma 3.3, each place $p_a \in P$ can be replaced by a place $p'_a$ obtained by multiplying $w(p_a)$, $v(p_a)$ and $M_0(p_a)$ by an integer $\alpha_a$. The idea is to build, for any unitary WEG $G = (T, P)$, another unitary WEG $G' = (T, P')$ using this last property and such that every transition is normalized.

Let us consider a transition $t_i \in T$. Then, by definition 4.1, this transition can be normalized iff there exists a vector $\alpha \in \{\mathbb{N}^{\star+}\}^{|P|}$ such that :

**vw** For any couple $(p_a, p_b) \in (\mathcal{P}^-(t_i), \mathcal{P}^+(t_i))$,

$$\alpha_a.v(p_a) = \alpha_b.w(p_b) \quad \Leftrightarrow \quad \begin{cases} \frac{\alpha_a}{\alpha_b} & \leqslant & \frac{w(p_b)}{v(p_a)} \\ \frac{\alpha_b}{\alpha_a} & \leqslant & \frac{v(p_a)}{w(p_b)} \end{cases}$$

**ww** For any couple $(p_a, p_b) \in (\mathcal{P}^+(t_i), \mathcal{P}^+(t_i))$,

$$\alpha_a.w(p_a) = \alpha_b.w(p_b) \quad \Leftrightarrow \quad \begin{cases} \frac{\alpha_a}{\alpha_b} & \leqslant & \frac{w(p_b)}{w(p_a)} \\ \frac{\alpha_b}{\alpha_a} & \leqslant & \frac{w(p_a)}{w(p_b)} \end{cases}$$

**vv** For any couple $(p_a, p_b) \in (\mathcal{P}^-(t_i), \mathcal{P}^-(t_i))$,

$$\alpha_a.v(p_a) = \alpha_b.v(p_b) \quad \Leftrightarrow \quad \begin{cases} \frac{\alpha_a}{\alpha_b} & \leqslant & \frac{v(p_a)}{v(p_b)} \\ \frac{\alpha_b}{\alpha_a} & \leqslant & \frac{v(p_b)}{v(p_a)} \end{cases}$$

In order to modelize this set of constraints, we build a valued directed graph $\mathcal{G} = (V \cup \{s\}, E)$ as follows :

- Any vertex $a \in V$ corresponds to a place $p_a \in P$;

- Every arc $e = (a, b) \in E$ is associated with a couple of adjacent places $(p_a, p_b)$. Let $t_e$ be their common transition. If we set

$$\beta(t_e, p_a) = \begin{cases} w(p_a) & \text{if} & p_a \in \mathcal{P}^+(t_e) \\ v(p_a) & \text{otherwise} \end{cases}$$

then, we can associate to $e$ the following constraint :

$$\log \alpha_b - \log \alpha_a \leqslant \mathcal{B}_{(a,b)}$$

with $\mathcal{B}_{(a,b)} = \log \frac{\beta(t_e, p_a)}{\beta(t_e, p_b)}$. Arc $(a, b) \in E$ is then valued by $\mathcal{B}_{(a,b)}$;

- $\forall a \in V$, we add the arcs $(s, a)$ valued by 0.

For any path $\mu$ of $\mathcal{G}$, we defined the value of the path as :

$$\mathcal{B}(\mu) = \sum_{e \in \mu \cap E} \mathcal{B}_e$$

Using Bellman-Ford algorithm [18], $\alpha \in \{\mathbb{N}^{\star+}\}^{|P|}$ can be polynomially computed iff $\mathcal{G}$ has no circuit of strictly negative value.

**Lemma 4.1.** *Every circuit of $\mathcal{G}$ with $q = 2$ vertices has a null value.*

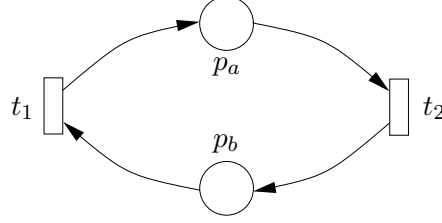*Proof.* We can distinguish three cases which can lead to a circuit of size $q = 2$ in $\mathcal{G}$ :

Figure 3: A normalized graph with two places

1. Two places $p_a = (t_1, t_2)$ and $p_b = (t_2, t_1)$ form a circuit in $G$ (as shown on figure 3). In $\mathcal{G}$, there are two arcs $(a, b)$ valued by respectively $\log \frac{w(p_a)}{v(p_b)}$ and by $\log \frac{v(p_a)}{w(p_b)}$. Since $G$ is unitary,
$$\frac{w(p_a)}{v(p_a)} \cdot \frac{w(p_b)}{v(p_b)} = 1$$
   so, these two arcs have the same value. On the same way, the two arcs $(b, a)$ are valued by $\log \frac{v(p_b)}{w(p_a)}$, so the value of the corresponding circuits of $\mathcal{G}$ is null.

2. Two places $p_a = (t_1, t_2)$ and $p_b = (t_1, t_2)$ in $G$ share the same initial and final transition (as shown on figure 4). In $\mathcal{G}$, there are two arcs $(a, b)$ valued by respectively $\log \frac{w(p_a)}{w(p_b)}$ and by $\log \frac{v(p_a)}{v(p_b)}$.

   Now, since $G$ is a strongly connected unitary graph, there is a path $\nu$ from $t_2$ to $t_1$ such that
$$W(\nu) \cdot \frac{w(p_a)}{v(p_a)} = W(\nu) \cdot \frac{w(p_b)}{v(p_b)} = 1$$

   So, the two arcs $(a, b)$ of $\mathcal{G}$ have the same value. On the same way, the arcs $(b, a)$ are valued by $\log \frac{w(p_b)}{w(p_a)}$, so the value of the corresponding circuits of $\mathcal{G}$ is null.
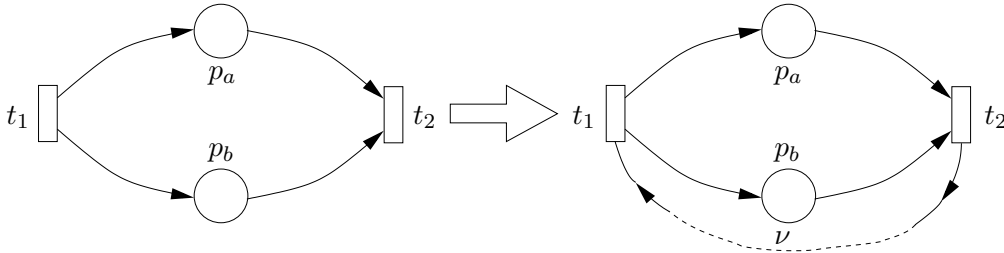


Figure 4: A normalized graph with two places

3. Two places $p_a$ and $p_b$ in $G$ share the same initial transition or the same final transition (as shown on figure 5).

   If they share the same final transition, then we get in $\mathcal{G}$ the arcs $(a, b)$ and $(b, a)$ valued respectively by $\log \frac{v(p_a)}{v(p_b)}$ and $\log \frac{v(p_b)}{v(p_a)}$. So, the value of the corresponding circuit of $\mathcal{G}'$ is null.

8

If they share the same initial transition, then we get in $\mathcal{G}'$ the arcs $(a, b)$ and $(b, a)$ valued respectively by $\log \frac{w(p_a)}{w(p_b)}$ and $\log \frac{wp_b)}{w(p_a)}$. So, the value of the corresponding circuit of $\mathcal{G}'$ is also null.
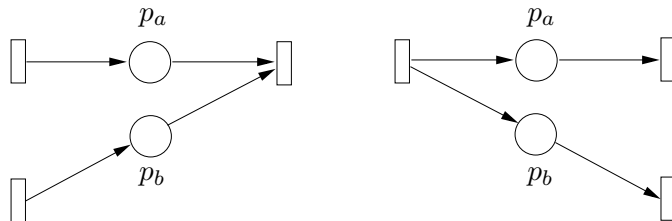


Figure 5: A normalized graph with two places

$\square$

**Lemma 4.2.** *Let be a vertex $i$ of the circuit $\mathcal{C} = (1, \ldots, q, 1)$ with $q > 2$ such that arcs $e_i = (i-1, i)$ and $e_{i+1} = (i, i+1)$ correspond to the same transition of $G$ (i.e. $t_{e_i} = t_{e_{i+1}}$). Then, $\mathcal{C}' = (1, 2, \ldots, i-1, i+1, \ldots, q, 1)$ is also a circuit of $\mathcal{G}$ and $\mathcal{B}(\mathcal{C}') = \mathcal{B}(\mathcal{C})$.*

*Proof.* Following the assumption of the lemma, let us denote by $p_{i-1}$, $p_i$ and $p_{i+1}$ the places associated respectively with the vertices $i-1$, $i$ and $i+1$ of $\mathcal{G}$. By construction of $\mathcal{G}$ and since $t_{e_i} = t_{e_{i+1}}$, we get :

$$
\begin{aligned}
\mathcal{B}_{e_i} + \mathcal{B}_{e_{i+1}} &= \log \frac{\beta(t_{e_i}, p_{i-1})}{\beta(t_{e_i}, p_i)} + \log \frac{\beta(t_{e_i}, p_i)}{\beta(t_{e_i}, p_{i+1})} \\
&= \log \frac{\beta(t_{e_i}, p_{i-1})}{\beta(t_{e_i}, p_{i+1})}
\end{aligned}
$$

Since $p_{i-1}$ and $p_{i+1}$ are adjacent to $t_{e_i}$, there exists an arc $(i-1, i+1)$ in $\mathcal{G}$ valued by $\log \frac{\beta(t_{e_i}, p_{i-1})}{\beta(t_{e_i}, p_{i+1})}$ (see figure 6). So, the sub-circuit $\mathcal{C}' = (1, 2, \ldots, i-1, i+1, \ldots, q, 1)$ exists
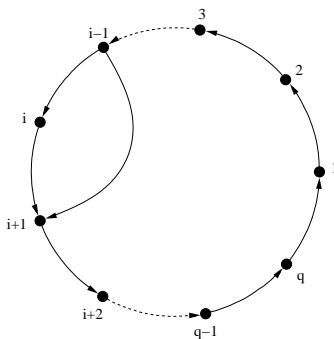


Figure 6: Illustration of a special case.

and has the same value as $\mathcal{C}$. $\square$

**Theorem 4.1.** *Every circuit of $\mathcal{G}$ has a null value.*

*Proof.* Let $\mathcal{C} = (1, ..., q, 1)$ be a circuit of $\mathcal{G}$. Applying lemma 4.2, we can suppose without loss of generality that $\mathcal{C}$ fulfills one of the following properties :

9

**Property 1** $q = 2$;

**Property 2** $q > 2$ and two consecutive edges of $\mathcal{C}$ are associated with different transitions :
$t_{e_1} \neq t_{e_q}$ and $\forall i \in \{1, ..., q-1\}$, $t_{e_i} \neq t_{e_{i+1}}$.

By lemma 4.1, if Property 1 holds, then $\mathcal{C}$ is null valued. So, we suppose now that $\mathcal{C}$ verifies Property 2. We denote by $e_i = (i-1, i)$ for $i \in \{2, ..., q\}$ and $e_1 = (q, 1)$. By
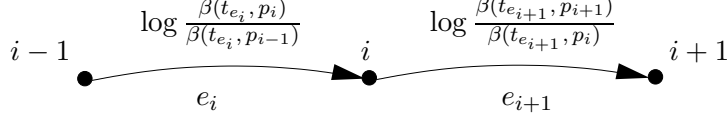
$$i-1 \quad \overset{\log \frac{\beta(t_{e_i}, p_i)}{\beta(t_{e_i}, p_{i-1})}}{\bullet \longrightarrow \atop e_i} \quad i \quad \overset{\log \frac{\beta(t_{e_{i+1}}, p_{i+1})}{\beta(t_{e_{i+1}}, p_i)}}{\bullet \longrightarrow \atop e_{i+1}} \quad i+1$$

Figure 7: Two weighted arcs of graph $\mathcal{G}$

definition of $\mathcal{G}$,

$$\mathcal{B}(\mathcal{C}) = \sum_{i=2}^{q} \log \frac{\beta(t_{e_i}, p_i)}{\beta(t_{e_i}, p_{i-1})} + \log \frac{\beta(t_{e_1}, p_1)}{\beta(t_{e_1}, p_q)}$$

$$= \sum_{i=1}^{q-1} \log \frac{\beta(t_{e_i}, p_i)}{\beta(t_{e_{i+1}}, p_i)} + \log \frac{\beta(t_{e_q}, p_q)}{\beta(t_{e_1}, p_q)}$$

Now, if $p_i$ is the place associated with $i$, then $p_i$ is adjacent to $t_{e_i}$ and $t_{e_{i+1}}$ and $t_{e_i} \neq t_{e_{i+1}}$. Then, we build a circuit $C$ of $G$ as follows :

- The transitions $t_{e_1}, ..., t_{e_q}$ belong to $C$;

- If $p_i = (t_{e_i}, t_{e_{i+1}})$ we get $\dfrac{\beta(t_{e_i}, p_i)}{\beta(t_{e_{i+1}}, p_i)} = \dfrac{w(p_i)}{v(p_i)} = W(p_i)$. Here, we add to $C$ the place $p_i$;

- If $p_i = (t_{e_{i+1}}, t_{e_i})$ then $\dfrac{\beta(t_{e_i}, p_i)}{\beta(t_{e_{i+1}}, p_i)} = \dfrac{v(p_i)}{w(p_i)}$. Since $G$ is a stongly connected unitary graph, there is a path $\mu_i$ from $t_{e_i}$ to transition $t_{e_{i+1}}$ such that $W(\mu_i) = \dfrac{w(p_i)}{v(p_i)}$. We add $\mu_i$ to $C$.
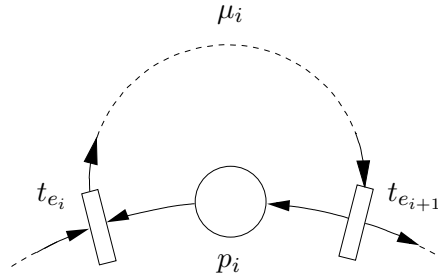
$$\mu_i$$

Figure 8: Construction of a ciruit of $G$

Now, setting

$$U_1 = \{i \in \{1, ..., q\}/p_i = (t_{e_i}, t_{e_{i+1}})\} \text{ and } U_2 = \{1, ..., q\} - U_1$$

10

we get :

$$
\begin{aligned}
W(C) &= \sum_{i \in U_1} \log \frac{w(p_i)}{v(p_i)} + \sum_{i \in U_2} \log W(\mu_i) \\
&= \sum_{i \in U_1} \log \frac{\beta(t_{e_i}, p_i)}{\beta(t_{e_{i+1}}, p_i)} + \sum_{i \in U_2} \log \frac{\beta(t_{e_i}, p_i)}{\beta(t_{e_{i+1}}, p_i)} \\
\\
&= \beta(\mathcal{C})
\end{aligned}
$$

As $G$ is unitary, $W(C) = 0$ so $\beta(\mathcal{C}) = 0$. $\qquad\square$

Surprisingly, there is a strong relation between the normalization and the expansion of a unitary WEG. This relation is another way to prove that any unitary graph is normalizable. Let us recall the definition of an expansible WEG [17] :

**Definition 4.2.** *Let $G$ be a WEG. $G$ is expansible iff there exists $(N_1, ..., N_n) \in \{\mathbb{N}^*\}^n$ such that :*

$$
\forall p = (t_i, t_j) \in P, \frac{N_i}{N_j} = \frac{v(p)}{w(p)}
$$

**Theorem 4.2.** *Let $G$ be a WEG. $G$ is expansible iff $G$ is normalizable.*

*Proof.* $A \Rightarrow B$ If $G$ is expansible, then there exists $(N_1, ..., N_n) \in \{\mathbb{N}^*\}^n$ such that $\forall p = (t_i, t_j) \in P, \frac{N_i}{N_j} = \frac{v(p)}{w(p)}$.

We set $N = lcm_{i \in \{1,...,n\}}(N_i)$, $\lambda = lcm_{a \in \{1,...,m\}}(w(p_a), v(p_a))$ and $\forall i \in \{1, ..., n\}, Z_i = \frac{N.\lambda}{N_i}$. For every place $p_a = (t_i, t_j)$, we set $\alpha_a = \frac{Z_i}{w(p_a)}$. We prove that the vector $\alpha$ is a solution to the previous system :

1. By definition of $Z_i$, $\alpha_a \in \mathbb{N}^\star$.

2. For any couple $(p_a, p_b) \in \mathcal{P}^-(t_i) \times \mathcal{P}^+(t_i)$,

$$
\alpha_b w(p_b) = Z_i = \frac{N\lambda}{N_i}
$$

   Setting $p_a = (t_j, t_i)$, we get :

$$
\frac{N\lambda}{N_i} = \frac{N\lambda}{N_j} \frac{v(p_a)}{w(p_a)} = \frac{Z_j}{w(p_a)} v(p_a) = \alpha_a v(p_a)
$$

3. For any couple $(p_a, p_b) \in \mathcal{P}^+(t_i) \times \mathcal{P}^+(t_i)$,

$$
\alpha_a w(p_a) = Z_i = \alpha_b w(p_b)
$$

4. For any couple $(p_a, p_b) \in \mathcal{P}^-(t_i) \times \mathcal{P}^-(t_i)$ with $p_a = (t_j, t_i)$,

$$
\alpha_a v(p_a) = \frac{Z_j}{w(p_a)} v(p_a) = \frac{N\lambda}{N_j} \frac{v(p_a)}{w(p_a)} = \frac{N\lambda}{N_i} = Z_i
$$

   So,

$$
\alpha_a v(p_a) = Z_i = \alpha_b v(p_b)
$$

$B \Rightarrow A$ Conversely, let us suppose that $\alpha$ is a solution to the previous system. Then, $\forall (p_a, p_b) \in \mathcal{P}^-(t_i) \times \mathcal{P}^+(t_i)$, $\alpha_a v(p_a) = \alpha_b w(p_b)$. Let $Z_i$ denotes this value.

Now, setting $Z = lcm_{i \in \{1,...,n\}}(Z_i)$, we prove that $\forall i \in \{1,...,n\}$, $N_i = \frac{Z}{Z_i}$ verifies the equations of the expansion. Indeed, for any place $p_a = (t_i, t_j) \in P$,

$$\frac{N_i}{N_j} = \frac{Z_j}{Z_i} = \frac{v(p_a)}{w(p_a)}$$

which completes the proof.

$\square$

**Corollary 4.1.** *Let $G$ be a strongly connected WEG. $G$ is normalized iff $G$ is unitary. Moreover, there exists $K \in \mathbb{N}^\star$ such that, $\forall i \in \{1,...,n\}$,*

$$N_i.Z_i = K$$

*Proof.* Let $G$ be a strongly connected WEG. It is proved in [17] that $G$ is a unitary iff $G$ is expansible. By theorem 4.2, we get the first part of the corollary. The relationship between vectors $(Z_i)_{i \in \{1,..,n\}}$ and $(N_i)_{i \in \{1,..,n\}}$ is in the proof of this last theorem. $\square$

We illustrate by an example the normalization's method and the strong link between normalization and expansion (see figure 9). One can check that $\forall i \in \{1,...,5\}$, $N_i.Z_i = 2520$ with the following expansion vector $^t N = (84, 70, 35, 70, 40)$.
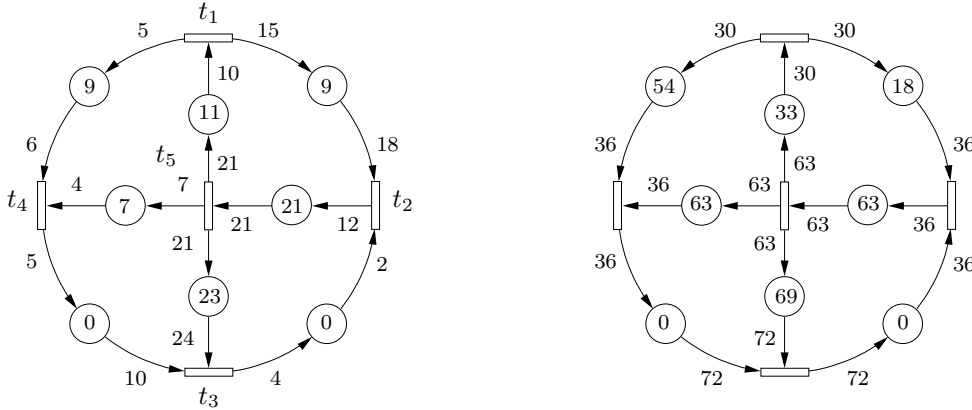


Figure 9: On the left, graph $G$ is non-normalized. Let us denote that the number of tokens on each place $p$ is a multiple of $gcd_p$. We show the associated normalized graph on the right side.

# 5 Sufficient condition for the liveness of a normalized WEG

We consider here a normalized WEG $G$. Firstly, we prove a sufficient condition on the liveness of $G$. Then, we show that this is a necessary condition for any circuit with two transitions. Finaly, we give an example to show that this condition is not necessary.

**Lemma 5.1.** *Let $G$ be a normalized WEG. Then, the number of tokens of every circuit remains constant.*

*Proof.* Let $C = (t_1, ..., t_q, t_1)$ be a circuit of $G$ and let $\nu_i$, $i \in \{1, ..., q\}$, be the number of firings of the transition $t_i$. We set $p_i = (t_i, t_{i+1})$ for $i \in \{1, ..., q-1\}$ and $p_q = (t_q, t_1)$. Then,

$$\sum_{i=1}^{q} M(p_i) = \sum_{i=1}^{q} \left( M_0(p_i) + \nu_i w(p_i) - \nu_{i+1} v(p_i) \right)$$

Now, since $G$ is normalized, $v(p_i) = w(p_{i+1})$ for $i \in \{1, .., q-1\}$ and $v(p_q) = w(p_1)$. So, $\sum_{i=1}^{q} \left( \nu_i w(p_i) - \nu_{i+1} v(p_i) \right) = 0$ and we proved the lemma. $\qed$

**Theorem 5.1 (Sufficient condition).** *Let $G$ be a normalized WEG. $G$ is live if for all circuit $C$ of $G$ :*

$$\sum_{p \in C \cap P} M_0(p) > \sum_{p \in C \cap P} (v(p) - gcd_p)$$

*Proof.* By contradiction, let us suppose that $G$ is not live and such that for all circuit $C$ of $G$, the inequality is true. Since $G$ is not live, every possible sequence of firings leads to a deadlock. It means that there is a circuit $C_{Dl}$ in $G$ such that it is not possible to fire any transition because of a lack of tokens on every place of this circuit, so :

$$\forall p \in C_{Dl} \cap P, M(p) < v(p)$$

As the number of tokens present in a place $p$ can be considered as a multiple of $gcd_p$ (cf. lemma 3.2), we get :

$$\forall p \in C_{Dl} \cap P, M(p) \leq v(p) - gcd_p$$

Summing these inequalities, we obtain that :

$$\sum_{p \in C_{Dl} \cap P} M(p) \leq \sum_{p \in C_{Dl} \cap P} (v(p) - gcd_p)$$

Since $G$ is normalized, by lemma 5.1, $\sum_{p \in C_{Dl} \cap P} M(p) = \sum_{p \in C_{Dl} \cap P} M_0(p)$ and we get a contradiction. $\qed$
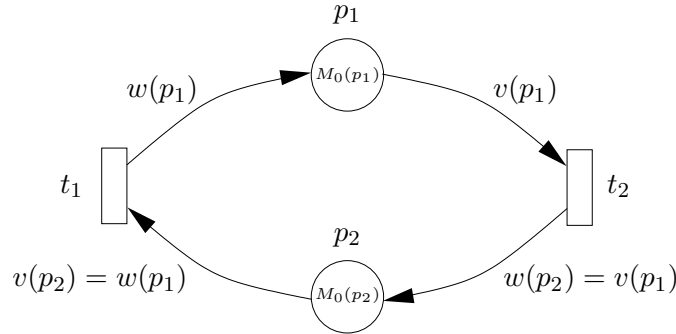


Figure 10: A normalized graph with two places.

**Theorem 5.2 (Special case).** *Let $C$ be a normalized circuit composed by two places $p_1$ and $p_2$ and two different transitions (see figure 10). Then, $C$ is live iff*

$$M_0(p_1) + M_0(p_2) > v(p_1) + v(p_2) - 2.gcd_{p_1}$$

13

*Proof.* Since $C$ is normalized, $w(p_1) = v(p_2)$ and $v(p_1) = w(p_2)$.

- If $w(p_1) = w(p_2) = w$, then, $gcd_{p_1} = gcd_{p_2} = w$. In this case, by lemma 3.3, the circuit $C$ is equivalent to an event graph with an initial marking for place $p_1$ and $p_2$ respectively equal to $\dfrac{M_0(p_1)}{w}$ and $\dfrac{M_0(p_2)}{w}$. This circuit is live iff

$$\frac{M_0(p_1)}{w} + \frac{M_0(p_2)}{w} > 0$$

  Since $M_0(p_1)$ and $M_0(p_2)$ are multiple of $w$, this is the condition of the theorem.

- We suppose here that $w(p_1) \neq w(p_2)$. In order to lighten our notations we set $p = p_1$ for the rest of our proof.

  We prove by contradiction that this condition is necessary : we suppose that the inequality is false and that $C$ is live. Let us consider that a state of the graph $C$ is given by the couple of values $(M(p_1), M(p_2))$. Let $\Lambda$ be the number of different reachable states of $C$.

  - $C$ is assumed to be live. Since the number of states is bounded, a sequence of firings containing at least one state twice can be built. So there are two integers $\nu_1$ and $\nu_2$ such that :

  $$\begin{cases} M(p_1) + \nu_1.w(p) - \nu_2.v(p) &=& M(p_1) &\implies& \nu_1.w(p) &=& \nu_2.v(p) \\ M(p_2) + \nu_2.v(p) - \nu_1.w(p) &=& M(p_2) &\implies& \nu_2.v(p) &=& \nu_1.w(p) \end{cases}$$

  The smallest values verifying this previous condition are $\nu_1^* = \frac{lcm_p}{w(p)}$ and $\nu_2^* = \frac{lcm_p}{v(p)}$. So, the number $\Lambda$ is bounded by :

  $$\Lambda \geq \frac{lcm_p}{w(p)} + \frac{lcm_p}{v(p)} - 1$$

  - The global number of intial tokens is lower to or equal than $\sigma = w(p) + v(p) - 2.gcd_p$. By lemma 3.2, the number of tokens held by a place $p$ is a multiple of $gcd_p$. Moreover, the number of couples of integers $(X, Y)$ multiples of $gcd_p$ and such that $X + Y = \sigma$ is exactly $\frac{\sigma}{gcd_p} + 1$.

  $$\begin{array}{c|c|c|c|c|c} 0 & gcd_p & 2.gcd_p & \ldots & \sigma - gcd_p & \sigma \\ \hline \sigma & \sigma - gcd_p & \sigma - 2.gcd_p & \ldots & gcd_p & 0 \end{array}$$

  $$\underbrace{\phantom{xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx}}_{\text{Couple } (X,Y) \text{ with } X+Y=\sigma}$$

  This yields to $\dfrac{\sigma}{gcd_p} + 1$ couples

  However, this number takes into consideration the values $M(p_1) = v(p) - gcd_p$ and $M(p_2) = v(p_2) - gcd_p = w(p) - gcd_p$, which generates a deadlock. So, we deduce that the number $\Lambda$ of different live-state that we can build with $\sigma$ tokens verifies :

  $$\Lambda \leq \frac{\sigma}{gcd_p} = \frac{w(p)}{gcd_p} + \frac{v(p)}{gcd_p} - 2$$

Nevertheless, we know that

$$v(p).w(p) = lcm_p.gcd_p$$

So we derive that :

$$\frac{w(p)}{gcd_p} = \frac{lcm_p}{v(p)} \quad \text{and} \quad \frac{v(p)}{gcd_p} = \frac{lcm_p}{w(p)}$$

So, we have :

$$\frac{w(p)}{gcd_p} + \frac{v(p)}{gcd_p} - 2 \leq \Lambda \leq \frac{w(p)}{gcd_p} + \frac{v(p)}{gcd_p} - 1$$

which is the contradiction.

□

In the general case, the condition of the theorem 5.1 is not necessary : indeed, let us consider the normalized WEG $G$ presented by figure 11. One can verify that this graph is live : the sequences of firings $s$ defined below can be repeated infinitely. However, the condition of the theorem is not true.
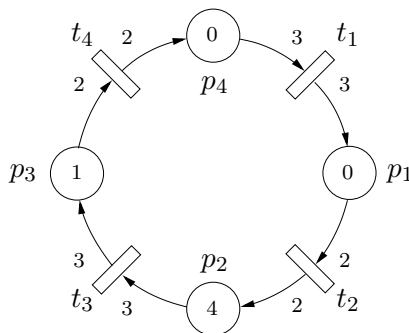


Figure 11: $G$ is live in spite of the fact that the sufficient condition of theorem 5.1 is not checked

|  |  | $t_3$ | $2.t_4$ | $t_1$ | $t_2$ | $t_3$ | $t_4$ | $t_1$ | $2.t_2$ |
|---|---|---|---|---|---|---|---|---|---|
| $M(p_1)$ | : | 0 | 0 | 0 | 3 | 1 | 1 | 1 | 4 | 0 |
| $M(p_2)$ | : | 4 | 1 | 1 | 1 | 3 | 0 | 0 | 0 | 4 |
| $M(p_3)$ | : | 1 | 4 | 0 | 0 | 0 | 3 | 1 | 1 | 1 |
| $M(p_4)$ | : | 0 | 0 | 4 | 1 | 1 | 1 | 3 | 0 | 0 |

The graph is live whereas the sufficient condition is not checked $\sum_{i=1}^{4} M_0(p_i) < \sum_{i=1}^{4} (v(p_i) - gcd_{p_i})$ (here :5<6).
We can check that the computation of the sufficient condition of liveness is in $\mathcal{O}(nm)$. Here, we consider again the example depicted on figure 9 on page 12.

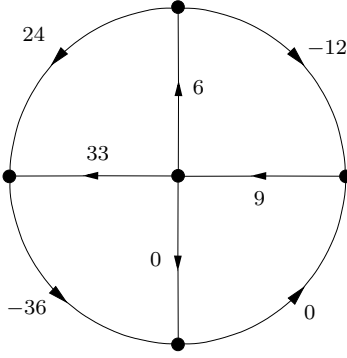- The normalization of a unitary graph is in $\mathcal{O}(nm)$.

Figure 12: The graph $\mathcal{G}'$ does not hold a circuit of negative weight so we can claim that $G$ is live.

- In the second step, we build a valued graph $\mathcal{G}' = (T, V)$ such any place $p' = (t_i, t_j) \in P'$ corresponds to an arc $(i, j) \in V$ valued by $M_0(p') - v(p') + gcd_{p'}$ (see figure 12 for the sequel of our example descibed on page 12).

  Then, using BELLMAN-FORD algorithm, we can check if $\mathcal{G}'$ has a circuit of strictly negative value. The complexity is also in $\mathcal{O}(nm)$.

  If there exists in $\mathcal{G}'$ a circuit of strictly negative value, then the condition of theorem 5.1 is false and we cannot conclude if the graph $G$ is live or not. Else, we build a partial subgraph $\mathcal{G}''$ of $\mathcal{G}'$ by removing all the non critical edges from $V$. Then we applied the dept search algorithm in order to determine if $\mathcal{G}''$ holds a circuit. This can be done in $\mathcal{O}(m)$.

  $\Rightarrow$ If $\mathcal{G}''$ contains a null valued circuit then we can not state anything for the liveness of $G$.

  $\Rightarrow$ Else, we can claim that the graph $G$ is live.

# 6  Conclusion

In this paper, we proved that any unitary weigthed event graph may be polynomially transformed into a normalized event graph, and we derived a sufficient condition of liveness. The normalization of a unitary weigthed event graph is, to our knowledge, a new concept and allows to reconsider the definition of work-in-process in a weighted Petri net. On the other hand, the main interest of the sufficient condition will be the possibility to speed-up future heuristics and branch-and-bound methods for the minimization function of the initial markings and the consideration of large size problems.

# References

[1] J-L Peterson. *Petri Net Theory and the Modeling of Systems.* Prentice Hall PTR, 1981.

[2] E. Teruel, P. Chrzastowski-Wachtel, J. M. Colom, and M. Silva. On weighted t-systems. pages 348–367.

[3] L. Recalde, E. Teruel, and M. Silva. On linear algebraic techniques for liveness analysis of pt systems. *Journal of Circuits Systems and Computers*, 8(1):223–265, 1998.

[4] Bruno Gaujal. Liveness in weighted routed nets. Technical report.

[5] P Chrétienne. *Les réseaux de Petri temporisés*. PhD thesis, Thèses d'état, Université P. et M. Curie, 1983.

[6] C Hanen. Study of an np-hard scheduling problem : the recurrent job-shop. *European Journal of Operational Research*, 72:82–101, 1994.

[7] C Hanen and A Munier. A study of the cyclic scheduling problem on parallel processors. *Discrete Applied Mathematics*, 57:167–192, 1995.

[8] H Hillion and J-M Proth. Performance evaluation of a job-shop system using timed event graph. *IEEE transactions on automatic control*, 34(1):3–9, 1989.

[9] S Gaubert. An algebraic method for optimizing resources in timed event graphs. In *9th conference on Analysis and Optimization of Systems, Lecture Notes in Computer Sciences*, volume 144, pages 957–966, 1990.

[10] S Laftit, J-M Proth, and X Xie. Optimization of invariant criteria for event graphs. *IEEE Transactions on Automatic Control*, 37(5):547–555, 1992.

[11] Edward A. Lee and David G. Messerschmitt. Synchronous data flow. *IEEE Proceedings of the IEEE*, 75(9), 1987.

[12] Edward A. Lee and David G. Messerschmitt. Static scheduling of synchronous data flow programs for digital signal processing. *IEEE Transaction on Computers*, C-36(1):24–35, 1987.

[13] M Adé. *Data memory minimization for synchronous data flow graphs emulated on DSP-FPGA targets*. PhD thesis, Université Catholique de Louvain, 1997.

[14] N Sauer. Marking optimization of weighted marked graphs. In *7th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'99)*, pages 719–725, Barcelone, Espagne, 1999.

[15] R. Govindarajan and Guang R. Gao. Rate-optimal shcedule for multi-rate dsp computations. *Journal of VLSI Signal Processing*, (9):211–235, 1995.

[16] G Cavory, R Dupas, and G Goncalves. A genetic approach to solving the problem of cyclic job shop scheduling with linear constraints. *European Journal of Operational Research*, 161:73–85, 2005.

[17] A Munier. Régime asymptotique optimal d'un graphe d'événements temporisé : application à un problème d'assemblage. *RAIRO*, 1993.

[18] T Cormen, C Leiserson, and R Rivest. *Introduction to Algorithms*. MIT Press, 1990.