Quantitative
Modeling
In
Parallel
Systems

Marking Optimization and Parallelism of Marked Graphs

Miguel Canales, Bruno Gaujal
INRIA-Sophia-Antipolis

D W2.T3.16a-2.v1

Sept 24, 1993

Distribution Level: All partners

Approved by: François Baccelli

Abstract The aim of the paper is to provide a formalization of the notion of parallelism of a marked graph exploitable by parallel simulation. We show that there exists an optimal starting point for equational simulations which gives a speed of simulation in the order of the intrinsic sequentiality of the system. Furthermore, under few assumptions, the modification of the marking will accelerate the simulation without altering its results for the stationary regime. We also derive algorithms to compute this optimal marking.

The QMIPS Project: The QMIPS project is a collaborative research project supported by the CEC as ESPRIT-BRA project no 7269. It is being carried out by the following organisations: CWI (Amsterdam), EHEI (University of Paris V), Imperial College (London), INRIA (Sophia-Antipolis), University of Erlangen, University of Newcastle, University of Torino and University of Zaragoza.

1 Introduction

Marked graphs constitute a good formalism to model manufacturing systems combining parallel tasks and synchronizations. they have been extensively studied either in the deterministic or in the stochastic context [9],[2]. These systems were shown to have a linear behavior if considered in the semi-field (max,+) and this property was the starting point of an extensive ergodic theory developed by F. Baccelli [2] and of a new method of parallel simulation based on recursive equations introduced in [9] and used in [3] and [12]. These parallel simulations of marked graph are of a new kind: they is not really event driven as in [13] or [16] but rather, equation "driven". The evolution is described by the successive application of linear transformations to the state variables. The aim of this paper is to improve the efficiency of these simulations by changing its starting state which is the initial marking of the graph.

Marking optimizations have been obtained in the deterministic case [18]. Here, we introduce the marking M^* which allows one to run parallel equational simulations more efficiently. Indeed, the cost of these algorithms depend on $L(M)$, the longest path without tokens in the marked graph under the initial marking M. The marking M^* will be chosen so that $L(M^*)$ is minimum. We show that $L(M^*)$ roughly equals the intrinsic sequentiality of the system so that little hope of increasing the speed of these algorithms is left. We also provide an algorithm to compute the couple $(M^*, L(M^*))$. Then, we show that changing the starting point of the system (i.e. its initial marking) will not alter the stationary behavior of the system, provided that the system is stable of course, under few assumptions on its topology.

In section 2, we give some preliminaries, in section 3, we give a formal meaning to the notion of intrinsic parallelism of a marked graph. In section 4, we define the marking M^* and we derive a computation of this marking. In the fifth section, we show that the evolution of a stochastic marked graph that satisfies the conditions of stability does not depend on the initial marking. This result is true for strongly connected graphs and for open systems with a single input. In the last section we apply the notions introduced in sections 3, 4 and 5 to give the optimal starting point for a parallel simulation of a marked graph.

2 Preliminaries

In the preliminaries, we will describe the model of marked graph that we will use in the following. This model is more precisely presented in [1].

A *marked graph* is a Petri Net where each place has at most one input transition and one output transition. We will denote it by $E = (P, T, \Lambda)$ where T is the set of transitions, P the set of places and Λ is the set of the links included in $P \times T \cup T \times P$. We denote by $\pi(p)$ the transition preceding place p and by $\pi(t)$ the set of the input places of transition t. The vector M denotes the marking of the net; M_i represents the number of tokens in place p_i. We will denote the graph along with the marking by $G = (E, M) = (P, T, \Lambda, M)$.

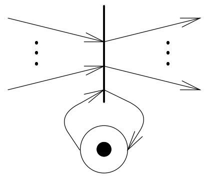

Figure 1: A token in the place recycling each transition guarantees the FIFO discipline.

Now we introduce the temporized model. Only the transitions will have associated temporization. We denote by $\sigma_t(n)$ the duration of the n-th firing time of transition t. The places have a null holding time. This simplification does not induce any loss of generality. A marked graph with temporized places and transitions can be emulated with a marked graph with temporizations on the transitions only. The semantic of firing times is given in [1] for example.

All the marked graphs considered in the following are connected. We will distinguish two classes of graphs. Strongly connected marked graphs will be called *closed systems* and usually model a device with no outside constraints. Non strongly connected graphs can model open systems with inputs. The inputs from the outside world will be represented as input transitions (with no incident arc). Their firing times follow a slightly different semantic than the one for regular transitions. An input transition with temporization $(u(n))_{n\in \mathbf{N}}$ sends one token in its output places for the ith time at the epoch $u(1) + \cdots + u(i)$.

We make some basic assumptions on our model.

FIFO Assumption:

First, all the transitions and are assumed to be FIFO. For every transition t, the n-th firing completion (token departure) of t corresponds to the n-th firing that t has started. A simple condition for a transition to be FIFO is to be *recycled* (i.e. it is the output and the input transition of a place with one token in the initial marking, as is shown in Figure 1). A transition with a constant holding time is FIFO too. In the following, the transitions are always recycled. The place recycling transition t is denoted p_t.

Initial Conditions:

- Once the initial marking M_0 is given, a non-timed net is defined by (E, M_0). We denote by $R(M)$ the set of all the markings reachable from M. It is well known that $M \in R(M')$ is parallelism relation if E is a marked graph [21].
- The initial tokens in places are not immediately available for the firing of the transitions. They remain in the places for a minimum duration which is called the *lag-time*. The lag-time of the i-th initial token in place p is denoted by $Y(p, i)$. The lag times must verify two conditions to be weakly compatible ([1] p. 70): first, the lag times must respect the FIFO feature; second, lag times of

tokens in place p cannot exceed the firing time of the transition $\pi(p)$. In the stochastic case, the lag time must belong to the support of the distributions of the firing times of the transition. Note that the lag times in the places recycling the transitions (there is only one token in these places) are not constraint by the first condition of weak compatibility. This remark will be useful in the following.

Stochastic Assumptions:

All the random variables considered here are defined on a common probability space (Ω, F, P).

The *holding time sequences* are sequences of non-negative real random vectors: $\{\sigma_1(n), \cdots, \sigma_{|T|}(n)\}_{n \in \mathbb{N}}$

We make the following assumptions (see [2] for further insight on these assumptions):

- *Stationarity and ergodicity:*

  The sequences $\{\sigma_t(n)\}$ are ergodic and stationary for all t.

- *Integrability:*

  The random variables $\sigma_t(n)$ are integrable.

- *Coupled ergodicity:*

  If the system has several inputs with temporizations $(u_1(n), \cdots, u_k(n))$, all the variables $(u_i(n))$, $(u_i(n) - u_j(n))$ are jointly ergodic and stationary for all i, j.

Finally, a Timed Marked Graph is defined by: $S = (E, \Sigma, Y, M_0)$ where E is the topology of the system, $\Sigma = (\sigma_1(n), \cdots, \sigma_{|T|}(n))_{n \in \mathbb{N}}$ is the temporization in the transitions, Y is the vector of the lag times and M_0 the initial marking.

We denote by $m(S, t)$ the marking in the system S by time t and by $m(p, t)$ the marking in place p at time t.

3 Parallelism in a Marked Graph

Measures of parallelism for discrete event systems are presented in [4], [14]. However, we are looking for a parallelism exploitable by parallel equation driven simulations and we present notions useful in this particular frame. Since the efficiency of parallel simulations depends on the availability of sufficient parallelism in the model itself, we present the notion of sequentiality of a marked graph. It will be further related to the complexity of the simulation algorithm we are interested in. In other words, we will see that the sequentiality is an appropriate measure of the parallelism present in the marked graph.

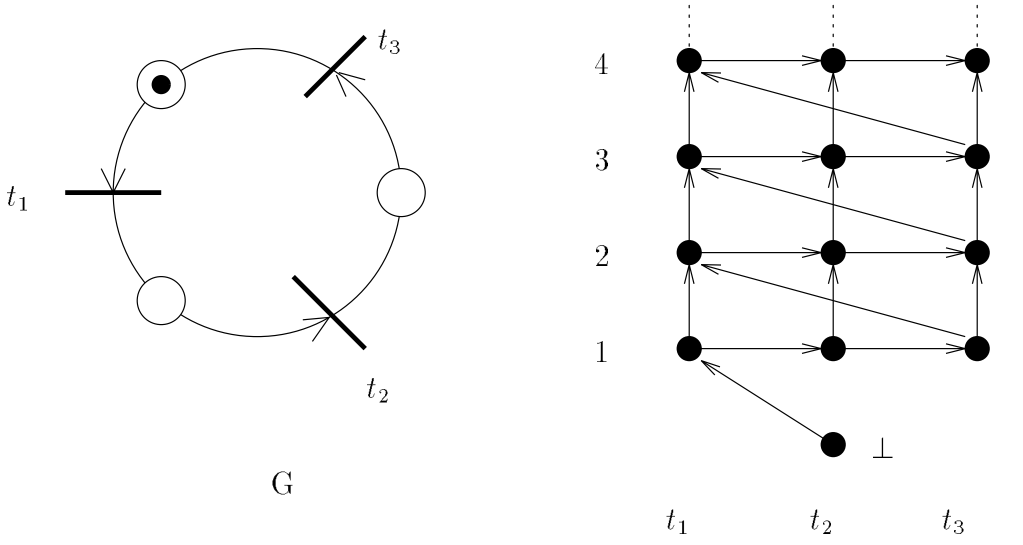

Figure 2: A marked graph and its associated graph of precedence. Since all the transitions are implicitly recycled, each column of τ is a total order.

3.1 Graph of Precedence

In this section we consider a marked graph without the timings: $G = (E, M_0)$.

We introduce a different structure based on the dependance relations in the marked graph. This graph is also called the developped graph and is presented in [7] for example. However our view is slightly different since we are not interested in analyzing the performance of the system but in discovering the available parallelism present in it.

The notion of parallelism present in a marked graph will be independent of the temporizations of the transitions and only be based on the precedence relations between the firing epochs.

In that purpose, we construct a graph of precedence τ which can be constructed in the following way. The graph τ has $T \times I\!N \cup \{\perp\}$ as the set of vertices. The vertex $\perp$ is special and represents the "initiation" of the graph of precedence. τ has an edge between the vertex (t, n) and the vertex (t', n') if the n'-th firing of transition t' uses a token produced by the n-th firing of transition t. We also put the edges $((\perp), (t, 1)), \cdots, (\perp), (t, m(t)))$ for all transition t, where $m(t) = \max_{p \in \pi(t)}(M_0(p))$. Figure 2 depicts a marked graph and its graph of precedence.

We establish a few properties that help to see the relations between the two structures.

Proposition 3.1 *The graph τ does not essentially depend on M_0 but rather on the class $R(M_0)$.*

Proof:

Let M_1 be a marking in $R(M_0)$ and τ_1 its associated graph of precedence.

We show that τ and τ_1 are nearly isomorphic. Consider the vector $(n_1, \cdots, n_{|T|})$, n_i being the number of times transition t_i has to be fired to go from marking M_0 to marking M_1.

We construct the transformation :

$$
\begin{array}{lcccll}
f: & \tau & \mapsto & \tau_1 & & \\
 & (t_i, n) & \rightarrow & (t_i, n - n_i) & if & n > n_i \\
 & (t_i, n) & \rightarrow & (\perp) & if & n \leq n_i \\
 & (\perp) & \rightarrow & (\perp) & &
\end{array}
$$

f is nearly an isomorphism between τ and τ_1. Indeed, only a finite number of vertices are contracted into $\perp$. ■

This proposition reinforces the fact that τ is a good model to study the parallelism of the marked graph. Indeed, the parallelism present in a marked graph should not depend on the starting point of the system but only of the overall " dependencies" between the events taking place during the evolution of the system.

Proposition 3.2 *The marked graph G is life if and only if τ is acyclic.*

Proof:

Suppose that τ contains a cycle $((t_1, n_1), \cdots, (t_k, n_k))$. This implies that $(t_1, \cdots, t_k)$ is a cycle in the Marked Graph G. By definition of τ, this means that the n_1-th firing of transition t_1 depends on the n_kth firing of t_k which depends on the $n_{k-1}-$th firing of t_{k-1} and so on. Eventually, the n_1-th firing of transition t_1 depends on the n_1-th firing of transition t_1. This imply that the cycle $(t_1, \cdots, t_k)$ does not contain any token which means that the marked Graph is not live.

Conversely, if G is not live, then it contains a cycle with no token, $t_1, \cdots, t_k$. In the graph of precedence we can exhibit the cycle $(t_1, 1), \cdots, (T_k, 1)$. ■

Remark: In the following, the marked graph will be live and τ may be considered as a partial order as a consequence. We will refer to antichains (anticliques of τ) and chains (totally ordered subsets), following the usual notations and definitions (presented in [11]).

We define the level n in τ as the set $\{(t, n), t \in T\}$. A chain up to level n in τ is an oriented path from $\perp$ to any element of the level n.

3.2 Degree of Parallelism

The *degree of parallelism* of a system is usually defined by the number of events than can take place in parallel in the system.

Definition 1 *The degree of parallelism δ of a marked graph G is the length of the maximum antichain in τ.*

This definition is equivalent to the usual one. Indeed, one can note that δ is the maximum number of enabled transitions that can occur in the marked graph under any reachable marking, independently with one another (these transitions can all fire at the same time).

In the general case, δ is difficult to compute. Indeed, it could be formulated as an integer linear programming problem (see [21]). However, some trivial bounds on δ are easy to find. $|T|$ is an upper bound of δ since $(t,n) <_\tau (t,k)$ whenever $0 < n < k$. This will happen to be good enough for parallel simulation of marked graphs on massive parallel machines as the Connection Machine. Indeed, for most systems, it is possible to allocate at least one processor per transition.

3.3 Sequentiality

We introduce a dual notion of the degree of parallelism which will be of interest in the following.

We consider the longest sequence of totally ordered firings during an evolution of the marked graph up to the level n in τ. The longer this sequence, the less parallel the marked graph. To get a finite value, we take the ratio of this sequence over n, the level in τ. To get rid off the influence of the initial part of the graph τ that depends on the initial marking, while it does not essentially depend on it (see proposition 3.1), we take the limit to infinity.

Definition 2 *The sequentiality $s(G)$ of a marked graph G is defined by:*

$$s(G) = \lim_{n\to\infty} s_n,$$

where $s_n = \dfrac{\text{length of the longest chain in } \tau \text{ up to level } n}{n}$.

We will show in the following that this limit exists. However, it is easy to see immediately that $s_n \leq |T|$ for all n, so that s is bounded.

We define the critical cycle C_r in the marked graph (G, M_0) as the cycle with the maximum average length, where the average length of a cycle C is its length $l(C)$

divided by the number of tokens it contains $w(C)$. We call λ the average length of the critical cycle in (G, M_0).

$$\lambda = \max_{C \in \mathcal{C}} \frac{l(C)}{w(C)}.$$

Proposition 3.3 *The average length of the critical cycle of a marked graph is equal to the sequentiality of the associated graph of task*

$$s(G) = \lambda.$$

Proof:

Let us add temporisations in the transitions of the marked graph G . Each firing of each transition is assumed to last for a time unit. Therefore the length of the longest chain in τ starting with $\perp$ and ending with the node (t, n) equals the time it takes in the marked graph to reach the nth firing of transition t, denoted $X_t(n)$.

But now, if one uses the language of marked graphs developped in [1] , one can use the periodicity result shown in [9] and write for any transition t, $X_t(n) = k\lambda + X_t(n-k), n > n_0$ for some bounded k and for n_0 large enough, in the case G is strongly connected. In the general case, one can also write $\max_t X_t(n) = k\lambda + \max_t X_t(n-k), \quad n > n_0$.

In terms of the graph τ, this last equality can be rewritten

$$s_n(\tau) = \frac{n_1 s_{n_1}(\tau) + \lambda(n - n_1)}{n}, \tag{1}$$

where n_1 is bounded and defined by : $n_1 = n - k\lfloor(n-n_0)/k\rfloor$. Finally when n goes to infinity in the equality (1), we get $s(\tau) = \lambda$. ∎

Proposition 3.4 *If $M' \in R(M)$, $G = (E, M)$ and $G' = (E, M')$, then $s(G) = s(G')$.*

Proof:

In marked graphs, the number of tokens remains constant in all the cycles (see [21]). Therefore, the critical cycles are the same in both G and G'. Since the sequentiality of a marked graph equals the average length of a critical cycle (proposition 3.3), we get the equality, $s(G') = \lambda' = \lambda = s(G)$. ∎

Remark: The sequentiality of a marked graph does not depend on the actual initial marking M_0 but rather on the class $R(M_0)$.

3.4 PRAM Model

In this section, we consider a parallel algorithm which graph of task is τ and that would run on a PRAM machine. This sort of models are studied in [8] for example, based on notions first introduced in [17].

The typical method to do so is to allocate one processor per "column" of τ (corresponding to one transition in G). Since the degree of parallelism δ of the system is smaller than $|T|$, this allocation is optimal.

Now the time it takes to compute the firing epochs up to level n is proportional to the longest path in the graph of task, that is proportional to $n.s(G)$.

In the following we will compare the complexity of real implementation of the simulation of the marked graph with this theoretical performance.

4 Shortest Longest Path Without Tokens

We introduce yet another notion: the longest path without tokens. Let $M \in R(M_0)$. $L(M)$ is the length of the longest path in the marked graph (E, M) with no tokens. $L(M)$ is finite because (E, M) is live.

Definition 3

$$L^* = \min_{M \in R(M_0)} (L(M))$$

and M^* is a marking such that $L(M^*) = L^*$.

Note that a marking M achieving the equality $L(M) = L^*$ is not unique. However we will denote by M^* any marking verifying this equality.

Proposition 4.1 $L^* = \lceil s(G) \rceil - 1$

Remark: This proposition establishes the relation between the sequentiality of the marked graph and the longest path without tokens under the appropriate marking.

Proof:

We will prove that $L^* = \lceil \lambda \rceil - 1$ which is equivalent to the result using proposition 3.3. The proof of this proposition will come from a way to compute L^*. We will show that L^* is obtained in a critical cycle which will be of the form shown in figure 3.

Let M^* be the marking in $R(M_0)$ achieving L^* as rarely as possible. Consider a path of length L^*. The first transition t on this path is enabled. Consider the paths preceding t. At least one of these path is of length at least $L^* - 1$ otherwise, the firing of t would reduce the number of paths of length L^*. We apply the same argument

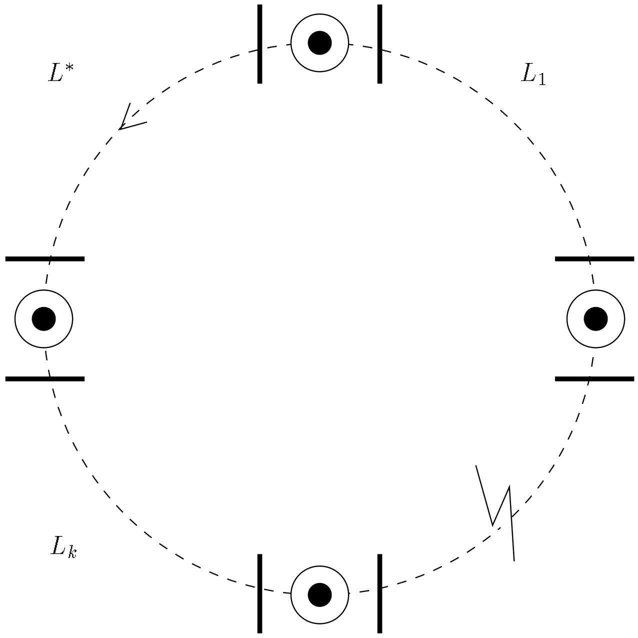

Figure 3: This is the form of a critical cycle in the graph under the marking M^*. For all i, $L_i = L^*$ or $L^* - 1$.

---

with this path and we can find at least one path of length at least $L^* - 1$ before at least on of these paths and so on. Eventually we get a cycle formed with such paths.

If none of these cycles contain the original path, we fire all the enabled transitions on all the cycles. All the paths in the cycles remain of the same length. However the paths at the "beginning" of the cycles have their respective lengths reduced by one. So we have paths which lengths go either from L^* to $L^* - 1$ or from $L^* - 1$ to $L^* - 2$. If one of these paths is in the first case, the number of paths of length L^* decreases. If all the paths are in the second case, we can fire the transitions at the end of these paths and get the same problem with the preceding paths and so on until the length of a path of length L^* decreases by one. This is a contradiction with the choice of the marking M^*.

Eventually, we get a cycle of the form depicted in figure 3.

Now, this cycle C has say k tokens and has a length greater than $k + (k-1)(L^* - 1) + L^*$. So $\lambda \geq 1 + L^* - 1 + 1/k = L^* + 1/k$. On the other hand, assume that $\lambda > L^* + 1 + \alpha$ with $\alpha > 0$. The critical cycle has say m tokens so its length is $m + mL^* + m\alpha$. As $\alpha > 0$ and $m\alpha$ is an integer, then, $m\alpha \geq 1$. But now, this means that one path without token in the critical cycle is longer than L^* which is impossible.

Finally, we have shown that $L^* + 1/k \leq \lambda \leq L^* + 1$. This implies that $L^* = \lceil \lambda \rceil - 1 = \lceil s(G) \rceil - 1$. ∎

Note that proposition 4.1 says that L^* is the integer approximation of the sequentiality of the system. This remark gives an insight on the reason why the complexity of the MIMD simulations of a marked graph are linear in L^* in the best case. See section 6 for a detailed discussion on this topic.

4.1 Computation of (M^*, L^*)

The previous proposition of L^* allows one to derive an algorithm to compute a couple (M^*, L^*).

Suppose M is a reachable marking but not a marking M^*. Then, using the previous characterization of the cycles containing a path of length L^* in M^*, there exists a sequence of transitions belonging to some cycle that can be fired so that either $L(M)$ is reduced or the number of paths with no token of length $L(M)$ is reduced. One can also see that this sequence is the following one: always fire all the transitions at the beginning of the longest paths.

On the other hand, if we can fire all the enabled transitions in the cycles containing the paths of length $L(M)$ without reducing $L(M)$, then the marking is M^*.

In order to describe the algorithm, we define the class $C(t)$ of transition t by: $C(t) = l$ if the longest empty path ending in transition t is of length l. We introduce $K = |T|$.

The algorithm is now the following:

The procedure $\mathcal{P}$ consists in the 3 following steps:

1 Update $C(t)$ for all $t \in T$.

2 Construct the set E_0 of all the transitions which are at the beginning of a path of length $L(M)$. Start with $E_{L(M)} = \{t/C(t) = L(M)\}$ and construct iteratively $E_{k-1} = \{t, C(t) = k-1 \text{ and } \exists t' \in E_k \text{ s.t. the place } (t', t) \text{ is empty } \}$.

3 Fire all the transitions in E_0.

The algorithm repeats this procedure until $L(M)$ has not been reduced in the last K iterations. Figure 4 shows an application of this algorithm to a small marked graph. The constant K could be improved (i.e. reduced) fairly easily. This would allow one to stop the algorithm earlier (as it is done in figure 4). However the gain is not substantial because only the number of computations made in the last marking is reduced.

Another possibility for the stopping test would be to compute λ using its definition (average length of a critical cycle) and then L^* using proposition 4.1. We stop the procedure when $L(M) = L^*$. Unfortunately, the extra step that is required in this case (compute the critical cycle) could be quite long to perform and could outweight the gain obtained by improving the stopping test.

A detailed and parallelized version of this algorithm is presented in [6].

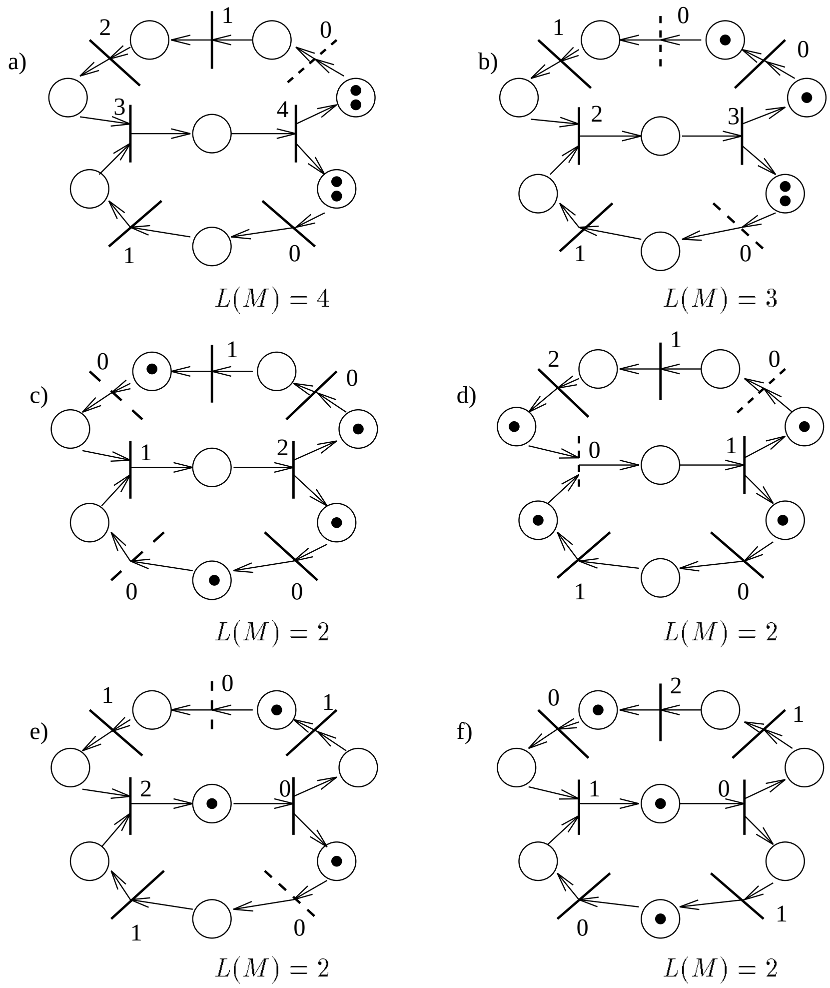

Figure 4: This picture shows the evolution of the marking of a marked graph when the algorithm of marking optimization is applied on it. Labels on the transitions denote $C(t)$, the class of transition t. The transitions belonging to E_0 are represented in dashed line. One can see that after three iterations without decreasing $L(M)$, the algorithm must stop. The stopping bound (3 iteration without reduction of $L(M)$) is given by $\max \frac{\text{length of the critical cycle}}{L(M)}$ which is equal to $\frac{5}{2}$ in this example.

5 Insensitivity To the Initial Marking

In this section, we show that for systems with a stationary regime that does not depend on the initial lag times, the initial marking does not alter this stationary regime.

For technical reasons, we distinguish two case: strongly connected networks and open networks. However the results will be similar in both cases.

5.1 Strongly Connected Networks

A marked graph E is strongly connected if there exists an oriented path from any transition of E to any other vertex. In the following we will denote strongly connected marked graphs by SCMG.

The following theorem has been shown in [10].

Theorem 5.1 *Let $G_1 = (E, M_1)$ and $G_2 = (E, M_2)$ be two live SCMG which differ only in their initial markings. If $M_2 \in R(M_1)$, then by blocking an arbitrary transition t, G_1 and G_2 will reach the same marking M where no transition but t is enabled.*

Now we consider a temporized system $S = (E, \Sigma, Y, M_0)$. In [19], a necessary and sufficient condition of stability of closed network is established:

A *Pattern* of a stochastic matrix A is a deterministic matrix $\tilde{A}$ which belongs to the support of A (which is considered as a random variable in $I\!R^{J \times J}$).

We can construct a sequence of stochastic matrices $A(n)$ that describes the evolution of the system. (see [9], [1])

Theorem 5.2 *If there exist an integer k and a matrix C which is a pattern of $A(k-1) \otimes \cdots \otimes A(0)$ with a unique critical cycle and verifying $C^n = \mu C^{n-1}$ for some μ, then the SCMG is stable and admits a unique stationary regime regardless of the initial condition Y.*

Note that most of the real systems verify this stability condition. For example if one transition has a firing sequence with an unbounded support distribution, the system is stable, see [1]. Only few and classified systems admit several stationary regimes depending on the initial lag-times (see [20]).

Now, we can formulate the following theorem:

Theorem 5.3 *Let $S_1 = (E, \Sigma, Y^1, M_1)$ and $S_2 = (E, \Sigma, Y^2, M_2)$ be two SCMG with weakly compatible lag-times and the same joint distribution of the sequence of the service times. Assume that the service times form jointly stationary and ergodic sequences of integrable r.v.'s and that the sequences of service times at different servers are mutually independent. Assume that one transition has an unbounded support firing distribution. Then, condition of stability are satisfied. If $M_2 \in R(M_1)$ then the stationary regimes of the two systems are identical.*

Proof:

Consider the systems $S_1' = (E, \Sigma, Y_1', M_1)$ and $S_2' = (E, \Sigma, Y_2', M_2)$ where $Y_1'(p_t) = Y_2'(p_t) = \infty$ for a transition t with unbounded firing distribution support and $Y_1'(p) = Y^1(p), Y_2'(p) = Y^2(p) \ \ \forall p \neq p_t$.

According to theorem 5.1, these two systems will reach the same marking M. We denote by k_j^1 and k_j^2 the numbers of times transition j has fired in S_1' and S_2' respectively before reaching the marking M.

We define T_0^1 and T_0^2 by:

$$T_0^1 = \sum_{j \in T} \sum_{l=1}^{k_j^1} \sigma_j^1(l) + \sum_{p \neq p_t} \sum_{l=1}^{M_1(p)} Y^1(p, l),$$

$$T_0^2 = \sum_{j \in T} \sum_{l=1}^{k_j^2} \sigma_j^2(l) + \sum_{p \neq p_t} \sum_{l=1}^{M_2(p)} Y^2(p, l),$$

and finally $T_0 = \max\{T_0^1, T_0^2\}$.

We consider the systems $S_1'' = (E, \Sigma, Y_1'', M_1)$ and $S_2'' = (E, \Sigma, Y_2'', M_2)$ where $Y_1''(p_t) = T_0$, $Y_2''(p_t) = T_0$ and $Y_1''(p) = Y^1(p) \ \ \forall p \neq p_t$, $Y_2''(p) = Y^2(p) \ \ \forall p \neq p_t$. T_0 is chosen large enough so that we obtain $m(S_1'', T_0) = m(S_2'', T_0) = M$. Now, systems S_1'' and S_2'' have weakly compatible lag-times. Furthermore, theorem 5.2 allows one to say that S_1'' and S_2'' are stable and that they have the same stationary regime than S_1 and S_2 respectively.

We just have to show that S_1'' and S_2'' have the same stationary regime.

Since the sequences of firing times are mutually independent and stationary, we can couple the service times in S_1'' and S_2'' in the following way:

$$\sigma_j^1(n + k_j^1) = \sigma_j^2(n + k_j^2) \ \ \forall j \in T, \ \forall n \geq 0.$$

These sequences are also mutually independent and stationary. Under such coupling, one sees that

$$m(S_1'', t) = m(S_2'', t) \ \forall t \geq T_0.$$

This implies that the two systems have the same stationary regime. ∎

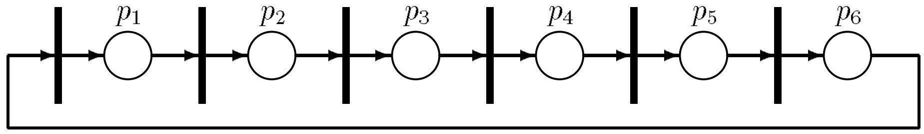

Figure 5: A circuit marked graph.

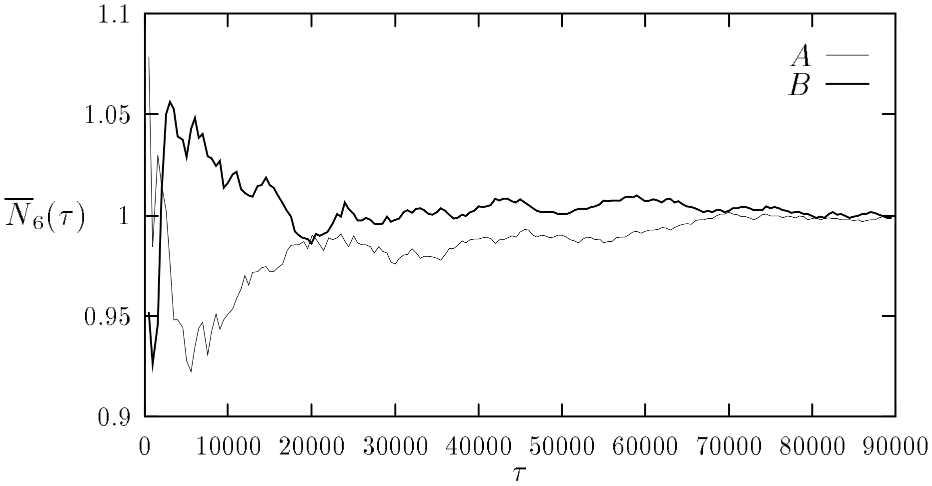

Figure 6: Stationary distribution of tokens for the graph depicted in Figure 5. A is the evolution of the marking in p_6 with $M_0 = (0,0,0,0,0,6)$ and B with $M_0 = (6,0,0,0,0,0)$.

We have run some experiment to give an idea of the speed of coupling. Indeed, the theorem 5.3 does not say anything on that feature.

Figure 5 depicts a petri net consisting in a single circuit. Figure 6 shows the evolution of the average number of tokens in the place p_6, for two differents simulations. In one, six tokens were assigned to place p_1 and 0 to the others, and in a second simulation the six tokens were assigned to the place p_6. The firing times of all the transitions are i.i.d. exponential variables with the same parameter.

5.2 General Networks

If a marked graph is not strongly connected, it is decomposable into strongly connected components interlinked by an acyclic oriented system (I). Therefore, the components can be partially ordered by I. In figure 7, the order of the components is $C_0 <_I C_1 <_I C_2$.

In particular, we call *input components* the minimal components according to I.

5.3 Networks With a Single Input

In this subsection, we give the theorems corresponding respectively to theorems 5.1, 5.2, 5.3 in the case of a marked graph with a single input.

First we extend theorem 5.1 to open networks with a single input component.

Theorem 5.4 *Let $G = (E, M)$ and $G' = (E, M')$ be two connected and live marked graphs and C_0 be the initial component of E. If one blocks any transition $t_0 \in C_0$, then if $M \in R(M')$, G and G' reach the same marking M_{t_0} where no transition can fire but t_0.*

Proof:

Since the graph E is connected, for any transition $t (t \neq t_0)$ in E, there is an oriented path from t_0 to t. Let us consider the path from t_0 to t that contains the smallest number of tokens under the current marking M. This path is simple because, since G is live, all the cycles contain tokens. However, it may not be unique. In this case, we choose the path with the smallest indices of transitions and we denote this path W_t. We shall denote by M_t the number of tokens on such a path under the marking M.

Now, block transition t_0. It is easy to see that transition t cannot fire more than M_t times before blocking. Indeed when a transition distinct from t_0 or t is fired no token is added or removed in W_t. If t is fired, tokens are removed from all the simple paths from t_0 to t and in particular from W_t. This implies that after some firings, the whole network will eventually block (no transition is enabled except t_0).

Now, we will see that t blocks after exactly M_t firings, which means that the path W_t is empty when t blocks. Let us say that the network reaches a complete deadlock under the marking D. Under D, let us follow the longest path without token which ends in transition t. This path begins in transition t_0 otherwise some transition in the net would be enabled. Now this path from t_0 to t is empty and by the token cycle conservation law in marked graphs, this path contained M_t tokens under the initial marking so the path W_t is also empty.

We construct a subgraph S of E in the following way: For each transition t, $t \neq t_0$ in G we only keep one empty path from t_0 to t, the path W_t.

1) The graph S is a spanning tree of G:

- S is connected: Each transition has a least one path which links it with t_0 in E.
- S contains no cycle: The existence of a cycle in S would mean that some transition in G is linked to t_0 by more than one path.

2) S does not depend on M. If we construct S' starting with another marking $M' \in R(M)$, we would get the same spanning tree, i.e. $S = S'$.

Let t be a transition in G. Let Σ_1 and Σ_2 be 2 path from t_0 to t. $M'(\Sigma_1) - M'(\Sigma_2) = M(\Sigma_1) - M(\Sigma_2)$. So the minimum path W_t is the same in both marking.

The next step is to describe the marking D. All the places in S are empty. Now, we add one place of E to the spanning tree S. We create a cycle. If this cycle is a circuit C then all the weight of the circuit $M(C)$ must be put in this place. Otherwise, if the edge is (t_1, t_2) consider the paths from t_0 to t_1 and t_2 respectively W_{t_1} and W_{t_2}. In the original marking, we know by construction of S that $M(W_{t_1}) + M(t_1, t_2) \geq M(W_{t_2})$. So there must be a non-negative weight $M(W_{t_1}) + M(t_1, t_2) - M(W_{t_2})$ on the edge (t_1, t_2). This marking is the same for any starting marking in $R(M)$. ∎

Now, we give the conditions of stability of an open system. These conditions are established in [1]. Here we give only the conditions of stability of a system with a single input.

Theorem 5.5 *If the input component verifies the condition of stability when considered in isolation as a strongly connected system given in theorem 5.2, and if for any components C_i and C_j, $C_i <_I C_j$ implies that the cycle time of C_i (isolated) is bigger than the cycle time of C_j (isolated), then the marked graph is stable and its unique stationary regime does not depend on the initial lag-times.*

These two results allow us to derive the insensitivity of the stationary regime for open networks with one initial component. The formulation of the theorem is similar to theorem 5.3 as well as its proof.

Theorem 5.6 *Let $S_1 = (E, \Sigma, Y^1, M_1)$ and $S_2 = (E, \Sigma, Y^2, M_2)$ be two MG with one input component and with the same joint distribution of the sequence of the service time. Assume that the service times form jointly stationary and ergodic sequences of integrable r.v.'s and that the sequences of service times at different servers are mutually independent. Assume also that the system satisfies the conditions of stability and that one transition in the input component has an unbounded firing distribution. If $M_2 \in R(M_1)$ then the stationary regimes of the two systems are identical.*

Proof: The proof is the same as for theorem 5.3

Consider the systems $S'_1 = (E, \Sigma, Y'_1, M_1)$ and $S'_2 = (E, \Sigma, Y'_2, M_2)$ where $Y'_1(p_t) = Y'_2(p_t) = \infty$ for some transition t with an unbounded support for its firing distributions and $Y'_1(p) = Y^1(p), Y'_2(p) = Y^2(p) \ \ \forall p \neq p_t$.

According to theorem 5.4, these two systems will reach the same marking M. We denote by k^1_j and k^2_j the numbers of times transition j has fired in S'_1 and S'_2 respectively before reaching the marking M.

We define T^1_0 and T^2_0 by:

$$T^1_0 = \sum_{j \in T} \sum_{l=1}^{k^1_j} \sigma^1_j(l) + \sum_{p \neq p_t} \sum_{l=1}^{M_1(p)} Y^1(p, l),$$
$$T^2_0 = \sum_{j \in T} \sum_{l=1}^{k^2_j} \sigma^2_j(l) + \sum_{p \neq p_t} \sum_{l=1}^{M_2(p)} Y^2(p, l),$$

and finally $T_0 = \max\{T^1_0, T^2_0\}$.

We consider the systems $S''_1 = (E, \Sigma, Y''_1, M_1)$ and $S''_2 = (E, \Sigma, Y''_2, M_2)$ where $Y''_1(p_t) = T_0$, $Y''_2(p_t) = T_0$ and $Y''_1(p) = Y^1(p) \ \ \forall p \neq p_t$, $Y''_2(p) = Y^2(p) \ \ \forall p \neq p_t$ .These two systems have weakly compatible lag-times. T_0 is chosen large enough so that we obtain $m(S''_1, T_0) = m(S''_2, T_0) = M$. Furthermore, theorem 5.5 allows one to say that S''_1 and S''_2 are stable and that they have the same stationary regime that S_1 and S_2 respectively.

We just have to show that S''_1 and S''_2 have the same stationary regime.

Since the sequences of firing times are mutually independent and stationary, we can couple the service times in S''_1 and S''_2 in the following way:

$$\sigma^1_j(n + k^1_j) = \sigma^2_j(n + k^2_j) \ \ \forall j \in T, \ \forall n \geq 0.$$

These sequences are also mutually independent and stationary. Under such coupling, one sees that

$$m(S''_1, t) = m(S''_2, t) \ \forall t \geq t_0.$$

Therefore, the two systems reach the same stationary regime. ∎

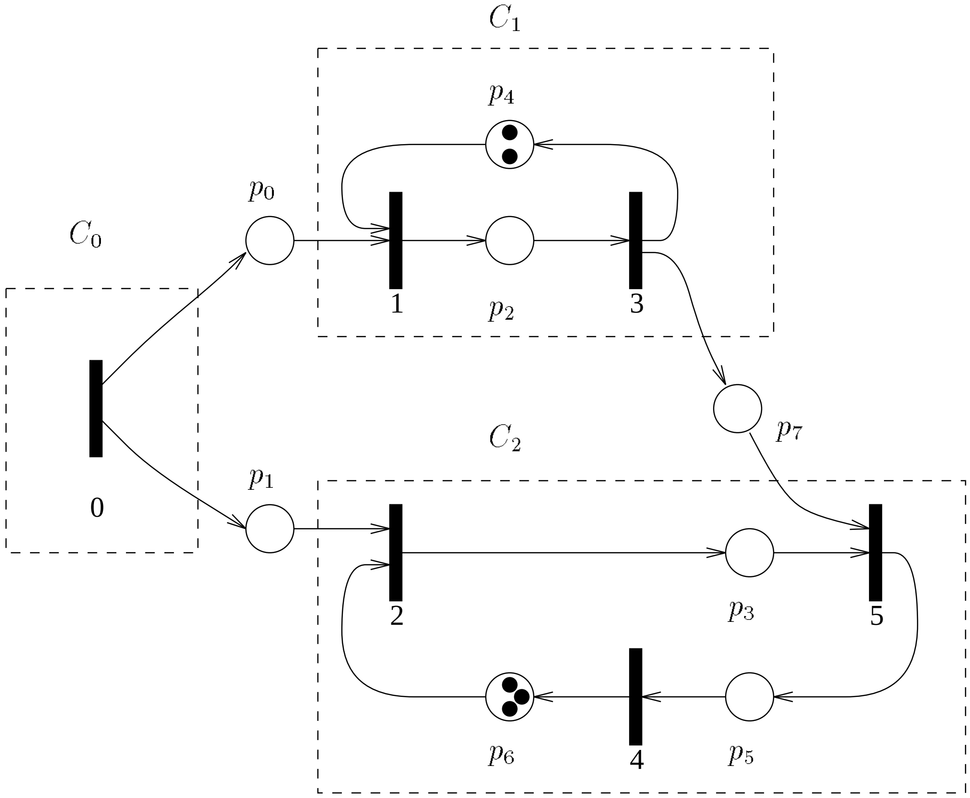

Figure 7: Open system with three components $C_0 < C_1 < C_2$. All the transition are recycled with places containing one token but this is not shown in the figure for simplicity.

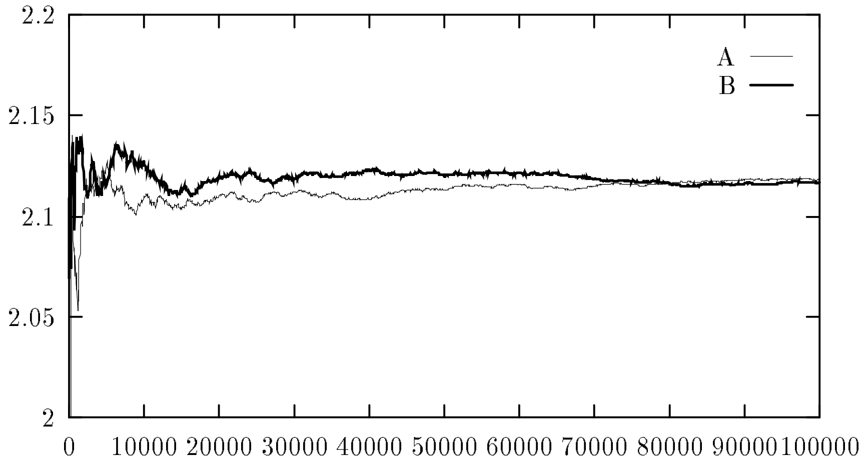

Figure 8: Evolution of the number of tokens in the place p_6 with two different initial markings. For A, we used the initial marking $(0,0,0,0,2,0,3,0)$. For B we used the initial marking $(1,1,1,1,1,1,1,0)$.

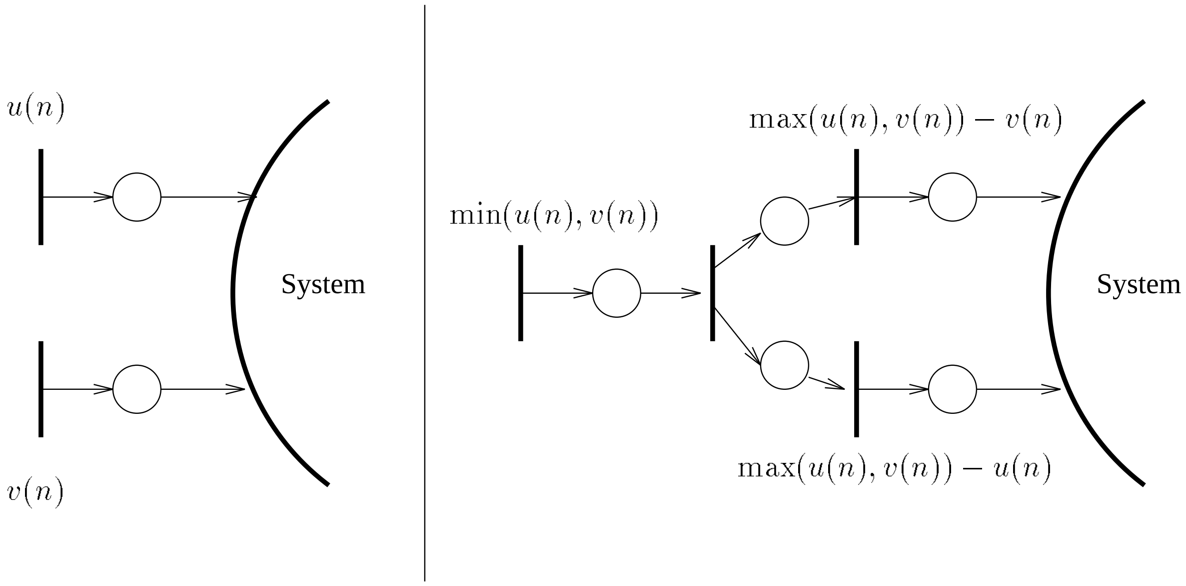

Figure 9: Transformation of a system with two inputs into a system with only one input. The firing times of the transitions are written on the figure.

The convergence is illustrated by an example depicted in figure 7. Figure 8 shows the evolution of the average marking in the place between transitions 4 and 5 with two different initial markings. For A we used the initial marking $(0,0,0,0,2,0,3,0)$. For B we used the initial marking $(1,1,1,1,1,1,1,0)$ which is reachable from the previous one.

5.4 Open Systems With Several Inputs

Let us recall that the theorem 5.6 applies only to open systems with one input component. However, for a system with several input components, the stationarity of the system depends of the coupled ergodicity of all the inputs that could be interpreted as a common dependency on a preceding phenomenon. Thus, under the right point of view, any stationary open systems can always be considered as a marked graphs with only one input component.

However, if we want to construct the detailed transformation, suppose that a system has two input sequences $u(n)$ and $v(n)$. We construct a system with entry $w(n) = \min(u(n), v(n))$ and two places with temporizations $\max(u(n), v(n)) - v(n)$ to get $u(n)$ and $\max(u(n), v(n)) - u(n)$ to get $v(n)$. See figure 9 for an illustration of this transformation.

Unfortunately, the firing times in some places are now strongly correlated and the theorem 5.6 are not applicable because the assumption of independance of the firing sequences is not satisfied. If we examine the proof of theorem 5.6, we can see that the coupling is done between the variables $\sigma_j^1(n+k_j^1)$ and $\sigma_j^2(n+k_j^2)$ for any transition j. If we impose the initial markings in the systems S^1 and S^2 in the input places to be the same, then, the coupling relation for the transitions in the input section of the system becomes $\sigma_j^1(n+k) = \sigma_j^2(n+k)$, where k is the same in the 2 systems and is equal to the number of tokens from the input to transition j. This coupling relation respects the correlations between the firing times presented in figure 9 and the result of theorem 5.6 holds under this restriction on the initial marking in the two systems.

From a practical point of view, we can try to modify the initial marking by firing any transition but the input transitions. The marking of the input places will not be changed and the stationary regime will remain the same with the new initial marking. However, we do not know how to compute a marking $\tilde{M}$ that would verify the constraint on the input places marking and that would minimize $L(\tilde{M})$. One can verify that in some cases, no marking M^* verifies the constraint on the initial marking. In such case, $L(M^*) < L(\tilde{M})$.

6 Optimal Marking of Marked Graphs

In this section we discuss the practical interest of the markings M^* for parallel simulation of marked graphs.

6.1 Equational Simulations

Marked graph can be very efficiently simulated using massive parallelism (see [3] and [12]). We describe briefly two kinds of equational simulation of a marked graph which is supposed stable with at most one input. In both cases we show that the complexity is linear in $L(M)$ and is close to the ideal case of a PRAM model.

The evolution of a Stochastic marked graph can be described by a linear system in the semi-field $\mathbb{R}(max,+)$:

$$X(n) = A(n).X(n-1).$$

The parallel algorithm developed in [3] uses these equations to compute the vector $X(n)$ of the firing times of the transitions. The computation of $A(n)$ involves $L(M)$ operations of cost $log(|T|)$ if made in parallel on a Connection Machine. Then, the matrix vector multiplication is done in parallel in $log(\overline{M}.|T|)$ where $\overline{M} = \max_p M(p)$. The complexity depends heavily on the initial marking. it is of the form $O(n.L(M).l(|T|))$ with l being a logarithmic function. Theorems 5.6 and 5.3 allow one to choose the marking which gives the best running time of the simulation, without altering the results of the simulation. This marking is a marking minimizing $L(M)$, i.e. a marking of type M^*.

This first algorithm only makes algebraic manipulations of the equations and ignores the underlying structure of the marked graph. A different approach uses the topology and the marking of the marked graph to establish an order on the utilization of these equations. The transitions are distributed in the $L(M_0)$ classes , $C_k = \{t/C(t) = k\}$. All the equations associated with places in a same class are used in parallel. The simulation algorithm consists in:

for ($n = 0$ to N)
    for ($i = 0$ to $L(M_0)$)
        fire all the transitions in C_i.

In this approach, each transition is assigned to a different processor. "fire a transition t" means the application of the equation involving X_t which requires d operations, d being the entry degree of t (i.e. $\#\pi(t)$). The complexity is yet again linear in $L(M_0)$: $n.L(M_0).\overline{d}$ where $\overline{d}$ is the max of all the degrees of the transitions. Once again, the marking M^* is the best initial marking of the system.

In both cases, the complexity of the algorithm is very close to the cost of any PRAM algorithm which graph of task is τ when started with a marking M^*. This seems to leave little hope for substantial improvement in this type of simulations (i.e. conservative) of marked graphs.

6.2 Applications

These results can be used in two different ways during a simulation of a given system. First, someone can *choose* an initial marking of type M^* that will satisfy the property $L(M^*) = L^*$, to start the simulation. Second, sometimes it is hard to find such a marking M^* by hand and furthermore, this marking may not correspond to the natural initial state of the system modeled by a marked graph. In these cases, the simulation may begin by a pre-computing step providing M^*. This initializing optimizer is available in the package MAGMAS© presented in [5] that provides simulation tools of marked graphs on a Connection Machine.

References

[1] F. Baccelli, *Ergodic Theory of Stochastic Petri Networks*, The Annals of Probability, Vol. 20, No. 1, 375-396, 1992.

[2] F. Baccelli, G. Cohen, J.D. Olsder and J.P. Quadrat, *Synchronization and Linearity.* Wiley, 1992.

[3] F. Baccelli and M. Canales, *Parallel Simulation of Stochastic Petri Nets using Recursive Equations*, ACM Transactions on Modeling and Computer Simulation, Vol. 3, No. 1, January 1993, pp. 20-41.

[4] B. Berard and L. Thimonier, *On a Concurrency Measure*, 2nd I.S.C.I.S proceedings, Istanbul, 1987.

[5] M. Canales, *Simulation parallèle de réseaux de Petri aléatoires utilisant des équations de récurrence*, Thèse de doctorat (in French), University of Nice, Sophia-Antipolis, to appear in Dec 1993.

[6] M. Canales and L. Finta, *A Parallel Algorithm To Optimize Initial Marking for Marked Graph Simulation*, In preparation.

[7] J. Carlier and P. Chretienne, *Problèmes d'ordonnancement*, Masson (in French), 1988.

[8] B. Charron-Bost, *Mesures de la concurrence et du parallélisme des calculs répartis.* Thèse de doctorat de l'université PARIS VII (in French), Sept 89.

[9] G. Cohen, D. Dubois, J.P. Quadrat and M. Viot, *A Linear-System-Theoric View of Discrete Event Processes and its Use for Performance Evaluation in Manufacturing.* IEEE Tr. Aut. Contr. ,Vol. 30, 210-220, 1985.

[10] Y. Dallery, Z. Liu, D. Towsley, *Equivalence, Reversibility, Symmetry and Concavity Properties in Fork/Join Queuing Networks with Blocking.* INRIA Report 1267, 1990, to appear in JCM.

[11] B. Dushnik and E.W. Miller, *Partially Ordered Sets.* Amer. Jour. of Math., 63:600-610, 1941.

[12] L. Finta, *Simulation parallèle de réseaux de Petri sur la Connection Machine*, Rapport de DEA I3S (in French), October 1992.

[13] R. M. Fujimoto, *Parallel Discrete Event Simulation.* In E. A. MacNair, K.J. Musselman and P. Heidelberger, editors, Winte Simulation Conference, 1989.

[14] D. Geniet, *AUTOMAF: un système de construction d'automates synchronisés et de calcul de mesure du parallélisme.* Thèse de doctorat de l'Université de Paris Sud, 1989.

[15] M. Gondran and M. Minoux, *Graphes et Algorithmes.* Eyrolles editor (in French), 1979.

[16] D. Jefferson, *Fast Concurrent Simulation using the Time Warp Mechanism.* The Rand. Corporation, Santa Monica, California.

[17] L. Lamport *Time, Clocks and the Ordering of Events in a Distributed System.* Comm. of ACM, 21, 7:558-564, 1978.

[18] S. Laftit, J.M. Proth and X.L. Xie, *Marking Optimization in Timed Event Graph.* Appl. and Th. of Petri Nets, 12th International Conference, p. 276-295, June 1991.

[19] J. Mairesse, *Products of Irreducible Random Matrices in the (Max,+) Algebra - Part I*, INRIA report No 1939, May 1993.

[20] J. Mairesse, *Products of Irreducible Random Matrices in the (Max,+) Algebra - Part II*, In preparation.

[21] T. Murata, *Petri Nets: Properties, Analysis and Applications.* Proc. of the IEEE, Vol 77, No 4, April 1989.