Signature of timed patterns in time Petri nets: a formal characterization

Camille Coquand[1], Audine Subias[1], and Yannick Pencolé[2]

[1] CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
Univ de Toulouse, INSA, LAAS, F-31400 Toulouse, France
firstname.lastname@laas.fr
[2] CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
Univ de Toulouse, LAAS, F-31400 Toulouse, France
yannick.pencole@laas.fr

Abstract

This paper investigates the problem of characterizing the signature of a time pattern in safe acyclic time Petri nets. While such signature contains an infinite number of elements because time is continuous, we propose a method to represent these elements as finite sets of constraints on the observable dates of firing of the observable transitions of the net.

1 Introduction

The study of systems requires the ability to differentiate between the different operating modes (critical, safety modes, failures, etc.). For this purpose, the signature of an operating mode is a relevant tool to know the observation resulting from the execution of the so called operating mode. It can be especially helpful in case of diagnosability, opacity or predictability analyses as it is a way to characterize the executions of the system [1], [2]. In discrete event systems, an operating mode is usually associated with the occurrence of a particular event (like a fault event switching the system from a normal mode to a failure mode). The signature is in this case the set of observable traces, i.e. the set of observations produced by the system in which the non-observable event of interest has occurred. More complex behavioural modes can be represented by a sequence of particular non-observable events [3]. Indeed, some behaviours of interest may only result from a succession of events which, taken independently, are not of interest. Such sequences of non-observable events are called *pattern*. These patterns can be atemporal as in [3],[4] where time only evolves with the occurrence of events, or they can be temporal, as introduced more recently in [5], in which case events are associated with quantitative firing time intervals. In this paper, the objective is to propose a formal characterization of the signature of temporal sequence patterns in systems that can be modeled by a specific class of time Petri nets. As time is a continuous quantity, the signature contains an infinite number of elements. This leads to a real problem in using the signature for further analyses of pattern such as diagnosability analyses. To solve this problem, given a system and a pattern as previously defined, the objective is to formally characterize the pattern signature by a finite set of constraints on the firing dates of the observable transitions of the system i.e. on the occurrence dates of the observable events that the system may produce when the pattern occurs.

In this paper, the formal characterization of a time pattern signature that we propose is based on the following assumptions. The considered systems and patterns are modelled by acyclic safe time Petri nets. The patterns are defined by unobservable events produced by the system. Finally, the pattern signature is formally characterized as finite sets of observable constraints only.

The paper is organized as follows. Section 2 recalls some formal prerequisites. Then the modeling of the problem and the characterization method are presented in Section 3. The synchronization method is described in Section 4. Section 5 develops the method to transform the synchronisation into observable contraints. Finally the characterization of the signature is given in Section 6.

2 Prerequisites

This section recalls some formal prerequisites on Time Petri Nets (TPN).

Definition 1. *A Labeled Time Petri Net (LTPN) is a 6-uple $N = \langle P, T, A, \Sigma, \ell, I_s \rangle$ where:*

- *P is a finite set of places*
- *T is a finite set of transitions ($P \cap T = \emptyset$)*
- *$A \subseteq (P \times T) \cup (T \times P)$ is a binary relation that models the arcs between the places and the transitions.*
- *Σ is a finite set of transition labels*
- *ℓ: $T \to \Sigma$ is the transition labelling function*
- *I_s: $T \to \mathbb{I}_{\mathbb{Q}_+}$ is a static closed interval function $I_s(t) = [a, b]$, its lower bound, also called the date of earlier firing is denoted $\lfloor I_s(t) \rfloor$, and its upper bound, also called the date of later firing, is denoted $\lceil I_s(t) \rceil$*

The *preset* of a transition t is the set of input places $pre(t) = \{p \in P \mid (p, t) \in A\}$, and similarly the *postset* of t is the set of output places $post(t) = \{p \in P \mid (t, p) \in A\}$. For a *safe* LTPN, a state is a couple $S = \langle M, I \rangle$ where M is the marking of the net (M: $P \to \{0, 1\}$) and I is the partial firing interval application (I: $T \to \mathbb{I}_{\mathbb{Q}_+}$) that associates to any enabled transition (*i.e.* a transition t for which $\forall p \in pre(t), M(p) > 0$) a time interval of $\mathbb{Q}_+$ in which t can be fired. $S_0 = \langle M_0, I_0 \rangle$ is the initial state of the net where M_0 is the initial marking of the net and I_0 is defined as follows: for any transition t enabled by M_0, $I_0(t) = I_s(t)$, else $I_0(t) = 0$. For a marking M, a transition t is firable at the date θ if and only if:

- t is enabled
- $\theta \in I(t)$ and for all t' enabled by M, $\theta \leq \lceil I(t') \rceil$

The fire of a transition *t* at a date θ is denoted: $\langle M, I \rangle \xrightarrow{\theta t} \langle M', I' \rangle$ and defined such that

- M' is such that $\forall p \in pre(t) \setminus post(t), M'(p) = 0$, $\forall p \in post(t) \setminus pre(t)$, $M'(p) = 1$ else $M'(p) = M(p)$
- for any transition $t' \in T$ ($t' \neq t$) enabled by M and still enabled by M', $I(t') = [a, b] \Rightarrow I'(t') = [max(0, a - \theta), b - \theta]$
- for every transition t' enabled by M' and not by M, $I'(t') = I_s(t')$

The set of final markings is denoted Q. A state S is reachable in a marked LTPN if there exists a run $r = \theta_1 t_1 \dots \theta_n t_n, n \in \mathbb{N}^*$ *such that* $S_0 \xrightarrow{\theta_1 t_1} S_1 \xrightarrow{\theta_2 t_2} S_2 \dots \xrightarrow{\theta_n t_n} S$. The set of reachable states of a LTPN *N* is denoted $R(N, S_0)$.

A run $r = \theta_1 t_1 \dots \theta_n t_n$ of a LTPN is said to be admissible if there exist $S_1, \dots S_n$ reachable states of N such that $S_0 \xrightarrow{\theta_1 t_1} S_1 \xrightarrow{\theta_2 t_2} S_2 \dots \xrightarrow{\theta_n t_n} S_n$.

A *timed sequence* over an alphabet Σ is a sequence of pairs $(d, e) \in \mathbb{R}_+ \times \Sigma$ where d corresponds to the date of firing of symbol e. A run produces a unique timed sequence.

Example 1. *Let us consider the alphabet* $\Sigma = \{a, b\}$. $\rho = 2a.3b.2a.2b$ *is a timed sequence over* Σ.

Definition 2. *The language* $\mathcal{L}(N)$ *of a LTPN* N *is composed of every timed sequence* $\rho = \theta_1 e_1 \dots \theta_n e_n$ *produced by an admissible run* $r = \theta_1 t_1 \dots \theta_n t_n$ *and* $\forall i \in [1, n], \ell(t_i) = e_i$ *for the LTPN* N*, leading from the initial state* S_0 *to a final state* $S = \langle M, I\rangle \wedge M \in Q$.

The projection on a set of transitions T is such that:

- if $r = \theta_1 t_1 \dots \theta_n t_n$, $\mathbf{P}_T(r) = \mathbf{P}_T((\theta_1 + \theta_2)t_2 \dots \theta_n t_n)$ if $t_1 \notin T$
- if $r = \theta_1 t_1 \dots \theta_n t_n$, $\mathbf{P}_T(r) = \theta_1 t_1.\mathbf{P}_T(\theta_2 t_2 \dots \theta_n t_n)$ if $t_1 \in T$
- if $r = \theta t$, $\mathbf{P}_T(r) = \epsilon$ if $t \notin T$

[6] defines the State Class Graph (SCG) of a TPN. It can be seen as an automaton that abstracts every behaviour of the TPN. A state of this automaton is a class of the TPN, containing a marking and a firing domain for every transition enabled by the marking. The initial firing domain is given by I_0 for every transition t enabled by M_0. It is an abstraction aggregating every state of the TPN sharing the same marking and a close firing domain.

Definition 3. *The State Class Graph (SCG) of a LTPN* $N = \langle P, T, A, \Sigma, \ell, I_s\rangle$ *is the 3-tuple* $(C, \alpha_0, \rightarrow)$ *such that:*

- $\alpha_0 = (M_0, F_0)$ *where* M_0 *is the initial marking of* N *and* $F_0 \in (\mathbb{R}^2_+)^T$ *is the initial firing domain of* N
- $C \in \{0, 1\}^P \times (\mathbb{R}^2_+)^T$ *is the set of all classes corresponding to states reachable in* N
- $\rightarrow \in C \times T \times C$ *is the transition function defined as follows :* $(M, F) \xrightarrow{t} (M', F')$ *iff*
  - t *is firable from* (M, F)
  - $M' = M - pre(t) + post(t)$
  - $F' = next(F, t)$

where $next : \mathbb{R}^T \times T \rightarrow \mathbb{R}^T$ is the procedure to build the firing domain F' associated with a reachable marking M' reached from M by the firing of t that is defined as follows:

1. for each transition t' enabled in M, compute the firing of t by adding the two constraints $\theta \leq \theta'$ and $\theta' = \theta + \theta'_{upd}$ (θ'_{upd} is a substitution variable)
2. eliminate variables relative to transitions enabled in M and not in M'
3. add the constraints relative to the newly enabled transitions (in M')
4. determine the canonical form of each constraint in F'

All along this paper, a path in the SCG of a LTPN N is a sequence of transition from the initial class C_0 to a class C_n associated with a final marking of the net.

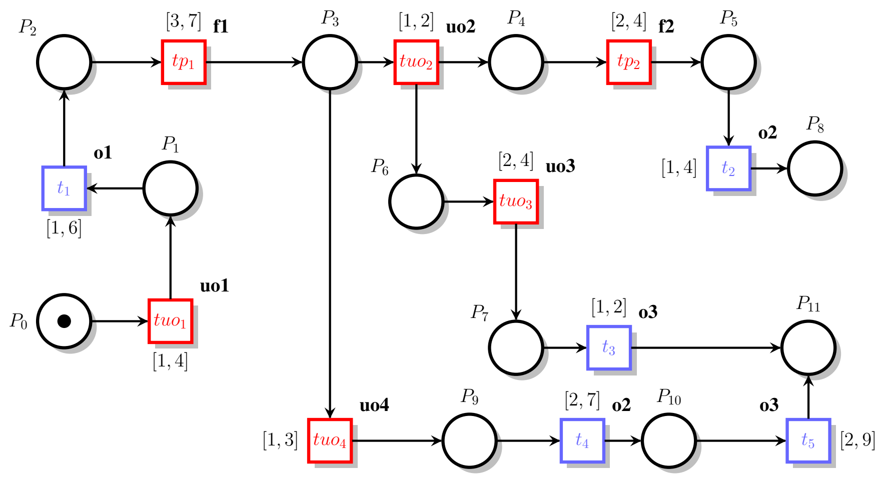

Figure 1: Θ_1 is a system composed of two behaviours : one is sequential (final marking P11) and one has concurrency (final marking P8 P11)

3 System and Pattern modeling

3.1 System and Pattern modeling

A system is a safe LTPN $\Theta = \langle P_\Theta, T_\Theta, A_\Theta, \Sigma_\Theta, \ell_\Theta, I_{s\Theta} \rangle$ that is partially observable. Some labels represent observable events ($\Sigma_{o\Theta} = \{o_1, ..., o_p\}$) others correspond to unobservable events ($\Sigma_{u\Theta} = \{uo_1, ..., uo_m\}$). The transition set is partitioned into two sets: $T_{o\Theta}$ the set of observable transitions (their label belongs to $\Sigma_{o\Theta}$) and $T_{u\Theta}$ the set of unobservable ones (their label belongs to $\Sigma_{u\Theta}$). Two assumptions are made about the system:

- **A0** The system is acyclic
- **A1** Every transition to reach a final state of Θ belongs to $T_{o\Theta}$ (it is labelled by an event of $\Sigma_{o\Theta}$).

A0 forbids an infinite number of transition firings in a finite amount of time. It also ensures that every run of the system is finite. Let us denote Q_Θ the set of final markings of Θ. **A1** ensures that a final marking is always reached by the firing of an observable transition.

Example 2. *Figure 1 represents a system Θ_1. The observable transitions (colored in blue) are t_1, t_2, t_3, t_4 and t_5 labeled by the events o_1, o_2, o_3, o_2 and o_3 respectively. The unobservable transitions (colored in red) are $t_{uo1}, t_{uo2}, t_{uo3}, t_{uo4}$, t_{p_1} and t_{p_2}, labeled by $uo_1, uo_2, uo_3, uo_4, f_1$ and f_2. Transition t_1 is enabled when there is a token in P_1 (i.e. after the firing of t_{uo1}) and can be fired between 1 and 6 time units (tu) after its enabling.*

Definition 4. *A timed pattern is an acyclic safe LTPN $\langle P_\Omega, T_\Omega, A_\Omega, \Sigma_\Omega, \ell_\Omega, I_{s\Omega} \rangle$ without parallelism (i.e. $\forall$ t and t' $\in T_\Omega$, such that $pre(t) \cap pre(t') = \emptyset$, t and t' are not enabled simultaneously) where:*

1. *$\Sigma_\Omega \subseteq \Sigma_{u\Theta}$*
2. *$M_{0\Omega} \notin Q_\Omega$ (with $M_{0\Omega}$ and Q_Ω respectively the initial marking and the set of final markings of Ω)*
3. *for every marking M reachable in Ω, there exists M' reachable from M such that $M' \in Q_\Omega$*

4. *from any reachable marking M, there is no event $f \in \Sigma_\Omega$ labelling more than one enabled transition*

5. *every run starting from a marking of Q_Ω necessarily leads to a marking of Q_Ω*

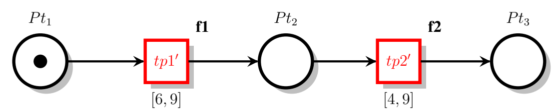

Figure 2: Ω_1 a pattern on Θ_1. Its final marking is $Pt_3 = 1$.

Time patterns represent unobservable behaviours (condition 1). In this work the study is restricted to pattern that does not contain parallelism. Condition 2 ensures that the language of a pattern does not contain the empty sequence i.e the pattern does not represent empty event sequences. Condition 3 ensures any run of the pattern is a prefix of a run for which the pattern is recognized. The patterns are deterministic (condition 4). Finally condition 5 ensures that the execution of the pattern is definitive.

Example 3. *Figure 2 gives an example of timed pattern Ω_1 on Θ_1 of Figure 1. The transitions t'_{p_1} and t'_{p_2} share their label with t_{p_1} and t_{p_2}. Back to the notion of admissible run, the run $6t'_{p_1}.7t'_{p_2}$ is admissible for Ω_1. It produces the timed sequence $\rho_1 = 6f_1.7f_2$.*

As a pattern is unobservable, its signature is given by the observable events that may be produced by the system when the pattern occurs. The pattern signature is then directly related to the (observable) executions of the system for which the pattern has occurred. The identification of such executions is formulated as a pattern-matching problem similarly to the one defined in [7]. Briefly speaking a system run matches a pattern Ω if it generates a word ρ that contains, as a subword, one of the words of Ω. For the sake of simplicity, it is said that a run matches a pattern if this run produces a timed sequence of $\mathcal{L}(\Theta)$ that matches this pattern.

Definition 5. *A timed sequence $\rho \in \mathcal{L}(\Theta)$ is matching Ω (denoted $\rho \Supset \Omega$) if there exists a subword $\rho' = (\sum_{i=1}^{j_1} \theta_i)e_{j_1} \dots (\sum_{i=j_{k-1}+1}^{j_k} \theta_i)e_{j_k}$ of ρ such that $\rho' \in \mathcal{L}(\Omega)$.*

Based on this matching notion the signature is defined as follows:

Definition 6. $Sig(\Omega) = \{\boldsymbol{P}_{\Sigma_{o\Theta}}(\rho) | \rho \in \mathcal{L}(\Theta) \wedge \rho \Supset \Omega\}$

In other words $Sig(\Omega)$ contains every observable trace ($\mathbf{P}_{\Sigma_{o\Theta}}(\rho)$ of every timed sequence ρ) of Θ (i.e $\rho \in \mathcal{L}(\Theta)$) that matches Ω ($\rho \Supset \Omega$).

Example 4. *The run $r_1 = 2t_{uo_1}.2t_1.4t_{p_1}.2t_{uo_2}.3t_{p_2}.3t_{uo_3}.2t_2.2t_3$ is an admissible run for Θ_1. It produces the timed sequence $\rho_1 = 2uo_1.2o_1.4f_1.2uo_2.3f_2.3uo_3.2o_2.2o_3$. $\rho' = 8f_1.5f_2$, ρ' is a subword of ρ_1, and $\rho' \in \mathcal{L}(\Omega_1)$, so $\rho_1 \Supset \Omega_1$.*

3.2 Characterization of a pattern signature

This approach that investigates the use of the SCGs relies on two main steps:

1. The first step is to express the behaviours of the system for which the pattern occurs as sets of two different types of constraints (Section 4):

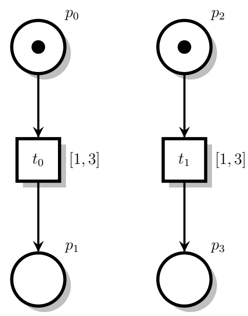

Figure 3: LTPN illustrating the problem of parallelism for a run of the SCG to be admissible for the system

- the constraints of admissibility, *i.e.* the constraints a system run must satisfy to be admissible. These constraints are obtained for every path of the system SCG taking into account the influence of each possible firing sequence on the other sequences (Section 4.1.)
- the constraints of synchronisation (Section 4.2), *i.e.* constraints that represent the different ways the system matches Ω on each path of the system SCG. For each of these ways, the path is reduced to the observable transitions and to the unobservable transitions involved in the pattern recognition (Section 4.2.1). Then the path is cut into blocks (Section 4.2.2) which are synchronized with the transitions of the pattern (Section 4.2.3)

2. The second step aims to express the constraints on firing dates of unobservable transitions of the system into constraints on the firing dates of observable ones (Section 5)

4 Pattern occurrences expressed as sets of constraints

The State Class Graph (SCG) is an abstraction of a Time Petri net that contains every behaviour of the system. Every sequence of transitions leading from the initial class to a final class (called a path in the rest of this document) is a sequence for which there exists at least a corresponding run of the system. The objective of this step is to extract from the SCG the runs of the system that are not only admissible but also that match the pattern. For this, the idea is to add additional time constraints on the firing dates of the transitions involved in the runs to guarantee these two requested characteristics.

4.1 Constraints of admissibility of a run of the system

From the different paths of the SCG it is possible to extract the runs of the system. Nevertheless, an extracted run may be a non admissible run due to the parallel behaviours in the Petri net. Therefore, it is necessary to add constraints on the transitions that are in parallel to take into account the influence of the firing of one transition on the others. Such constraints are called admissibility constraints.

Example 5. *Figure 3 shows a system with parallelism. Its SCG has 4 classes :* C_0 *(*$P_0P_2, 1 \leq t_0 \leq 3, 1 \leq t_1 \leq 3$*),* C_1 *(*$P_1P_2, 0 \leq t_1 \leq 2$*),* C_2 *(*$P_0P_3, 1 \leq t_1 \leq 2$*) and* C_3 *(*P_1P_3*). The run* $3t_0.2t_1$ *satisfies the firing domains of the SCG but is not admissible as if* t_0 *is fired at date 3,* t_1 *cannot be fired 2 time units after. An admissible run must also satisfy the following admissibility constraint* $1 \leq t_0 + t_1 \leq 3$*.*

Definition 7. *An admissibility constraint on a transition* t_m*, for a run* $r = \theta_1 t_1, \cdots, \theta_k t_k, \cdots, \theta_m t_m$*, is a constraint* $\lfloor I(t_m) \rfloor \leq \sum_{j=k+1}^{m} d_j \leq \lceil I(t_m) \rceil$ *where the variables* d_j *are the firing dates of transitions* t_j *(*$j = k+1, \cdots, m$ *), and* d_k *the enabling date of the transition* t_m *due to the firing of* t_k *at* d_k*.*

It leads to Proposition 1:

Proposition 1. *A run r of Θ is admissible if and only if it satisfies the constraints of the SCG and the admissibility constraints of the path from which this run is extracted.*

.

Example 6. *From the net given Figure 1 let us consider the path π_1: $[1,4]t_{uo_1}.[1,6]t_1.$ $[3,7]t_{p_1}.$ $[1,2]t_{uo_2}.$ $[2,4]t_{p_2}.[1,4]t_2.[0,1]t_{uo_3}.[1,2]t_3.$ t_{uo_3} is enabled after the firing of t_{uo_2} and fired after t_{p_2}. The admissibility constraint concerning t_{uo_3} will be $2 \le d_{p_2} + d_2 + d_{uo_3} \le 4$.*

4.2 Constraints of synchronisation of an admissible run with the pattern

4.2.1 Reduction of the paths

As previously said the constraints of synchronization represent the different ways the system matches a pattern, by taking into account the different paths in the SCG. Each path is defined by a sequence of transitions with their corresponding firing intervals.

On a given path, it is possible that the system matches several times the pattern. In the remainder it is considered that the matching of a pattern on a path is the first to occur in a run, *i.e.* if the pattern can be recognized multiple times in a run, but only the first recognition is considered.

The algorithm to capture this first matching is not presented here but is inspired from the chronicle recognition algorithm of [8]. Briefly, the algorithm explores the path and the pattern in parallel as binary trees also called assumptions trees. For each transition examined in a path there are three cases: (1) a transition with the same label is necessarily fired in the pattern, in such cases the algorithm explores the next transition of the pattern. (2) The transition may fire in the pattern for certain dates but not for every possible ones. In such cases the algorithm splits the exploration into two branches, one for which the algorithm considers the transition has been fired in the pattern, and another one for which it considers that the transition has not been fired yet. (3) The transition analysed in the path of the system path cannot be fired in the pattern thus the algorithm goes to the next transition in the path without evolving in the pattern exploration. At the end the algorithm returns the different paths of the SCG for which there exists a run that matches Ω. Note that, every path in the SCG of a system following assumption **A0** is finite, and the SCG contains a finite number of paths.

The paths matching the pattern are extracted from the SCG of the system and then can include observable transitions but also unobservable transitions. The next step is then to reduce this path by projection to the observable transitions and to the unobservable transitions involved into the execution of the pattern. The firing intervals of the transitions in the projected sequence are updated to take into account the projection in a way inspired by [9].

Example 7. *Let us consider the path $\pi_5 = [1,4]t_{uo1}.[1,6]t_1.[3,7]t_{p_1}.$ $[1,2]t_{uo2}.[2,4]t_{uo3}.$ $[0,2]t_{p_2}.$ $[0,2]t_3.$ $[0,4]t_2$ which is a path of the SCG of Θ_1. Its admissibility constraints are $\{2 \le d_{u_3} + d_{p_2} \le 4, 1 \le d_{p_2} + d_3 \le 2, 1 \le d_2 + d_3 \le 4\}$. The run $3t_{uo1}.2t_1.3t_{p_1}.2t_{uo2}.$ $2t_{uo3}.1t_{p_2}.$ $2t_3.1t_2$ matches the pattern Ω_1. The projection of π_5 onto the set of transitions $T_{proj} = \{t_1, t_2, t_3, t_{p_1}, t_{p_2}\}$ is given by: $\boldsymbol{P}_{T_{proj}}(\pi_5) = t_1.t_{p_1}.t_{p_2}.t_3.t_2$.*

The classes visited following π_5 are C_0($M : P0, D_0 : 1 \le t_{uo1} \le 4$), C_1 ($M : P1, D_1 : 1 \le t_1 \le 6$), C_2 ($M : P2, D_2 : 3 \le t_{p_1} \le 7$), C_3 ($M : P3, D_3 : 1 \le t_{uo2} \le 2, 1 \le t_{uo4} \le 3$), C_4 ($M : P4 * P6, D_4 : 2 \le t_{p_2} \le 4, 2 \le t_{uo3} \le 4$), C_7 ($M : P4 * P7, D_7 : 1 \le t_3 \le 2, 0 \le t_{p_2} \le 2$), C_{12} ($M : P5 * P7, D_{12} : 1 \le t_2 \le 4, 0 \le t_3 \le 2$) and C_{19} ($M : P11 * P5, D_{19} : 0 \le t_2 \le 4$).

In the reduced path obtained by projection of π_5, the first projected transition to be fired is t_1. In π_5, t_1 is fired after t_{uo1}. The firing observable interval for t_1 becomes then [2,10]. The firing interval of the next transition of the reduced path (t_{p_1}) is unchanged as it is involved in the execution of Ω as its first *event. Next, the firing of t_{p_2} follows the firing of t_{uo2} and t_{uo3} in π_5. To calculate the new firing interval of t_{p_2} after the projection, the variables y_{uo2}, y_{uo3} and y_{p2} are associated respectively with t_{uo2}, t_{uo3} and t_{p_2} such that:*

$$\begin{cases} 1 \leq y_{uo2} \leq 2 \\ 2 \leq y_{uo3} - y_{uo2} \leq 4 \\ 2 \leq y_{p2} - y_{uo2} \leq 4 \\ y_{p2} - y_{uo3} \leq 2 \end{cases} \tag{1}$$

The first equation indicates that t_{uo2} is fired between 1 and 2 tu after the firing of t_{p_1} according to C_3. The second one models that t_{uo3} is fired between 2 and 4 tu after the firing of t_{uo2} and the third one does the same for t_{p_2} according to C_4. Then the last one models the firing of t_{p_2} after t_{uo3} and the concurrency between these two transitions. The solution of this system of inequations is $3 \leq y_{p2} \leq 6$, so [3,6] is the new firing interval of t_{p_2}. Finally concurrency between t_3 and t_2 does not change their firing intervals, and the new intervals are [0,2] and [0,4] respectively.

The projection of the path π_5 gives then the reduced path $\pi = [2,10]t_1.[3,7]t_{p_1}.\ [3,6]t_{p_2}.\ [0,2]t_3.[0,4]t_2$.

4.2.2 Partitioning of a path

The partitioning aims to highlight the different synchronization points in terms of transitions between two paths. In this study the partitioning is applied on both types of reduced paths: those issued from the SCG of the system and those extracted from the SCG of the pattern, in both cases using the reduction step presented in Section 4.2.1.

Let us consider two paths π and π' such that $\pi \Supset \pi'$ (π matches π', *i.e.* there exists a run r of π such that $r \Supset \pi'$). In order to synchronize both paths, they are partitioned into blocks. The split into blocks is guided by the (unobservable) transitions that must be synchronized i.e transitions sharing the same labels in π and π'.

Definition 8. *A block w is a sequence of pairs $(\mathcal{I}, t) \in \mathcal{I}_{\mathbb{Q}_+} \times T$ where T is a set of transitions.*

Following the splitting principle two types of blocks are considered:

1. Type 1: blocks for which only the last transition of the block is an unobservable transition that must be synchronized. It is possible that the block contains only such a transition.
2. Type 2: blocks for which every transition is observable

The partitioning of a path leads for the system's path to a sequence of n blocks such that the $n-1$ first blocks include a set of observable transitions and an unobservable one, or only one unobservable transition. The n^{th} block, due to assumption **A1** contains observable transitions.

Example 8. *The partitioning of the reduced path $\pi_1 = [2,10]t_1.\ [3,7]t_{p_1}.[3,6]t_{p_2}.\ [1,2]t_2.[1,3]t_3$ is : $w_1 = [2,10]t_1.[3,7]t_{p_1}$, $w_2 = [3,6]t_{p_2}$, $w_3 = [1,2]t_2.[1,3]t_3$.*
Simillary the partitioning of π_5 is : $w_1 = [1,4]t_{uo_1}.[1,6]t_1.[3,7]t_{p_1}$, $w_2 = [1,2]t_{uo_2}.[2,4]t_{uo_3}.[0,2]t_{p_2}$, $w_3 = [0,2]t_3.[0,4]t_2$.

The same partitioning is applied to every path issued from the SCG of the pattern. The blocks obtained are all of type 1 and constituted with one couple of firing interval/unobservable transition as every transition of the pattern is unobservable.

Example 9. *For Ω_1 only one path is extracted from the SCG of the pattern, and this path is partitioned into two blocks : $w'_1 = [6,9]t'_{p_1}$ and $w'_2 = [4,9]t'_{p_2}$.*

4.2.3 Synchronization of two blocks

A block of the system is denoted w and a block of the pattern is denoted w'. Then the partionning of a path from the system is given by $\pi = w_1 \dots w_n$ and the one from the pattern is $\pi' = w'_1 \dots w'_{n-1}$ (containing one block less due to the last block of π being observable).

- For $i < n$, every block w_i is going to be synchronized with w'_i as the last transition of each block shares the same label.

  Proposition 2. *The synchronisation of two blocks $w = \mathcal{I}_1 t_1 \dots \mathcal{I}_k t_k$ and $w' = \mathcal{I}_\Omega t_\Omega$ with $\ell(t_n) = \ell(t_\Omega)$, defines the following set of constraints to be satisfied:*

$$\forall r = d_1 t_1 \dots d_k t_k \subseteq w, r \Supset \mathcal{I}_\Omega t_\Omega \Leftrightarrow \begin{cases} d_1 \in \mathcal{I}_1 \\ \dots \\ d_k \in \mathcal{I}_k \\ \sum_{i=1}^{k} d_i \in \mathcal{I}_\Omega \end{cases} \quad (2)$$

  Proof. Let us consider $r = d_1 t_1 \dots d_k t_k$ a run of $w = \mathcal{I}_1 t_1 \dots \mathcal{I}_k t_k$ such that $\forall i \in [1,k], d_i \in \mathcal{I}_i$ (imposed by the system).
  $r \Supset \mathcal{I}_\Omega t_\Omega \Leftrightarrow \exists d \in \mathcal{I}_\Omega \mid (\sum_{i=1}^{k} d_i)\ell(t_k) = d\ell(t_\Omega), \Leftrightarrow \sum_{i=1}^{k} d_i = d$ □

- For $i = n$ (i.e for the last block of π) , because there is no transition to be synchronized as it is composed of observable transitions, the set of constraints corresponds to the constraints imposed by the system on its transitions: $w_n = \mathcal{I}_1 t_{o_1} \dots \mathcal{I}_m t_{o_m}, \{d_1 \in \mathcal{I}_1, \dots d_m \in \mathcal{I}_m\}$.

Definition 9. *Let's consider the set of constraints resulting in the synchronization of two blocks defined in Equation 2. The canonical form of this set of constraints, denoted $\mathcal{P} = \{d_1 \in \mathcal{I}'_1, \dots, d_k \in \mathcal{I}'_k, \sum_{i=1}^{k} d_i \in \mathcal{I}'_\Omega\}$ is such that :*

- $\forall i \in [1,k], \forall d_i \in \mathcal{I}'_i, \exists (d_j)_{j \in [1,k], j \neq i} \in \mathcal{I}'_1 \times \dots \times \mathcal{I}'_k \setminus \mathcal{I}'_i$ *such that* $\sum_{j=1}^{k} d_j \in \mathcal{I}_\Omega$

- $\forall d \in \mathcal{I}'_\Omega, \exists (d_i)_{i \in [1,k]} \in \mathcal{I}'_1 \times \dots \times \mathcal{I}'_k$ *such that* $\sum_{i=1}^{k} d_i = d$

The canonical form of the set of constraints resulting in the synchronization of two blocks is considered in the rest of this work in order to prevent from overapproximation of the signature of Ω.

The constraints of $\mathcal{P}$ are called synchronization constraints.

Example 10. *The synchronization of $w_1 = [2,10]t_1.[3,7]t_{p_1}$ and $w'_1 = [6,9]t'_{p_1}$ results in $\mathcal{P}_1 = \{2 \leq d_1 \leq 6, 3 \leq d_{p_1} \leq 7, 6 \leq d_1 + d_{p_1} \leq 9\}$.*

If the considered path π is composed of n blocks ($n \in \mathbb{N}^*$), the synchronization with π' consists of a sequence of n sets of constraints $\mathcal{P}_1, \dots \mathcal{P}_n$. A set of constraints defines a polyhedron. The satisfaction of the sequence of polyhedra $(\mathcal{P}_1, \dots \mathcal{P}_n)$ by a run r is denoted $r \Rightarrow (\mathcal{P}_1, \dots \mathcal{P}_n)$.

According to the two types of blocks the set of polyhedra can be partitioned into three sets: $\mathcal{C}_o$ the set of polyhedra containing constraints on observable transitions, $\mathcal{C}_{o^*u}$ the set of polyhedra containing constraints on observable transitions and one constraint on an unobservable transition and $\mathcal{C}_u$ the set containing only one constraint on an unobservable transition (when there are two constraints one the same date of an unobservable transition the constraints can be merge into one).

The block partitioning induces that the last polyhedron of the sequence $(\mathcal{P}_1, \dots \mathcal{P}_n)$ contains a contraint on an observable transition, hence, it is possible to define a so called *well-formed* sequence of a resulting from the partitioning step:

Definition 10. $(\mathcal{P}_i)_{i \in [1,n]}$ *is a well-formed sequence of polyhedra if:*

- $\mathcal{P}_n \in \mathcal{C}_o$
- $\forall i \in [1, n-1], \mathcal{P}_i \in \mathcal{C}_{o^*u} \cup \mathcal{C}_u$

Proposition 3. *The sequence of polyhedra resulting from the synchronization of* π *and* π' *is well-formed.*

Proposition 3 is true according to the considered partitioning.

5 Translation into observable constraints

The pattern signature contains observable traces. The constraints of admissibility and synchronization calculated in the previous sections need then to be rewritten to be expressed as observable constraints i.e constraints on observable dates only. The mechanism of this translation is presented first in the case of the admissibility constraints. Then it is presented for the synchronisation constraints in the case of two blocks, and generalized to the case of a path containing n blocks.

5.1 Towards observable admissibility constraints

The following exposes the translation of the admissibility constraints defined in Section 4.1. The aim of this step is to express the admissibility constraints into contraints on the observable transitions of a path.

Let us consider a path $\pi = \mathcal{I}_0 t_0 \dots \mathcal{I}_h t_h . \mathcal{I}_i t_i . \mathcal{I}_j t_j . \mathcal{I}_k t_k . \mathcal{I}_l t_l .... \mathcal{I}_n t_n$ each time interval is relative to the previous firing date of transition. Let us suppose t_h, t_k and t_l being observable transitions and t_i, t_j unobservable ones. $\mathcal{I}_k$ is relative to the firing date of t_j which is relative to the firing date of t_i and so on. That means that the firing date of t_k depends on the firing dates of all the previous transitions in the path π. Therefore, if t_i is concerned by an admissibility constraint (i.e $\alpha \leq d_i \leq \beta$), the firing date of t_k is impacted. The admissibility constraint $\alpha \leq d_i \leq \beta$ ($d_i \in [\alpha, \beta]$) is then expressed by a new constraint given by $[\alpha', \beta'] = [\alpha, \beta] + I_j + I_k$. This corresponds to the translation of the admissibility constraint into an observable constraint. Same reasoning can be applied if both t_i and t_j are involved in the admissibility constraint ($\alpha \leq d_i + d_j \leq \beta$).

Let us now consider an admissibility constraint on t_k and t_l: $\alpha \leq d_k + d_l \leq \beta$. As t_l is relative to t_k which is observable, no change is needed. For t_k its firing depends on all the unobservable transitions fired between t_h and t_k, so the admissibility constraint becomes $[\alpha', \beta'] = [\alpha, \beta] + \mathcal{I}_i + \mathcal{I}_j$ with $(\mathcal{I}_i + \mathcal{I}_j)$ given eventually by an admissibility constraint.

This translation is applied to every admissibility constraint, that leads to a new set of observable constraints. The set of observable admissibility constraints of a path π is noted Π^a.

Example 11. *The set of admissible constraints of* $\pi_5 = [1,4]t_{uo1}.[1,6]t_1.[3,7]t_{p_1}.\ [1,2]t_{uo2}.[2,4]t_{uo3}.\ [0,2]t_{p_2}.[0,2]t_3.\ [0,4]t_2$ *is* $\{(1)2 \leq d_{uo_3} + d_{p_2} \leq 4, (2)1 \leq d_{p_2} + d_3 \leq 2, (3)1 \leq d_2 + d_3 \leq 4\}$. *Let us consider the constraint (1). As the transitions of (1) are unobservable, it is deported onto the next observable transition (t_3 in this case), relatively to the previous one (t_1 here). Between t_1 and t_3 the transitions fired are $t_{p_1}, t_{uo_2}, t_{uo_3}$ and t_{p_2}. Thus the new constraint will be* $d_3 \in [\alpha', \beta'] = [2,4] + \mathcal{I}_{p_1} +$

$\mathcal{I}_{uo_2} + \mathcal{I}_3$, *i.e.* $d_3 \in [7, 15]$. *Let us now consider (2). There is one observable transition involved in (2), thus the updated constraint will be* $d_3 \in [\alpha', \beta'] = [1, 2] + \mathcal{I}_{p_1} + \mathcal{I}_{uo_2} + \mathcal{I}_{uo_3}$, *i;e.* $d_3 \in [7, 15]$. *Finally, when it comes to (3),* t_2 *is relative to* t_3 *so no change is needed.* t_3 *is relative to* t_1*, and so depends on the firing (as before) of* $t_{p_1}, t_{uo_2}, t_{uo_3}$*. As the admissibility constraint (1) gives the new sum of firing intervals* $(\mathcal{I}_{uo_3}+\mathcal{I}_{p_2})$*, the new constraint will be* $(d_3+d_2) \in [\alpha', \beta'] = [1, 4]+\mathcal{I}_{p_1}+\mathcal{I}_{uo_2}+(\mathcal{I}_{uo_3}+\mathcal{I}_{p_2})$*, i.e.* $(d_3 + d_2) \in [7, 17]$.

The set of observable admissibility constraints of the path π_5 *is* $\Pi^a_5 = \{7 \leq d_3 \leq 15, 7 \leq d_2+d_3 \leq 17\}$.

5.2 Towards observable synchronization constraints

The aim of this step is to translate the synchronisation constraints into constraints involving only observable transitions. For this, the firing dates of unobservable transitions that appear in the sets of synchronisation constraints (i.e in the sets of polyhedra $\mathcal{P}_i$) are eliminated from the constraints using a variable change.

5.2.1 Case of 2 polyhedra

Let us consider the couple of paths $\pi = \mathcal{I}_1 t_1 \dots \mathcal{I}_{n-1} t_{n-1}.\mathcal{I}_p t_p.\mathcal{I}_n t_n$ (from the system) and $\pi' = \mathcal{I}'_p t'_p$ (from the pattern) such that $\ell(t_p) = \ell(t'_p)$ i.e t_p and t'_p are unobservable . As described in Section 4.2.2, π can be partitioned into two blocks: $w_1 = \mathcal{I}_1 t_1 \dots \mathcal{I}_p t_p$ and $w_2 = \mathcal{I}_n t_n$. The polyhedra resulting from the synchronization of w_1 with $w'_1 = \pi'$ and the constraint of w_2 (see Section 4.2.3) are:

$$\mathcal{P}_1 = \{d_1 \in \mathcal{I}_1, \dots, d_p \in \mathcal{I}_p, (\sum_{j=1}^{n-1} d_i) + d_p \in \mathcal{I}'_p\}, \mathcal{P}_2 = \{d_n \in \mathcal{I}_n\}$$

The constraint on d_p in $\mathcal{P}_1$ (i.e. on the firing date of t_p) can be eliminated by the variable change $d'_n = d_p + d_n$, leading to two new polyhedra $\mathcal{P}_{o,1}$ and $\mathcal{P}_{o,2}$, with constraints related to firing dates of observable transitions only:

$$\mathcal{P}_{o,1} = \{d_1 \in \mathcal{I}_1, \dots, d'_n \in (\mathcal{I}_p + \mathcal{I}_n), (\sum_{j=1}^{n-1} d_i) + d'_n \in \mathcal{I}'_p + \mathcal{I}_n\} \quad (3)$$

$$\mathcal{P}_{o,2} = \{d'_n \in \mathcal{I}_p + \mathcal{I}_n\} \quad (4)$$

Proposition 4. *(1) Let* r_o *be a sequence of observable transitions and their dates of firing satisfying* $(\mathcal{P}_{o,1}, \mathcal{P}_{o,2})$*. There exists* r *a run of* π *such that* $\boldsymbol{P}_{T_{o\Theta}}(r) = r_o$ *and* $r \models (\mathcal{P}_1, \mathcal{P}_2)$.
(2) Let r *be a run of* π *such that* $r \models (\mathcal{P}_1, \mathcal{P}_2)$*, then* $\boldsymbol{P}_{T_{o\Theta}}(r) \models (\mathcal{P}_{o,1}, \mathcal{P}_{o,2})$.

5.2.2 Generalisation

The transformation process described for a synchronisation step leading to two sets of synchronisation constraints (i.e two polyhedra) (Section 5.2.1) can be generalized to the case of a sequence of n polyhedra $(\mathcal{P}_1, \dots \mathcal{P}_n)$. For this the variable change is applied as many times as the number of constraints that relate only to an unobservable transition in the sequence.

More precisely, the variable change is applied to the first polyhedron of the sequence and then propagated from one polyhedron to another until the end of the sequence .

According to Definition 10 and Proposition 3 two applications of translation, $Trans_1$ and $Trans_2$, are defined:

Definition 11. *$Trans_1$: $\mathcal{C} \times \mathcal{C} \to \mathcal{C}$ and $Trans_2$: $\mathcal{C} \times \mathcal{C} \to \mathcal{C}$ such that:*

- $Trans_1(\mathcal{P}_1, \mathcal{P}_2) =$
  1. *$\mathcal{P}_{o,1}$ if $(\mathcal{P}_1, \mathcal{P}_2) \in (\mathcal{C}_{o^*u} \times (\mathcal{C}_{o^*u} \setminus \mathcal{C}_u)) \cup (\mathcal{C}_{o^*u} \times \mathcal{C}_o)$ where $\mathcal{P}_{o,1}$ is obtained processing the variable change $d'_n = d_p + d_n$ in $\mathcal{P}_1 = \{d_1 \in \mathcal{I}_1, \dots d_{n-1} \in \mathcal{I}_{n-1}, d_p \in \mathcal{I}_p, (\sum_{j=1}^{n-1} d_j) + d_p \in \mathcal{I}'_p\}$, with $\mathcal{P}_2 = \{d_n \in \mathcal{I}_n, \dots\}$ (see 3)*
  2. *$\emptyset$ if $(\mathcal{P}_1, \mathcal{P}_2) \in (\mathcal{C} \times \mathcal{C}_u) \cup (\emptyset \times \mathcal{C})$*
- $Trans_2(\mathcal{P}_1, \mathcal{P}_2) =$
  1. *$\mathcal{P}_{o,2}$ if $(\mathcal{P}_1, \mathcal{P}_2) \in (\mathcal{C}_{o^*u} \times (\mathcal{C}_{o^*u} \setminus \mathcal{C}_u)) \cup (\mathcal{C}_{o^*u} \times \mathcal{C}_o)$ where $\mathcal{P}_{o,2}$ is obtained processing the variable change $d'_n = d_p + d_n$ in $\mathcal{P}_2 = \{d_n \in \mathcal{I}_n, \dots\}$, with $\mathcal{P}_1 = \{d_1 \in \mathcal{I}_1, \dots d_{n-1} \in \mathcal{I}_{n-1}, d_p \in \mathcal{I}_p, (\sum_{j=1}^{n-1} d_j) + d_p \in \mathcal{I}'_p\}$ (see 4)*
  2. *$\mathcal{P}_{o,1}$ if $(\mathcal{P}_1, \mathcal{P}_2) \in (\mathcal{C} \times \mathcal{C}_u)$ where $\mathcal{P}_{o,1}$ is obtained processing the variable change $d'_{p_2} = d_{p_1} + d_{p_2}$ in $\mathcal{P}_1 = \{d_1 \in \mathcal{I}_1, \dots d_{n-1} \in \mathcal{I}_{n-1}, d_{p_1} \in \mathcal{I}_{p_1}, (\sum_{j=1}^{n-1} d_j) + d_{p_1} \in \mathcal{I}'_{p_1}\}$ with $\mathcal{P}_2 = \{d_{p_2} \in \mathcal{I}_{p_2} \cap \mathcal{I}'_{p_2}\}$ (see 3)*
  3. *$\emptyset$ if $(\mathcal{P}_1, \mathcal{P}_2) \in \emptyset \times \mathcal{C}$*

When the variable change is applied to the first polyhedron of the sequence the assigned variable appears in the second polyhedron as well. Then as the unobservable constraints of this polyhedron need another variable change, the application $P(\mathcal{P}_1, \dots, \mathcal{P}_n)$ =($Trans_1(\mathcal{P}_1, \mathcal{P}_2), P(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3, \dots \mathcal{P}_n)$) is the application that propagates the variable changes towards a sequence of polyhedra (if $n = 2$ $P(\mathcal{P}_1, \mathcal{P}_2) = (Trans_1(\mathcal{P}_1, \mathcal{P}_2), Trans_2(\mathcal{P}_1, \mathcal{P}_2))$.

Proposition 5. *Let $(\mathcal{P}_i)_{i\in[1,n]}$ be the sequences of polyhedra resulting from the synchronisation of π with π'. Let's also consider $(\mathcal{P}_{o,j})_{j\in[1,n]}$ such that $(\mathcal{P}_{o,j})_{j\in[1,n]} = P(\mathcal{P}_1, \dots, \mathcal{P}_n)$.*

1. *$\forall r_o$ such that $r_o \models (\mathcal{P}_{o,j})_{j\in[1,n]}$, there exists r such that $\boldsymbol{P}_{\Sigma_o}(r) = r_o$ and $r \models (\mathcal{P}_i)_{i\in[1,n]}$.*
2. *$\forall r$ such that $r \models (\mathcal{P}_i)_{i\in[1,n]}$, $\boldsymbol{P}_{\Sigma_o}(r) \models (\mathcal{P}_{o,j})_{j\in[1,n]}$.*

Proof. By developing the application P, we have $P(\mathcal{P}_1, \dots, \mathcal{P}_n)$ =($Trans_1(\mathcal{P}_1, \mathcal{P}_2)$, $P(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3, \dots \mathcal{P}_n)$) = ($Trans_1(\mathcal{P}_1, \mathcal{P}_2)$, $Trans_1(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3)$, $P(Trans_2(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3), \mathcal{P}_4, \dots \mathcal{P}_n)$). Every $\mathcal{P}_{o,j}, j \in [1, n]$ will be a succession of applications of $Trans_2$ $(j-1)$ times and one application of $Trans_1$. Then by recurrence on $Trans_2$ it can be shown that the variable change induced by $(j-1)$ applications of $Trans_2$ does not affect the observable dates of the observable transitions, and then points 1 and 2 are true. □

We denote $\Pi^o = \cup_{j=1}^{n} \mathcal{P}_{o,j}$ the set of polyhedra resulting from the translation of the synchronisation constraints of path.

Example 12. *The synchronisation of π with π' leads to the sequence $(\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3)$ where $\mathcal{P}_1 = \{2 \le d_1 \le 6, 3 \le d_{p_1} \le 7, 6 \le d_1 + d_{p_1} \le 9\}, \mathcal{P}_2 = \{4 \le d_{p_2} \le 6\}, \mathcal{P}_3 = \{1 \le d_2 \le 2, 1 \le d_3 \le 3\}$. The application of P to $(\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3)$ is explicited:*

- $P(\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3)$ = ($Trans_1(\mathcal{P}_1, \mathcal{P}_2), P(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3)$ =
  ($Trans_1(\mathcal{P}_1, \mathcal{P}_2)$, $Trans_1(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3)$, $Trans_2(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3)$)

- *$Trans_1(\mathcal{P}_1, \mathcal{P}_2) = \emptyset$ because $\mathcal{P}_2 \in \mathcal{C}_u$*
- *$Trans_2(\mathcal{P}_1, \mathcal{P}_2) = \{2 \leq d_1 \leq 6, 3+4 \leq d_{p_1} + d_{p_2} \leq 9+6, 6+3 \leq d_1 + d_{p_1} + d_{p_2} \leq 9+6\}$ $= \{2 \leq d_1 \leq 6, 7 \leq d'_{p_2} \leq 9, 9 \leq d_1 + d'_{p_2} \leq 15\}$*
- *$Trans_1(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3) = \{2 \leq d_1 \leq 6, 8 \leq d_2 \leq 15, 10 \leq d_1 + d_2 \leq 17\}$*
- *$Trans_2(Trans_2(\mathcal{P}_1, \mathcal{P}_2), \mathcal{P}_3) = \{8 \leq d_2 \leq 15, 1 \leq d_3 \leq 3\}$*

The translation into observable constraints of the synchronisation constraints of the path π_1 is $(\mathcal{P}_{o,1}, \mathcal{P}_{o,2}, \mathcal{P}_{o,3})$ with: $\mathcal{P}_{o,1} = \emptyset$, $\mathcal{P}_{o,2} = \{2 \leq d_1 \leq 6, 8 \leq d_2 \leq 15, 10 \leq d_1 + d_2 \leq 17\}$, $\mathcal{P}_{o,3} = \{8 \leq d_2 \leq 15, 1 \leq d_3 \leq 3\}$.

6 Finite characterization of the pattern Signature

As it as been shown previously, it is possible for a path to represent the first occurrences of the pattern as a set of observable constraints. Repeating this operation as many times as the number of paths for which there exists at least one run that matches the pattern, leads to the signature of the pattern for the system.

Let us consider $\pi_1, \dots \pi_n$ the n paths of the SCG of the system Θ that correspond to the different recognitions of Ω (see Section 4.2.1) . For each $\pi_i, i \in [1, n]$, the set of constraints issued from the synchronisation of π_i with Ω translate into observable constraints plus the admissibility constraints of π_i is denoted Π_i ($\Pi_i = \Pi_i^o + \Pi_i^a$).

To each solution X_s of Π_i is associated a sequence of dates/transitions, as each variable in the inequations of Π_i is related to an observable transition, and the order of the inequations gives us a particular transition sequence. In other words, every π_i has a sequence of observable transitions associated denoted $\mathcal{T}_o^i$. Let us denote $\Pi = \cup_{i=1}^n \Pi_i$ the set containing every set of constraints representing the occurrence of Ω in Θ.

Proposition 6. *$X_s = \{\theta_1, \dots \theta_k\}$ is solution of $\Pi = \cup_{i=1}^n \Pi_i \Leftrightarrow \exists i \in [1, n] \mid r_o = \theta_1 t_1 \dots \theta_k t_k, \mathcal{T}_o^i = t_1.t_2 \dots t_k$ is such that $\rho_o = \theta_1 \ell(t_1) \dots \theta_k \ell(t_k) \in Sig(\Omega)$.*

Proof. ($\Rightarrow$) Let's consider $i \in [1, n]$ and $X_s = \{\theta_1, \dots \theta_k\}, \mathcal{T}_o^i = t_1.t_2 \dots t_k$ a solution of Π_i. r_o is an admissible observable run of Θ because its transition sequence comes from the system and r_o satisfies the admissibility constraints of the system ($\Pi_i^a \subseteq \Pi_i$). r_o also satisfies Π_i^o thus by Proposition 5 there exists a run r of Θ such that $r \Supset \Omega$ and $\mathbf{P}_{T_o\Theta}(r) = r_o$. So if $r_o = \theta_1 t_1 \dots \theta_k t_k$, so $\rho_o = \theta_1 \ell(t_1) \dots \theta_k \ell(t_k) \in Sig(\Omega)$.

($\Leftarrow$) If $\rho_o = \theta_1 \ell(t_1) \dots \theta_k \ell(t_k) \in Sig(\Omega)$ there exists a run r of Θ such that $r \Supset \Omega$ and $r_o = \theta_1 t_1 \dots \theta_k t_k = \mathbf{P}_{T_o\Theta}(r)$. Thus by Propositions 2 and 5 there exists $\Pi_i, i \in [1, n]$ representing this run. Thus X_s the set of dates of ρ_o is a solution of Π_i. □

Proposition 6 shows that there exists a finite set of constraints characterizing $Sig(\Omega)$, and so there exists a finite characterization of $Sig(\Omega)$.

Example 13. *There are 6 paths in the SCG of Θ_1 for which it is possible to match Ω_1:*
$\pi_1 = t_{uo_1}.t_1.t_{p_1}.t_{uo_2}.t_{p_2}.t_2.t_{uo_3}.t_3$, $\pi_2 = t_{uo_1}.t_1.t_{p_1}.t_{uo_2}.t_{uo_3}.t_3.t_{p_2}.t_2$, $\pi_3 = t_{uo_1}.t_1.t_{p_1}.t_{uo_2}.t_{p_2}.t_{uo_3}.t_3.t_2$, $\pi_4 = t_{uo_1}.t_1.t_{p_1}.t_{uo_2}.t_{p_2}.t_{uo_3}.t_2.t_3$, $\pi_5 = t_{uo_1}.t_1.t_{p_1}.t_{uo_2}.t_{uo_3}.t_{p_2}.t_3.t_2$ *and* $\pi_6 = t_{uo_1}.t_1.t_{p_1}.t_{uo_2}.t_{uo_3}.t_{p_2}.t_2.t_3$.
Π *the set of constraints characterizing the recognition of Ω_1 toward Θ_1 is :* $\Pi = \{$
$\Pi_1 = \{2 \leq d_1 \leq 6, 8 \leq d_2 \leq 15, 10 \leq d_1 + d_2 \leq 17, 1 \leq d_3 \leq 3, 7 \leq d_2 + d_3 \leq 15\}$,
$\Pi_2 = \{2 \leq d_1 \leq 6, 9 \leq d_3 \leq 15, 12 \leq d_1 + d_3 \leq 17, 1 \leq d_2 \leq 5, 7 \leq d_2 + d_3 \leq 13\}$,

$\Pi_3 = \{2 \leq d_1 \leq 6, 10 \leq d_3 \leq 15, 12 \leq d_1 + d_3 \leq 19, 0 \leq d_2 \leq 1, 7 \leq d_2 + d_3 \leq 17\},$
$\Pi_4 = \{2 \leq d_1 \leq 6, 7 \leq d_2 \leq 17, 9 \leq d_1 + d_2 \leq 19, 0 \leq d_3 \leq 2, 7 \leq d_2 + d_3 \leq 15\},$
$\Pi_5 = \{2 \leq d_1 \leq 6, 7 \leq d_3 \leq 15, 9 \leq d_1 + d_3 \leq 17, 0 \leq d_2 \leq 4, 7 \leq d_2 + d_3 \leq 17\},$
$\Pi_6 = \{2 \leq d_1 \leq 6, 8 \leq d_2 \leq 15, 10 \leq d_1 + d_2 \leq 17, 0 \leq d_3 \leq 1, 7 \leq d_2 + d_3 \leq 15\}\}$

7 Conclusion

This paper develops a method to characterize in a finite way the signature of a timed pattern in an acyclic safe Time Petri Net. The proposed characterization is constructive and is divided into two main steps. First, the executions of the system containing the pattern are synchronized with the pattern itself. The synchronizations are based on the state class graph of the system and are formulated as a pattern matching problem. The synchronized executions are then represented as sets of constraints on the firing dates of the system transitions. In a second step, these constraint sets are transformed into observable constraints, i.e. constraints on the dates of the system observable transitions that must be satisfied so that the pattern occurs.

Future work includes the extension of this characterization to a larger number of systems and different types of patterns, for example pattern with parallelism. Another challenging issue is the signature based diagnosability. The comparison of the signature of the pattern and the signature of the non-occurence of the pattern could be investigated. Finally, in the case where the pattern is not diagnosable, the signature can be relevant to identify why the pattern is not diagnosable in order to repair the system.

References

[1] Ramla Saddem, Armand Toguyéni, and Tagina Moncef. Algorithme d'interprétation d'une base de signatures temporelles causales pour le diagnostic en ligne des systèmes à événements discrets. In *9th International Conference on Modeling, Optimization & SIMulation*, 2012.

[2] Ramla Saddem and Alexandre Philippot. Causal Temporal Signature from diagnoser model for online diagnosis of Discrete Event Systems. In *2014 International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 551–556, Metz, France, November 2014. IEEE.

[3] T. Jéron, H. Marchand, S. Pinchinat, and M.-O. Cordier. Supervision patterns in discrete event systems diagnosis. In *2006 8th International Workshop on Discrete Event Systems*, pages 262–268, 2006.

[4] HE. Gougam, Y. Pencolé, and A. Subias. Diagnosability analysis of patterns on bounded labeled prioritized Petri nets. *Discrete Event Dyn Syst 27*, pages 143–180, 2017.

[5] Yannick Pencolé and Audine Subias. Timed pattern diagnosis in timed workflows: a model checking approach. *IFAC-PapersOnLine*, 51(7):94–99, 2018. 14th IFAC Workshop on Discrete Event Systems WODES 2018.

[6] Bernard Berthomieu and Miguel Menasche. An enumerative approach for analyzing time Petri nets. In *Proceedings IFIP*, pages 41–46. Elsevier Science Publishers, 1983.

[7] Yannick Pencolé and Audine Subias. Diagnosability of event patterns in safe labeled time Petri nets: A model-checking approach. *IEEE Transactions on Automation Science and Engineering*, pages 1–12, 2021.

[8] Christophe Dousson, Paul Gaborit, and Malik Ghallab. Situation recognition: Representation and algorithms. In *IJCAI*, volume 93, pages 166–172, 1993.

[9] Xu Wang, Cristian Mahulea, and Manuel Silva. Fault diagnosis graph of time Petri nets. In *2013 European Control Conference (ECC)*, pages 2459–2464, 2013.