Diagnosabilization of Time Petri net for timed fault

Camille Coquand, Yannick Pencolé, Audine Subias

Diagnosabilization of Time Petri net for timed fault

Camille Coquand * **Yannick Pencolé** ** **Audine Subias** *

* *CNRS, LAAS, Univ de Toulouse, INSA, LAAS, F-31400 Toulouse, France (e-mail: firstname.lastname@laas.fr).*
** *CNRS, LAAS, Univ de Toulouse, LAAS, F-31400 Toulouse, France (e-mail: yannick.pencole@laas.fr)*

Abstract: Diagnosability is the property of a system to have sufficient observable information to guarantee the diagnosis of a fault. Here, the considered fault is a timed fault, i.e. an unobservable event that occurs in bounded time since the start of the system. Starting with a system modeled as a Time Petri net that is not diagnosable, this work proposes a method that provides adjustments by restriction of static time intervals to ensure the system becomes Δ-diagnosable for that fault. These adjustments are characterized by a set of constraints over interval bounds and then provide a set of solutions, if any, to ensure the diagnosability of the system.

Keywords: Discrete event system, Time Petri nets, diagnosabilization, critical pair.

1. INTRODUCTION

Fault diagnosis of a partially observable dynamic discrete event system is the problem of determining the occurrence in the system of unobservable faulty events based on the sequence of observations emitted by the system (Zaytoon and Lafortune (2013)). In this paper, the complete behaviour of the system is modeled as a Time Petri Net (Berthomieu and Menasche (1983)) and a fault is defined as the occurrence of a single faulty event in a *bounded time interval*. In other words, a fault is characterized by the occurrence of single event that is considered to be abnormal as soon as it occurs within a fixed set of time bounds: such a *timed fault* can model, for instance, time shifts of critical events, delays... This paper deals with the Δ-diagnosability of such timed faults. If the system is Δ-diagnosable, it means that the occurrence of the timed fault is always determined with certainty after Δ time units (Tripakis (2002); Basile et al. (2016); Pencolé and Subias (2021)). Here, we suppose that the system is not yet Δ-diagnosable and we aim at providing a set of *adjustments* in the system to ensure that the system becomes Δ-diagnosable. In this paper, we firstly propose a formal definition of this problem that we call *diagnosabilization*. More precisely, the type of modifications that we propose is to *restrict* some static time intervals associated with some transitions of the Time Petri nets. This paper then proposes a first and original method that is based on a diagnosability analysis detailed in Coquand et al. (2022) and that characterizes a set of solutions as constraints over interval bounds in static intervals that ensures, if it exists, that the system can be Δ-diagnosable by just applying time restrictions complying with this set of constraints.

The problem of synthesizing requirements to design a discrete event system so that it is diagnosable has not yet received a lot of attention in the literature. Recently, He et al. (2022) proposes to use a Sat Modulo Theory method to add delay blocks into an untimed automaton to disambiguate observable events and ensure the overall diagnosability of the system. As far as timed discrete event systems are concerned, closest contributions are not directly focussing on fixing diagnosability but more on repairing the model wrt to the real process via processing mining techniques (Fahland and van der Aalst (2012)). Basile et al. (2015) proposes to repair Time Petri Nets by adjusting the static intervals, using Mixed-Integer Linear Programming techniques.

The paper is organized as follows. Section 2 recalls the necessary formal background and introduces the diagnosabilization problem. Then Section 3 presents a diagnosability analysis based on the enumeration of critical pairs. Using this enumeration, Section 4 proposes time constraint synthesis for which if the system satisfies them it will be Δ-diagnosable.

2. DIAGNOSABILIZATION OF TIME PETRI NETS

After some reminders on Time Petri nets and their semantics, this section formally introduces the problem of diagnosabilization.

2.1 Safe Labeled Time Petri Nets

Definition 1. A safe 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 modeling the arcs between the transitions and the places
- Σ is a finite alphabet of transition labels
- $\ell : T \to \Sigma$ is the transition labeling function
- $I_s : T \to I_{\mathbb{Q}_+}$ is a static interval function $I_s(t)$, for which the lower bound, also called the date of earlier firing is denoted $\downarrow(I_s(t)) \in \mathbb{Q}_+$, and its upper bound, also called the date of later firing, is denoted $\uparrow(I_s(t)) \in \mathbb{Q}_+ \cup \{+\infty\}$

A marking M of the net is a function M: $P \to \{0,1\}$.

The *preset* of a transition t is the set of input places $pre(t) = \{p \in P |\ (p,t) \in A\}$, and similarly the *postset* of t is the set of output places $post(t) = \{p \in P |\ (t,p) \in A\}$. For a *safe* LTPN, a state is a couple $S = \langle M, I\rangle$ where I is the partial firing interval application (I: $T \to I_{\mathbb{Q}_+}$) that maps to any transition a time interval of $\mathbb{Q}_+$ in which t can be fired as soon as it is enabled. $S_0 = \langle M_0, I_0\rangle$ 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 $I_0(t) = \varnothing$. For a marking M, a transition t is firable at the date θ if and only if:

- t is enabled (*i.e.* $\forall p \in pre(t)$, $M(p) = 1$)
- $\theta \in I(t)$ and for all t' enabled by M, $\theta \leq \uparrow(I(t'))$

The firing 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 in M and still enabled in M', $I(t') = [a,b] \Rightarrow I'(t') = [\max(0, a-\theta), b-\theta]$
- for every transition t' enabled by M', and for each transition disabled by the firing of t and newly enabled by it (loops), $I'(t') = I_s(t')$

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 N is said to be admissible if there exists $\{S_1, \dots S_n\} \in R(N, S_0)^n$ 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 relative date of firing of the label e. A run produces a unique timed sequence.

Definition 2. The language $\mathcal{L}(N)$ of a LTPN N is the set composed of every timed sequence ρ such that there exists an admissible run for N $r = \theta_1 t_1 \dots \theta_n t_n$ with $\rho = \theta_1 \ell(t_1) \dots \theta_n \ell(t_n)$.

2.2 System and fault modelings

A system is modeled as a partially observable safe LTPN. The set of observable (resp. unobservable) events is denoted Σ^o (resp. Σ^u). The following assumptions are considered:

- **A0** each static firing interval $I_s(t)$ is bounded ($\forall t \in T$, $\uparrow(I_s(t)) < +\infty$);
- **A1** the system is ultimately observable, that is at any time, from any reachable state, every run will eventually lead to the fire of an observable transition in bounded time;
- **A2** the system has no zeno run (a zeno run is an infinite sequence of transitions that can occur in a finite amount of time).

Example 1. Figure 1 shows an example of a system. The observable alphabet of the system is $\Sigma^o_\Theta = \{o1, o2, o3\}$. The unobservable one is $\Sigma^u_{\Theta_0} = \{f, uo, l\}$. The blue (resp. the red) transitions are labeled by observable (resp. unobservable) events. $r = 3t_0.7t_1.1t_2.3t_3.1t_4.5t_6.3t_0$ is a run of Θ. Its associated *timed sequence* is $\rho = 3o1.7f.1o2.3f.1o2.5o3.3o1$.

In untimed DES, a fault is modeled as a single event that may occur in the system. One way to refine this modeling is to add a time constraint on the occurrence of the event, as the event may only become troublesome after a certain time in the system's life. A timed fault is modeled here as an unobservable event that may occur in the system in a finite time window.

Definition 3. A timed fault Ω_f over a system Θ is an unobservable event $f \in \Sigma^u$ associated with a bounded rational interval $I_{\Omega_f} \in \mathbb{Q}^2_+$. The bounds are denoted a_{Ω_f} and b_{Ω_f}. The language associated with Ω_f is $\mathcal{L}(\Omega_f) = \{d_i f | d_i \in I_{\Omega_f}\}$.

As most of the systems are maintained on a regular basis, dealing with a fault occurring in a finite time window is a reasonable hypothesis.

The occurrence of the fault in the system is formulated as a fault matching problem (Coquand et al. (2021)). It can be seen as a synchronization between the system and the fault.

Definition 4. A timed sequence $\rho \in \mathcal{L}(\Theta)$ matches a timed fault Ω_f (denoted $\rho \ni \Omega_f$) if there exists a sub-word ρ' of ρ (i.e ρ' is an ordered set of events extracted from ρ) such that $\rho' \in \mathcal{L}(\Omega_f)$.

A run matches a fault if its timed sequence matches the fault. If a run matches many times a timed fault, only the first matching is considered.

Example 2. Figure 2 shows an example of a timed fault on Θ. The timed sequence ρ of Example 1 matches Ω_f ($\rho \ni \Omega_f$) as $\rho' = 10f$ is a sub-word of ρ, and $\rho' \in \mathcal{L}(\Omega_f)$. As the first occurrence of f in ρ matches Ω_f, no other matching is considered.

The projection of a timed sequence onto the observable alphabet of the system (also called observable timed trace) is defined as follows:

- $\mathbf{P}_{\Sigma_\Theta \to \Sigma^o_\Theta}(\theta_1 e_1.\theta_2 e_2 \dots \theta_n e_n) = \theta_1 e_1.\mathbf{P}_{\Sigma_\Theta \to \Sigma^o_\Theta}(\theta_2 e_2 \dots \theta_n e_n)$ if $e_1 \in \Sigma^o_\Theta$
- $\mathbf{P}_{\Sigma_\Theta \to \Sigma^o_\Theta}(\theta_1 e_1.\theta_2 e_2 \dots \theta_n e_n) = \mathbf{P}_{\Sigma_\Theta \to \Sigma^o_\Theta}((\theta_1 + \theta_2) e_2 \dots \theta_n e_n)$ otherwise

In this work a method is proposed to guarantee the diagnosis of a timed fault on a time system. A guaranteed diagnosis is a diagnosis for which there is no ambiguity whether the fault has occurred or not. In the diagnosis community this notion is called diagnosability. Diagnosability is the property of a system that guarantees that there is enough observations to decide that a fault has definitively occurred a bounded amount of time after its occurrence. In timed systems it can be reformulated in terms of timed elapsed since the occurrence of the fault. Based on the definition of Pencolé and Subias (2021), diagnosability for timed fault can be defined as:

Definition 5. Θ is said to be Ω_f-diagnosable if $\exists \tau \in \mathbb{R}_+$ *s.t.* $\forall (\rho_1, \rho_2) \in \mathcal{L}(\Theta)^2, \rho_1 = \rho'_1.\rho''_1, time(\rho''_1) \geq \tau, \rho'_1 \ni \Omega_f \wedge \mathbf{P}_{\Sigma \to \Sigma^o_\Theta}(\rho_2) = \mathbf{P}_{\Sigma \to \Sigma^o_\Theta}(\rho_1) \Rightarrow \rho_2 \ni \Omega_f$.

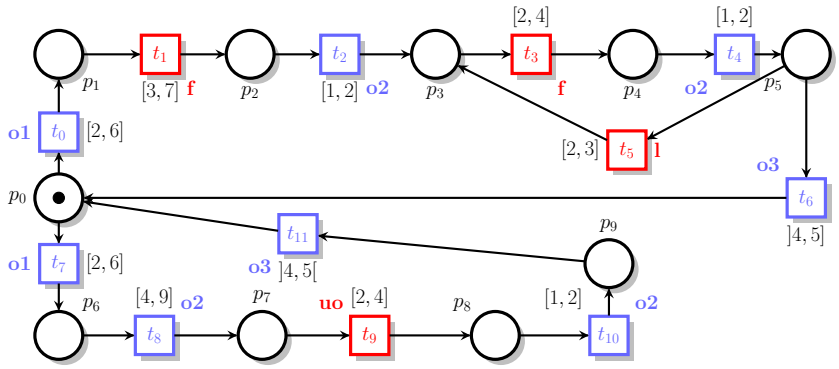

Fig. 1. System Θ

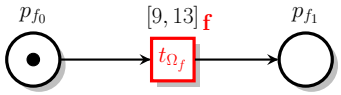

Fig. 2. Fault Ω_f on Θ: event f between dates 9 and 13 after the start of the system

Where *time* is the function that associates with each timed sequence its duration. There also exists a restricted version of the diagnosability definition where the time bound τ is fixed ($\tau = \Delta$) which is a property that ensures the timed fault will be diagnosed with certainty Δ time units after its occurrence in any case: this property is the Δ-diagnosability.

Definition 6. Let be $\Delta \in \mathbb{R}_+$. Θ is said to be Δ-Ω_f-diagnosable if $\forall(\rho_1, \rho_2) \in \mathcal{L}(\Theta)^2$, $\rho_1 = \rho_1'.\rho_1'', time(\rho_1'') \geq \Delta$, $\rho_1' \ni \Omega_f \wedge \mathbf{P}_{\Sigma \to \Sigma_\Theta^o}(\rho_2) = \mathbf{P}_{\Sigma \to \Sigma_\Theta^o}(\rho_1) \Rightarrow \rho_2 \ni \Omega_f$.

2.3 Problem statement

If a system Θ is not diagnosable, the diagnosis of a timed fault Ω_f cannot be guaranteed. Therefore guaranteeing the diagnosis of Ω_f requires changes in the model of Θ. As it can be expensive to add sensors on the system, we propose to focus on the development of methods that only modify the scheduling of the system, *i.e.* modifying the static time intervals of the system. To do so, we suppose in the following that there exists a subset of transitions in the system whose static time intervals can be adjusted: in other words, the bounds of these intervals are now considered as parameters. This set of parameterized transitions is denoted $T_p \subseteq T$. The label of a transition of T_p can either be observable or not, as there are cases in which the date of some observable transitions may not be modified and some unobservable transitions may be controllable before the start of the system. The proposed adjustments will be only restrictions of time intervals so that we ensure the causality of the system is preserved. In the following, such a solution will be called a diagnosabilisation by time restrictions (TR-diagnosabilisation for short).

Definition 7. A timed system is (Δ-)TR-diagnosabilisable for a timed fault Ω_f and a set of parameterized transitions T_p if there exists at least a valuation set $\mathcal{V}_d = \{I_s^d(t), t \in T_p, I_s^d(t) \subseteq I_s(t)\}$ such that, by replacing $I_s(t)$ with $I_s^d(t)$ for every transition $t \in T_p$ in Θ, Θ becomes (Δ-)Ω_f-diagnosable. Such a solution S_d is also called a (Δ-)TR-diagnosabilisation of Θ.

Through the rest of this paper, we now propose a method that provides a solution that ensures that Δ-Ω_f-diagnosability holds in the system Θ. For the sake of simplicity in the following, we will omit $\Delta-$ and only expressions like TR-diagnosabilisation, TR-diagnosabilisable...

3. Δ-DIAGNOSABILITY ANALYSIS: CRITICAL PAIR BASED ANALYSIS

Consider $\Delta \in \mathbb{R}_+$, the proposed method aims at guaranteeing that Δ-Ω_f-diagnosability holds in the system. This TR-diagnosabilization method is based on the enumeration of so-called *critical pairs* (Pecheur et al. (2002)). In the context of Δ-diagnosability, a critical pair is a couple of runs, one faulty and one safe, that shares the same observable trace.

Definition 8. A critical pair is a couple (r, r') of runs *such that*:

- let ρ (resp. ρ') be the timed sequence produced by r (resp. r'),
- $\rho = \rho_0\rho_1$, $\rho_0 \ni \Omega_f$, no prefix ρ_0' of ρ_0 matches Ω_f, $time(\rho_1) \geq \Delta$
- $\rho' \not\ni \Omega_f$
- $time(\rho) = time(\rho')$
- $\mathbf{P}_{\Sigma_\Theta \to \Sigma_\Theta^o}(\rho) = \mathbf{P}_{\Sigma_\Theta \to \Sigma_\Theta^o}(\rho')$

The existence of a critical pair characterizes an ambiguity in the system's behaviour so that it is sometimes impossible to decide whether the timed fault has occurred or not after Δ time units, based on the system's observations.

Proposition 1. A system Θ is Δ-Ω_f-diagnosable iff there is no critical pair in Θ.

The enumeration of critical pairs in the system relies on a finite abstraction of the system behaviours inspired by Coquand et al. (2022). This abstraction is a set of 3-uples (π, Π, Π^o) called *path* in the following, where π is a sequence of firable transitions from the system also called a *transition support*, Π is the set of constraints representing the earliest and latest dates of firing of the transitions of π and Π^o is the observable projection of Π, *i.e.* the set of time constraints representing the earliest and latest firing dates of every observable transition in π relatively to the firing date of the previous observable transition. Section 3.1 briefly presents and illustrates the computation of this abstraction.

The proposed TR-diagnosabilization method is a two step process. First, critical pairs involving one path only are identified (see Section 3.2) and a first TR-diagnosabilization is obtained as detailed in Section 4.1. Then critical pairs involving two paths are identified (see Section 3.3) and a final TR-diagnosabilization is obtained as detailed in Section 4.2.

3.1 Abstraction of system behaviours: paths

The objective of a path is to represent a possibly infinite set of runs that share the same transition support π and that might be involved in a critical pair. The computation of the constraint set representing the earliest and latest firing dates of each transition is detailed in Coquand et al. (2022). A support $\pi = t_0 \dots t_n \dots . t_{n+l} \dots . t_{n+k}$ of a given path has the following characteristics:

(1) t_n, t_{n+l}, t_{n+k} are transitions labeled with observable events ($0 \leq l \leq k$), no transition $t_{n+m}, (l < m < k)$ is observable.
(2) the earliest firing date of t_n is greater than b_{Ω_f}, so that after t_n there cannot be any transition in the path involved in the matching of the timed fault;
(3) if $k \neq 0$, the earliest firing date of t_{n+l} is lower than $\Delta + b_{\Omega_f}$;
(4) the earliest firing date of t_{n+k} is greater or equal to $\Delta + b_{\Omega_f}$.

The support π can be seen as a path in the State Class Graph of the system (SCG) defined in Berthomieu and Menasche (1983). There are a finite number of supports to extract as a consequence of Assumptions **A0-A1-A2** and $b_{\Omega_f} < +\infty$ and any run r of the system such that $time(r) \leq \Delta + b_{\Omega_f}$ is represented by exactly one of these supports, especially every run r with $time(r) \leq \Delta + b_{\Omega_f}$ such that $r \Subset \Omega_f$. Finally, as Ω_f is a timed fault whose occurrence date is bounded by b_{Ω_f}, any faulty run with $time(r) > \Delta + b_{\Omega_f}$ has, as a prefix, a run r' such that $time(r') \leq \Delta + b_{\Omega_f}$.

Example 3. Considering the system of Figure 1 and $\Delta = 1$, $\pi_0 = t_0.t_1.t_2.t_3.t_4.t_6.t_0$ is a possible support. The firing dates of the second occurrence of t_0 relatively to the start of the system are necessarily greater than 15 (the execution of a loop in the system to fire t_0 is greater than the sum of the lower bounds of the static time intervals of the transitions) so it is greater than b_{Ω_f}, moreover t_0 is observable. Note also here that as $\Delta = 1$, the earliest date of t_0 is also greater than $\Delta + b_{\Omega_f} = 14$ (special case where $k = 0$ so this occurrence of t_0 corresponds to transition $t_n = t_{n+l} = t_{n+k}$ as defined in the characteristics of the support hereabove). The constraints representing the earliest and latest firing dates of π_0 are in the set Π_0 $=\{2 \leq \theta_0 \leq 6,\ 3 \leq \theta_1 \leq 7,\ 1 \leq \theta_2 \leq 2,\ 2 \leq \theta_3 \leq 4,\ 1 \leq \theta_4 \leq 2,\ 4 < \theta_5 \leq 5,\ 2 \leq \theta_6 \leq 6\}$. (Here θ_i (resp. θ_i^o) corresponds to the date of firing of the *i-th* transition of the sequence (resp. of the observable sequence)). The observable constraint set associated with Π_0 finally is Π_0^o $= \{2 \leq \theta_0^o \leq 6,\ 4 \leq \theta_1^o \leq 9,\ 3 \leq \theta_2^o \leq 6,\ 4 < \theta_3^o \leq 5,\ 2 \leq \theta_4^o \leq 6\}$. [1]

[1] As the running example in this paper is simple for the sake of readability, there is actually no parallelism so the computation of the constraints is straightforward. However, in the general case, Π_0 and Π_0^o also contain parallelism constraints that require more complex computations, see Coquand et al. (2021) for details.

3.2 Critical pair extracted from one path only

Let us consider a path (π, Π, Π^o) such that there is a run of this path that matches Ω_f. This means that there is at least one transition label by the faulty event and whose firing fits the constraint of Ω_f. Such a transition occurrence is called a *faulty transition candidate* in the following of this paper. For example, let us consider again the support π_0, occurrences of t_1 and t_3 in π_0 are faulty transition candidates. Given the same path π it is also possible that it abstracts another run such that it does not match Ω_f, which means that the path itself characterizes some critical pairs.

Example 4. Back to Figure 2, run $r_1 = 5t_0.4t_1.1t_2$ matches the fault Ω_f while run $r_1' = 5t_0.3t_1.2t_2$ does not match Ω_f. Both runs share their observable projection $\mathbf{P}_{\Sigma_\Theta \to \Sigma_\Theta^o}(r_1)$ $= \mathbf{P}_{\Sigma_\Theta \to \Sigma_\Theta^o}(r_1') = 5o_1.5o_2$. Now, it can be noticed that there exists a common continuation r_2 of r_1 and r_1', that is $r_2 = 2t_3.1t_4.5t_6.2t_0$ so that run r $(= r_1 r_2)$ and run r' $(= r_1' r_2)$ are both represented in the path (π_0, Π_0, Π_0^o) and (r, r') is a critical pair.

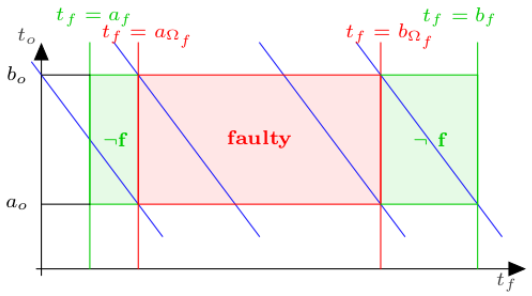

Fig. 3. Faulty and safe areas for a timed fault occurring between a_{Ω_f} and b_{Ω_f} t.u.

A faulty transition candidate can be source of an ambiguity, *i.e.* it is involved in critical pairs that can be extracted from the same support π. Figure 3 represents this situation. Transition t_f is a faulty transition candidate and t_o is the first observable transition that is fired after t_f in the considered path. a_f (resp. b_f) is the earliest (resp. latest) firing date of t_f relatively to the start of the system, and a_o (resp. b_o) is the earliest (resp. latest) firing date of t_o relatively to the firing of t_f. The red and green polyhedra correspond to admissible firing of t_f and t_o in the considered path. The red area corresponds to faulty firing of t_f and the green ones correspond to faultless firings. The observation of the system that captures the firing of t_f corresponds to the firing of t_o, i.e. the first observed date t_o^{obs} after the fire of t_f relatively to the start of the system is such that $t_o^{obs} = t_f + t_o$. A straight blue line in Figure 3 then represents the set of couples of firing dates (t_f, t_o) that are possible for a given value θ^{obs} of the observed date t_o^{obs}. Obviously, if θ^{obs} is associated with a blue line that only intersects the red (resp. green) area then the firing date of t_f ensures the run is faulty (resp. safe). On the contrary, if θ^{obs} is associated with a blue line that intersects both the red and the green areas, it means

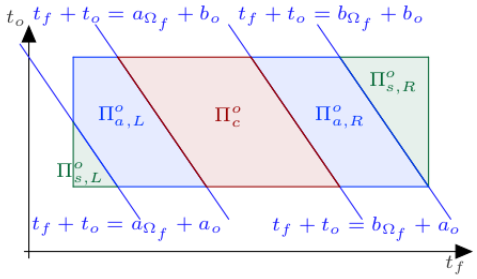

Fig. 4. Ambiguous and non-ambiguous polyhedra for the timed fault of Figure 3

that observing $t_o^{obs} = \theta^{obs}$ leads to an ambiguity as the firing date of t_f may be faulty or not. From this analysis, it follows that it is possible to characterize polyhedra by a set of constraints over $\theta^{obs}, a_{\Omega_f}, b_{\Omega_f}, a_o, b_o$ that assert for any couple (t_f, t_o) whether the situation is safe, certainly faulty or ambiguous (see Figure 4). The polyhedra are defined as follows:

safe ($\Pi_{s,L}$ and $\Pi_{s,R}$):

$$\theta_o^{obs} < a_{\Omega_f} + a_o \tag{1a}$$
$$b_{\Omega_f} + b_o < \theta_o^{obs} \tag{1b}$$

certain (Π_c):

$$a_{\Omega_f} + b_o \leq \theta_o^{obs} \leq b_{\Omega_f} + a_o \tag{1c}$$

ambiguous ($\Pi_{a,L}$ and $\Pi_{a,R}$):

$$a_o + a_{\Omega_f} \leq \theta_o^{obs} < a_{\Omega_f} + b_o \tag{1d}$$
$$a_o + b_{\Omega_f} < \theta_o^{obs} \leq b_{\Omega_f} + b_o \tag{1e}$$

Note that some areas may be empty regarding the different values of a_f b_f, a_{Ω_f}, b_{Ω_f}, a_o and b_o. Such constraints, if added to the observable projection Π^o of the path, lead to the definition of a set of polyhedra representing three different types of run in a path

- $\Pi^o_a = \Pi^o \cup$ ***ambiguous***: the set of observable timed sequences leading to an ambiguity
- $\Pi^o_s = \Pi^o \cup$ ***safe*** the set of observable timed sequences involving safe runs only
- $\Pi^o_c = \Pi^o \cup$ ***certain*** the set of observable timed sequences involving faulty runs only

The existence of critical pairs inherent to one path only is then determined by the following result.

Proposition 2. Considering a path (π, Π, Π^o), $\mathcal{S}(\Pi^o_a) = \emptyset$ iff

- there exists a faulty transition candidate t_f such that $a_{\Omega_f} \leq a_f \leq b_f \leq b_{\Omega_f}$, or
- for each faulty transition candidate t_f, the earliest and latest firing dates of t_o are equal ($a_o = b_o$).

Based on Proposition 2, we can apply the first step of the TR-diagnosabilization to ensure that $\mathcal{S}(\Pi^o_a) = \emptyset$, this step is described in Section 4.1.

3.3 Critical pair extracted from two different paths

We assume that a first TR-diagnosabilization has been successfully performed based on Section 4.1 to get a new set of parameters S_d such that $\mathcal{S}(\Pi^o_a) = \emptyset$. By using again Equations 1a,1b and 1c, it is then possible to update Π_c, Π^o_c and Π_s, Π^o_s for each path to take into account this set of parameters S_d. But, still, this first TR-diagnosabilization is not sufficient as there may be some critical pairs involving a faulty run from a path and a safe run from another path. The faulty (resp. faultless) polyhedron for a path (π, Π, Π^o) can be written as (π, Π^o_c) (resp. (π, Π^o_s)). Coquand et al. (2022) shows that the search for critical pairs between two different paths is then equivalent to checking for the intersection between their safe and certain polyhedron:

Proposition 3. If there exists $((\pi, \Pi, \Pi^o), (\pi', \Pi', \Pi^{o'}))$ *such that* $\exists (r, r') \in (\pi, \Pi^o_s) \times (\pi', \Pi^{o'}_c), \mathbf{P}_{\Sigma \to \Sigma_o}(r) = \mathbf{P}_{\Sigma \to \Sigma_o}(r')$, then the system is not Δ-Ω_f-diagnosable.

Example 5. The observable timed sequence $\rho_o = 2o_1.9o_2.3o_2.4, 5o_3.4o_1.5o_2.5o_2.4, 2o_3$ is produced by a faulty run of π_0 and by a non-faulty run of $\pi_1 = t_7.t_8.t_9.t_{10}.t_{11}.t_7$. It belongs to the intersection between the certain polyhedron of π_0 and the safe polyhedron of π_1.

Based on Proposition 3, the second and final step of the TR-diagnosabilization is finally described in Section 4.2.

4. DISAMBIGUATION OF CRITICAL PAIR

Section 3 shows that the sources of critical pair, *i.e.* the reasons of undiagnosability of a system can be divided in two classes: the first is the ambiguities inherent to a path ($\mathcal{S}(\Pi^o_a) \neq \emptyset$), the second is the ambiguities coming from the intersection of the safe and certain polyheron abstracting two different paths. As the number of paths and the number of faulty candidate transitions are bounded, there are a bounded number of sources of critical pairs. After enumerating the different sources of critical pairs using the results of the previous section, the proposed method synthetises new time constraints the system must satisfy to delete ambiguities, *i.e.* to make the system Δ-diagnosable.

4.1 Ambiguities inherent to a path ($\mathcal{S}(\Pi^o_a) \neq \emptyset$)

Considering a support $\pi = t_0 \dots t_{f_1} \dots t_{o_{f_1}} \dots t_{f_2} \dots t_{f_n} \dots t_m$, its faulty transition candidates $\{t_{f_1}, \dots, t_{f_n}\}$, and suppose that there is at least one faulty transition candidate that induces that $\mathcal{S}(\Pi^o_a) \neq \emptyset$. For each t_{f_i}, the first observable transition following it in π is denoted $t_{o_{f_i}}$.

Proposition 2 defines the conditions of emptiness of $\mathcal{S}(\Pi^o_a)$. As a consequence, the type of adjustments by time restriction to empty $\mathcal{S}(\Pi^o_a)$ are the following ones:

S0: trivialization of the firing date of $t_{o_{f_i}}$ relatively to the firing of t_{f_i} for each t_{f_i} that is source of an ambiguity
S1: enforcement of the ambiguous transition to become always faulty (by adjusting the set of its firing dates to always match the fault behaviour for this particuliar sequence)

Lemma 1. Let's consider t_{f_i}, $t_{o_{f_i}}$ and $t_{uo_0}, \dots t_{uo_k}$ the unobservable transitions between t_{f_i} and $t_{o_{f_i}}$ in π. **S0** can be formalized as the set of constraints:

$$\forall j \in [0, k], \downarrow (t_{uo_j}) = \uparrow (t_{uo_j}), \downarrow (t_{o_{f_i}}) = \uparrow (t_{o_{f_i}})$$

Solution **S0** can be seen as a determinisation of the time elapsed since the firing of the faulty transition candidate,

for each faulty transition candidate. This solution, however, cannot be applied as soon as one of the transitions $t_{o_{f_i}}, t_{uo_0}, \dots t_{uo_k}$ is not in T_p (a transition cannot be parameterized). Consider now solution **S1**.

Lemma 2. Let $F_\pi = \{t_{f_1}, \dots, t_{f_n}\}$ be the set of faulty transition candidates of π. **S1** can be formalized as:

$$\exists t_f \in F_\pi, a_{\Omega_f} \leq a_{t_f} \leq b_{t_f} \leq b_{\Omega_f}$$

where a_{t_f} and b_{t_f} are respectively the earliest and latest dates of firing of t_f relatively to the start of the system.

Solution **S1** can be seen as forcing the system to be faulty if its behaviour follows a particuliar support. As for **S0**, its implementation depends on the transitions involved in the time constraints and their belonging to T_p.

For a path (π, Π, Π^o), the constraint set that characterizes the disambiguation from Π_a can be written $\Pi \wedge (\mathbf{S0} \vee \mathbf{S1})$. Considering that a system contains k ambiguous path (k sequences of transition for which there are inherent ambiguities), the constraints set providing a non-ambiguous solution can be written $\bigwedge_{j=0.}^{k-1} (\Pi_j \wedge (\mathbf{S0}_j \vee \mathbf{S1}_j)$.

Example 6. Let's consider $T_p = \{t_0, t_4, t_5, t_7, t_{11}\}$. For the path π_0, solution **S1** can be implemented as the following constraints: $a_{\Omega_f} \leq a_{t_0} + a_{t_1}$ and $b_{t_0} + b_{t_1} \leq b_{\Omega_f}$. As t_0 is parameterized, there is a solution for these constraints: $a_{t_0} = b_{t_0} = 6$.

4.2 Ambiguities from two different paths

Considering two paths (π, Π, Π^o) and $(\pi', \Pi', \Pi^{o'})$ such that $(\pi, \Pi^o_s) \cap (\pi', \Pi^{o'}_c) \neq \emptyset$, Proposition 3 states that there is a source of critical pairs. This source of critical pairs comes from the fact that for each observable transition of the two paths there is at least one common observable date. In order to delete such an ambiguity, one need:

S2: modification for one of the observable transition of π of its earliest observable date *such that* it becomes greater than the latest observable date of the corresponding transition in π'.

Lemma 3. Considering two polyhedra (π, Π^o_c) and $(\pi', \Pi^{o'}_s)$ sharing their untimed observable trace, **S2** can be formalized as follows. Let $(t_{n+l}, t'_{n+l'})$ be a couple of observable transitions with same label and common firing dates from π and π':

$$\mathbf{S2} = \bigvee_{\text{set of } (t_{n+l}, t'_{n+l'})} (a_{t_{n+l}} > b_{t'_{n+l'}}) \vee (a_{t'_{n+l'}} > b_{t_{n+l}}).$$

To disambiguate pair (π, π') such that $(\pi, \Pi_s) \cap (\pi', \Pi'_c) \neq \emptyset$, constraint **S2** must be added to the previous constraints.

Example 7. Considering the paths π_0 and $\pi_1 = t_7.t_8.t_9.t_{10}.t_{11}.t_7$, there is an ambiguity as π_0 can be faulty and π_1 is safe. Solution **S2** can be implemented as $a_{t_7} > b_{t_0} \vee a_{t_0} > b_{t_7}$. Using $a_{t_0} = b_{t_0} = 6$ and $b_{t_7} = 4$, Θ is 1-diagnosable, so for the set of parametrised transitions T_p, the valuation set $\mathcal{V}_d = \{I_s(t_0) = [6,6], I_s(t_4) = [1,2], I_s(t_5) = [2,3], I_s(t_7) = [2,4], I_s(t_{11}) =]4,5[\}$ is a solution of TR-diagnosabilisation.

5. CONCLUSION

This paper introduces the problem of TR-diagnosabilisation of a system modeled as a safe LTPN and proposes a first resolution method that ensures Δ-diagnosability. Based on a Δ-diagnosability analysis that enumerates the sources of critical pairs, time constraints that delete critical pairs are synthetized. Those constraints are added to the time constraints relative to the structure of the system, and if there is a solution to this constraints set, this solution is a solution for TR-diagnosabilisation.

The proposed method relies on a sufficient condition for TR-diagnosabilisation. Future works will focus on the determination of a necessary and sufficient condition for TR-diagnosabilization and the extension to Ω_f-diagnosability.

REFERENCES

Basile, F., Cabasino, M.P., and Seatzu, C. (2016). Diagnosability analysis of labeled time Petri net systems. *IEEE Transactions on Automatic Control*, 62(3), 1384–1396.

Basile, F., Chiacchio, P., and Coppola, J. (2015). Model repair of time petri nets with temporal anomalies. volume 48, 85–90. doi:10.1016/j.ifacol.2015.06.477.

Berthomieu, B. and Menasche, M. (1983). An enumerative approach for analyzing time Petri nets. In *Proceedings IFIP*, 41–46. Elsevier Science Publishers.

Coquand, C., Subias, A., and Pencolé, Y. (2021). Signature of timed patterns in time Petri nets: a formal characterization. In *Modélisation des Systèmes Réactifs (MSR'21)*. Paris, France.

Coquand, C., Subias, A., Pencolé, Y., and Lubat, É. (2022). Critical pairs based diagnosability analysis of timed fault in time petri nets. In *16th IFAC Workshop on Discrete Event Systems*.

Fahland, D. and van der Aalst, W.M.P. (2012). Repairing process models to reflect reality. In A. Barros, A. Gal, and E. Kindler (eds.), *Business Process Management*, 229–245. Springer Berlin Heidelberg, Berlin, Heidelberg.

He, L., Dague, P., and Ye, L. (2022). Using Delay Blocks to Make Non-Diagnosable Discrete Event Systems Diagnosable. In *33rd International Workshop on Principle of Diagnosis – DX 2022*. LAAS-CNRS-ANITI, Toulouse, France. URL `https://hal.archives-ouvertes.fr/hal-03773712`.

Pecheur, C., Cimatti, A., and Cimatti, R. (2002). Formal verification of diagnosability via symbolic model checking. In *Workshop on Model Checking and Atificial intelligence (MoChArt-2002), Lyon, France.*

Pencolé, Y. and Subias, A. (2021). Diagnosability of event patterns in safe labeled time Petri nets: A model-checking approach. *IEEE Transactions on Automation Science and Engineering*, 1–12. doi: 10.1109/TASE.2020.3045565.

Tripakis, S. (2002). Fault diagnosis for timed automata. In W. Damm and E.R. Olderog (eds.), *Formal Techniques in Real-Time and Fault-Tolerant Systems*, 205–221. Springer Berlin Heidelberg, Berlin, Heidelberg.

Zaytoon, J. and Lafortune, S. (2013). Overview of fault diagnosis methods for discrete event systems. *Annual Reviews in Control*, 37(2), 308–320. doi: 10.1016/j.arcontrol.2013.09.009.