



HAL
open science

Failure Detection and Localization for Timed Event Graphs in $(\max, +)$ -Algebra

Euriell Le Corrond, Yannick Pencolé, Alexandre Sahuguède, Claire Paya

► **To cite this version:**

Euriell Le Corrond, Yannick Pencolé, Alexandre Sahuguède, Claire Paya. Failure Detection and Localization for Timed Event Graphs in $(\max, +)$ -Algebra. *Discrete Event Dynamic Systems*, 2021, 31 (4), pp.513-552. 10.1007/s10626-020-00329-7. hal-03122966

HAL Id: hal-03122966

<https://hal.science/hal-03122966>

Submitted on 27 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Failure Detection and Localization for Timed Event Graphs in $(max, +)$ -Algebra

Euriell Le Corronc · Yannick Pencolé ·
Alexandre Sahuguède · Claire Paya.

Received: date / Accepted: date

Abstract In this paper, we address the problem of failure detection and localization in a Timed Discrete Event System (TDES) such $(max, +)$ -linear system graphically modeled by a Timed Event Graph (TEG). The considered failures are changes on holding times or tokens of the TEG places that can provoke shifts between an observed outgoing timed flow and an expected outgoing timed flow (for a given incoming timed flow). Indicators are built to first detect such shifts relying on the $(max, +)$ algebraic framework and the residuation theory. An analysis of the indicators' values provides information about time or event failure that could have happen. Then, thanks to the knowledge of the behavior of the system through its corresponding TEG, sets of failures that could explain the detected shifts are obtained. It comes from matrices of signatures for each indicator built on each observable output of the system. An example of application is proposed to experiment exhaustively failures of type time and event on each place of the TEG.

Keywords Failure diagnosis · Detection and Localization · Timed Event Graphs · $(max, +)$ -Linear Systems · Dioid and Residuation Theory · Toolbox

1 Introduction

Smart manufacturing gathers a large category of manufacturing processes that intensively use computers to control, supervise and adapt the manufacturing processes by taking advantage of advanced information (from raw data sensors on supply chains and manufacturing processes to client demands and market

E. Le Corronc, Y. Pencolé, A. Sahuguède, C. Paya
LAAS-CNRS, Université de Toulouse
CNRS, UPS, Toulouse, France
Tel.: +33-5-61-33-69-53
Fax: +33-5-61-33-63-00
E-mail: {elecorro,ypencole,asahugue,cpaya}@laas.fr

trend analyses). Within this context, automated failure diagnosis of supply chains and manufacturing processes plays a crucial role as it is a way to improve the flexibility of the overall manufacturing process by quickly isolating the failure in the chain and then taking actions like reconfigurations or maintenance depending on the type of diagnosed failure. Based on new available sensing technologies (IoT), smart manufacturing aims at deploying more and more connected sensors on the processes so that supervision systems, and more specifically automated diagnosis systems, have more and more data to analyse and exploit. Among the possible dysfunctions, manifest breakdowns, that lead to complete stop of the manufacturing process, have to be distinguished from performance problems within the process and the resource flow. While our proposal can deal with failures of the former type, this paper focuses on failures of the latter type. A manufacturing process can firstly be affected by *timed shift failures*, such as unexpected transmission delays of a resource from a machine to another, late arrival of deliveries from the supply chain, decreasing of production over time. This type of failure is mostly due to degraded functioning of machines. Another specific type of failures, called *event shift failures*, has also to be considered: typical event shift failures are unexpected arrival/removal of resources in the process (for instance, a human operator taking a resource for quality testing, and bringing back later or not; a machine that rejects the input resource or a machine that always produces a potentially malformed output even if one of its resource input is missing...).

This paper presents an original framework for the automated detection and localisation of event/time shift failures within a manufacturing process. We propose in this framework to model such a process as a fault-free $(max, +)$ -linear system: the model only represents normal behaviours of the underlying system. This $(max, +)$ formalism is based on the idempotent semiring theory ([1, 7, 17]). Such models aim at representing Timed Discrete Event Systems (TDES) and particularly synchronization between equipments, process durations and transmission times which are the typical phenomena that are investigated in this paper. Another advantage of such a formalism is that this type of models can be implemented with help of efficient tools that take advantage of the underlying $(max, +)$ -linear algebra. More precisely, we propose to model a manufacturing process as a Timed Event Graph (TEG), a sub-class of time Petri nets, to automatically interpret it as an equation system in the idempotent semiring $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$.

The paper is organized as follows. Section 2 presents some related works. Section 3 describes the general problem of failure detection in timed discrete event systems and then introduces the subclass of TEGs that is investigated throughout this paper. The necessary mathematical background introducing the $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ dioid is then detailed in Section 4 as well as the way TEGs are represented within this $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ algebraic framework. Section 5 then focuses on the detection of event/time shifts in the observable signals of the supervised process. This detection system actually compares the real observed event flows with predicted event flows and mainly relies on the residuation theory in $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ introduced in Section 4.1. Section 6 proposes a charac-

terization of event/time shift failures within this framework and defines a set of indicators for their detection. Section 7 explains how can the indicators be used to perform failure localization in the manufacturing process. Finally, Section 8 presents an exhaustive set of simulations and experimental results on an example of manufacturing process that shows the detection/localization capabilities of our proposed indicators. Section 9 finally concludes this work.

2 Related work

As opposed to the classical fault diagnosis in untimed discrete event systems [29], there are definitely fewer works dealing with failure diagnosis in timed discrete event systems and they are more recent. The first attempts to deal with time in an event-based system is to model observable time patterns that are relevant to symptomatic abnormal behaviours without a complete model of the underlying system to supervise. Among these pattern-based techniques, there are temporal causal signatures [24], [20] that directly map faults to observable effects defined as timed event patterns. A closer formalism is also the one of chronicles [9] that provides an effective and automated way to recognize such patterns on-line from an observable flow of timed events.

Problems of failure diagnosis relying on a complete model of *autonomous* timed system (i.e. without inputs) have then been defined by the use of time automaton in [27, 28, 25, 5, 6]. These works extend the classical fault diagnosis problem of discrete event systems with time, a fault is usually modeled as the occurrence of a specific event. More recent contributions also use time Petri nets with specific algorithms [10, 12, 16, 26] to deal with the space and time complexity of the problem and [3] investigates the diagnosability of such systems. [18] also proposes a diagnosis algorithm on time workflows that handles more complex fault than single fault event. This algorithm uses a model-checking technique.

Our contribution is based on systems that depend on observable inputs (in this sense systems are not autonomous). It is typical of smart manufacturing processes where the resources to be processed are firstly sensed before moving to the first processing machines. We investigate a subclass of time Petri nets that are the Timed Event Graphs: thanks to this restriction, we can take advantage of the linearity of the models to get effective and efficient computations in an algebraic way by the use of the specific $(max, +)$ -algebra denoted $\mathcal{M}_{in}^{ax}[\gamma, \delta]$. Contrary to the set of works listed hereabove, we consider that the available model of the system is faultless (no behavioural model of faults is available). Our work is an extension of [21] and [13]. The previous work [21] is a seminal work that exploits $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ to only focus on time shift failure detection while [13] introduces the time shift failure localization. This paper fully benefits from $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ and extends both works to also deal with event shift failures and shows how effective is the method on smart manufacturing processes. The use of $(max, +)$ algebraic approaches for fault diagnosis is very new, and to the best of our knowledge, there are only three other seminal

works for fault diagnosis using a $(max, +)$ algebraic approach. In [23], they propose a two-layered diagnosis architecture where only the high layer relies on another type of $(max, +)$ algebra to perform fault detection in a set of components modeled as an autonomous Timed Event Graph. In [19], they exploit a similar $(max, +)$ algebra to define a framework for the fault diagnosis of autonomous discrete time hybrid systems. Finally, the work of [2] also exploits a $(max, +)$ method to predict trajectories in durational graphs that are a sub-class of timed automata.

3 Problem statement

3.1 Failure Detection in Timed Discrete Event System

This paper addresses the failure detection problem in a specific type of Timed Discrete Event Systems (TDES) that relies on a formal framework based on the $(max, +)$ algebra theory. Generally speaking, a TDES Π is a system that generates a *timed language* $\mathcal{L}(\Pi) \subseteq \mathcal{T}(\Sigma)$ that is a set of *timed sequences* over an alphabet of events Σ . In this paper, we assume that time is global and discrete (one global clock). A timed sequence of $\mathcal{T}(\Sigma)$ is then a sequence ρ mixing durations, events of Σ and time points:

$$\mathcal{T}(\Sigma) \subseteq (\Sigma \cup \{\lambda\} \times \mathbb{N})^+.$$

For example, if $\Sigma = \{a, b, c\}$ then $\rho = (a, 3).(c, 4).(\lambda, 2).(b, 1).(b, 2).(\lambda, 4)$ is such a timed sequence of $\mathcal{T}(\Sigma)$. Event a occurs at date 3 after the origin of time, c occurs 4 units of time after a (so c occurs at date 7), the time point representing two units of time after the occurrence of c is represented by $(\lambda, 2)$ (it is thus date 9), b occurs 1 unit of time later (so at date 10), and so on. To simplify the notations, ρ is simply written as $\rho = 3a4c2\lambda 1b2b4\lambda$. We denote $P_{\Sigma'}$ the *temporal projection* operator from $\mathcal{T}(\Sigma)$ to $\mathcal{T}(\Sigma')$ that keeps only the events from Σ' in a run of $\mathcal{T}(\Sigma)$: for instance, if $\Sigma' = \{a, b\}$ then $P_{\Sigma'}(\rho) = 3a7b2b4\lambda$ and if $\Sigma' = \{c\}$ then $P_{\Sigma'}(\rho) = 7c9\lambda$.

To setup the detection problem, Σ is partitioned into two subsets: the set of observable events (namely Σ_o) and the set of unobservable events (namely Σ_{uo}). In this paper, we assume that time is fully observable. It follows that any possible run ρ of the system can be associated with its observation σ by temporal projection. Back to the previous instance, if $\Sigma_o = \{a, b\}$ and $\Sigma_{uo} = \{c\}$ then the *observation* of $\rho = 3a4c2\lambda 1b2b4\lambda$ is $\sigma = P_{\Sigma_o}(\rho) = 3a7b2b4\lambda$.

All along this paper, we consider that the model of the system is *fault-free*: there is no information about how the system abnormally behaves in case of failures. The objectives are:

1. to detect the occurrence of a failure, i.e. to detect that the system has generated a run that is not modelled based on the observations;
2. to localise the part of the system and the possible class of failures that can explain the observations.

Definition 1 (Failure) A TDES Π is *failing* if it produces a *real* (but partially-observed) timed sequence ρ that is not part of $\mathcal{L}(\Pi)$.

A system is failing if it produces an *abnormal run* ρ ($\rho \notin \mathcal{L}(\Pi)$). As the system is partially observable, the system may generate an abnormal run ρ whose observation is the same as the one of a run ρ' from the normal behaviour of the system: $P_{\Sigma_o}(\rho) = P_{\Sigma_o}(\rho')$. In this case, the failure is not detectable by any kind of indicator.

Definition 2 (Detectable Failure) The failure of Π is *detectable* if it produces a *real* (but partially-observed) timed sequence ρ such that:

$$\forall \rho' \in \mathcal{L}(\Pi), P_{\Sigma_o}(\rho) \neq P_{\Sigma_o}(\rho').$$

Consider σ a timed sequence of $\mathcal{T}(\Sigma_o)$, the *failure detection problem* then consists in checking whether there is no run ρ from the normal behaviour that can produce the observation σ :

$$\text{Detection}(\Pi, \sigma) \equiv (\forall \rho \in \mathcal{L}(\Pi), P_{\Sigma_o}(\rho) \neq \sigma).$$

Suppose now that σ ensures that the system is failing, so it means that the underlying run ρ of the system is not in $\mathcal{L}(\Pi)$. Two classes of failures can then be distinguished depending on the sequence ρ based on the following remark. It is always possible to characterize ρ as a concatenation of two timed sequences $\rho_1.\delta e.\rho_2$ ($e \in \Sigma, \delta \in \mathbb{R}^+$) such that ρ_1 is in $\mathcal{L}(\Pi)$ but $\rho_1.\delta e$ is not. If $\rho_1.\delta e$ is not in $\mathcal{L}(\Pi)$ two cases hold:

1. in a run $\rho \in \mathcal{L}(\Pi)$ starting with ρ_1 , event e should occur after ρ_1 but not after a duration δ : it is a *time-shift failure* that is located at the occurrence of e (e might normally occur before δ in the run so in this case the failure is a delay, or e might normally occur after δ in the run, in this case, the occurrence of e is too early);
2. there is no run in $\mathcal{L}(\Pi)$ starting with ρ_1 followed by an occurrence of event e : it is an *event-shift failure*. Either the occurrence of e has been added or the occurrence of a different e' is missing.

3.2 Timed Event Graph

The aim of the paper is to exploit the $(max, +)$ theory to perform failure detection and localisation on systems that can be represented in this theory, that is *Timed Event Graphs* (TEG). A TEG is a subclass of Petri nets where each place has only one upstream transition and one downstream transition, that represents a subclass of timed discrete event systems (see Figure 1).

Definition 3 (Timed Event Graph) A Timed Event Graph is a tuple $\langle P, T, A, M_0, H \rangle$ such that

- P is a finite set of places;

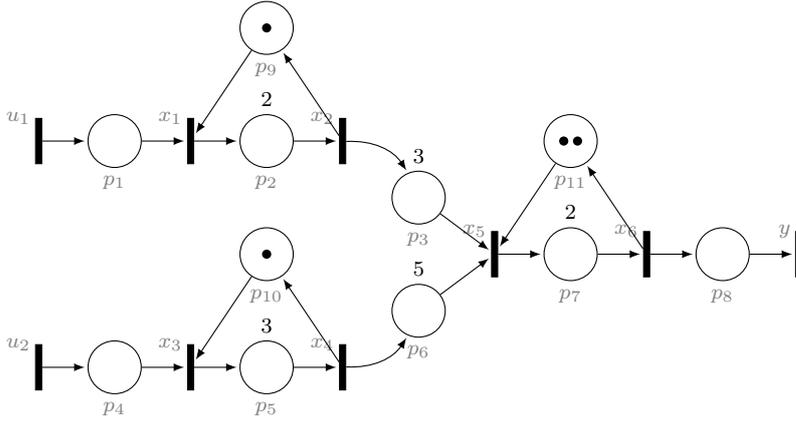


Fig. 1: A MISO TEG of an automated assembly line.

- T is a finite set of transitions;
- $A \subseteq (P \times T) \cup (T \times P)$ is a finite set of arcs;
- $M_0 : P \rightarrow \mathbb{N}$ is the initial marking;
- $H : P \rightarrow \mathbb{N}$ is the holding time function;
- $\forall p \in P, |\{(p, t) \in A\}| \leq 1$ and $|\{(t, p) \in A\}| \leq 1$.

The alphabet of the system associated with a TEG is $\Sigma = T$. A marking M of a TEG is a function $P \rightarrow \mathbb{N}$ that maps a number of tokens to a place p . The firing rule of a TEG is defined as follows. A transition t is enabled by the marking M if $\forall p, (p, t) \in A, M(p) \geq 1$. A transition t is *firable* at date $\delta \in \mathbb{N}$ if:

1. t is enabled in the marking of the TEG at date δ ;
2. $\forall p, (p, t) \in A$, if $H_p(\delta)$ denotes the amount of time before δ when p holds at least a token, then $H_p(\delta) \geq H(p)$;
3. δ is minimal (i.e. conditions 1 and 2 do not hold for $\delta - 1$), this is the *earliest time firing condition*.

Finally, TEG semantics ensures that a transition is fired as soon as it is firable.

Throughout this paper, we consider the subclass of TEGs that represent non-autonomous systems, which means that they are characterized by a set of *input transitions* u_1, \dots, u_n with $\{(p, u) \in A\} = \emptyset$ and a set of *output transitions* y_1, \dots, y_m with $\{(u, p) \in A\} = \emptyset$. Input transitions typically represent the sensed flows of resources to be processed by a manufacturing system while the output transitions represent the sensed flows of products after processing. The system starts operating as soon as input resources are available (trigger of an input transition).

All along the paper, we will consider the following assumptions meaning that there is always at least one firing of any input transition and that each input flow is a finite sequence.

Assumption 1 (Non-empty input sequences) *The sequence of events that triggers any input transition u_i is not empty. Formally, given the temporal projection $P_{\{u_i\}}$:*

$$\forall \rho \in \mathcal{L}(\Pi), \forall k \in \mathbb{N}, P_{\{u_i\}}(\rho) \neq k\lambda. \quad (1)$$

Assumption 2 (Finite input resources) *The sequence of events that triggers any input transition u_i is finite, formally:*

$$\forall \rho \in \mathcal{L}(\Pi), \exists n \in \mathbb{N}, P_{\{u_i\}}(\rho) = k_1 u_i k_2 u_i \dots k_n u_i k_{n+1} \lambda. \quad (2)$$

Any transition that is neither an input transition nor an output transition is called an *internal* transition. An *observable transition* is a transition that is associated with a sensor on the real system that is able to record any event of the system represented by the fire of the transition t and the date of the event. Throughout this paper, we consider that both input and output transitions t are observable ($t \in \Sigma_o$). We may also consider that some sensors are also available to record events and dates related to internal transitions. Such a sensor for an internal transition t is modeled in the TEG as a *measurable output transition* $mo(t)$.

Definition 4 (Measurable output transition) *A measurable output transition $mo(t)$ associated with an internal transition t is a transition such that:*

- $\exists p \in P, (t, p) \in A, (p, mo(t)) \in A$;
- $H(p) = 0$;
- $mo(t)$ is an output transition ($\{(mo(t), p') \in A\} = \emptyset$).

By definition, t fires at date δ if and only if $mo(t)$ fires at date δ so $mo(t)$ represents the sensor associated with t as an output observable transition.

Example 1 Figure 1 presents a MISO (Multiple Inputs - Single Output) Timed Event Graph that represents an automated assembly line. It is composed of 3 machines M_1 , M_2 and M_3 . Machines M_1 and M_2 independently process respectively incoming parts u_1 and u_2 . For each incoming part, the duration of the process is 2 for machine M_1 ($H(p_2) = 2$) and 3 for machine M_2 ($H(p_5) = 3$). The end of the processing of machines M_1 and M_2 is respectively represented by the firing of transition x_2 and x_4 . Machine M_3 then assembles one part coming from machine M_1 with one part coming from machine M_2 , taking respectively 3 and 5 time units to arrive there ($H(p_3) = 3, H(p_6) = 5$). The firing of transition x_6 represents the end of the assembling process. Two assembling processes can be done at the same time and the duration of this process is 2 time units. The complete product is then delivered as the output y of the line. In this example, only u_1, u_2 and y are observable. Suppose now that another sensor is available and records any end of the process of machine M_2 (transition x_2), then we can add a measurable output transition $y_2 = mo(x_2)$.

The way a TEG can be defined as a $(max, +)$ -linear system is detailed in the second subsection of next section.

4 Models of $(max, +)$ -linear systems

This section recalls the mathematical background that will be used throughout the paper. For this section, interested reader is invited to peruse [1, 7, 17].

4.1 Dioid and residuation theory

Definition 5 (Idempotent semiring) An idempotent semiring, also called dioid, \mathcal{D} is a set endowed with two inner operations denoted \oplus and \otimes . The sum \oplus is associative, commutative, idempotent (*i.e.* $\forall a \in \mathcal{D}, a \oplus a = a$) and admits a neutral element denoted ε . The product¹ \otimes is associative, distributes over the sum and admits a neutral element denoted e .

An idempotent semiring is said to be complete if it is closed for infinite sums and if the product distributes over infinite sums too. Due to the sum idempotency, an order relation can be associated with \mathcal{D} by the following equivalences: $\forall a, b \in \mathcal{D}, a \succcurlyeq b \iff (a = a \oplus b \text{ and } b = a \wedge b)$. Because of the lattice properties of a complete idempotent semiring (see [4]), $a \oplus b$ is the least upper bound of \mathcal{D} (also denoted \top) whereas $a \wedge b$ is its greatest lower bound. Finally, the *Kleene star* operator is defined as follows: $a^* = \bigoplus_{i \geq 0} a^i$ with $a^{i+1} = a \otimes a^i$ and $a^0 = e$.

Theorem 1 ([17]) *Implicit equation $x = ax \oplus b$, defined over a complete dioid \mathcal{D} , admits $x = a^*b$ as least solution.*

Example 2 The set $\overline{\mathbb{Z}}_{max} = \mathbb{Z} \cup \{-\infty, +\infty\}$, endowed with the max operator as sum \oplus and the classical sum as product \otimes , is a complete idempotent semiring where $\varepsilon = -\infty$, $e = 0$ and $\top = +\infty$. On $\overline{\mathbb{Z}}_{max}$, the greatest lower bound \wedge takes the sense of the min operator.

Example 3 The set of formal series with two commutative variables γ and δ , Boolean coefficients in $\{\varepsilon, e\}$ and exponents in \mathbb{Z} , is a complete idempotent semiring denoted $\mathbb{B}[[\gamma, \delta]]$ where $\varepsilon = \bigoplus_{n, t \in \mathbb{Z}} \varepsilon \gamma^n \delta^t$ (null series) and $e = \gamma^0 \delta^0$. A series $s \in \mathbb{B}[[\gamma, \delta]]$ is written in a single way by $s = \bigoplus_{n, t \in \mathbb{Z}} s(n, t) \gamma^n \delta^t$ where $s(n, t) = e$ or ε .

Graphically, a series of $\mathbb{B}[[\gamma, \delta]]$ is described by a collection of points of coordinates (n, t) in \mathbb{Z}^2 with γ as horizontal axis and δ as vertical axis. Boolean coefficients indicate the presence of a point $s(n, t) = e$ or its absence $s(n, t) = \varepsilon$.

Example 4 The quotient set of $\mathbb{B}[[\gamma, \delta]]$ by the modulo $\gamma^*(\delta^{-1})^*$ equivalence relation provides the complete idempotent semiring $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$. This means that an element of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ is an equivalence class denoted² $[a]_{\gamma^*(\delta^{-1})^*}$ gathering all the elements of $\mathbb{B}[[\gamma, \delta]]$ equivalent modulo $\gamma^*(\delta^{-1})^*$. Neutral elements ε

¹ As in usual algebra, \otimes will be omitted when no confusion is possible.

² Notation a without the bracket will be adopted in the sequel.

and e are identical to those of $\mathbb{B}[[\gamma, \delta]]$. Element³ \top is equal to $(\gamma^{-1}\delta)^*$. The equivalence relation induces these computation rules:

$$\begin{aligned}\gamma^n \delta^t \oplus \gamma^{n'} \delta^t &= \gamma^{\min(n, n')} \delta^t, \\ \gamma^n \delta^t \oplus \gamma^n \delta^{t'} &= \gamma^n \delta^{\max(t, t')}, \\ \gamma^n \delta^t \otimes \gamma^{n'} \delta^{t'} &= \gamma^{n+n'} \delta^{t+t'}.\end{aligned}$$

Remark 1 Equivalence $\gamma^*(\delta^{-1})^*$ naturally induces a monotony feature of series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$. Indeed, let $s = \bigoplus_{i \geq 0} \gamma^{n_i} \delta^{t_i} \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, then $n_i \leq n_{i+1}$ and $t_i \leq t_{i+1}$.

Remark 2 A C++ library called `minmaxgd` enables series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ to be handled (see [8]).

Graphically, the product of a monomial $\gamma^n \delta^t \in \mathbb{B}[[\gamma, \delta]]$ by $\gamma^*(\delta^{-1})^*$ goes to consider this element in $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ as a southeast cone with coordinates (n, t) containing all the monomials equivalent to $\gamma^n \delta^t$. The description of all the points of all the cones induced by the monomials of a series $s \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ corresponds to the *maximal representation* of s . Its *minimal representation* is obtained by writing only the monomials representing the cones' vertices.

Example 5 Let $s = \gamma^1 \delta^3 \oplus \gamma^2 \delta^4 \oplus \gamma^5 \delta^7$ be a series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ illustrated on Figure 2.

- *Maximal representation of s* : all the points contained in the cones induced by the monomials of s (among them, points of coordinates: $(1, 3), (2, 2), (3, 4), (6, 7), \dots$).
- *Minimal representation of s* : only the points of the cones' vertices (only the points of coordinates: $(1, 3), (2, 4), (5, 7)$).

Two other interesting representations of a series $s \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ are its *dater function* and its *counter function*.

Definition 6 (Dater function) The dater function of a series $s \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ is the non-decreasing function denoted $\mathcal{D}_s(n) : \mathbb{Z} \mapsto \overline{\mathbb{Z}}$ and obtained from the maximal representation of s by conserving for each γ^n of the series, the **largest** corresponding δ^t . Thus, $s = \bigoplus_{n \in \mathbb{Z}} \gamma^n \delta^{\mathcal{D}_s(n)}$.

Definition 7 (Counter function) The counter function of a series $s \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ is the non-decreasing function denoted $\mathcal{C}_s(t) : \mathbb{Z} \mapsto \overline{\mathbb{Z}}$ and obtained from the maximal representation of s by conserving for each δ^t of the series, the **smallest** corresponding γ^n . Thus, $s = \bigoplus_{t \in \mathbb{Z}} \gamma^{\mathcal{C}_s(t)} \delta^t$.

³ A simplification of writing gives $\varepsilon = \gamma^{+\infty} \delta^{-\infty}$ and $\top = \gamma^{-\infty} \delta^{+\infty}$.

Example 6 Let us take back series $s = \gamma^1\delta^3 \oplus \gamma^2\delta^4 \oplus \gamma^5\delta^7 \in \mathcal{M}_{in}^{ax}[\gamma, \delta]$ of Figure 2. The dater function of s is

$$\begin{aligned} \forall n \in \mathbb{Z}, \mathcal{D}_s(n) &= \{\dots, -\infty, 3, 4, 4, 4, 7, 7, \dots\} \\ \text{with } \mathcal{D}_s(n) &= -\infty \text{ for } n \leq 0, \\ \mathcal{D}_s(1) &= 3, \mathcal{D}_s(2) = 4, \mathcal{D}_s(3) = 4, \mathcal{D}_s(4) = 4, \\ \mathcal{D}_s(n) &= 7 \text{ for } n \geq 5. \end{aligned}$$

Its counter function is

$$\begin{aligned} \forall t \in \mathbb{Z}, \mathcal{C}_s(t) &= \{\dots, 1, 1, 2, 5, 5, 5, +\infty, \dots\} \\ \text{with } \mathcal{C}_s(t) &= 1 \text{ for } t \leq 3, \\ \mathcal{C}_s(4) &= 2, \mathcal{C}_s(5) = 5, \mathcal{C}_s(6) = 5, \mathcal{C}_s(7) = 5, \\ \mathcal{C}_s(t) &= +\infty \text{ for } t \geq 8. \end{aligned}$$

Remark 3 Because of equivalence $\gamma^*(\delta^{-1})^*$, this equality holds from the minimal representation:

$$\begin{aligned} s &= \gamma^1\delta^3 \oplus \gamma^2\delta^4 \oplus \gamma^5\delta^7 \\ &= \dots \oplus \gamma^1\delta^{-1} \oplus \gamma^1\delta^0 \oplus \gamma^1\delta^1 \oplus \gamma^1\delta^2 \\ &\quad \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^4 \oplus \gamma^3\delta^4 \oplus \gamma^4\delta^4 \oplus \gamma^5\delta^5 \oplus \gamma^5\delta^6 \\ &\quad \oplus \gamma^5\delta^7 \oplus \gamma^6\delta^7 \oplus \gamma^7\delta^7 \oplus \gamma^8\delta^7 \oplus \gamma^9\delta^7 \oplus \dots \end{aligned}$$

Indeed, the minimal representation does not need to detail all the monomials: some of them are covered by the equivalence class of others as $\gamma^1\delta^2$ by $[\gamma^1\delta^3]_{\gamma^*(\delta^{-1})^*}$ or $\gamma^3\delta^4$ by $[\gamma^2\delta^4]_{\gamma^*(\delta^{-1})^*}$. This is the representation mostly chosen.

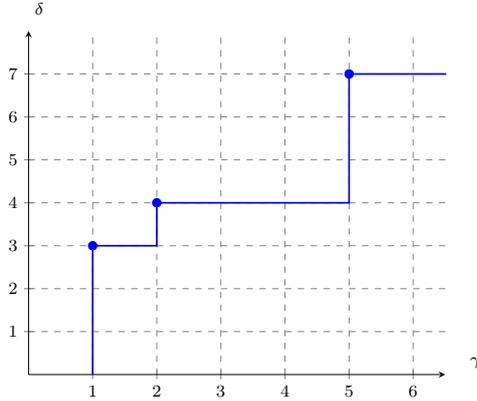


Fig. 2: Graphical representation of the series $s \in \mathcal{M}_{in}^{ax}[\gamma, \delta]$ from Example 5.

In a complete dioid, the product operation is not reversible. However, the residuation theory, coming from the lattice theory [4], allows the definition of “pseudo-inverse” of some isotone maps, such as this product, to be done. In particular, the residuation theory provides optimal solutions to inequalities such as $f(x) \preceq b$, where f is an order-preserving mapping defined over ordered sets. This theory can then be applied over complete idempotent semiring as follows.

Definition 8 (Residuated and residual mappings) Let $f : \mathcal{D} \rightarrow \mathcal{C}$ be an isotone mapping, where \mathcal{D} and \mathcal{C} are complete idempotent semirings. Mapping f is said to be *residuated* if $\forall b \in \mathcal{C}$, the greatest element of subset $\{x \in \mathcal{D} \mid f(x) \preceq b\}$, denoted $f^\sharp(b)$, exists and belongs to this subset. Mapping f^\sharp is called the *residual* of f .

When f is residuated, f^\sharp is the unique isotone mapping such that $f \circ f^\sharp \preceq \text{Id}_{\mathcal{C}}$ and $f^\sharp \circ f \succeq \text{Id}_{\mathcal{D}}$, where $\text{Id}_{\mathcal{C}}$ and $\text{Id}_{\mathcal{D}}$ are respectively the identity mappings on \mathcal{C} and \mathcal{D} .

Theorem 2 ([17]) Let \mathcal{D} be a complete dioid and $A \in \mathcal{D}^{n \times n}$ be a square matrix. Then, $A \phi A \in \mathcal{D}^{n \times n}$ is a matrix which verifies

$$A \phi A = (A \phi A)^*. \quad (3)$$

Example 7 Mapping $R_a : x \mapsto x \otimes a$ defined over a complete idempotent semiring \mathcal{D} is residuated. Its residual is usually denoted $R_a^\sharp : x \mapsto x \phi a$ and called *right quotient*. Therefore, $b \phi a$ is the greatest solution to inequality $x \otimes a \preceq b$, i.e. $b \phi a = \hat{x} = \bigoplus \{x \mid x \otimes a \preceq b\}$. This example can be applied for the product of matrices such as $X \mapsto X \otimes A \in \mathcal{D}^{p \times m}$ with $A \in \mathcal{D}^{n \times m}$ and $X \in \mathcal{D}^{p \times n}$, that is:

$$R_A = X \otimes A : (X \otimes A)_{ij} = \bigoplus_{k=1}^n X_{ik} \otimes A_{kj},$$

and the computation of $B \phi A \in \mathcal{D}^{p \times n}$ with $B \in \mathcal{D}^{p \times m}$ is given by:

$$R_A^\sharp(B) = B \phi A : (B \phi A)_{ij} = \bigwedge_{k=1}^m B_{ik} \phi A_{jk}. \quad (4)$$

Example 8 By introducing the idempotent semiring of dater functions (see [1, Section 5.3]) endowed with the pointwise maximum as sum and the following convolution as product:

$$\forall m, (f \otimes g)(m) = \bigoplus_{\nu \in \mathbb{Z}} \{f(\nu) \otimes g(m - \nu)\} = \sup_{\nu \in \mathbb{Z}} \{f(\nu) + g(m - \nu)\},$$

the residual of this convolution is

$$\forall m, (f \phi g)(m) = \bigwedge_{\nu \in \mathbb{Z}} \{f(\nu) \phi g(\nu - m)\} = \inf_{\nu \in \mathbb{Z}} \{f(\nu) - g(\nu - m)\}. \quad (5)$$

In this dioid, $e = 0$, $\varepsilon = -\infty$ and $\top = +\infty$. Now with the idempotent semiring of counter functions endowed with the pointwise minimum as sum and the following convolution as product:

$$\forall s, (f \otimes g)(s) = \bigoplus_{\tau \in \mathbb{Z}} \{f(\tau) \otimes g(s - \tau)\} = \inf_{\tau \in \mathbb{Z}} \{f(\tau) + g(s - \tau)\},$$

the residual of this convolution is

$$\forall s, (f \not\otimes g)(s) = \bigwedge_{\tau \in \mathbb{Z}} \{f(\tau) \not\otimes g(\tau - s)\} = \sup_{\tau \in \mathbb{Z}} \{f(\tau) - g(\tau - s)\}. \quad (6)$$

In this dioid, $e = 0$, $\varepsilon = +\infty$ and $\top = -\infty$.

4.2 ($\max, +$)-linear systems and Timed Event Graph modeling

The complete dioid $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ aims at modeling timed discrete event systems as timed sequences of events. Indeed, thanks to equivalence $\gamma^*(\delta^{-1})^*$, series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ are non-decreasing and can be used to represent the accumulation of event occurrences over time. For instance, suppose that a system only generates one type of event, let us say, a , then the series $s = \gamma^0 \delta^2 \oplus \gamma^1 \delta^3 \oplus \gamma^2 \delta^6 \oplus \gamma^3 \delta^{+\infty}$ fully represents the run $r_s = (a, 2).(a, 1).(a, 3) \equiv 2a1a3a$ where $\gamma^3 \delta^{+\infty}$ is a monomial that means that the 4th occurrence of a never happens. Suppose now that a is actually a transition of a TEG (see Definition 3) then s represents a firing sequence of this transition. Generally speaking, for each transition of the TEG, we can write a firing sequence of this transition, also called an event flow or a trajectory, as a series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$. A monomial $\gamma^n \delta^t$ of any of these series is interpreted as follows: *its $(n + 1)$ event occurrence happens at earliest at time t* . When trajectories describe a finite number n of events, they contain $n + 1$ monomials in which the last monomial is written $\gamma^n \delta^{+\infty}$ meaning that the $(n + 1)$ th event occurrence never happens.

Given a flow $u \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^p$ as an input vector of size p , we can model the system's response as a flow $y \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^q$ represented as an output vector of size q that is ruled by the following equation:

$$y = h \otimes u \quad (7)$$

where h is the *transfer function* image. This input/output relation can be obtained from the following state representation:

$$\begin{cases} x = Ax \oplus Bu \\ y = Cx \end{cases} \quad (8)$$

where $x \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^n$ is the state vector of the system, $A \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{n \times n}$, $B \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{n \times p}$ and $C \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{q \times n}$. Indeed, by applying Theorem 1 to state equation $x = Ax \oplus Bu$, the input/output relation is $y = CA^*Bu = hu$, and:

$$h = CA^*B. \quad (9)$$

Systems that are fully characterized by Equation 7 or Equations 8 are commonly called $(max, +)$ -linear systems and they also describe Timed Event Graphs and their semantics (see Section 3.2) as detailed below.

- Each series u_i (respectively y_i) of the input vector u (respectively output vector y) in a $(max, +)$ -linear system is associated with the timed sequence produced by the fire of the input transition u_i (respectively output transition y_i) in the corresponding TEG Π .⁴ Let $r \in \mathcal{L}(\Pi)$ be a run of the TEG Π then u_i (respectively y_i) represents $P_{\{u_i\}}(r) \in \mathcal{T}(\{u_i\})$ (respectively $P_{\{y_i\}}(r) \in \mathcal{T}(\{y_i\})$).
- Each series x_i of the state vector x in a $(max, +)$ -linear system is associated with the timed sequence produced by the fire of the respective internal transition x_i . Let $r \in \mathcal{L}(\Pi)$ be a run of the TEG Π then x_i represents $P_{\{x_i\}}(r) \in \mathcal{T}(\{x_i\})$.
- Equation $x = Ax \oplus Bu$ describes both the structure and the firing rules of the internal transitions of the TEG. From this equation, we have $x_i = \bigoplus_{j=1}^n A(i, j)x_j \oplus \bigoplus_{j=1}^p B(i, j)u_j$. If $A(i, j) \neq \varepsilon$ then $A(i, j)$ is a monomial $\gamma^m \delta^t$ where m is the backward event shift between the transitions x_j and x_i and t is their backward time shift which means that if x_j fires at time τ then x_i will fire at a time greater or equal to $\tau + t$ (time shift) and after m other fires of x_i (event shift, only the $(m + 1)^{th}$ next fire of x_i depends on this fire of x_j). It follows that $A(i, j)$ corresponds to a place p_k such that $(x_j, p_k) \in A$ and $(p_k, x_i) \in A$ with $H(p_k) = t$ and $M_0(p_k) = m$. Similarly, if $B(i, j) = \gamma^m \delta^t \neq \varepsilon$, then it corresponds to a place p_l such that $(u_j, p_l) \in A$ and $(p_l, x_i) \in A$ with $H(p_l) = t$ and $M_0(p_l) = m$. Finally, the \oplus operator between the monomials $A(i, j)$'s and $B(i, j)$'s ensures that the *earliest firing rule condition* of the transition x_i holds.
- Equation $y = Cx$ describes both the structure and the firing rules of the output transitions of the TEG in a similar way as for the internal transitions. For each measurable output transition, corresponding row of C contain only one element different from ε , this element is the monomial $\gamma^0 \delta^0$.

Looking back to the TEG defined by Figure 1, the state representation of the corresponding $(max, +)$ -system is given in $\mathcal{M}_{in}^{ax}[\gamma, \delta]$:

$$\begin{cases} x = \begin{pmatrix} \cdot & \gamma^1 \delta^0 & \cdot & \cdot & \cdot & \cdot \\ \gamma^0 \delta^2 & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \gamma^1 \delta^0 & \cdot & \cdot \\ \cdot & \cdot & \gamma^0 \delta^3 & \cdot & \cdot & \cdot \\ \cdot & \gamma^0 \delta^3 & \cdot & \gamma^0 \delta^5 & \cdot & \gamma^2 \delta^0 \\ \cdot & \cdot & \cdot & \cdot & \gamma^0 \delta^2 & \cdot \end{pmatrix} x \oplus \begin{pmatrix} \gamma^0 \delta^0 & \cdot \\ \cdot & \cdot \\ \cdot & \gamma^0 \delta^0 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{pmatrix} u \\ y = (\dots \gamma^0 \delta^0) x \end{cases} .$$

⁴ We use the same notation for a transition of the TEG and its corresponding series in the $(max, +)$ -linear system.

The entries of matrices $A \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{6 \times 6}$, $B \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{6 \times 2}$ and $C \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{1 \times 6}$ show the firing relations between transitions. $A(1, 2) = \gamma^1 \delta^0$ means that there is one token between x_2 and x_1 , so the $(n+1)^{th}$ firing of x_1 depends on the n^{th} firing of x_2 . Because of the synchronization on x_1 , the n^{th} firing of this transition also depends of the n^{th} firing of u_1 as indicated by the monomial $B(1, 1) = \gamma^0 \delta^0$. Monomial $A(2, 1) = \gamma^0 \delta^2$ indicates that the firing date of x_2 depends on the firing date of x_1 plus 2 time units. The absence of connection between transitions can be seen for instance between x_1 and y with $C(1, 1) = . = \varepsilon$.

The transfer function $h \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{1 \times 2}$ of the system is computed:

$$h = CA^*B = (\gamma^0 \delta^7 (\gamma^1 \delta^2)^* \quad \gamma^0 \delta^{10} (\gamma^1 \delta^3)^*).$$

In $h(1, 0)$, the monomial $(\gamma^1 \delta^2)^*$ shows the process of 1 piece by 2 time units on the machine M_1 whereas the monomial $(\gamma^1 \delta^3)^*$ of $h(1, 1)$ shows the process of 1 piece by 3 time units on M_2 . Now the system's behavior is characterized by h in $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, flows of events over time can be proceed. Thus, if the following finite input trajectory is given:

$$u = \begin{pmatrix} \gamma^0 \delta^2 \oplus \gamma^1 \delta^3 \oplus \gamma^2 \delta^5 \oplus \gamma^4 \delta^{+\infty} \\ \gamma^0 \delta^2 \oplus \gamma^1 \delta^3 \oplus \gamma^2 \delta^5 \oplus \gamma^4 \delta^{+\infty} \end{pmatrix}$$

in which $\gamma^0 \delta^2$ means that the 1^{st} event occurrence of u_1 and u_2 happens at time 2 but their 5^{th} event occurrence never happens ($\gamma^4 \delta^{+\infty}$). The corresponding output trajectory is:

$$y = hu = \gamma^0 \delta^{12} \oplus \gamma^1 \delta^{15} \oplus \gamma^2 \delta^{18} \oplus \gamma^3 \delta^{21} \oplus \gamma^4 \delta^{+\infty}$$

meaning that the first complete product is delivered at $t = 12$. This time corresponds to the arrival time on u_1 and u_2 (at $t = 2$) plus the duration of the longest path between the inputs and the output, that is 10 between u_2 and y .

Remark 4 It is important to notice that dater function of y is

$$\begin{aligned} \forall n \in \mathbb{Z}, \mathcal{D}_y(n) &= \{\dots, -\infty, 12, 15, 18, 21, +\infty, \dots\} \\ \text{with } \mathcal{D}_y(n) &= -\infty \text{ for } n \leq -1, \\ \mathcal{D}_y(0) &= 12, \mathcal{D}_y(1) = 15, \mathcal{D}_y(2) = 18, \mathcal{D}_y(3) = 21, \\ \mathcal{D}_y(n) &= +\infty \text{ for } n \geq 4. \end{aligned}$$

Its counter function is

$$\begin{aligned} \forall t \in \mathbb{Z}, \mathcal{C}_y(t) &= \{\dots, 0, 1, 1, 1, 2, 2, 2, 3, 3, 3, 4, \dots\} \\ \text{with } \mathcal{C}_y(t) &= 0 \text{ for } t \leq 12, \\ \mathcal{C}_y(t) &= 1 \text{ for } 13 \leq t \leq 15, \\ \mathcal{C}_y(t) &= 2 \text{ for } 16 \leq t \leq 18, \\ \mathcal{C}_y(t) &= 3 \text{ for } 19 \leq t \leq 21, \\ \mathcal{C}_y(t) &= 4 \text{ for } t \geq 22. \end{aligned}$$

Dater and counter functions of such series differ from those of Example 6 since some event occurrences never happened (those from numbering 4). In such case, the dater function always begins and ends by infinite value whereas the counter function only contains finite values.

Example 9 Let us now consider the TEG of a MIMO (Multiple Inputs - Multiple Outputs) represented by Figure 3. It is composed of two inputs u_1 and u_2 and two outputs y_1 and y_2 . We also suppose that internal transitions x_1 and x_3 can be observed so we add two measurable output transitions $y_3 = mo(x_1)$ and $y_4 = mo(x_3)$ (see Definition 4). The state representation of this system is:

$$\begin{cases} x = \begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \gamma^0\delta^2 & \cdot & \cdot & \gamma^0\delta^1 \\ \gamma^0\delta^2 & \cdot & \gamma^1\delta^1 & \cdot \\ \cdot & \cdot & \gamma^0\delta^1 & \cdot \end{pmatrix} x \oplus \begin{pmatrix} \gamma^0\delta^1 & \cdot \\ \cdot & \cdot \\ \cdot & \gamma^0\delta^3 \\ \cdot & \cdot \end{pmatrix} u \\ y = \begin{pmatrix} \cdot & \gamma^0\delta^3 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \gamma^0\delta^1 \\ \gamma^0\delta^0 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \gamma^0\delta^0 & \cdot \end{pmatrix} x \end{cases}$$

that gives the following transfer function $h \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{4 \times 2}$

$$h = CA^*B = \begin{pmatrix} \gamma^0\delta^8(\gamma^1\delta^1)^* & \gamma^0\delta^8(\gamma^1\delta^1)^* \\ \gamma^0\delta^5(\gamma^1\delta^1)^* & \gamma^0\delta^5(\gamma^1\delta^1)^* \\ \gamma^0\delta^1(\gamma^1\delta^1)^* & \cdot \\ \gamma^0\delta^3(\gamma^1\delta^1)^* & \gamma^0\delta^3(\gamma^1\delta^1)^* \end{pmatrix}.$$

By giving the following input:

$$u = \begin{pmatrix} \gamma^0\delta^2 \oplus \gamma^1\delta^4 \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^3 \oplus \gamma^1\delta^5 \oplus \gamma^3\delta^{+\infty} \end{pmatrix},$$

the output is:

$$y = \begin{pmatrix} \gamma^0\delta^{11} \oplus \gamma^1\delta^{13} \oplus \gamma^2\delta^{14} \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^8 \oplus \gamma^1\delta^{10} \oplus \gamma^2\delta^{11} \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^3 \oplus \gamma^1\delta^5 \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^6 \oplus \gamma^1\delta^8 \oplus \gamma^2\delta^9 \oplus \gamma^3\delta^{+\infty} \end{pmatrix}.$$

5 Time and event shift detection

As presented in Subsection 4.2, a $(max, +)$ -linear system is fully characterized by its transfer function h and its corresponding TEG. As described in Section 3.1, the model of the system that we consider is fault-free, i.e. the model represents the relationship between the inputs and the outputs only when the system normally behaves.

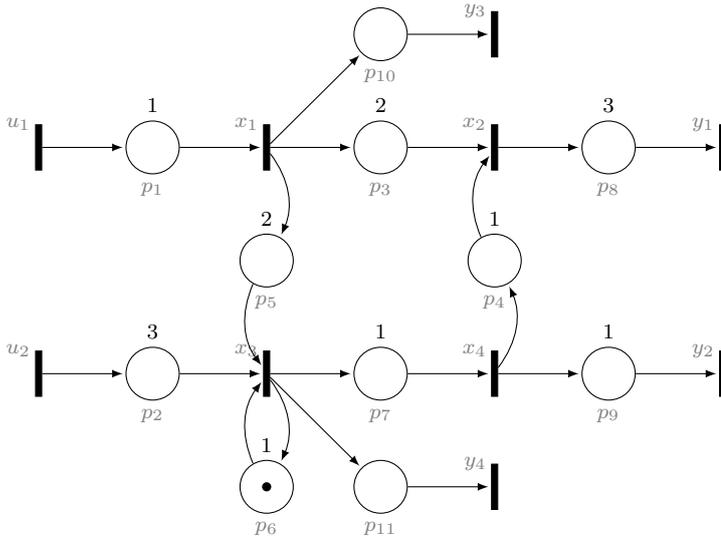


Fig. 3: A TEG of a MIMO $(max, +)$ -linear system.

From now, and through the rest of this paper, we use the following notations. The real input of the system is denoted u , the real output of the system is denoted y and we assume that inputs u and outputs y are fully observable. As inputs u are fully observable, we can compute the output flows $\tilde{y} = h \otimes u$ that should be normally produced by the system given the real inputs u . A failure is then detectable if and only if the real output y does not match the output \tilde{y} (\tilde{y} is part of the observable projection of a run from the normal behaviour of the system as explained in Subsection 3.1).

The challenge now is to find a way to actually compare y and \tilde{y} and determine the presence of time and event shifts between y and \tilde{y} as described in Subsection 3.1. Subsections 5.1 and 5.2 propose to perform these comparisons by using the residuation theory in particular the use of the right quotient ϕ between y and \tilde{y} . Information about shifts between date occurrences or number of events for these output transition firings are directly given by this computation. Such correlation between series of $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ previously comes from the second order theory of $(max, +)$ -linear systems introduced in [17]. The work of Subsection 5.1 was first proposed in [21]. Details of computation and analyses of the results are added here.

Moreover, since we deal with $(max, +)$ -linear systems, the impact on outputs y and \tilde{y} of Assumptions 1 and 2 made for this paper and given in the problem statement (Subsection 3.2), means that:

- I. Series y and \tilde{y} never equal to ε and contain at least one monomial each (see Assumption 1);
- II. Series y and \tilde{y} are finite flows of events, they always end with a monomial $\gamma^n \delta^{+\infty}$ (see Assumption 2).

5.1 Detection of time shifts

In a $(max, +)$ -linear system, when failures happened, this can provoke a real output trajectory y different than the expected output trajectory \tilde{y} . Then, some comparison needs to be done to detect shifts between series y and \tilde{y} . The right quotient is used and so are dater functions for the time shifts in this subsection, counter functions for the event shifts in the next subsection.

Definition 9 (Time shift function) Let a and b be series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, $\mathcal{D}_a(n)$ and $\mathcal{D}_b(n)$ their respective dater functions. The *time shift function* representing the time shifts between a and b for each $n \in \mathbb{Z}$ is defined by:

$$\mathcal{T}_{a,b}(n) = \mathcal{D}_b(n) - \mathcal{D}_a(n).$$

Intuitively speaking, the time shift function associates the time difference between the n^{th} event occurrence of series a and the n^{th} event occurrence of series b . Obviously, $\mathcal{T}_{a,a}(n) = 0, \forall n \in \mathbb{Z}$. If $\mathcal{T}_{a,b}(n)$ is equal to a positive, respectively negative, number then the n^{th} event occurrence of b will happen later, respectively earlier, than the n^{th} event occurrence of a . About the infinite values (see [17]), these equalities hold for all finite $\mathcal{D}(n) \in \overline{\mathbb{Z}}$:

$$(-\infty) - \mathcal{D}(n) = -\infty \quad (+\infty) - \mathcal{D}(n) = +\infty, \quad (10)$$

$$\mathcal{D}(n) - (-\infty) = +\infty \quad \mathcal{D}(n) - (+\infty) = -\infty, \quad (11)$$

$$(-\infty) - (-\infty) = +\infty \quad (+\infty) - (+\infty) = +\infty. \quad (12)$$

Example 10 Let a and b be series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, let us deal with situations we will have for the detection step.

- If $a = \gamma^0 \delta^2 \oplus \gamma^1 \delta^{+\infty}$ so $\mathcal{D}_a(n) = \{\dots, -\infty, 2, +\infty, \dots\}$, and $b = \gamma^0 \delta^1 \oplus \gamma^2 \delta^{+\infty}$ so $\mathcal{D}_b(n) = \{\dots, -\infty, 1, 1, +\infty, \dots\}$. For $n = 1$ one has

$$\mathcal{D}_b(1) - \mathcal{D}_a(1) = 1 - (+\infty) = -\infty$$

representing the fact that a stops before b (b has one more event occurrence regarding a).

- If $a = \gamma^0 \delta^4 \oplus \gamma^2 \delta^{+\infty}$ so $\mathcal{D}_a(n) = \{\dots, -\infty, 4, 4, +\infty, \dots\}$, and $b = \gamma^1 \delta^3 \oplus \gamma^2 \delta^{+\infty}$ so $\mathcal{D}_b(n) = \{\dots, -\infty, 3, +\infty, \dots\}$. For $n = 0$ one has

$$\mathcal{D}_b(0) - \mathcal{D}_a(0) = (-\infty) - 4 = -\infty$$

representing the fact that a starts before b (a has one more event occurrence regarding b).

The time shift function $\mathcal{T}_{a,b}(n)$ gives all the time differences between two series a and b of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ for the same number of event occurrence. Particularly, obtaining the smallest and the largest time shifts can be computed:

$$\mathcal{T}^- = \inf_{\nu \in \mathbb{Z}} \{\mathcal{D}_b(\nu) - \mathcal{D}_a(\nu)\}, \quad (13)$$

$$\mathcal{T}^+ = \sup_{\nu \in \mathbb{Z}} \{\mathcal{D}_b(\nu) - \mathcal{D}_a(\nu)\}. \quad (14)$$

However, regarding Equation (5) describing the residuation operation of two dater functions, these values \mathcal{T}^- and \mathcal{T}^+ are actually obtained by that computation of residuation with $m = 0$ and by taking $\sup = -\inf$ for \mathcal{T}^+ . Indeed, the time shift function only concerns dates of the same number of event occurrences whereas Equation (5) computes all the differences between all the event occurrences. On a series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, $m = 0$ goes to take the exponent of γ equals to 0. When it is not directly given in the minimal representation, it can be obtained in its dater function. Thus, the following theorem uses the residuation computation between series a and b to obtain these bounds \mathcal{T}^- and \mathcal{T}^+ .

Theorem 3 ([17]) *Let $a, b \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, the time shift function given in Definition 9 can be bounded as follows:*

$$\forall n \in \mathbb{Z}, \quad \mathcal{D}_{b\phi a}(0) \leq \mathcal{T}_{a,b}(n) \leq -\mathcal{D}_{a\phi b}(0) \quad (15)$$

where $\mathcal{D}_{b\phi a}(0)$ and $-\mathcal{D}_{a\phi b}(0)$ are directly extracted from series $b\phi a$ and $a\phi b$ by only looking at the monomials containing γ^0 :

$$\begin{aligned} \mathcal{D}_{b\phi a}(0) &\text{ comes from } \gamma^0 \delta^{\mathcal{D}_{b\phi a}(0)} \in b\phi a, \\ -\mathcal{D}_{a\phi b}(0) &\text{ comes from } \gamma^0 \delta^{\mathcal{D}_{a\phi b}(0)} \in a\phi b. \end{aligned}$$

Thus, needed comparison to detect time shifts between two series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ is reduced to compute bounds $\mathcal{D}_{b\phi a}(0)$ and $-\mathcal{D}_{a\phi b}(0)$. Series a and b will be replaced by outputs \tilde{y} and y in the detection of failures of the next section.

Remark 5 In this paper, only the difference between dates of the same number of event occurrences are considered. On further work, the residuation computation could give results on time shifts between different numbers of event occurrences.

Example 11 Let $a, b \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ such that

$$\begin{aligned} a &= \gamma^0 \delta^{12} \oplus \gamma^1 \delta^{15} \oplus \gamma^2 \delta^{18} \oplus \gamma^3 \delta^{21} \oplus \gamma^4 \delta^{+\infty}, \\ b &= \gamma^0 \delta^{12} \oplus \gamma^1 \delta^{15} \oplus \gamma^2 \delta^{19} \oplus \gamma^3 \delta^{23} \oplus \gamma^4 \delta^{+\infty}. \end{aligned}$$

As illustrated in Figure 4, it is clear that the smallest time shift is equal to 0 (reached for γ^0 and γ^1 of both a and b) whereas the largest time shift is equal to 2 (reached for γ^3). Computations of $b\phi a$ and $a\phi b$ provide mathematically these shifts:

$$\begin{aligned} b\phi a &= \gamma^0 \delta^0 \oplus \gamma^1 \delta^3 \oplus \gamma^2 \delta^7 \oplus \gamma^3 \delta^{11} \oplus \gamma^4 \delta^{+\infty}, \\ a\phi b &= \gamma^0 \delta^{-2} \oplus \gamma^1 \delta^2 \oplus \gamma^2 \delta^6 \oplus \gamma^3 \delta^9 \oplus \gamma^4 \delta^{+\infty}. \end{aligned}$$

The smallest time shift $\mathcal{D}_{b\phi a}(0)$ is equal to 0 and is found in the monomial $\gamma^0 \delta^{\mathcal{D}_{b\phi a}(0)} \in b\phi a$. This means that for at least one point, series a and b have the same time occurrence for the same event occurrence numbering (event occurrences 0 and 1 with times 12 and 15 to be specific). The largest time

shift $-\mathcal{D}_{a\cancel{b}}(0)$ is equal to 2 (between monomials $\gamma^3\delta^{21} \in a$ and $\gamma^3\delta^{23} \in b$ to be specific) and is found in the monomial $\gamma^0\delta^{\mathcal{D}_{a\cancel{b}}(0)} \in a\cancel{b}$. This latter series is illustrated in Figure 5 and shows that the largest shift between a and b in the time domain is 2 time units.

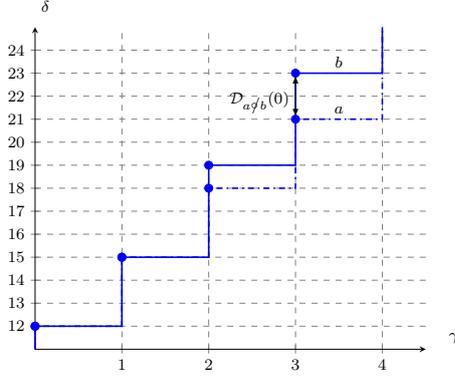


Fig. 4: Series $a, b \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ of example 11.

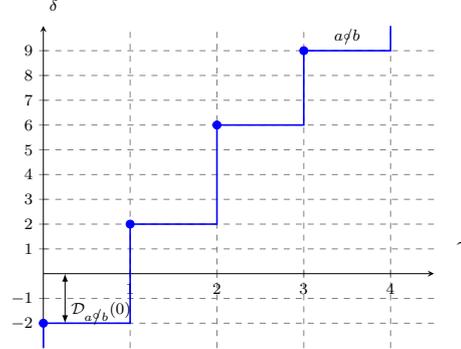


Fig. 5: Maximal bound of the time shift between a and b .

The time shift function $\mathcal{T}_{a,b}(n)$ is about vertical distances of a series b in comparison to a series a . The analysis of the bounds of this time shift function provides indications on the graphical position of b regarding the position of a under Assumptions I and II given at the beginning of this section, where a and b are series representing \tilde{y} and y . This position can be translated into information on time delay or advance of event occurrences of b compared to ones of a .

Property 1 When $\mathcal{D}_{b\cancel{a}}(0)$ and $\mathcal{D}_{a\cancel{b}}(0)$ of Theorem 3 are finite values, three analyses can be obtained:

- if the bounds are both positives

$$\mathcal{D}_{b\cancel{a}}(0) \geq 0 \quad \text{and} \quad -\mathcal{D}_{a\cancel{b}}(0) \geq 0,$$

then b is graphically above a ($a \preceq b$ in the dioid order), meaning that at least one event occurrences of b happened later than occurrences of a (the others happened at the same time);

- if the bounds are both negatives

$$\mathcal{D}_{b\cancel{a}}(0) \leq 0 \quad \text{and} \quad -\mathcal{D}_{a\cancel{b}}(0) \leq 0,$$

then b is graphically under a ($a \succcurlyeq b$ in the dioid order), meaning that at least one event occurrence of b happened before than occurrences of a (the others happened at the same time);

– finally, if one bound is negative and the other is positive

$$\mathcal{D}_{b\neq a}(0) \leq 0 \quad \text{and} \quad -\mathcal{D}_{a\neq b}(0) \geq 0,$$

then b graphically crosses a (they are not comparable in the dioid order), meaning that some event occurrences of b happened later, some before, some possibly at the same time.

Remark 6 When $\mathcal{D}_{b\neq a}(0) = -\mathcal{D}_{a\neq b}(0)$, the time shift between a and b is always the same from the beginning to the end.

A particular analysis has to be made when bounds (only one, no matter the lower or the upper, or the two) is (are) infinite. By taking back Example 10, two cases are possible to obtain infinite bounds:

- one series stops before the other in the numbering of event occurrences;
- one series starts before the other in the numbering of event occurrences.

This introduces an infinite time shift since a comparison is made between a finite number and the infinite value. Moreover, if the two bounds are infinite, then the two series share no number of event occurrence at all.

Property 2 When one bound of Theorem 3 is infinite ($\mathcal{D}_{b\neq a}(0) = -\infty$ or $-\mathcal{D}_{a\neq b}(0) = -(-\infty) = +\infty$), an exponent of γ in one series does not exist in the other⁵ meaning that an event shift happened on event occurrences between a and b . According to the other bound one has:

- if $\mathcal{D}_{b\neq a}(0) \geq 0$ and $-\mathcal{D}_{a\neq b}(0) = +\infty$, then $a \preceq b$;
- if $\mathcal{D}_{b\neq a}(0) = -\infty$ and $-\mathcal{D}_{a\neq b}(0) \leq 0$, then $a \succeq b$;
- if $\mathcal{D}_{b\neq a}(0) = -\infty$ and $-\mathcal{D}_{a\neq b}(0) \geq 0$ or if $\mathcal{D}_{b\neq a}(0) \leq 0$ and $-\mathcal{D}_{a\neq b}(0) = +\infty$, then a and b are not comparable;
- if $\mathcal{D}_{b\neq a}(0) = -\infty$ and $-\mathcal{D}_{a\neq b}(0) = +\infty$, then the event numbering of one series starts just after or more the numbering of the last event of the other series.

The following property points out that even if time shifts happened, series can be equal for some monomials.

Property 3 When one bound of Theorem 3 is equal to 0 (no matter the value of the other bound), series a and b share at least one identical monomial (if several, they do not necessary follow each other). An exception exists when a and b crosses each other: the value 0 is not a bound on minimal and maximal shifts between a and b and will so not appear in neither in $\mathcal{D}_{b\neq a}(0)$ nor in $-\mathcal{D}_{a\neq b}(0)$.

Finally, some combinations of bounds are impossible because of their order.

Property 4 Because bounds $\mathcal{D}_{b\neq a}(0)$ and $-\mathcal{D}_{a\neq b}(0)$ of Theorem 3 are completely ordered, if $\mathcal{D}_{b\neq a}(0)$ is positive, then $-\mathcal{D}_{a\neq b}(0)$ is positive too (it can not be negative). So $\mathcal{D}_{b\neq a}(0) \geq 0$ and $-\mathcal{D}_{a\neq b}(0) \leq 0$ is a combination that can never happen, neither can the combination $\mathcal{D}_{b\neq a}(0) = +\infty$ and $-\mathcal{D}_{a\neq b}(0) = -\infty$.

⁵ But this is not because of the equivalence relation in $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ (see Remark 3).

5.2 Detection of event shifts

To know if series y and \tilde{y} of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ have shifts in the event domain, counter functions are now used with again the right quotient $\not\leftarrow$. In [17], [11] and [22], such event shifts are considered as the representation of some stock in the system.

Definition 10 (Event shift function) Let a and b be series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, $\mathcal{C}_a(t)$ and $\mathcal{C}_b(t)$ their respective counter functions. The *event shift function* representing the event shifts between a and b for each $t \in \mathbb{Z}$ is defined by

$$\mathcal{S}_{a,b}(t) = \mathcal{C}_b(t) - \mathcal{C}_a(t).$$

The event shift function associates the difference of numbering between event occurrences of a and b for the same time t . Obviously, $\mathcal{S}_{a,a}(t) = 0, \forall t \in \mathbb{Z}$. If $\mathcal{S}_{a,b}(t)$ is equal to a positive, respectively negative, value then at the same time t , the numbering of the event occurrence of b is smaller, respectively larger, than the numbering of the event occurrence of a . The same relations than those of Equations (10) and (11) hold for infinite values with finite $\mathcal{C}(t) \in \overline{\mathbb{Z}}$ (see [17]) but not those when both terms are equal to infinite values:

$$(-\infty) - \mathcal{C}(t) = -\infty \quad (+\infty) - \mathcal{C}(t) = +\infty, \quad (16)$$

$$\mathcal{C}(t) - (-\infty) = +\infty \quad \mathcal{C}(t) - (+\infty) = -\infty, \quad (17)$$

$$(-\infty) - (-\infty) = -\infty \quad (+\infty) - (+\infty) = -\infty. \quad (18)$$

Remark 7 Unlike dater functions, infinite values will not be obtained with counter functions during the detection step since series we deal with only contain finite values (see Proposition 1 further).

The event shift function $\mathcal{C}_{a,b}(t)$ gives all the event differences between two series a and b of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ at the same date. Particularly, obtaining the smallest and the largest event shifts can be computed:

$$\mathcal{C}^- = \inf_{\tau \in \mathbb{Z}} \{\mathcal{C}_b(\tau) - \mathcal{C}_a(\tau)\}, \quad (19)$$

$$\mathcal{C}^+ = \sup_{\tau \in \mathbb{Z}} \{\mathcal{C}_b(\tau) - \mathcal{C}_a(\tau)\}. \quad (20)$$

However, regarding Equation (6) describing the residuation operation of two counter functions, these values \mathcal{C}^- and \mathcal{C}^+ are actually obtained by that computation of residuation with $s = 0$ and by taking $\inf = -\sup$ for \mathcal{C}^- . Indeed, the event shift function only concerns event occurrences of the same date whereas Equation (6) computes all the differences between all the dates. On a series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, $s = 0$ goes to take the exponent of δ equals to 0. When it is not directly given in the minimal representation, it can be obtained in its counter function. Thus, the following theorem uses the residuation computation between series a and b to obtain these bounds \mathcal{C}^- and \mathcal{C}^+ .

Theorem 4 ([17]) Let $a, b \in \mathcal{M}_{in}^{ax}[\gamma, \delta]$, the event shift function given in Definition 10 can be bounded as follows:

$$\forall t \in \mathbb{Z}, \quad -\mathcal{C}_{b\phi a}(0) \leq \mathcal{S}_{a,b}(t) \leq \mathcal{C}_{a\phi b}(0). \quad (21)$$

where $-\mathcal{C}_{b\phi a}(0)$ and $\mathcal{C}_{a\phi b}(0)$ are again extracted from the series $b\phi a$ and $a\phi b$ with here a look at their γ exponents when their δ exponents are equals to 0:

$$\begin{aligned} -\mathcal{C}_{b\phi a}(0) &\text{ comes from } \gamma^{\mathcal{C}_{b\phi a}(0)} \delta^0 \in b\phi a, \\ \mathcal{C}_{a\phi b}(0) &\text{ comes from } \gamma^{\mathcal{C}_{a\phi b}(0)} \delta^0 \in a\phi b. \end{aligned}$$

As for the time shift function, needed comparison in the event domain between two series of $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ use the computation of the bounds $-\mathcal{C}_{b\phi a}(0)$ and $\mathcal{C}_{a\phi b}(0)$. Series a and b will be replaced by outputs \tilde{y} and y in the detection of failures of the next section.

Remark 8 In this paper, only the difference between event occurrences of the same date are considered. On further work, the residuation computation could give results on event shifts between different dates of event occurrences.

Example 12 Let $a, b \in \mathcal{M}_{in}^{ax}[\gamma, \delta]$ such that

$$\begin{aligned} a &= \gamma^1 \delta^{12} \oplus \gamma^3 \delta^{15} \oplus \gamma^5 \delta^{18} \oplus \gamma^6 \delta^{+\infty}, \\ b &= \gamma^0 \delta^{12} \oplus \gamma^2 \delta^{15} \oplus \gamma^4 \delta^{18} \oplus \gamma^5 \delta^{+\infty}. \end{aligned}$$

As illustrated Figures 6 and 7, the largest event shift between a and b is equal to 1. It happened for instance at $t = 15$ where the 2^{nd} event occurrence of a happened whereas at the same time, a is already to its 3^{rd} event occurrence. This value is obtained by computations $b\phi a$ and $a\phi b$:

$$\begin{aligned} b\phi a &= \gamma^{-1} \delta^0 \oplus \gamma^1 \delta^3 \oplus \gamma^3 \delta^6 \oplus \gamma^4 \delta^{+\infty}, \\ a\phi b &= \gamma^1 \delta^0 \oplus \gamma^3 \delta^3 \oplus \gamma^5 \delta^6 \oplus \gamma^6 \delta^{+\infty}, \end{aligned}$$

meaning that $-\mathcal{C}_{b\phi a}(0) = 1$ and $\mathcal{C}_{a\phi b}(0) = 1$.

Remark 9 In this example, $\mathcal{D}_{b\phi a}(0) = 0$ and $-\mathcal{D}_{a\phi b}(0) = +\infty$. Then $a \preceq b$ (see Property 2).

The event shift function $\mathcal{S}_{a,b}(t)$ is about horizontal distances of a series b in comparison to a series a . As for the time shifts, the analysis of the bounds of the event shifts provides indications on the graphical position of b regarding the position of a under the same assumptions than those referenced I. and II at the beginning of this section (again, a and b are series representing \tilde{y} and y) and used for the time shift function. This position can be translated into information on delay or advance in the event occurrences of b compared to ones of a .

Property 5 When $-\mathcal{C}_{b\phi a}(0)$ and $\mathcal{C}_{a\phi b}(0)$ of Theorem 4 are finite values, three analyses can be obtained:

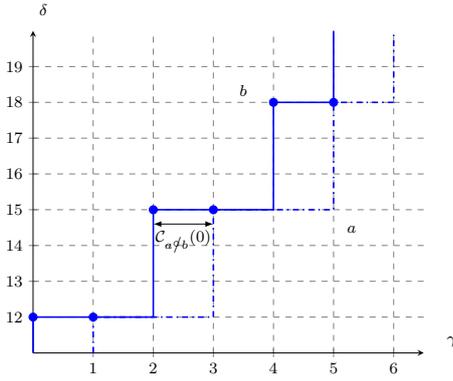


Fig. 6: Series $a, b \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ of Example 12.

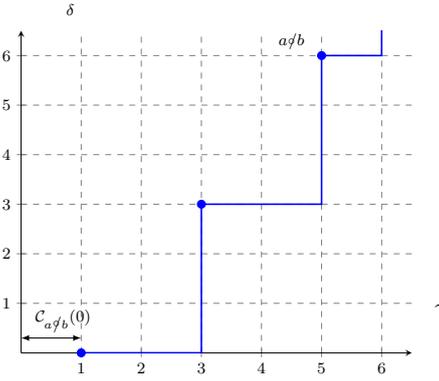


Fig. 7: Maximal bound of the event shift between a and b .

- if the bounds are both positives

$$-\mathcal{C}_{b \neq a}(0) \geq 0 \quad \text{and} \quad \mathcal{C}_{a \neq b}(0) \geq 0,$$

then b is graphically above a ($a \preceq b$ in the dioid order), meaning that b is always equal or late in the numbering of its event occurrences than a ;

- if the bounds are both negatives

$$-\mathcal{C}_{b \neq a}(0) \leq 0 \quad \text{and} \quad \mathcal{C}_{a \neq b}(0) \leq 0,$$

then b is graphically under a ($a \succeq b$ in the dioid order), meaning that b is always equal or in advance in the numbering of its event occurrences than a ;

- finally, if one bound is negative and the other is positive

$$-\mathcal{C}_{b \neq a}(0) \leq 0 \quad \text{and} \quad \mathcal{C}_{a \neq b}(0) \geq 0,$$

then b graphically crosses a (they are not comparable in the dioid order).

Remark 10 When $-\mathcal{C}_{b \neq a}(0) = \mathcal{C}_{a \neq b}(0)$, the event shift between a and b is always the same from the beginning to the end.

The following proposition deals with the result of the assumption of having a and b never equal to ε , meaning that there is always at least one monomial in each series.

Proposition 1 *Bounds $-\mathcal{C}_{b \neq a}(0)$ and $\mathcal{C}_{a \neq b}(0)$ will never reach infinite values.*

Proof As soon as even only one event happened in each a and b , counters $\mathcal{C}_a(t)$ and $\mathcal{C}_b(t)$ are equal to finite values (see Remark 4) the result of their subtraction cannot be infinite.

Now again, even if event shifts happened, series can be equal for some monomials.

Property 6 When one bound of Theorem 4 is equal to 0 (no matter the value of the other bound), series a and b share at least one same numbering of event occurrences in both series.

Example 13 Let a and b be series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$.

- If $a = \gamma^0\delta^1 \oplus \gamma^2\delta^{+\infty}$ and $b = \gamma^0\delta^2 \oplus \gamma^1\delta^{+\infty}$, one has $-\mathcal{C}_{b\neq a}(0) = 0$ meaning that at least one numbering of event occurrences is reached in both series. Indeed, at $t = 1$, series a already has its first event ($\gamma^0\delta^1$) whereas the first event of b will necessarily happen later (at $t = 2$ according to monomial $\gamma^0\delta^2$). More precisely:

$$\begin{array}{c} \dots \\ \mathcal{C}_b(0) - \mathcal{C}_a(0) = 0 - 0 = 0 \\ \mathcal{C}_b(1) - \mathcal{C}_a(1) = 0 - 0 = 0 \\ \mathcal{C}_b(2) - \mathcal{C}_a(2) = 0 - 2 = -2 \\ \mathcal{C}_b(3) - \mathcal{C}_a(3) = 1 - 2 = -1 \\ \mathcal{C}_b(4) - \mathcal{C}_a(4) = 1 - 2 = -1 \\ \dots \end{array}$$

and from Equation (19):

$$\inf_{\tau \in \mathcal{Z}} \{\mathcal{C}_b(\tau) - \mathcal{C}_a(\tau)\} = - \sup_{\tau \in \mathcal{Z}} \{\mathcal{C}_b(\tau) - \mathcal{C}_a(\tau)\} = 0.$$

- If $a = \gamma^1\delta^1 \oplus \gamma^2\delta^{+\infty}$ and $b = \gamma^0\delta^2 \oplus \gamma^1\delta^{+\infty}$, one has $-\mathcal{C}_{b\neq a}(0) = 1$ and $\mathcal{C}_{a\neq b}(0) = 2$ meaning that some numberings of event occurrences achieved by one series is never reached by the other. For instance, at $t = 0$, series b already has its first event ($\gamma^0\delta^2$) whereas the first event of a never happened (the series begins with $\gamma^1\delta^1$).

Finally, as for time shifts, some combinations of bounds are impossible.

Property 7 Because $-\mathcal{C}_{b\neq a}(0)$ and $\mathcal{C}_{a\neq b}(0)$ are completely ordered, if $-\mathcal{C}_{b\neq a}(0)$ is positive, then $\mathcal{C}_{a\neq b}(0)$ is positive too (it can not be negative). So $-\mathcal{C}_{b\neq a}(0) \geq 0$ and $\mathcal{C}_{a\neq b}(0) \leq 0$ is a combination that can never happen.

6 Indicator for the detection of failures

Section 5 proposes a method to mathematically compare series of $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ in order to perform event and time shift detection. Here, we build an indicator for each output of a $(max, +)$ -linear system, so for each output of its TEG, that will be raised if a shift failure is detectable between the real observed output y and the expected output \tilde{y} . This indicator uses the bounds of Equations (15) and (21) and relies on all the analyses done over them in Section 5 by taking $\tilde{y} = a$ and $y = b$.

6.1 Definition of the indicator

The indicator defined here deals with both time and event shifts between y and \tilde{y} provoked by both time and event failures. As explained in Section 3, the indicator relies on the following assumptions:

- all the inputs of the system are observable, so input u is known;
- none of the inputs u_i is empty (see Assumption 1 of Subsection 3.2);
- all the outputs of the system are observable, so output y is known;
- as u and y represent the real inputs and outputs of the system, they are finite (i.e. all series representing these observations end with a monomial $\gamma^n \delta^{+\infty}$, see Assumption 2 of Subsection 3.2).

The proposed indicator relies on the off-line computation of the transfer function h of the system and is defined as follows (general case with $y \neq \varepsilon \neq \tilde{y}$).

Definition 11 (Indicator of shift failures) Let $h \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^{q \times p}$ be the transfer function of a $(max, +)$ -linear system, let $u \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^p$ be the observable input of the system such that $hu \neq \varepsilon$ and $y \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^q \neq \varepsilon$ be its output. Indicator $I(u, y_i)$ is the function⁶:

$$I(u, y_i) = \begin{cases} false & \text{if for } \tilde{y}_i = hu, \tau(y_i, \tilde{y}_i) = \nu(y_i, \tilde{y}_i) = [0; 0] \\ true & \text{otherwise} \end{cases}$$

with

$$\begin{aligned} \tau(\tilde{y}_i, y_i) &= [\mathcal{D}_{y_i \neq \tilde{y}_i}(0) ; -\mathcal{D}_{\tilde{y}_i \neq y_i}(0)], \\ \nu(\tilde{y}_i, y_i) &= [-\mathcal{C}_{y_i \neq \tilde{y}_i}(0) ; \mathcal{C}_{\tilde{y}_i \neq y_i}(0)]. \end{aligned}$$

\mathcal{I} is the set of the q failure indicators of the system.

To obtain these indicators, expected output \tilde{y}_i is first computed with the knowledge of h and of the observable input u . Then, thanks to the right quotient operation between observed output y_i and expected output \tilde{y}_i , time shifts $\tau(\tilde{y}_i, y_i)$ and event shifts $\nu(\tilde{y}_i, y_i)$ are obtained. The indicator is raised when at least one bound of the intervals $\tau(\tilde{y}_i, y_i)$ and $\nu(\tilde{y}_i, y_i)$ is different from 0; it returns false when all the bounds are equal to 0. For the sake of completeness of the indicator definition, we also need to handle specific cases where the predicted and/or real outputs are empty, that are: $I(u, y_i = \varepsilon) = true$ if and only if $\tilde{y}_i \neq \varepsilon$ and if $\tilde{y}_i = \varepsilon$, $I(u, y_i \neq \varepsilon) = true$.

Now, let $\sigma \in \Sigma_o^*$ be a non-empty observation of the system (see Section 3.1) respecting the previous assumptions, we denote:

$$\text{Indicator}(II, \sigma) \equiv \bigvee_{i=1}^q I(u, y_i)$$

⁶ When there is only one output, such as in SISO (Single Input - Single Output) or MISO systems, notation $I(u, y)$ with $\tau(y, \tilde{y})$ and $\nu(y, \tilde{y})$ holds.

where $u = [u_j]_{j \in \{1, \dots, p\}}$ and u_j is the series that represents $P_{\{u_j\}}(\sigma)$ and y_i is the series that represents $P_{\{y_i\}}(\sigma)$ (see Section 4.2).

The next result shows that the defined set of indicators provides a way to solve the failure detection problem (see Section 3.1) that is sound and complete.

Theorem 5 *For any non-empty observation $\sigma \in \Sigma_o^*$ defined as above,*

$$\text{Indicator}(\Pi, \sigma) \equiv \text{Detection}(\Pi, \sigma).$$

Proof (\Rightarrow) Suppose first that $\text{Indicator}(\Pi, \sigma)$ is true. So it means that there exists at least an output y_i such that $I(u, y_i)$ is true. If $y_i \neq \varepsilon$ and $\tilde{y}_i \neq \varepsilon$, either $\tau(y_i, \tilde{y}_i) \neq [0; 0]$ or $\nu(y_i, \tilde{y}_i) \neq [0; 0]$. Suppose $\tau(y_i, \tilde{y}_i) \neq [0; 0]$, Theorem 3 asserts that there exists $t \in \mathbb{Z}$ such that $\mathcal{T}_{y_i, \tilde{y}_i}(t) \neq 0$ so $y_i \neq \tilde{y}_i$, thus $y \neq \tilde{y} = hu \neq \varepsilon$. The real run ρ' of the system must then be such that it has generated u as observable inputs but not hu as observable outputs. It follows that σ cannot be generated by any run $\rho \in \mathcal{L}(\Pi)$ of the system. Therefore, $\text{Detection}(\Pi, \sigma)$ is true. If $\nu(y_i, \tilde{y}_i) \neq [0; 0]$, using a similar reasoning by using Theorem 4, the result also holds. Now, if $y_i = \varepsilon$, then $\tilde{y}_i \neq \varepsilon$, and if $\tilde{y}_i = \varepsilon$ then $y_i \neq \varepsilon$, $\text{Detection}(\Pi, \sigma)$ is also true.

(\Leftarrow) Suppose that $\text{Detection}(\Pi, \sigma)$ is true, it means that there does not exist a run of Π whose observable projection is σ (see Section 3). As input u is observable, the real run of Π that has generated σ must have generated u which implies that there does not exist a run of Π that can generate both u and y . If there exists $y_i = \varepsilon$ then $\tilde{y}_i \neq \varepsilon$ and $\text{Indicator}(\Pi, \sigma)$ is true. If there exists $\tilde{y}_i = \varepsilon$ then $y_i \neq \varepsilon$, $\text{Indicator}(\Pi, \sigma)$ is also true. In the general case, $y \neq \tilde{y} = hu$. Thus, there is at least one output $y_i \neq \tilde{y}_i$ so there exists $t \in \mathbb{Z}$ such that either $\mathcal{S}_{y_i, \tilde{y}_i}(t) \neq 0$ or $\mathcal{T}_{y_i, \tilde{y}_i}(t) \neq 0$ or both. Theorem 3 and Theorem 4 then ensure that either $\nu(y_i, \tilde{y}_i) \neq [0; 0]$ or $\tau(y_i, \tilde{y}_i) \neq [0; 0]$ or both. By Definition 11, $\text{Indicator}(\Pi, \sigma)$ must be true.

Complexity analysis

To determine the complexity of the indicator, we first need to distinguish the off-line part of the computation (that does not rely on any observed input u) and the on-line part. The offline part of the computation consists in computing the transfer function $h = CA^*B$ whose worst-case complexity is based on the computation of A^* . A^* is exclusively composed of monomials and represents the n_T internal transitions of the TEG. Based on the algorithms implemented in the library `minmaxgd` [8], A^* is then a linear number of \otimes products, each product being in $O(n_T^2 \log(n_T))$ (effective product combined with the sorting of the monomials to get a canonical representation of the resulting series). Therefore, the offline computational part is in $O(n_T^3 \log(n_T))$. As far as the online part is concerned, given an input u , the computation of the indicator both requires the computation of $\tilde{y}_i = hu$ and the residuals $\tilde{y}_i \not\neq y_i$ and $\tilde{y}_i \neq y_i$. Let n_u denote the number of input observed events, the number of monomials in any polynomial u_i linearly depends on n_u . The complexity of the computation

of $\tilde{y}_i = hu$ then relies on the complexity of a product between a series of h and a polynomial of u that is in $O(n_h n_u \log(n_h))$ where n_h is the maximal number of monomials in a series of h : the prediction $\tilde{y}_i = hu$ linearly depends on the size of u from a complexity point of view. Finally, the residuals $\tilde{y}_i \not\neq y_i$ and $\tilde{y}_i \neq y_i$ involve only polynomials. The number of monomials in \tilde{y}_i linearly depends on n_u by definition of a TEG. If we suppose that the number of monomials in y_i also linearly depends on n_u then the computation of any residuals is finally in $O(n_u^2 \log(n_u))$.

6.2 Indicator analysis

The proposed indicator gathers the use of Equations (15) and (21) to treat both time and event shift detection. To explain the indications of such indicator, a coupling of the time and event shift analyses made separately in Subsections 5.1 and 5.2 has to be done under assumptions referenced I. and II. at the beginning of Section 5 that still hold. Moreover, remember that we took $\tilde{y} = a$ and $y = b$ to obtain

$$\tau = [\mathcal{D}_{y \neq \tilde{y}}(0); -\mathcal{D}_{\tilde{y} \neq y}(0)] = [\underline{\tau}; \bar{\tau}] \quad \text{and} \quad \nu = [-\mathcal{C}_{y \neq \tilde{y}}(0); \mathcal{C}_{\tilde{y} \neq y}(0)] = [\underline{\nu}; \bar{\nu}].$$

Let τ and ν be shift intervals with finite values as illustrated in Table 1, three combinations, coming from Properties 1 and 5, are translated into failure detection information:

- case 1.a. means that y is slower than \tilde{y} , the shift failure delays the event occurrences of this output;
- case 2.b. means that y is faster than \tilde{y} , the shift failure makes the events of this output be occurring earlier than normally expected;
- case 3.c. means that \tilde{y} and y cross each other, delay and advance are produced by the shift failure.

Moreover, for cases 1.a and 2.b, if τ , respectively ν , contains the same value for its two bounds, the delay or the advance is always the same during all the duration of the observation (see Remarks 6 and 10).

1.	$\tau = [\geq 0; \geq 0]$	a.	$\nu = [\geq 0; \geq 0]$
2.	$\tau = [\leq 0; \leq 0]$	b.	$\nu = [\leq 0; \leq 0]$
3.	$\tau = [\leq 0; \geq 0]$	c.	$\nu = [\leq 0; \geq 0]$

Table 1: Cases of τ and ν with finite values in intervals.

Proposition 2 *In combinations 1.a, 2.b and 3.c of Table 1, because τ is composed with finite values, all the numbering of event occurrences of \tilde{y} are present in y and vice versa.*

Proof According to Theorem 3, $\underline{\tau}$ and $\bar{\tau}$ are the extreme time shifts between y and \tilde{y} among all the time shifts between event occurrences of y and \tilde{y} . If an event occurrence exists in \tilde{y} but not in y , the corresponding time shift is an infinite value and becomes the smallest or the largest time shift as proposed in Property 2. Thus, all the event occurrences of \tilde{y} exist in y and vice versa.

This proposition points out that when a τ interval with finite values is obtained, the observed number of event occurrences is equal to the expected number of event occurrences. In that case, even if an event shift happened, only a time shift is really detected. This particular case can happen when the event failure introduces some tokens on a place involved in the longest duration path of a synchronization. Then, the tokens are ready to be used by the synchronization sooner that if the event shift does not happen and the dates of the real outputs can be in advance regarding the dates of the expected output.

Proposition 3 *When τ and ν are shift intervals with finite values as illustrated in Table 1, no other combination than 1.a, 2.b and 3.c can be done. Moreover, non-ordered bounds in τ or ν cannot exist.*

Proof Because of Theorem 3, a τ positive interval is equivalent to $y \succcurlyeq \tilde{y}$. Because of Theorem 4, a ν positive interval is equivalent to $y \succcurlyeq \tilde{y}$. Then, a τ positive interval implies a ν positive interval and vice versa. The same reasoning holds for negative intervals (cases 2. and b.) and mixed intervals (cases 3. and c.). For the non-ordered bound case, Properties 4 and 7 show that bounds of τ and ν are ordered.

Now, one or both bounds of τ can be equal to an infinite value as explained in Property 2 and recalled in Table 2. Remember that interval ν never contains infinite values (see Proposition 1). One bound of τ equals to ∞ necessary signifies that for at least one event occurrence of y , respectively \tilde{y} , there is no event occurrence of \tilde{y} , respectively y . The number of effectively produced output is lower or upper than the number of desired output. Such a shift can appear when for instance a product is taken from the line to be checked by the quality team: one product will miss at the output regarding the number of expected product. It can also happened when a product is put back in the line after a quality check. The two bounds of τ equal to ∞ signifies that y and \tilde{y} share no number of event occurrence at all. Expected and observed output are completely shifted one from another in the counting of event number point of view.

i. $\tau = [\geq 0 ; +\infty]$	ii. $\tau = [-\infty ; \leq 0]$
iii. $\tau = [\leq 0 ; +\infty]$	iv. $\tau = [-\infty ; \geq 0]$
v. $\tau = [-\infty ; +\infty]$	

Table 2: Cases of τ with infinite values in intervals.

Let τ be time shift intervals with infinite values and ν be event shift intervals with finite values as illustrated in Tables 1 and 2, these combinations, coming from Properties 2 and 5, are translated into failure detection information:

- case i.a. (only positive values in τ and ν) means that y is slower than \tilde{y} , the shift failure delays the event occurrences of this output;
- case ii.b. (only negative values in τ and ν) means that y is faster than \tilde{y} , the shift failure makes event of this output à vérifier: be occurring earlier than normally expected;
- cases iii., iv. or v. with c. (mixed bounds for τ and ν) means that \tilde{y} and y cross each other, delay and advance are produced by the shift failure.

Proposition 4 *When τ contains infinite values as illustrated in Table 2, and with the different configurations of ν as given in Table 1, no other combination than i.a., ii.b., iii.c., iv.c. and v.c. can be done. Moreover, non-ordered bounds in τ or ν can not exist.*

Proof The proof is the same than the one of Proposition 3.

Finally, Table 3 summarizes the possible and impossible cases for intervals τ and ν with:

- ok = possible case because of Propositions 3 and 4;
- \square = impossible case because of Proposition 3;
- \boxtimes = impossible case because of Proposition 4.

		a.	b.	c.
		$[\geq 0; \geq 0]$	$[\leq 0; \leq 0]$	$[\leq 0; \geq 0]$
$\nu =$				
$\tau =$				
1.	$[\geq 0; \geq 0]$	ok	\square	\square
2.	$[\leq 0; \leq 0]$	\square	ok	\square
3.	$[\leq 0; \geq 0]$	\square	\square	ok
i.	$[\geq 0; +\infty]$	ok	\boxtimes	\boxtimes
ii.	$[-\infty; \leq 0]$	\boxtimes	ok	\boxtimes
iii.	$[\leq 0; +\infty]$	\boxtimes	\boxtimes	ok
iv.	$[-\infty; \geq 0]$	\boxtimes	\boxtimes	ok
v.	$[-\infty; +\infty]$	\boxtimes	\boxtimes	ok

Table 3: Summary of cases for τ (with finite and infinite values) and ν (finite values).

All individual bound analyses made in Section 5 still hold for a complete analysis of the detected failure.

6.3 Simple examples of uses

Let us take back the MISO $(max, +)$ -linear system of Figure 1. When the system is not failing, remember that for the following input:

$$u = \begin{pmatrix} \gamma^0 \delta^2 \oplus \gamma^1 \delta^3 \oplus \gamma^2 \delta^5 \oplus \gamma^4 \delta^{+\infty} \\ \gamma^0 \delta^2 \oplus \gamma^1 \delta^3 \oplus \gamma^2 \delta^5 \oplus \gamma^4 \delta^{+\infty} \end{pmatrix}$$

the expected output is:

$$\tilde{y} = hu = \gamma^0\delta^{12} \oplus \gamma^1\delta^{15} \oplus \gamma^2\delta^{18} \oplus \gamma^3\delta^{21} \oplus \gamma^4\delta^{+\infty}.$$

Firstly, assume that there is a time failure from transition x_1 to transition x_2 that generates a delay of 2 time units (machine M_1 unexpectedly spends 4 time units to process instead of 2). The observed output⁷ is then:

$$y = \gamma^0\delta^{12} \oplus \gamma^1\delta^{15} \oplus \gamma^2\delta^{19} \oplus \gamma^3\delta^{23} \oplus \gamma^4\delta^{+\infty}.$$

The computations of $y\phi\tilde{y}$ and $\tilde{y}\phi y$ give

$$\begin{aligned} y\phi\tilde{y} &= \gamma^0\delta^0 \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^7 \oplus \gamma^3\delta^{11} \oplus \gamma^4\delta^{+\infty}, \\ \tilde{y}\phi y &= \gamma^0\delta^{-2} \oplus \gamma^1\delta^2 \oplus \gamma^2\delta^6 \oplus \gamma^3\delta^9 \oplus \gamma^4\delta^{+\infty} \\ &= \gamma^0\delta^{-2} \oplus \gamma^1\delta^0 \oplus \gamma^1\delta^1 \oplus \gamma^1\delta^2 \oplus \gamma^2\delta^6 \oplus \gamma^3\delta^9 \oplus \gamma^4\delta^{+\infty}. \end{aligned}$$

Thus, $I(u, y) = true$ because

$$\begin{aligned} \tau(\tilde{y}, y) &= [\mathcal{D}_{y\phi\tilde{y}}(0); -\mathcal{D}_{\tilde{y}\phi y}(0)] = [0; 2], \\ \nu(\tilde{y}, y) &= [-\mathcal{C}_{y\phi\tilde{y}}(0); \mathcal{C}_{\tilde{y}\phi y}(0)] = [0; 1]. \end{aligned}$$

The shift failure is detected and corresponds to the case 1.a. of previous subsection. Moreover, since bounds are both composed of positive and finite values, this failure has slowed down the production rate of the system. Value 0 in τ tells us that at some point, the real output was exactly as expected. The largest time shift is equal to 2 and the largest event shift is equal to 1. Value 0 in ν indicates that at least one numbering of event occurrences is reached in both series. This is largely confirmed by τ since it contains only finite numbers: all the numberings of \tilde{y} are actually reached by y .

Now, assume that an event failure produces the add of a token on place p_6 between x_4 and x_5 . The observed outputs is:

$$y = \gamma^0\delta^9 \oplus \gamma^1\delta^{12} \oplus \gamma^2\delta^{15} \oplus \gamma^3\delta^{18} \oplus \gamma^4\delta^{+\infty}.$$

The computations of $y\phi\tilde{y}$ and $\tilde{y}\phi y$ give

$$\begin{aligned} y\phi\tilde{y} &= \gamma^0\delta^{-3} \oplus \gamma^1\delta^0 \oplus \gamma^2\delta^3 \oplus \gamma^3\delta^6 \oplus \gamma^4\delta^{+\infty}, \\ \tilde{y}\phi y &= \gamma^0\delta^3 \oplus \gamma^1\delta^6 \oplus \gamma^2\delta^9 \oplus \gamma^3\delta^{12} \oplus \gamma^4\delta^{+\infty} \\ &= \gamma^0\delta^0 \oplus \gamma^0\delta^3 \oplus \gamma^1\delta^6 \oplus \gamma^2\delta^9 \oplus \gamma^3\delta^{12} \oplus \gamma^4\delta^{+\infty}. \end{aligned}$$

Thus, $I(u, y) = true$ because

$$\begin{aligned} \tau(\tilde{y}, y) &= [\mathcal{D}_{y\phi\tilde{y}}(0); -\mathcal{D}_{\tilde{y}\phi y}(0)] = [-3; -3], \\ \nu(\tilde{y}, y) &= [-\mathcal{C}_{y\phi\tilde{y}}(0); \mathcal{C}_{\tilde{y}\phi y}(0)] = [-1; 0]. \end{aligned}$$

⁷ \tilde{y} and y are respectively equal to series a and b from Example 11.

The shift failure is detected and corresponds to the case 2.b. of previous subsection. Its effect is such that y is always faster than \tilde{y} of 3 time units. This comes from the token added in the place with the longest duration just before the synchronization illustrated by transition x_5 . Moreover, since τ contains only finite numbers, all the numbering of \tilde{y} is reached by y . Thus, there is no direct detection of the event shift since it only accelerates the process without loss of event occurrence of y regarding \tilde{y} (of which we had a preview since one bound of ν is equal to 0).

An event failure produces the loss of a token on place p_{11} between x_6 and x_5 . The observed outputs is:

$$y = \gamma^0 \delta^{12} \oplus \gamma^1 \delta^{15} \oplus \gamma^2 \delta^{18} \oplus \gamma^3 \delta^{21} \oplus \gamma^4 \delta^{+\infty}.$$

The computations of $y\phi\tilde{y}$ and $\tilde{y}\phi y$ give

$$\begin{aligned} y\phi\tilde{y} &= \gamma^0 \delta^0 \oplus \gamma^1 \delta^3 \oplus \gamma^2 \delta^6 \oplus \gamma^3 \delta^9 \oplus \gamma^4 \delta^{+\infty}, \\ \tilde{y}\phi y &= \gamma^0 \delta^0 \oplus \gamma^1 \delta^3 \oplus \gamma^2 \delta^6 \oplus \gamma^3 \delta^9 \oplus \gamma^4 \delta^{+\infty}. \end{aligned}$$

Thus, $I(u, y) = false$ because

$$\begin{aligned} \tau(\tilde{y}, y) &= [\mathcal{D}_{y\phi\tilde{y}}(0); -\mathcal{D}_{\tilde{y}\phi y}(0)] = [0; 0], \\ \nu(\tilde{y}, y) &= [-\mathcal{C}_{y\phi\tilde{y}}(0); \mathcal{C}_{\tilde{y}\phi y}(0)] = [0; 0]. \end{aligned}$$

The event shift is not detected because the input was sufficiently slowed regarding the system behavior to not need the use of the complete machine M_3 for doing the assembly of the products. So, even if an event failures provokes the breakdown of one part of this machine, that does not slow the produced output.

Finally, an event failure produces the add of a token in place p_8 between x_6 and y . The observed output is:

$$y = \gamma^0 \delta^0 \oplus \gamma^1 \delta^{12} \oplus \gamma^2 \delta^{15} \oplus \gamma^3 \delta^{18} \oplus \gamma^4 \delta^{21} \oplus \gamma^5 \delta^{+\infty}.$$

The computations of $y\phi\tilde{y}$ and $\tilde{y}\phi y$ give

$$\begin{aligned} y\phi\tilde{y} &= \gamma^1 \delta^0 \oplus \gamma^2 \delta^3 \oplus \gamma^3 \delta^6 \oplus \gamma^4 \delta^9 \oplus \gamma^5 \delta^{+\infty}, \\ \tilde{y}\phi y &= \gamma^0 \delta^3 \oplus \gamma^1 \delta^6 \oplus \gamma^2 \delta^9 \oplus \gamma^3 \delta^{+\infty} \\ &= \gamma^0 \delta^0 \oplus \gamma^0 \delta^3 \oplus \gamma^1 \delta^6 \oplus \gamma^2 \delta^9 \oplus \gamma^3 \delta^{+\infty}. \end{aligned}$$

Thus, $I(u, y) = true$ because

$$\begin{aligned} \tau(\tilde{y}, y) &= [\mathcal{D}_{y\phi\tilde{y}}(0); -\mathcal{D}_{\tilde{y}\phi y}(0)] = [-\infty; -3], \\ \nu(\tilde{y}, y) &= [-\mathcal{C}_{y\phi\tilde{y}}(0); \mathcal{C}_{\tilde{y}\phi y}(0)] = [-1; 0]. \end{aligned}$$

The event shift is detected and corresponds to the case ii.b. of previous subsection. This detection shows that the number of observed output is upper than the number of expected output and that it is faster than expected output. Indeed, since a token is ready to be used by the output without any action on the input, the output fires it as soon as the system starts (that is monomial $\gamma^0 \delta^0$) and then the tokens given by the input are used by the output.

7 Localization of failures

Once the step of detection of failures is achieved by the detection of shifts between the observed outputs and the expected outputs, the next step in the failure diagnosis process is about the localization of the source of such shifts. This work was previously presented in [13] but for only time shift failures. Here, both failures, time and event, are gathered in the indicator and so in the localization step. Before presenting the localization method, we firstly propose a formal characterization of these types of failures in the $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ framework that shows that any type of failures investigated here has an effect on a place the TEG.

7.1 Time and event shift failure

The type of failures that are considered in this paper are time shifts, having an impact on the holding time of the TEG places, and event shifts, having an impact on their number of tokens. In a TEG of a $(max, +)$ -linear system representing its normal behavior, places are characterized by a monomial non-equal to ε in matrices A , B and C of its state representation. A time shift failure has for impact that the holding time of the corresponding place is increased or decreased as defined below. Such a failure represents for instance delay in the availability of a piece of equipment in an assembly line, delay for luggage delivery in the network of conveyors or unexpected advance in the transfer of a box.

Definition 12 (Time shift failure) In a TEG of a $(max, +)$ -linear system, by writing $\gamma^n \delta^t$ the monomial representing a place p in matrices A , B or C of its state representation, the impact of a time shift failure f_τ on this place is written as follows:

$$\gamma^n \delta^t \otimes f_\tau \quad \text{with} \quad f_\tau = \begin{cases} \gamma^0 \delta^{t'} & \text{if a time shift failure occurs,} \\ e & \text{otherwise,} \end{cases}$$

where $t' \in \mathbb{Z}$ and $t' \geq -t$.

Example 14 The transfer of a box takes 3 time units to be done: $\gamma^0 \delta^3$ as illustrated Figure 8 by a single place. If a delay of 2 time units happens $f_\tau = \gamma^0 \delta^2$, the time of the transfer is slowed and last 5 time units:

$$\gamma^0 \delta^3 \otimes f_\tau = \gamma^0 \delta^3 \otimes \gamma^0 \delta^2 = \gamma^0 \delta^{3+2} = \gamma^0 \delta^5.$$

Another case is when a time shift of 2 time units accelerate the box transfer:

$$\gamma^0 \delta^3 \otimes f_\tau = \gamma^0 \delta^3 \otimes \gamma^0 \delta^{-2} = \gamma^0 \delta^{3-2} = \gamma^0 \delta^1.$$

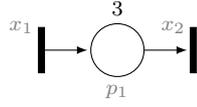


Fig. 8: TEG of a box transfer.

About the event shift failure, it has for impact that the number of tokens in a place is increased or decreased (but is never negative). Such a failure represents for instance loss or add of resources such as the breakdown or the add of machines among others doing the same process, or the removal of a product to be checked by the quality team in a production line.

Definition 13 (Event shift failure) In a TEG of a $(max, +)$ -linear system, by writing $\gamma^n \delta^t$ the monomial representing a place p in matrices A , B or C of its state representation, the impact of an event shift failure f_ν on this place is written as follows:

$$\gamma^n \delta^t \otimes f_\nu \quad \text{with} \quad f_\nu = \begin{cases} \gamma^{n'} \delta^0 & \text{if an event shift failure occurs,} \\ e & \text{otherwise,} \end{cases}$$

where $n' \in \mathbb{Z}$ and $n' \geq -n$.

Example 15 Figure 9 shows a simple process of a product working at a rate of 2 products ($\gamma^2 \delta^0$ for p_4) per 2 time units (holding time in p_2). The breakdown of a machine in p_4 allows now only 1 product to be processed per 2 time units. The impact on p_4 is the loss of a token and can be written with its corresponding monomial:

$$\gamma^2 \delta^0 \otimes f_\nu = \gamma^2 \delta^0 \otimes \gamma^{-1} \delta^0 = \gamma^{2-1} \delta^0 = \gamma^1 \delta^0.$$

On the opposite, if a machine is added to the normal behavior, the production rate is increased and the impact on p_4 is seen on γ :

$$\gamma^2 \delta^0 \otimes f_\nu = \gamma^2 \delta^0 \otimes \gamma^1 \delta^0 = \gamma^{2+1} \delta^0 = \gamma^3 \delta^0.$$

Now, if a product already processed is added after the machines on the production chain, this can for instance have an impact on p_3 as follows:

$$\gamma^0 \delta^0 \otimes f_\nu = \gamma^0 \delta^0 \otimes \gamma^1 \delta^0 = \gamma^{0+1} \delta^0 = \gamma^1 \delta^0.$$

7.2 Failure signature

As explained in the previous subsection, a shift failure in a $(max, +)$ -linear system actually causes a modification of the holding times of places (failure of type f_τ) or on its number of tokens (failure of type f_ν) of the corresponding TEG. Thus, there are as many possible time-shift failures (respectively event-shift failures) as places. Let \mathcal{F} denote the set of such possible failures in the

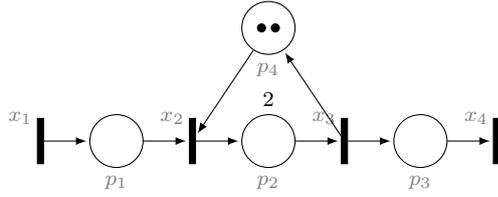


Fig. 9: Nominal behavior of a loop in a TEG.

TEG. The effect of such a failure can be observed on several outputs and so several indicators can be involved. From the knowledge of the system TEG, it is possible to obtain links between the output designated by the indicator and the failure that causes the shift if the TEG is structurally observable.

Definition 14 (Structural observability, [17]) A TEG is structurally observable if, from every internal transition x_i , there exists a path to at least one output transition y_i .

From this definition, the studied TEG of this paper are all structurally observable. In the following, we denote by \mathcal{I} the set of available indicators.

Definition 15 (Failure signature) Let f_i be a failure (either an event-shift or a time-shift failure) associated with a place i of the TEG, the *failure signature* of f_i is the set of indicators $\mathcal{I}_{f_i} \subseteq \mathcal{I}$ such that for any indicator $I(u, y_j) \in \mathcal{I}_{f_i}$, there exists a path from place i to the output transition y_j in the TEG.

Property 8 If f_i occurs in the system, then only indicators from \mathcal{I}_{f_i} may return true.

This property is due to the structural observability, it means in particular that if an indicator $I(u, y_k) \in \mathcal{I}_{f_j} \setminus \mathcal{I}_{f_i}$ returns true, it cannot be explained by the occurrence of f_i .

Example 16 With the TEG illustrated Figure 3, the failure signature \mathcal{I}_{f_1} is composed by the four indicators $I(u, y_1)$, $I(u, y_2)$, $I(u, y_3)$ and $I(u, y_4)$. Now, by reasoning from the failures to the indicators, one can see that output y_4 may be affected by failures $f_1, f_2, f_5, f_6, f_{11}$ (respectively associated to places $p_1, p_2, p_5, p_6, p_{11}$). The presence of a failure on one of these places can produce a shift detected by indicator $I(u, y_4)$ and indicator $I(u, y_4)$ is part of the signature of all these failures.

Thus, each shift failure has a signature containing all the indicators that can be raised if it occurs. They are all gathered in the *matrix of signatures* of the system.

Definition 16 (Signature Matrix) The *signature matrix* of the system is the relation $\mathcal{I} \times \mathcal{F}$ denoted $M \in \mathbb{B}^{|\mathcal{I}| \times |\mathcal{F}|}$ for which each element is defined by:

$$M_{ij} = \begin{cases} 1 & \text{if } I(u, y_i) \in \mathcal{I}_{f_j}, \\ 0 & \text{otherwise.} \end{cases}$$

Example 17 The signature matrix of the TEG of Figure 3 is the following:

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Lines are indicators on the observed outputs: $\mathcal{I} = \{I(u, y_1), I(u, y_2), I(u, y_3), I(u, y_4)\}$. Columns are the time shift failures that can happen in this system: $\mathcal{F} = \{f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}\}$.

Thanks to this matrix and with the state of indicators, a set of candidate failures is proposed.

Definition 17 (Candidate failure) A failure f_j is a *candidate failure* if it exists an indicator $I(u, y_i)$ with $M_{ij} = 1$ that returns true.

Example 18 In Figure 3, if indicators $I(u, y_1)$ and $I(u, y_3)$ return true whereas the other indicators return false, then $f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_{10}$ are the candidate failures. By construction, no failure among f_9 and f_{11} can explain the value of indicators.

7.3 Characteristic signature

The signature matrix points out different failures that can explain the raise of indicators. However, it is about a set of possibilities, not a strict implication. Indeed, in a $(max, +)$ -linear system, when a shift failure occurs in a path upstream a synchronization, it is possible that its effect is totally hidden by the synchronization. For a time failure, the time shift can be counterbalanced by a higher process duration in another path. For instance, in the TEG of Figure 3, several synchronizations are present between several paths that arrive on outputs except y_3 . For an event failure, the event shift can also be hidden if the token is added in the path with the slowest process duration as illustrated in the second example of Subsection 6.3.

So, a shift failure f_i can raise the indicator $I(u, y_i)$ but *not necessarily* because of downstream synchronizations. But, if one (and only one) failure happens in a place of the TEG necessarily involved in a path leading from one input to one output of the system but without synchronization, the indicator of this output will be necessarily raised. For instance, in TEG of Figure 3, if f_8 occurs, it will necessarily have an effect on the output y_1 . Similarly, if f_1 or f_{10} occurs, a shift will necessarily appear on y_3 without the possibility to be counterbalanced by any synchronization. Thus, the definition of a characteristic signature illustrating this statement is possible.

Definition 18 (Characteristic signature) The set $\mathcal{I}_{f_i}^c \subseteq \mathcal{I}_{f_i} \subseteq \mathcal{I}$ is the set of indicators returning *necessarily* true when a shift failure f_i occurs ($\forall I(u, y) \in \mathcal{I}_{f_i}^c, I(u, y) \equiv true$). $\mathcal{I}_{f_i}^c$ is called the characteristic signature of f_i .

Definition 19 (Characteristic signature matrix) The characteristic signature matrix of the system is the relation $\mathcal{I} \times \mathcal{F}$ denoted $M^c \in \mathbb{B}^{|\mathcal{I}| \times |\mathcal{F}|}$ for which each element is defined by:

$$M_{ij}^c = \begin{cases} 1 & \text{if } I(u, y_i) \in \mathcal{I}_{f_j}^c, \\ 0 & \text{otherwise.} \end{cases}$$

Example 19 The characteristic signature matrix of the TEG of Figure 3 is the following:

$$M^c = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

As in the previous signature matrix, lines are the indicators over the observed outputs: $\mathcal{I} = \{I(u, y_1), I(u, y_2), I(u, y_3), I(u, y_4)\}$. Columns are the shift failures that can happen in this system: $\mathcal{F} = \{f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}\}$.

Such characteristic signature matrix can be used to refine the localization made earlier. Indeed, under the *single failure assumption* meaning that only one failure has happened in the system, if an indicator of the characteristic signature of a failure f_i has not raised, then f_i surely has not happened and can be removed from the set of candidates.

Definition 20 (Minimal set of candidate failures) Under the *single failure assumption*, the minimal set of candidate failures is the set of failures f_i such that:

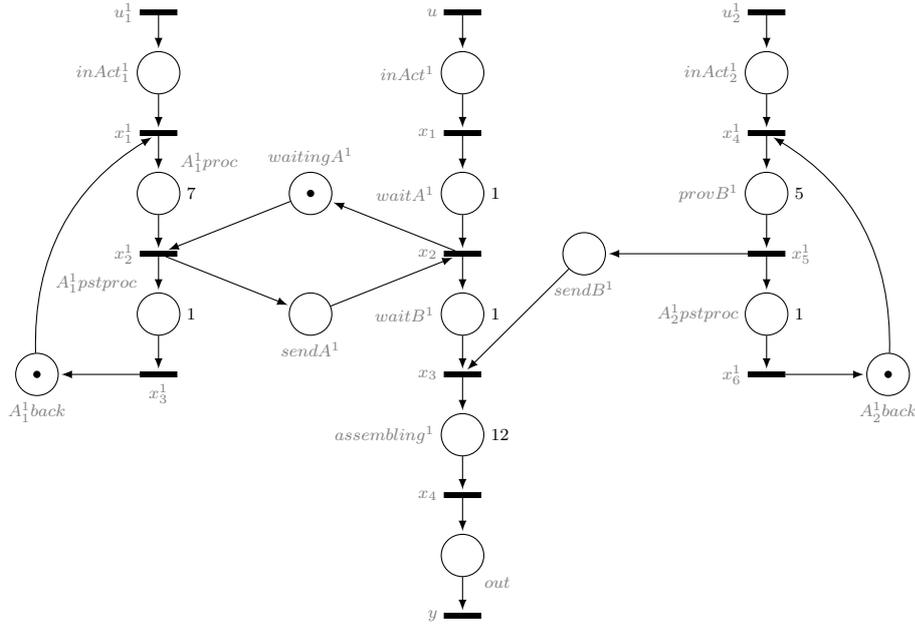
- for all indicators $I(u, y_j)$ such that $M_{ij}^c = 1$, $I(u, y_j)$ is true;
- in the case when $\mathcal{I}_{f_i}^c = \emptyset$, there exists *at least one* indicator such that $M_{ik} = 1$ that returns true.

Example 20 On Figure 3, if indicators $I(u, y_1)$ and $I(u, y_3)$ return true, the set of candidate failures found in M is $\{f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_{10}\}$. But, the analysis of the characteristic signature matrix M^c tells us that there is no observation of time shift on y_2 that excludes failure f_7 . So, the minimal set of candidate failures is $\{f_1, f_2, f_3, f_4, f_5, f_6, f_8, f_{10}\}$.

8 Example of application

As an illustrative exemple, this section presents some experimental results on a TEG that is depicted on Figure 10. This entire experiment is fully implemented in a C++ toolbox, called MAXPLUSDIAG, that relies on the `minmaxgd` library.

TEG of Figure 10 represents a typical manufacturing process we aim at diagnosing. The process is composed of a main line (from u to y). At some steps, the line is waiting for a resource of type A that is provided by another

Fig. 10: A manufacturing process MP .

line called *Activity 1*. Similarly, the main line is also waiting for a resource of type B provided by *Activity 2* so that the main line can finally assemble the final product.

This subsection presents an exhaustive experimental analysis of single event/time shift failures in the process MP . This process, as depicted in Figure 10, is composed of 3 inputs, namely, u_1^1 , u_2^1 and the main input u . It also has one output y : it is thus composed of four observable series. In this experimental setting, we call a configuration of MP a subset of internal transitions of MP that are observed with the help of measurable output transitions (see Definition 4). As there are 11 internal transitions, there are 1024 configurations (from the configuration with no measurable output transitions till the configuration with 11 measurable output transitions). In order to show how the proposed set of indicators behaves on this process, for any possible configuration of MP , we simulate for each possible place of MP :

- a time shift failure of $+2$ (on 16 places);
- a time shift failure of -1 (on the 7 places that have a time greater or equal to 1);
- an event shift failure of $+2$ (on 16 places);
- an event shift failure of -1 (on the 3 places that have a token).

To sum up, for every configuration, there are 42 fault injection problems to be solved. As the number of possible configurations is 1024, the presented

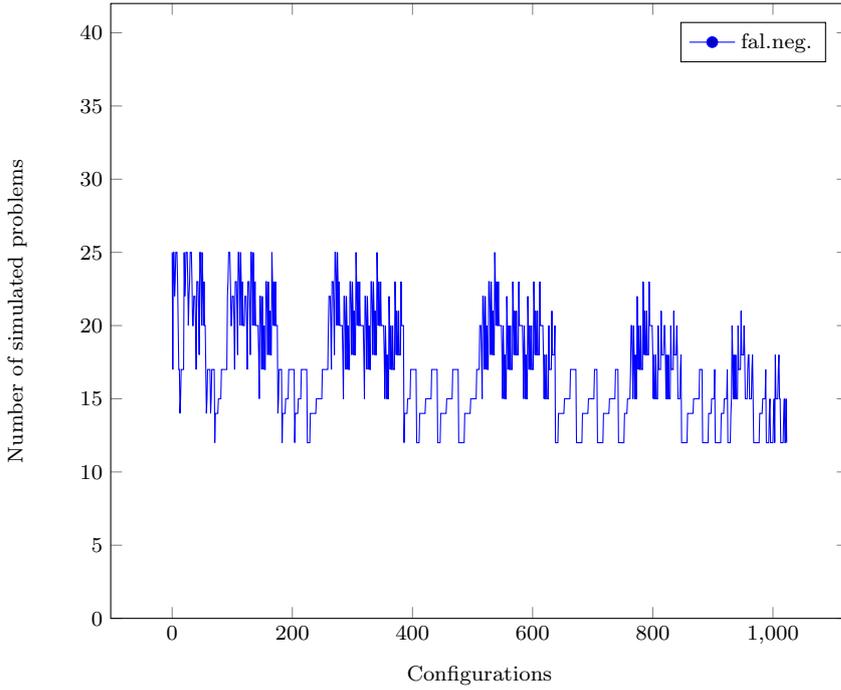


Fig. 11: False negative detections

simulation solves 43008 problems. The inputs that used for these simulations are:

$$u_1^1 = u_2^1 = u = \gamma^0 \delta^0 \oplus \gamma^1 \delta^{20} \oplus \gamma^2 \delta^{40} \oplus \gamma^3 \delta^{60} \oplus \gamma^4 \delta^{+\infty}.$$

Figure 11 presents the rate of false negative detections over the 1024 possible configurations: configurations are ordered from the configurations with less measurable output transitions till the configuration with the maximal set of measurable output transitions. As expected, the number of false negative detections decrease with the increase of the number of measurable output transitions. However, due to synchronization phenomena within the process, some shift failures are not detectable. Indeed, if a shift failure occurs in a place but its output transition waits for another token from another place, the shift failure has no effect over the propagation time so has no effect on the observed outputs. Simulations then shows there are 12 non detectable shift failures (time shift (+2) on places A_1^1back , A_2^1back , $waitA^1$, $sendB^1$, $waitingA^1$, time shift (-1) on place $waitA^1$, event shift (+2) on places A_1^1back , A_2^1back , $waitA^1$, $sendB^1$, $waitingA^1$, and event shift (-1) on place $waitingA^1$).

Simulations show that:

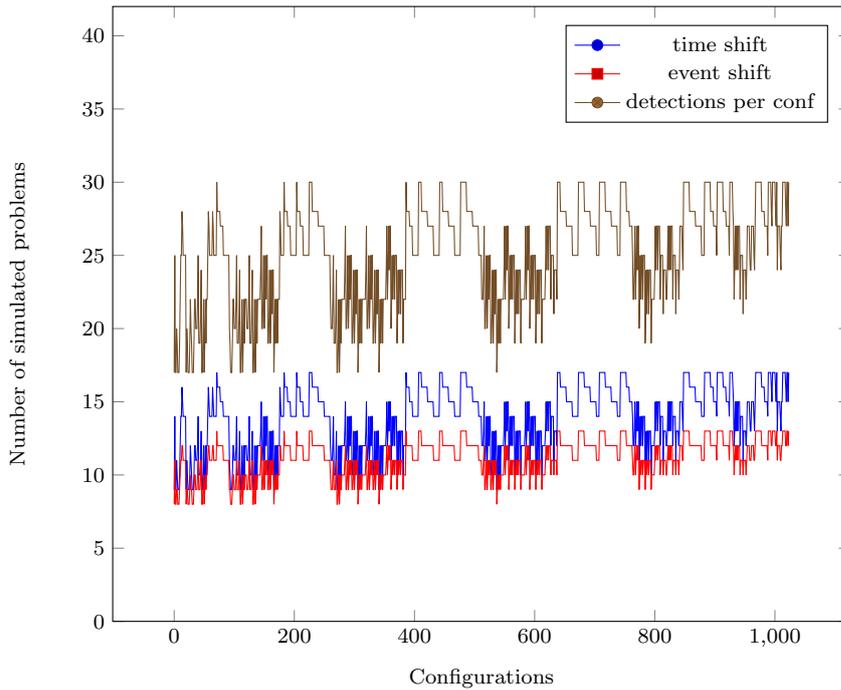


Fig. 12: Time shift detections

- as soon as a time shift failure has been injected, the analysis of the indicator as presented in Section 6.2 always leads to the detection of a time shift failure as a possible diagnosis candidate.
- similarly, as soon as an event shift failure has been injected, the analysis of the indicator always leads to the detection of an event shift failure as a possible diagnosis candidate.

Note that, for some problems, both event shift/time shift failures are possible diagnosis candidates.

Figure 12 presents this result. The curve named *detection per conf* presents the number of problems that have been detected (in this sense, it is obtained by subtracting the number of false negatives of Figure 11 to the 42 simulated problems). The curve named *time shift* returns the number of time shift failure problems that have been detected as a time shift failure. The curve named *event shift* returns the number of event shift failure problems that have been detected as an event failure. It is noticeable that the first curve is actually the sum of both last curves.

As far as the localization is concerned, whatever the problem is, in case of a detection then any injected failing place is always part of the candidates. If the failure is not part of the minimal set of candidates, it simply means that its characteristic signature is empty. Figure 13 shows the percentage of cases

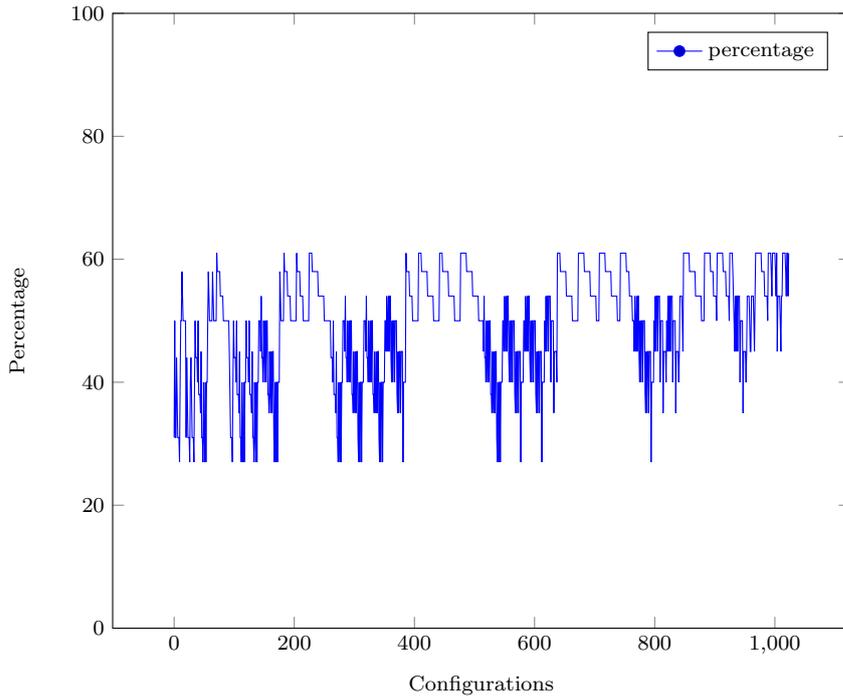


Fig. 13: Percentage of cases where the injected fault is part of the minimal set of candidates.

where the injected failure is part of the minimal set of candidates. Figure shows that the percentage increases with the number of measurable output transitions and more particularly measurable output transitions linked to internal transitions that are not synchronised transitions. Indeed, an observed synchronization transition cannot be part of the characteristic signature of a failure so its observation cannot improve the quality of the minimal set of candidates. Meanwhile, the observation of a transition that is not a synchronization has a direct impact of the characteristic signatures and improves the detection. In any configuration when this subset of transitions is observed, the part of injected failures being part of the minimal set is 61 percent.

This section presented an exhaustive analysis about how the proposed detection system behaves given all the possible indicator configurations. The results of these set of 43008 experiments can be summarized as follows:

1. as soon as a failure is detectable (i.e. its effect is propagated until at least one observable transition), it is detected by one indicator at least;
2. based on the indicator analyses, the diagnosis system always returns at least the correct type of failure (event shift or time shift failure);
3. any injected failure that is detected is always part of the failure candidates;

4. as the conducted experiments are based on a single failure, any detected injected failure that has a characteristic signature is always part of the minimal set of failure candidates.

9 Conclusion

This paper deals with a method to detect and localize time and event shift failures in systems modeled by Timed Event Graphs and represented by $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ equations. These systems are well-suited to manage timed processes with synchronization of equipments typical of smart manufacturing.

Automated failure detection of performance problems such as time and event shifts is proposed through a failure indicator on each output. The set of indicators is built by using $(max, +)$ -algebra, more precisely the residuation of $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ series. Given an input flow of timed observations, it formally compares the expected and observed outputs of the system. Then, numerical values of indicators, that are interval of time units and numbering of event occurrences, give us information about the detected shifts.

After the detection step, the localization of failures that made indicators raised is proposed. Based on the structural analysis of the TEGs and the set of available indicators, we introduce the notion of failure signatures for TEG based on which it is then possible to determine the possible sources of the time or event shift in the system.

There are many perspectives of this work to build a complete diagnosis process for TEG and $(max, +)$ -linear systems.

A first way of investigation is to extend the TEG model to handle bounded holding time or tokens uncertainties in places thanks to the dioid of intervals (see [14] and [15]). In such modeling, minimal and maximal bounds of holding times or tokens allows uncertain or variable duration process to be considered that is more realistic in a smart manufacturing context. Then, new indicators have to be found and under the assumption of exact observations (so no interval of time or of event for the observations of input and output trajectories) some first simple leads are to use shifts of observed output with both maximal and minimal expected output. In that case, failures that do not make the real output go out the interval of expected output will not be detected.

Another very interesting lead is about the diagnosability question as well as the sensor placement problem (that can be formalized through observers). Indeed, faults can be compensated by synchronization phenomena so that they might totally be silent from a global point of view, and adding local sensors is then necessary. Moreover, such new information could drive more efficiently discrimination of time shifts regarding event shifts.

Finally, the indicator proposed here is only binary as it only relies on checking whether a property holds on a set of monomials from the computed residuals. However residuals also contain quantitative pieces of information as differences for the same event occurrences or for the same dates but also differences shifted by one, by two, ... occurrences or dates. Our objective,

in case of a failure detection, is then to exploit the quantitative pieces of information stored in the residuals to better characterise the event and time shift failures and especially check whether there are cumulative effects of shift failures (if so, it would indicate the cause of a failure are permanent and generates more and more time/event shifts) or not (the failure occurred at a time point and then disappeared, may be due to underlying intermittent/latent equipment faults).

References

1. Baccelli, F., Cohen, G., Olsder, G.J., Quadrat, J.-P.: Synchronization and linearity: an algebra for discrete event systems. Wiley and sons (1992)
2. Baniardalani, S., Askari, J.: Fault diagnosis of timed discrete event systems using dioid algebra. *International Journal of Control, Automation and Systems* pp. 1095–1105 (2013)
3. Basile, F., Cabasino, M.P., Seatzu, C.: Diagnosability analysis of labeled time petri net systems. *IEEE Transactions on Automatic Control* **62**(3), 1384–1396 (2017). DOI 10.1109/TAC.2016.2588736
4. Blyth, T.: Lattices and ordered algebraic structures. Springer-Verlag London (2005)
5. Bouyer, P., Chevalier, F., D'Souza, D.: Fault diagnosis using timed automata. In: *Proceedings of the 8th International Conference on Foundations of Software Science and Computation Structures* (2005)
6. Bouyer, P., Jaziri, S., Markey, N.: Efficient timed diagnosis using automata with timed domains. In: C. Colombo, M. Leucker (eds.) *Runtime Verification*, pp. 205–221. Springer International Publishing, Cham (2018)
7. Cohen, G., Moller, P., Quadrat, J.-P., Viot, M.: Algebraic tools for the performance evaluation of discrete event systems. *Proceedings of the IEEE* **77**(1), 39–85 (1989)
8. Cottenceau, B., Lhommeau, M., Hardouin, L., Boimond, J.-L.: Data processing tool for calculation in dioid. In: *5th International Workshop on Discrete Event Systems* (2000). <http://www.istia.univ-angers.fr/hardouin/outils.html>
9. Dousson, C., Duong, T.V.: Discovering chronicles with numerical time constraints from alarm logs for monitoring dynamic systems. In: *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence* (1999)
10. Ghazel, M., Toguyéni, A., Yim, P.: State observer for des under partial observation with time petri nets. *Discrete Event Dynamic Systems* **19**(2), 137–165 (2009)
11. Hardouin, L., Maia, C., Cottenceau, B., Santos-Mendes, R.: Max-plus linear observer: application to manufacturing systems. In: *10th International Workshop on Discrete Event Systems* (2010)
12. Jiroveanu, G., Schutter, B., Boel, R.: Taming Heterogeneity and Complexity of Embedded Control, chap. *The On-line Diagnosis of Time Petri Nets Based on Partial Orders*, p. 21. John Wiley & Sons (2013)
13. Le Corrond, E., Sahuguède, A., Pencolé, Y., Paya, C.: Localization of time shift failures in (max, +)-linear systems. In: *14th Workshop on Discrete Event Systems* (2018)
14. Lhommeau, M., Hardouin, L., Cottenceau, B., Jaulin, L.: Interval analysis and dioid: application to robust controller design for timed event graphs. *Automatica* **40**(11), 1923–1930 (2004)
15. Litvinov, G., Sobolevskii, A.: Idempotent interval analysis and optimization problems. *Reliable computing* **7**(5), 353–377 (2001)
16. Liu, B., Ghazel, M., Toguyéni, A.: Diagnosis of labeled time petri nets using time interval splitting. In: *19th World Congress of the International Federation of Automatic Control* (2014)
17. MaxPlus: Second order theory of min-linear systems and its application to discrete event systems. In: *Proceedings of the 30th IEEE Conference on Decision and Control* (1991)
18. Pencolé, Y., Subias, A.: Timed pattern diagnosis in timed workflows: A model checking approach. In: *14th Workshop on Discrete Event Systems* (2018)

19. Provan, G.: An algebraic approach for diagnosing discrete-time hybrid systems. In: M. Zanella, I. Pill, A. Cimatti (eds.) 28th International Workshop on Principles of Diagnosis (DX'17), *Kalpa Publications in Computing*, vol. 4, pp. 37–51. EasyChair (2018). DOI 10.29007/n87m. URL <https://easychair.org/publications/paper/9Nth>
20. Saddem, R., Philippot, A.: Causal temporal signature from diagnoser model for online diagnosis of discrete event systems. In: International Conference on Control, Decision and Information Technologies (2014)
21. Sahuguède, A., Le Corronc, E., Pencolé, Y.: Design of indicators for the detection of time shift failures in $(max, +)$ -linear systems. In: 20th World Congress of the International Federation of Automatic Control (2017)
22. Santos-Mendes, R., Cottenceau, B., Hardouin, L.: Adaptive feedback control for $(max, +)$ -linear systems. In: 10th IEEE Conference on Emerging Technologies and Factory Automation, vol. 2 (2005)
23. Schullerus, G., Supavatanakul, P., Krebs, V., Lunze, J.: Modelling and hierarchical diagnosis of timed discrete-event systems. *Mathematical and Computer Modelling of Dynamical Systems* **12**(6), 519–542 (2006)
24. Toguyéni, A., Craye, E., Gentina, J.C.: A method of temporal analysis to perform online diagnosis in the context of flexible manufacturing system. In: 16th Annual conference of IEEE Industrial Electronics Society, pp. 445–450. Pacific Grove, California, United States (1990)
25. Tripakis, S.: Fault diagnosis for timed automata. In: 7th International Symposium of Formal Techniques in Real-Time and Fault-Tolerant Systems (2002)
26. Wang, X., Mahulea, C., Silva, M.: Diagnosis of time petri nets using fault diagnosis graph. *IEEE Transactions on Automatic Control* **60**, 1–1 (2015). DOI 10.1109/TAC.2015.2405293
27. Yi-Liang Chen, Provan, G.: Modeling and diagnosis of timed discrete event systems—a factory automation example. In: Proceedings of the 1997 American Control Conference (Cat. No.97CH36041), vol. 1, pp. 31–36 vol.1 (1997)
28. Zad, S., Kwong, R., Wonham, W.: Topics in Control and its Applications, chap. Fault Diagnosis in Finite-State Automata and Timed Discrete-Event Systems. Springer (1999)
29. Zaytoon, J., Lafortune, S.: Overview of fault diagnosis methods for discrete event systems. *Annual Reviews in Control* **37**(2), 308 – 320 (2013)