Active Diagnosis Algorithm for the Localization of Time Failures in (Max,+)-Linear Systems

Ibis Velasquez, Euriell Le Corronc, Yannick Pencolé

Active Diagnosis Algorithm for the Localization of Time Failures in (Max,+)-Linear Systems

Ibis Velasquez * **Euriell Le Corronc** * **Yannick Pencolé** *

* *LAAS-CNRS, Université de Toulouse, UT3, CNRS, Toulouse, France (e-mail: {ibis.velasquez,euriell.le.corronc,yannick.pencole}@laas.fr)*

Abstract: This paper deals with active diagnosis in Timed Event Graphs represented as (Max,+)-Linear Systems. Based on the control theory of (max,+)-linear system, an offline algorithm is proposed that aims at better localizing the source of detected time failures within the supervised system. The proposed algorithm is divided into several steps that consists in successively synthesizing, testing and analyzing the injection of new input flows in the system to provide a better localization of the detected time failure.

Keywords: time failure, Timed Event Graph, (max,+)-linear systems, detection, localization, active diagnosis, optimal control, algorithm

1. INTRODUCTION

This paper introduces the active diagnosis problem in Timed Event Graphs. Timed event graphs (TEGs) are a subclass of timed Petri nets which can be represented in a (max,+) algebraic linear system (Baccelli et al. (1992)). They are characterized by the fact that each place has precisely one upstream and one downstream transition and all arcs have weight 1. TEGs are well suited to model timed discrete event systems with synchronization and delay phenomena (manufacturing/logistics/transportation systems, digital twins, communication networks, embedded microcontrollers, ...). This formalism has its own theory of control (Menguy et al. (2000); Cottenceau et al. (2001); Schafaschek et al. (2020)) and more recently some contributions on failure diagnosis have also been developped (Sahuguède et al. (2017); Le Corronc et al. (2018); Provan (2018); Le Corronc et al. (2021)). The objective of this paper is to combine previous control and diagnosis theories to define and solve the time failure active diagnosis problem over TEGs. Active diagnosis is the problem of setting up and applying a control policy in the system that ensures that the system's observable response is enough informative to better identify the source of any previously detected malfunctions. In discrete event systems, this problem has been introduced in Sampath et al. (1998). In Chanthery and Pencolé (2009), the control policy relies on a diagnosability pre-analyses to ensure that the active diagnosis result is definitive (the failure is definitely identified or will never be identified). Active diagnosis has also been investigated in switched systems (Van Gorp et al. (2013)) based on an event-based diagnoser and a testing procedure.

The proposed algorithm focuses on the active diagnosis of time failures, i.e. unexpected delays that propagate throughout the system that is only partially observable. The algorithm relies on the detection method previously developed in Sahuguède et al. (2017); Le Corronc et al. (2021)). Once a failure has been detected at operating time, an active diagnosis session is opened on the system (Chanthery and Pencolé (2009)) to ensure the algorithm has full control to actively perform the diagnosis (offline method) and identify the source of the failure within the system.

The proposed method is based on a structural analysis of the Timed Event Graph that is step by step. At each step, based on the result of the current structural analysis, the method then synthesizes new inputs to apply on the TEG (control step) that aims at refining the localization of the detected failure. By successively applying these control steps and looking at the successive indicators'responses, the localization results are more precise than the results that could be obtained by applying the methods proposed in Le Corronc et al. (2018, 2021). Indeed, these latter methods are based on one run of the system only and can produce a very large set of diagnostic candidates to be the source of the detected failure.

The paper is organized as follows. Section 2 deals with TEG and models of (max,+)-linear systems through the specific dioid $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ (see Baccelli et al. (1992); Max-Plus (1991) for details). Section 3 then presents the necessary background about the time failure detection and the control theory in TEG. The proposed active diagnosis algorithm is detailed in Section 4. Section 5 concludes and gives some perspectives.

2. TEG MODELS OF (MAX,+)-LINEAR SYSTEMS

2.1 Dioid and residuation theories

The dioid theory is the mathematical framework for modeling Timed Event Graphs (TEG) as (max,+)-linear systems. A *dioid* $\mathcal{D}$ is a set composed of two internal operations $\oplus$ and $\otimes$. The sum $\oplus$ is associative, commutative, idempotent (i.e. $\forall a \in \mathcal{D}, a \oplus a = a$) and admits ε as neutral

element. The product $\otimes$ is associative, distributive on the right and the left over the addition $\oplus$ and admits e as neutral element. Element ε is absorbing for $\otimes$. A dioid is said to be *complete* if it is closed for infinite sums and if $\otimes$ is distributive over infinite sums. In a complete diod $\mathcal{D}$, $x = a^*b$ is the least solution of $x = ax \oplus b$ where $a^* = \bigoplus_{i\geq 0} a^i$ is the *Kleene star* operator with $a^{i+1} = a \otimes a^i$ and $a^0 = e$. Due to the sum idempotency, an *order relation* is associated with $\mathcal{D}$ by the following equivalences: $\forall a, b \in \mathcal{D}$, $a \succeq b \Leftrightarrow a = a \oplus b$.

Example 1. 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 dioid where $\varepsilon = -\infty$ and $e = 0$.

The product of a dioid is not an inversible operator but a "pseudo-inverse" can be defined and will be used in this paper. It is called *residuation*. Let $f : \mathcal{D} \to \mathcal{C}$ be an isotone mapping, where $\mathcal{D}$ and $\mathcal{C}$ are complete dioids. Mapping f is said to be *residuated* if $\forall b \in \mathcal{C}$, the greatest element of subset $\{x \in \mathcal{D} | f(x) \preceq b\}$, 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 $f \circ f^\sharp \preceq \mathsf{Id}_{\mathcal{C}}$ and $f^\sharp \circ f \succeq \mathsf{Id}_{\mathcal{D}}$, where $\mathsf{Id}_{\mathcal{C}}$ and $\mathsf{Id}_{\mathcal{D}}$ are respectively the identity mappings on $\mathcal{C}$ and $\mathcal{D}$.

Example 2. Mapping $R_a : x \mapsto x \otimes a$ defined over a complete dioid $\mathcal{D}$ is residuated. Its residual is usually denoted $R_a^\sharp : x \mapsto x/a$ and called *right quotient*. Therefore, $b/a = \bigoplus\{x \mid x \otimes a \preceq b\}$ meaning that b/a is the greatest solution to inequality $x \otimes a \preceq b$.

2.2 Dioid and TEG modeling of (max,+)-linear systems

Timed Event Graphs (TEG, see Fig. 1) are a subclass of Timed Petri Nets in which each place has exactly one upstream and one downstream transition and for which arcs have weight one. In TEG, each place has a minimal time duration for the tokens (can be 0) and the earliest firing rule is applied. The tokens already present in places at $t = 0$ are considered present at the "origin of time". Consequently, the durations associated with the tokens are already consumed when the system starts. TEG can be decomposed into elementary structures: tandem, synchronization, parallelism, loops. Such structures are mixed to obtain more complex TEG as in Fig. 1. Let m_i and m_j be two nodes of a TEG (a node can be either a place or a transition), a path $m_i \rightsquigarrow m_j$ is a succession of nodes from m_i to m_j led by the directed arcs.

To model TEG as (max,+)-linear systems, the dioid $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is defined. First, the set of formal series with two commutative variables γ and δ, Boolean coefficients in $\{\varepsilon, e\}$ and exponents in $\mathbb{Z}$ constitutes the complete dioid $\mathbb{B}[\![\gamma, \delta]\!]$. Neutral elements are $\varepsilon(\gamma, \delta) = \bigoplus_{(n,t)\in\mathbb{Z}} \varepsilon\gamma^n\delta^t$ and $e(\gamma, \delta) = \gamma^0\delta^0$. A series $s \in \mathbb{B}[\![\gamma, \delta]\!]$ is written $s = \bigoplus_{(n,t)\in\mathbb{Z}} s(n,t)\gamma^n\delta^t$ where $s(n,t) = e$ or ε (respectively representing the presence or the absence of the monomial $\gamma^n\delta^t$). Now, the complete dioid $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is the quotient of $\mathbb{B}[\![\gamma, \delta]\!]$ modulo $\gamma^*(\delta^{-1})^*$ where $\forall a, b \in \mathbb{B}[\![\gamma, \delta]\!]$, $a = b \Leftrightarrow a\gamma^*(\delta^{-1})^* = b\gamma^*(\delta^{-1})^*$, meaning that an element of $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is an equivalence class $[a]_{\gamma^*(\delta^{-1})^*}$ (simply denoted a hereafter) gathering all the elements of $\mathbb{B}[\![\gamma, \delta]\!]$ equivalent modulo $\gamma^*(\delta^{-1})^*$. Neutral elements ε and e are identical to those of $\mathbb{B}[\![\gamma, \delta]\!]$.

Thus, a TEG is mathematically modeled by the following state representation in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$:

$$\begin{cases} x = Ax \oplus Bu \\ y = Cx \end{cases} \tag{1}$$

where $A \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]^{m\times m}$, $B \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]^{m\times p}$, $C \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]^{q\times m}$; m, p and q refer respectively to the *state vector* size of the system (x), the *input vector* size (u) and the *output vector* size (y). Entries of matrices A, B and C represent places of the TEG by the mean of a monomial $\gamma^n\delta^t \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ where n is the backward event shift between two transitions (number of initial tokens in the place) and t is their backward time shift (minimum duration time of the tokens in the place). When there is no connection between transitions, the entry is equal to ε. For each transition of a TEG, that is for each element of vectors x, u and y, one can write the cumulative trajectory [1] of its event occurrences over time (its dated firings) by a series of $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ in which a monomial $\gamma^n\delta^t$ is interpreted as follows: *its* $(n+1)^{th}$ *event occurrence (the numbering starts at* 0*) happens at earliest at time* t. Trajectories can describe a finite number of transition firings by a series ending by $\gamma^m\delta^{+\infty}$ meaning that the $(m+1)^{th}$ event occurrence never happens.

From Eq. (1), the relationship between input u and output y is computed through the Kleene star operator:

$$y = CA^*Bu = Hu. \tag{2}$$

$H \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]^{q\times p}$ is the transfer function of the TEG (lines are outputs and columns are inputs). An entry H_{ij} of H is the dynamic between the input u_j and the output y_i. If such an entry is a series with a Kleene star, there is at least one loop in the path between u_j and y_i. Systems that are fully characterized by Eq. (1) or (2) are commonly called $(max, +)$-linear systems. A C++ library called `minmaxgd` enables series of $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ to be handled (see Cottenceau et al. (2000)).

Example 3. The TEG of Fig. 1 has the following transfer function for which the Kleene star on monomials $\gamma^2\delta^1$ represents the presence of the loop on transition x_3.

$$H = CA^*B = \begin{pmatrix} \gamma^0\delta^2(\gamma^1\delta^1)^* & \gamma^0\delta^1(\gamma^1\delta^1)^* \\ \gamma^0\delta^2(\gamma^1\delta^1)^* & \gamma^0\delta^1(\gamma^2\delta^1)^* \end{pmatrix}. \tag{3}$$

For these inputs $u_1 = u_2 = \gamma^0\delta^0 \oplus \gamma^1\delta^1 \oplus \gamma^2\delta^{+\infty}$ where a first event is produced at time $t = 0$ ($\gamma^0\delta^0$), a second event at $t = 1$ ($\gamma^1\delta^1$), there is no third event ($\gamma^2\delta^{+\infty}$) ; the output is $y_1 = y_2 = \gamma^0\delta^2 \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^{+\infty}$. The first events of both y_1 and y_2 go out at $t = 2$, the second at $t = 3$, there is no third event.

3. DIAGNOSIS AND CONTROL THEORIES

This section presents the diagnosis and control theories required to solve the active diagnosis problem (Section 4).

3.1 Time failure detection in TEG

The proposed active diagnosis method aims at actively determining the source (a place) of *time failures* in a TEG, it relies on the time failure detection method from (Sahuguède et al. (2017); Le Corronc et al. (2021)).

[1] Equivalence $\gamma^*(\delta^{-1})^*$ makes series of $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ non-decreasing.

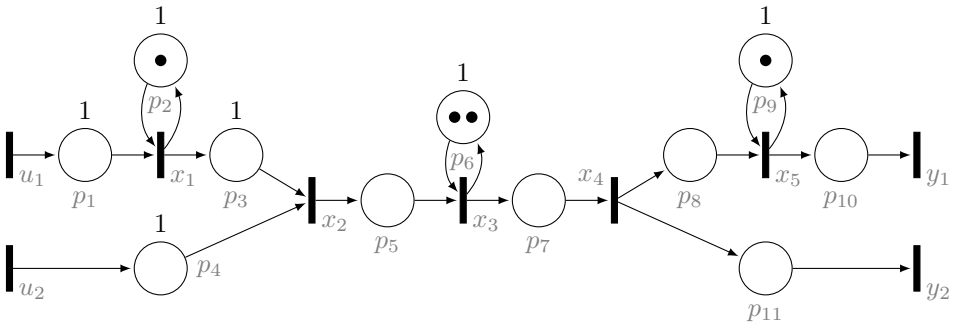

Fig. 1. TEG with two inputs u_1 and u_2, two outputs y_1 and y_2, one synchronization on x_2, one parallelism on x_4, three loops on x_1, x_3 and x_5 and some tandems such as the one between p_5 and p_7

Definition 1. (Time failure). A *time failure* held by a place p whose normal duration is d, is a relative delay $\theta > 0 \in \mathbb{Z}$ so that the real duration associated with p is $d + \theta$.

The TEG is assumed to be partially observable: only input/output series u and y are observable (i.e. known at operating time). Based on the transfer function H of the TEG and inputs u, the expected output $\tilde{y} = Hu$ can be computed. A time failure is said to be *detectable* if it leads at least to the production of a real output $y_i \in y$ that is different from the normal output $\tilde{y}_i \in \tilde{y}$. All along this paper, only detectable time failures are considered. The time failure detection is performed by an *indicator* that compares the real output y with the expected one $\tilde{y}$.

Such numerical comparison between series of $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ uses dater functions. The *dater function* of a series $s \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is the non-decreasing function $\mathcal{D}_s(n)$ from $\mathbb{Z}$ to $\overline{\mathbb{Z}}$ such that $s = \bigoplus_{n\in\mathbb{Z}} \gamma^n \delta^{\mathcal{D}_s(n)}$. Then, a theorem of MaxPlus (1991) establishes that the time shifts between two series a and b, *i.e.* the time difference between the n^{th} event occurrences of series a and b defined by $\mathcal{T}_{a,b}(n) = \mathcal{D}_b(n) - \mathcal{D}_a(n)$ for each $n \in \mathbb{Z}$, is bounded as follows:

$$\forall n \in \mathbb{Z}, \quad \mathcal{D}_{b \not\vartriangleleft a}(0) \leq \mathcal{T}_{a,b}(n) \leq -\mathcal{D}_{a \not\vartriangleleft b}(0).$$

So, the comparison between series a and b can be reduced to determining the bounds $\mathcal{D}_{b \not\vartriangleleft a}(0)$ and $-\mathcal{D}_{a \not\vartriangleleft b}(0)$ of $\mathcal{T}_{a,b}$. Thanks to the `minmaxgd` C++ library, these computations are made easily and do not need a specific algorithm.

Definition 2. (Indicator of time failures). Let H be the transfer function of a $(\max, +)$-linear system, u be its measurable input such that $Hu \neq \varepsilon$ and $y \neq \varepsilon$ be its measurable output. Indicator $I(u, y_i)$ for output y_i is the Boolean function:

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

with

$$\Delta(\tilde{y}_i, y_i) = [\mathcal{D}_{y_i \not\vartriangleleft \tilde{y}_i}(0)\ ;\ -\mathcal{D}_{\tilde{y}_i \not\vartriangleleft y_i}(0)]. \tag{4}$$

the *time interval* of y_i.

Example 4. Consider again the TEG of Fig. 1 with data of Example 3 (notation y of that previous example becomes $\tilde{y}$ in this one). A time failure $\theta = 1$ on p_2 occurs and produces the observed output:

$$y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} \gamma^0\delta^2 \oplus \gamma^1\delta^4 \oplus \gamma^2\delta^{+\infty} \\ \gamma^0\delta^2 \oplus \gamma^1\delta^4 \oplus \gamma^2\delta^{+\infty} \end{pmatrix}.$$

The minimal time shift between y_1 and $\tilde{y_1}$ is $\mathcal{D}_{y_1 \not\vartriangleleft \tilde{y_1}}(0) = 0$ and is found in $y_1 \not\vartriangleleft \tilde{y_1} = \gamma^0\delta^{\mathbf{0}} \oplus \gamma^1\delta^2 \oplus \gamma^2\delta^{+\infty}$. The maximal time shift is $-\mathcal{D}_{\tilde{y_1} \not\vartriangleleft y_1}(0) = 1$ and is found in $\tilde{y_1} \not\vartriangleleft y_1 = \gamma^0\delta^{-\mathbf{1}} \oplus \gamma^1\delta^2 \oplus \gamma^2\delta^{+\infty}$. Then, the time failure is detected since $I(u, y_1) = I(u, y_2) = \mathit{true}$ because $\Delta(y_1, \tilde{y_1}) = \Delta(y_2, \tilde{y_2}) = [0; 1]$.

3.2 Optimal control theory

Generally speaking, active diagnosis is the problem of controlling the system to improve the diagnosis quality. In this paper, the proposed method relies on the control theory of (max,+)-linear systems. It aims to delay the input flows to achieve pre-specified behavior and performances according to a criterion to be optimized (Cottenceau et al. (2001)). A generally used criterion is the just-in-time criterion that is the right amount of output events at the right time. In other words, this criterion is meant to satisfy the requirement on or before the date of the requirement. In TEG, it aims to perform the minimum firing number of input transitions and to delay these firings as much as possible.

Among existing controllers, the so-called *optimal control* for TEG is an open-loop strategy that consists in computing the largest input flow u_{opt} for which the obtained output flow $y_{opt} = Hu_{opt}$ is lower or equal to a known reference output y_r ($y_{opt} \preceq y_r$, Menguy et al. (2000)). With series of $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$, the "largest series" is the most delayed series and a lower series is faster. So, the use of this type of control reduces useless waiting times of tokens inside a TEG, that is particularly useful in TEG with synchronizations and loops. The input u_{opt} is optimal regarding the just-in-time criterion. Formally:

$$u_{opt} = H \backslash\!\!\backslash y_r.$$

Example 5. Back to the TEG of Fig. 1 with the following reference output:

$$y_r = \begin{pmatrix} \gamma^0\delta^{10} \oplus \gamma^2\delta^{11} \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^{10} \oplus \gamma^2\delta^{11} \oplus \gamma^3\delta^{+\infty} \end{pmatrix},$$

the optimal input is computed:

$$u_{opt} = H \backslash\!\!\backslash y_r = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} \gamma^0\delta^7 \oplus \gamma^1\delta^8 \oplus \gamma^2\delta^9 \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^8 \oplus \gamma^1\delta^9 \oplus \gamma^2\delta^{10} \oplus \gamma^3\delta^{+\infty} \end{pmatrix}$$

and its corresponding output $y_{opt} = Hu_{opt}$ is:

$$y_{opt} = \begin{pmatrix} y_{opt1} \\ y_{opt2} \end{pmatrix} = \begin{pmatrix} \gamma^0\delta^9 \oplus \gamma^1\delta^{10} \oplus \gamma^2\delta^{11} \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^9 \oplus \gamma^1\delta^{10} \oplus \gamma^2\delta^{11} \oplus \gamma^3\delta^{+\infty} \end{pmatrix}.$$

This output is faster than the reference output y_r ($y_{opt} \preceq y_r$). The first events of y_{opt} are produced at $t = 9$ ($\gamma^0\delta^9$) and respect the output y_r that requires a first event at or before $t = 10$. Moreover, u_1 starts earlier than u_2 to avoid the accumulation of tokens before the synchronization on x_2 because the path $u_2 \rightsquigarrow x_2$ is faster than $u_1 \rightsquigarrow x_2$.

4. OFFLINE ACTIVE DIAGNOSIS IN TEG

This section details the active diagnosis algorithm proposed in this paper. We suppose first that a time failure has been detected in the system at operating time by the indicators from Section 3.1. Then the system is stopped and an active diagnostic session starts which is controlled by the proposed algorithm.

4.1 Assumptions

Using the parcimony principle, the proposed algorithm assumes that the source of the detected failure is unique (i.e. it is characterized by a permanent delay θ on a unique place p). Let G be the TEG that models the supervised system, the active diagnosis algorithm also requires that:

- G is *empty*: any existing initial token must be in a place that is part of a loop, place that is in the preset of a synchronization transition that is not yet enabled,
- Input and output transitions are not involved in loops and are not synchronized transitions.

These assumptions lead to the following Properties 1–3 in the TEG G.

Property 1. As G is empty, no internal or output transition can be fired before the first event of any input transition.

Also, as a result of G's emptiness, the number of loops of a TEG is exactly the number of its places that hold tokens at initial time. Let A, B and C be the matrices of G (see Eq.(1)) and H be its transfer function.

Property 2. Matrices B and C contain only monomials with γ^0. In other words, there are only empty places (with no token) from any input transition u_i to any internal transition x_j, and from any internal transition x_r to any output transition y_s.

Property 3. Let $u_j \rightsquigarrow y_i$ be a path of G between an input u_j and an output y_i, then y_i cannot be fired before the occurrence of first event u_j. In other words, the first event of the series H_{ij} is a monomial with γ^0.

4.2 Algorithm

The algorithm is composed of several steps (see Fig. 2). Some of them (denoted Ana) perform analyses such as structural analyses of G (are there loops in G? any synchronizations? what are the paths between inputs and outputs?...). The others are control steps (optimal control or simple control, denoted as Ctrl), that consist in the synthesis of some input flows and their application to the system G in order to obtain a new set of indicator results that would improve the localization results.

There are three control steps which have different objectives:

- Ctrl1: simple optimal control as an initial test,
- Ctrl2: adaptive optimal control for a better localization in loops,
- Ctrl3: control for a better localization in path involving synchronizations;

and three analysis steps:

- Ana1: analysis of the indicators returned by Ctrl1,
- Ana2: analysis of the TEG loops based on the conclusion of Ana1,
- Ana3: analysis of the TEG the synchronizations based on either the results of Ctrl1, Ana1 or Ana2.

They are all combined in Fig. 2 where the conditions from one step to another are denoted by ♣*condX* and are explained in the following subsections.

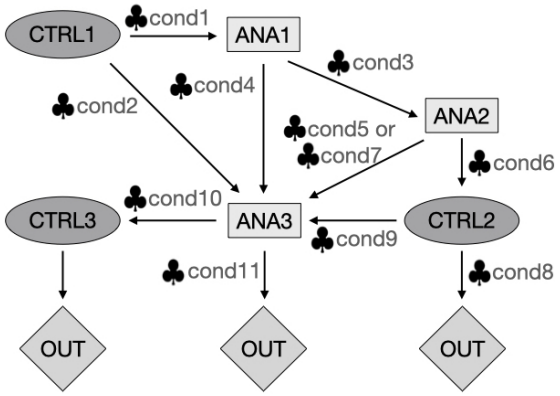

Fig. 2. Active Diagnosis Algorithm

About the complexity of the proposed algorithm: any analysis step is linear in the number of transitions/places and token numbers in the underlying TEG. The most complex part of the algorithm is in the computation of the residuations (indicators and computation of the optimal inputs) that is quadratic.

4.3 Optimal control for Ctrl1 *and* Ctrl2

In Ctrl1-2, an optimal control is performed that requires a reference output based on which the optimal input is computed. This reference output has to be the fastest output with the least number of events so that the time failure in G is detected by the indicators as quickly and as precisely as possible. This output is called the *minimal reference output*. While the time information of this minimal reference output is mathematically computed through H, the event information however depends on the number of tokens that are initially present in G. As G is empty (Property 1), the only tokens initially present in G are in loops in places that belong to the preset of a synchronised transition. Due to the semantics of a TEG, at time $t = 0$, such tokens are considered to be present in their respective place forever which means they already stayed in their place longer than the duration requested by the place. Therefore, a time failure on such a place cannot be detected as long as all the tokens initially present in the place are not used. The minimal reference output must then contain one event more than the largest number of tokens initially present in these places. This minimal number of events is denoted ν and depends on whether we are in step Ctrl1 or step Ctrl2.

Definition 3. (Minimal reference output). Let H be the transfer function of G and $E, Z \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]^q$ two column vectors where every $E_i = \gamma^0\delta^0$ and $Z_i = \gamma^\nu\delta^{+\infty}$ with ν computed according to the need of the step CTRL-X. The minimal reference output is $y_r \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]^q$ (it is a column vector) such that:

$$y_r = HE \oplus Z \tag{5}$$

where each series of y_r (each row) ends with $\gamma^\nu\delta^{+\infty}$.

Then, the computation of the optimal input is $u_{opt} = H \backslash y_r$ and its corresponding expected output is $\tilde{y} = y_{opt} = Hu_{opt}$ (see Subsection 3.2). The optimal input is then applied to the system $u = u_{opt}$ that operates them to produce the real output y and a new set of indicator results (see Definition 2). Indicators that return true are gathered in the set $\mathcal{I}_{true}$, their corresponding time intervals are in the set $\mathcal{TI}$.

4.4 CTRL1 - *Simple optimal control*

This optimal control CTRL1 is the first step of the active diagnosis algorithm. The procedure described in Subsection 4.3 is applied with the following definition of ν, that is the minimal number of events needed by the minimal reference output.

Definition 4. (Minimal number of events ν for CTRL1). Let N_l be the number of loops in G and $\mathcal{P}_t$ be the set of places that contain at least one token [2]. Let $(p_i, o_i) \in (\mathcal{P}_t \times \mathbb{Z} \setminus \{0\})$ be the ordered pair [3] of a place p_i that contains $o_i > 0$ tokens with $i = \{1, 2, \ldots, N_l\}$. The minimal number of events ν needed by CTRL1 is:

$$\nu = \begin{cases} 1 & \text{if } N_l = 0 \text{ (no loop in } G), \\ \max(o_i) + 1 & \text{otherwise.} \end{cases} \tag{6}$$

It is possible that, at this stage, all the indicators return false, meaning that this first optimal control does not reveal the time failure because it is hidden by some synchronizations. Another control step has to be performed to compute a more efficient input control.

Next:

- ◇ If $|\ \mathcal{I}_{true}\ | \geq 1$ (at least one indicator returns true), the corresponding time interval(s) is(are) analyzed by ANA1 ♣*cond1.*
- ◇ If $|\ \mathcal{I}_{true}\ | = 0$ (no indicator is true), the time failure is not revealed by this first optimal control. The next step is ANA3 ♣*cond2.*

Example 6. Consider the TEG of Fig. 1 with data of Examples 3 and 4 (transfer function H of Eq. (3), output $\tilde{y}$ such that $\tilde{y}_1 = \tilde{y}_2 = \gamma^0\delta^2 \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^{+\infty}$, a time failure $\theta = 1$ on p_2). When the active diagnosis session starts, CTRL1 is applied with $N_l = 3$ and $\mathcal{P}_t = \{p_2, p_6, p_9\}$ with $o_2 = 1$, $o_6 = 2$, $o_9 = 1$. Therefore, $\nu = \max(1, 2, 1) + 1 = 2 + 1 = 3$. Thus, the reference output of Eq.(5) is:

$$y_r = HE \oplus Z = \begin{pmatrix} \gamma^0\delta^2 \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^4 \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^2 \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^4 \oplus \gamma^3\delta^{+\infty} \end{pmatrix}.$$

with $E = (\gamma^0\delta^0 \quad \gamma^0\delta^0)^t$ and $Z = (\gamma^3\delta^{+\infty} \quad \gamma^3\delta^{+\infty})^t$. The optimal control is computed:

$$u_{opt} = H \backslash y_r = \begin{pmatrix} \gamma^0\delta^0 \oplus \gamma^1\delta^1 \oplus \gamma^2\delta^2 \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^1 \oplus \gamma^1\delta^2 \oplus \gamma^2\delta^3 \oplus \gamma^3\delta^{+\infty} \end{pmatrix}$$

and its corresponding optimal output is:

$$y_{opt} = Hu_{opt} = \begin{pmatrix} \gamma^0\delta^2 \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^4 \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^2 \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^4 \oplus \gamma^3\delta^{+\infty} \end{pmatrix}.$$

When the optimal control u_{opt} is applied to the system, one can observe this new output:

$$y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} \gamma^0\delta^2 \oplus \gamma^1\delta^4 \oplus \gamma^2\delta^6 \oplus \gamma^3\delta^{+\infty} \\ \gamma^0\delta^2 \oplus \gamma^1\delta^4 \oplus \gamma^2\delta^6 \oplus \gamma^3\delta^{+\infty} \end{pmatrix}.$$

Both indicators of y_1 and y_2 return true ($\mathcal{I}_{true} = \{I(u, y_1), I(u, y_2)\}$) with $\Delta(\tilde{y}_1, y_1) = \Delta(\tilde{y}_2, y_2) = [0; 2]$ so $\mathcal{TI} = \{[0; 2], [0; 2]\}$ and the next step is ANA1.

4.5 ANA1 - *Can the source of the failure be in a loop?*

ANA1 is performed after CTRL1 if there is at least an indicator that returns true (a time failure has been detected). The question is now: *can the source of failure be in a loop or not?* At this stage, it is possible to answer this question by analyzing the intervals $[a, b]$ from $\mathcal{TI}$. As detailed in Le Corronc et al. (2021), such an interval may be a degenerated interval $[b, b], b > 0$ stating that the associated measured output y is continuously delayed with b time units with respect to $\tilde{y}$ or it can be an interval $[a, b], a \geq 0, b > a$ stating that the minimal delay between both the occurrence of an event n in $\tilde{y}$ and y is a and the maximal one is b.

Proposition 1. If there is no non-degenerated interval $[0, b], b > 0$ in $\mathcal{TI}$, the source of the time failure cannot be in a loop.

Proof. Suppose the source to the unique time failure is in a loop l. Remember that in a TEG, the n tokens present at $t = 0$ in the place of the loop (see for instance place p_2 in Figure 1) are not impacted by any time shift, as they are considered to be there for already an infinite amount of time. Let $Y_l = \{y_1, \ldots, y_m\}$ be the outputs that are downstream from l. So the first n^{th} occurrences of event in any output $y \in Y_l$ are not impacted by any time shift. However, by definition of CTRL1, ν occurrences of event are expected in any $y \in Y_l$. As $\nu > n$ and the failure is detectable, there must exist at least an output $y \in Y_l$ such that the ν^{th} occurrence of its corresponding event is impacted by the time failure from the loop, so the interval $\Delta(\tilde{y}, y)$ of the corresponding indicator must look like $[0, b], b > 0$, hence the result.

In other words, a non-degenerated interval $[0, b], b > 0$ on an output y may indicate that the time failure is located inside a loop upstream.

Next:

- ◇ If the time failure is potentially in a loop, go to ANA2 to complete the analysis on loops ♣*cond3.*
- ◇ Otherwise go to the step ANA3 ♣*cond4.*

Example 7. On previous Example 6, $\mathcal{TI} = \{[0; 2], [0; 2]\}$, two non-degenerated intervals so the time failure may be in a loop (the one with place p_2 or the one with place p_6. In this example, the algorithm then performs step ANA2.

[2] Because of Property 1, $N_l = |\ \mathcal{P}_t\ |$.

[3] Notation (p, o) gathered all the pairs (p_i, o_i).

4.6 ANA2 - *Can we distinguish among the loops?*

Step ANA1 concludes that a time failure may be in a loop among a subset of *candidate loops* $\{cl_1, \ldots, cl_{N_l}\}$ of size N_l. If $N_l = 1$, there is only one loop in this subset, the analysis ANA2 is finished. If the time failure is in a loop, it is definitely in this one. Next, analysis ANA3 is performed to check whether the time failure may not be in a loop but involved in a synchronization.

If $N_l > 1$, there are several candidate loops: which one could hold the time failure? Let o_i denotes the initial number of tokens in the place of the loop cl_i. If there exists cl_i and cl_j such that $o_i \neq o_j$, it is possible to have better localization by using another adaptive optimal control.

Next:

- ◇ If there is only one suspected loop, go to ANA3 for further investigations ♣*cond5*.
- ◇ If there are several candidate loops and they do not contain the same number of tokens, go to CTRL2 ♣*cond6*.
- ◇ If there are several candidate loops and they all contain the same number of tokens o_l, go to ANA3 for further investigations ♣*cond7*.

Example 8. Since $N_l > 1$, there are several candidate loops, the one with p_2 and the other with p_6 and they do not contain the same number of tokens. The next step is then CTRL2.

4.7 CTRL2 - *Adaptive optimal control for loops*

This optimal control step CTRL2 is applied when these 3 conditions are true:

(i) a time failure is assumed to be in a loop according to ANA1,
(ii) there are several candidate loops $\{cl_1, \ldots, cl_{N_l}\}$ according to ANA2,
(iii) not all the candidate loops contain the same number of tokens according to ANA2.

Then, to know in which loop the time failure is, a solution is to drain step by step the tokens already present in G by applying control with an increasing number of events, *i.e.* control with an increasing ν as defined below, until a non-degenerated $[0, b], b > 0$ time interval is obtained meaning that the time failure is in a loop in which less than ν tokens evolve (see the proof of Proposition 1). Recall that o_i denotes the number of tokens involved in the loop cl_i. Let $\mathcal{O} = \{o_i, 1 \leq i \leq N_l\}$ and let $o^1 < o^2 < \ldots < o^{|\mathcal{O}|}$ be the numerical order of the elements of $\mathcal{O}$.

Definition 5. (Adaptive number of events ν for CTRL2). The adaptive number of events ν needed by CTRL 2 for each step $j = \{1, 2, \ldots, |\mathcal{O}|\}$ is:

$$\nu_j = o^j + 1. \tag{7}$$

The following procedure is then applied:

(1) Compute the optimal control of Subsection 4.3 with the adaptive number of events ν_j of Definition 5.
(2) If all the time intervals of $\mathcal{TI}$ are degenerated, repeat (1) with ν_{j+1}. Stops when at least one time interval of $\mathcal{TI}$ is non-degenerated.

Remark 1. To that point, this procedure must end with at least one non-degenerated time interval. In the worst case, it is indeed the one that was obtained at step ANA1.

Let ν_f denotes the last index computed by the previous algorithm and $o_f = \nu_f - 1$. It follows that only candidate loops cl_i such that $o_i = o_f$ remain candidates.

Next:

- ◇ Go to ANA3 for further investigations ♣*cond9*.

Example 9. At step ANA2, it is assumed that the time failure is in a loop, either the one with p_2 or the one with p_6. From Definition 5, $N_l = 2$ and $\mathcal{O} = \{1, 2\}$. Thus, $\nu_1 = o^1 + 1 = 1 + 1 = 2$ and $\nu_2 = o^2 + 1 = 2 + 1 = 3$. For $j = 1$ so $\nu_1 = 2$, it results that:

$$u_{opt} = \begin{pmatrix} \gamma^0\delta^0 \oplus \gamma^1\delta^1 \oplus \gamma^2\delta^{+\infty} \\ \gamma^0\delta^1 \oplus \gamma^1\delta^2 \oplus \gamma^2\delta^{+\infty} \end{pmatrix}.$$

and $\Delta(\tilde{y}_1, y_1) = \Delta(\tilde{y}_2, y_2) = [0; 1]$. It is already non degenerated intervals so the control procedure stops. The time failure is in a loop with $o_f = 1$ token. As a consequence, if the time failure is in a loop then it must be on place p_2. Next step is ANA3.

4.8 ANA3 - *Can the source of a failure be in the upstream of a synchronization?*

Step ANA3 is always the last step of the algorithm. While the previous steps focus on the presence of time failure in loops, this step is complementary and analyses the structure of G in order to determine whether the source of a time failure is in the upstream of a synchronization that would consequently have an impact on how the time failure propogates through this synchronization depending on the inputs. Based on the previous analyses, Step ANA3 determines the synchronizations in G that are relevant so that a new control procedure CTRL3 can be applied on them. Synchronizations between paths from several inputs to a single output are characterized in H by:

$$\exists i, r, s \in \mathbb{N} \text{ with } r \neq s \quad \text{s.t.} \quad H_{ir} \neq \varepsilon \text{ and } H_{is} \neq \varepsilon. \tag{8}$$

In other words, if for one row i of H (*i.e.* for one output), two columns r and s (*i.e.* two inputs) are different from ε, there exists a synchronization between two paths from two different inputs u_r and u_s leading to that output y_i ($u_r \rightsquigarrow y_i$ and $u_s \rightsquigarrow y_i$). If ANA3 is applied just after CTRL1, it means that the first optimal control has not yet detected the time failure while it is present by assumption, so in this case, every synchronization characterized by Eq. (8) is relevant. Let $\mathcal{Y}_{ctrl}$ be the set of outputs to be measured by CTRL3 then $\mathcal{Y}_{ctrl} = \{y_i | u_r \rightsquigarrow y_i \in G \wedge \text{Eq. (8) holds}\}$. Let $\mathcal{U}_{ctrl}$ be the set of inputs to be controlled, then $\mathcal{U}_{ctrl} = \{u_r | u_r \rightsquigarrow y_i \wedge y_i \in \mathcal{Y}_{ctrl} \wedge \text{Eq. (8) holds}\}$. If CTRL1 has detected the failure, the relevant outputs are then the ones with an indicator that is true, therefore $\mathcal{Y}_{ctrl} = \{y_i | u_r \rightsquigarrow y_i \in G \wedge \text{Eq. (8) holds}\} \cap \{y_i | I(u_{opt}, y_1) = true\}$ and $\mathcal{U}_{ctrl}$ is similar as in the previous case.

Next:

- ◇ If $\mathcal{U}_{ctrl} \neq \varnothing$, go to CTRL3 ♣*cond10*.
- ◇ If $\mathcal{U}_{ctrl} = \varnothing$, go OUT ♣*cond11*.

Example 10. In the current example, a time failure has been detected by CTRL1. Both indicators are true. So

all synchronizations x_1, x_2, x_3, x_4, x_5 are relevant. $\mathcal{Y}_{ctrl} = \{y_1, y_2\}$ and $\mathcal{U}_{ctrl} = \{u_1, u_2\}$.

4.9 CTRL *3 - Control for synchronizations*

This control step deals with synchronizations in G between inputs of $\mathcal{U}_{ctrl} = \{u^1, \ldots, u^{N_c}\}$ and outputs of $\mathcal{Y}_{ctrl}$ selected by ANA3 but is not an optimal control. An *ad hoc* control has to be built to find in which path among these synchronizations the time failure is. The principle of the control procedure is to check whether the effect of the time failure in G is impacted (masked or not) by a synchronization in a path from $\mathcal{U}_{ctrl}$ to $\mathcal{Y}_{ctrl}$. To do so, N_c independant controls are setup, each control k with $1 \leq k \leq N_c$ consists in applying to the input u^k a control c_t that is slow ($u^k = c_t$) while for all the other inputs $u^j \neq u^k$, the applied control c_0 is the fastest ($u^j = c_0$). By doing so for every input u^k, it is possible to determine the different effects of the unique time failure in G on the relevant synchronizations in G to better localize the time failure.

For each control k, input trajectories c_t and c_0 are defined as follows:

$$\begin{cases} c_t = \gamma^0 \delta^{t^k_{max}} \oplus \gamma^\eta \delta^{+\infty}, \\ c_0 = \gamma^0 \delta^0 \oplus \gamma^\eta \delta^{+\infty}. \end{cases}$$

with t^k_{max} and η defined below.

Definition 6. (Number of events η for CTRL3). The number of events η needed for CTRL3 in c_t and c_0 is:

$$\eta = \begin{cases} \nu & \text{if previous steps are CTRL1-ANA3,} \\ \nu & \text{if previous steps are ANA1-ANA3,} \\ o_l + 1 & \text{if previous steps are ANA2-ANA3,} \\ o_f + 1 & \text{if previous steps are CTRL2-ANA3.} \end{cases}$$

where ν comes from Definition 4, o_l from ANA2 and o_f from CTRL2.

Definition 7. (Time t^k_{max} for each step k of CTRL3). For an input $u^k \in \mathcal{U}_{ctrl}$, let $t_i^{k'}$ be the traversal time imposed by the entry $H_{ik'}$ of the transfer function for η tokens to travel along the path $u^{k'} \rightsquigarrow y_i$, with $k' \neq k$, $u^{k'} \in \mathcal{U}_{ctrl}$ and $y_i \in \mathcal{Y}_{ctrl}$. The time t^k_{max} needed at each step k of CTRL3 in c_t is:

$$t^k_{max} = \max\{t_i^{k'}\}.$$

So, that time t^k_{max} is not built from durations downstream u^k but from all the other inputs $u^{k'}$.

When applying control k, η tokens are sent at $t = 0$ through the inputs $u^j \neq u^k$. These tokens will be stopped by synchronizations. Then, η tokens are sent through u^k at $t = t^k_{max}$. If the time failure is on a path leaving from u^k, it generates as much delay as possible on the affected outputs in $\mathcal{Y}_{ctrl}$.

Now, after applying all the controls, consider the inputs u^k from $\mathcal{U}_{ctrl}$ so that there exists at least an output $y_k \in \mathcal{Y}_{ctrl}$ associated with an indicator such that $\Delta(\tilde{y}_k, y_k) = [a, b]$ with b being maximal. Such an input u^k is then suspected to be in the upstream of the time failure and then blelongs to $\mathcal{U}_{upstream}$. Then the time failure can be better localized by a structure analysis to be in the downtream of every $u^k \in \mathcal{U}_{upstream}$ and in the upstream of every impacted synchronization.

Example 11. According to Example 10, $\mathcal{U}_{ctrl} = \{u_1, u_2\}$ and $\mathcal{Y}_{ctrl} = \{y_1, y_2\}$. Thus, $N_c =| \mathcal{U}_{ctrl} |= 2$, meaning that two steps of CTRL3 are achieved ($k = 1$ and $k = 2$) and $\mathcal{U}_{ctrl} = \{u^1, u^2\}$ with $u_1 = u^1$, $u_2 = u^2$ according to the new notation. The number of events is $\eta = o_f + 1 = 1 + 1 = 2$ (previous steps are CTRL2-ANA3). For u^1, $t_1^2 = 2$ since the dynamic between u_2 and y_1 is $H_{12} = \gamma^0\delta^1(\gamma^1\delta^1)^* = \gamma^0\delta^1 \oplus \gamma^1\delta^2 \oplus \ldots$ so it takes 2 time units for $\eta = 2$ tokens to travel along the path $u^2 \rightsquigarrow y_1$. The same for the path $u^2 \rightsquigarrow y_2$ with $H_{22} = \gamma^0\delta^1(\gamma^2\delta^1)^* = \gamma^0\delta^1 \oplus \gamma^2\delta^2 \oplus \ldots$ (again, it takes 2 time units for $\eta = 2$ tokens to travel along the path $u^2 \rightsquigarrow y_2$) so $t_2^2 = 2$. Thus, $t^1_{max} = max\{t_1^2, t_2^2\} = max\{2, 2\} = 2$. Then, for $k = 1$, $u^1 = c_t = \gamma^0\delta^{t^1_{max}} \oplus \gamma^\eta\delta^{+\infty} = \gamma^0\delta^2 \oplus \gamma^2\delta^{+\infty}$ and $u^2 = c_0 = \gamma^0\delta^0 \oplus \gamma^\eta\delta^{+\infty} = \gamma^0\delta^0 \oplus \gamma^2\delta^{+\infty}$. The obtained time intervals are $\Delta(\tilde{y}_1, y_1) = \Delta(\tilde{y}_2, y_2) = [0; 1]$. Same reasoning for u^2 where $t_1^1 = 3$ and $t_2^1 = 3$ (see H_{11} and H_{21}). Thus $t^2_{max} = max\{t_1^1, t_2^1\} = max\{3, 3\} = 3$. Then, for $k = 2$, $u^2 = c_t = \gamma^0\delta^{t^2_{max}} \oplus \gamma^\eta\delta^{+\infty} = \gamma^0\delta^3 \oplus \gamma^2\delta^{+\infty}$ and $u^1 = c_0 = \gamma^0\delta^0 \oplus \gamma^\eta\delta^{+\infty} = \gamma^0\delta^0 \oplus \gamma^2\delta^{+\infty}$. The obtained time intervals are $\Delta(\tilde{y}_1, y_1) = \Delta(\tilde{y}_2, y_2) = [0; 0]$. The maximal upper bound of the time intervals $\mathcal{TI}$ is obtained when $k = 1$ so $\mathcal{U}_{upstream} = \{u_1\}$. The only synchronization between u_1 and another input of $\mathcal{U}_{ctrl}$ leading to an output of $\mathcal{Y}_{ctrl}$ is on x_2. Thus, the time failure is located between u_1 and x_2, that is in one of the places p_1, p_2, p_3.

4.10 OUT - Final localization of the time failure

This final step gather the conclusions from all the previous steps and provides a final conclusion:

(i) CTRL2: may be in the only loop with o_f tokens.
(ii) ANA3: may be in the only loop of G or in a loop with o_l or o_f tokens.
(iii) CTRL3: must be in a path between $\mathcal{U}_{upstream}$ and some synchronizations.

Example 12. At the end of the algorithm, we know that if the time failure is in a loop, it must be the one with the place p_2 (so the time failure should be in p_2). This is also confirmed by the fact that the result of CTRL3 states that time failure is localized in a place within $\{p_1, p_2, p_3\}$.

5. CONCLUSION

This paper introduces the problem of active diagnosis for the localization of time failures in (max,+)-linear systems modelled by Timed Event Graphs and proposes a first method to solve the problem. It is executed offline after an online time failure is detected and offers to localize it through several controls such as optimal control. This seminal work leads to several perspectives. First, the method does not address the estimation of the time failure value but only the localization of its source. Second, the method also shows that it may sometimes be impossible to disambiguate between two possible sources due to a lack of observability of the system. This leads to the question of defining active diagnosability in TEGs, as it is defined for discrete event systems in Chanthery and Pencolé (2009), and developping a method that checks whether the underlying TEG is actively diagnosable (structural analysis methods, model-checking).

REFERENCES

Baccelli, F., Cohen, G., Olsder, G.J., and Quadrat, J.-P. (1992). *Synchronization and linearity: an algebra for discrete event systems*. Wiley and sons.

Chanthery, E. and Pencolé, Y. (2009). Monitoring and active diagnosis for discrete-event systems. In *7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*.

Cottenceau, B., Hardouin, L., Boimond, J.-L., and Ferrier, J.-L. (2001). Model reference control for timed event graphs in dioids. *Automatica*, 37, 1451–1458.

Cottenceau, B., Lhommeau, M., Hardouin, L., and Boimond, J.-L. (2000). Data processing tool for calculation in dioid. In *5th International Workshop on Discrete Event Systems*. `http://www.istia.univ-angers.fr/ hardouin/outils.html`.

Le Corronc, E., Pencolé, Y., Sahuguède, A., and Paya, C. (2021). Failure detection and localization for timed event graphs in (max,+)-algebra. *Journal of Discrete Event Dynamic Systems*, 31, 513–552.

Le Corronc, E., Sahuguède, A., Pencolé, Y., and Paya, C. (2018). Localization of time shift failures in (max,+)-linear systems. In *14th Workshop on Discrete Event Systems*.

MaxPlus (1991). Second order theory of min-linear systems and its application to discrete event systems. In *30th IEEE Conference on Decision and Control*.

Menguy, E., Boimond, J.-L., Hardouin, L., and Ferrier, J.-L. (2000). Just-in-time control of timed event graphs: update of reference input, presence of uncontrollable input. *IEEE Transactions on Automatic Control*, 45(11), 2155–2159.

Provan, G. (2018). An algebraic approach for diagnosing discrete-time hybrid systems. In *28th International Workshop on Principles of Diagnosis*.

Sahuguède, A., Le Corronc, E., and Pencolé, Y. (2017). Design of indicators for the detection of time shift failures in (max, +)-linear systems. In *20th IFAC World Congress*.

Sampath, M., Lafortune, S., and Teneketzis, D. (1998). Active diagnosis of discrete-event systems. *IEEE Transactions on Automatic Control*, 43(7), 908–929.

Schafaschek, G., Hardouin, L., and Raisch, J. (2020). Optimal control of timed event graphs with resource sharing and output-reference update. *Automatisierungstechnik*, 68(7), 512–528.

Van Gorp, J., Giua, A., Defoort, M., and Djemai, M. (2013). Active diagnosis for a class of switched systems. In *52nd IEEE Conference on Decision and Control*.