Fault diagnosis in labelled Petri nets: A Fourier-Motzkin based approach

Ahmed Al-Ajeli a, David Parker b,

aCollege of Information Technology, University of Babylon, Babylon, Iraq
bSchool of Computer Science, University of Birmingham, Birmingham, UK

Abstract

We propose techniques for fault diagnosis in discrete-event systems modelled by labelled Petri nets, where fault events are modelled as unobservable transitions. The proposed approach combines an offline and an online algorithm. The offline algorithm constructs a diagnoser in the form of sets of inequalities that capture the legal, normal and faulty behaviour. To implement the offline algorithm, we adopt the Fourier-Motzkin method for elimination of variables from these sets of inequalities. Upon observing an event, the diagnoser is used to determine whether a fault occurred or might have occurred. The occurrence of a fault can be verified by checking the observed sequence against the sets of inequalities. This approach has the advantage that the tradeoff between the size of the diagnoser and the time for computing the diagnosis is achieved. In addition, fault diagnosis in both bounded and unbounded Petri nets can be addressed.

Key words: Discrete-event systems; Petri nets; Fault diagnosis; Fourier-Motzkin elimination.

1 Introduction

The safety and reliability of large complex systems play an important role in the availability of the services provided by them. Unfortunately, fault occurrences in such systems are usually unavoidable. Fault diagnosis addresses the problem of detecting and isolating these fault occurrences. Thus, developing automatic approaches to obtain accurate and timely diagnosis decisions in such systems enhances their safety and reliability. It is well known that the problem of fault diagnosis in partially-observed discrete-event systems (DES) is a complex problem; it has been studied by many researchers in order to develop methods in which the time and space complexity are balanced.

The traditional approach to solving this problem is by assuming that there is a model capturing the behaviour of the system to be diagnosed (also called the plant). Two formalisms are usually used in the literature: automata and Petri nets (Basile et al. (2008); Cabasino et al. (2010); Dotoli et al. (2009); Sampath et al. (1995)). In this formalism, faults are modelled as unobservable events. The problem of fault diagnosis under partial observation was first investigated by Sampath et al. (1995). The authors modelled the system behaviour as a regular language captured by an automaton and the solution starts by creating, from this model, an automaton called a diagnoser in which all events are observable. One of the limitations of this approach, however, is the inability to handle infinite systems (i.e., unbounded state spaces). Petri net models provide more attractive graphical and mathematical features which can be used for the purpose of dealing with both finite and infinite systems. An extension to the idea introduced in the automata context has been proposed for Petri nets (Cabasino et al. (2010); Jiroveanu et al. (2008); Zhu et al. (2018)). The aim was to reduce the computational cost by only enumerating a subset of the reachable markings in the system being diagnosed.

A different idea has been proposed in Basile et al. (2009); Dotoli et al. (2009), where they use equations to address the diagnosis problem, rather than representing the diagnoser as an automaton. More specifically, the fault diagnosis problem has been reduced to an integer linear programming (ILP) problem, which is solved online every time an event is observed. Using this idea, the space complexity is reduced at the cost of the time complexity, which could be exponential. For a review of approaches for fault diagnosis in DES, we refer the reader to Basile (2014); Cabasino et al. (2012); Zaytoon and Lafortune (2013).

The above contributions have been demonstrated in the context of Petri nets where no two transitions in the model of the system share the same label. Extensions to the work of Cabasino et al. (2011) and Fantii et al. (2013) have been...
reported in Cabasino et al. (2010), Dotoli et al. (2009) and Wang et al. (2020) to the cases of labelled Petri nets (LPN) in which there is no restriction on having unique labels associated with transitions. These transitions can be simultaneously enabled (indistinguishable transitions), but only one of them can fire. In addition, Basile et al. proposed an approach for both diagnosability and fault detection in labelled Petri nets exploiting the ILP approach (Basile et al. (2012)). Recently, a diagnostic technique using an online count vector estimation was designed (Chouchane et al. (2020); Zhu et al. (2020)). These techniques are based on solving a fewer number of LP problems for an observed sequence of events.

Alternatively, a new approach adopting the idea of variable elimination from a set of inequalities has been developed for fault diagnosis in Petri nets (Al-Ajeli and Bordbar (2016); Al-Ajeli and Parker (2018)). The integer Fourier-Motzkin elimination method (IFME) has been used for the elimination (Pugh (1991); Williams (1976)). IFME is an extension of the Fourier-Motzkin elimination (FME) method used for inequalities in real variables (Comini et al. (2014); Duffin (1974); Kohler (1967)).

In this paper, we further extend the previous work based on the IFME method to the case of labelled Petri nets under the assumption that observable transitions might be indistinguishable. The proposed solution is in two parts: offline and online. The diagnoser is constructed offline as sets of inequalities. During the online step, a sequence of observed events (labels) is obtained and verified against the sets of inequalities constructed in the offline step to make the diagnosis decisions. It is worth mentioning that the present approach does not use the IFME method for solving an ILP problem, neither online nor offline. Instead, the method is used for the purpose of projecting the space described by a set of inequalities by eliminating variables.

This paper is structured as follows. In Section 2, a general background of Petri nets and the IFME method is provided. Section 3 presents a description of the fault diagnosis problem in DES. The details of the proposed approach and a proof of correctness for this approach on the fault diagnosis problem are covered in Section 4. Conclusions and future directions are discussed in Section 5.

2 Background

2.1 Petri nets

A Petri net (Murata (1989)) is defined as a four tuple $\mathcal{N} = (P, T, Pre, Post)$, where $P$ and $T$ are non-empty finite sets of places and transitions, respectively; $Pre : P \times T \rightarrow N$ and $Post : P \times T \rightarrow N$ are the weights of the arcs from places to transitions and from transitions to places. We use $m = |P|$ and $n = |T|$ for the number of places and transitions. For a given transition $t \in T$, an input (resp. output) place of $t$ is a place $p$ such that $Pre(p,t)$ (resp. $Post(p,t)$) is positive. $A = Post - Pre$ is the incidence matrix of a net.

A state of a Petri net, known as a marking, is represented as $M : P \rightarrow N$ capturing the number of tokens in each place. We sometimes represent a marking as an $m \times 1$ matrix of non-negative integers. A transition $t$ is enabled at a marking $M$ if $M(p) \geq Pre(p,t)$ for each input place $p$ of $t$. An enabled transition can fire, resulting in a new marking $M'$, denoted by $M \rightarrow M'$. We can find the reachable marking $M'$ by $M' = M + Au$, where $u$ is the n-dimensional firing vector of the transition $t$. A sequence of transitions $\sigma = t_1, \ldots, t_l$ of $T$ is called enabled at a marking $M$ if there are markings $M_1, \ldots, M_l$ so that $M \rightarrow M_1 \rightarrow M_2 \rightarrow \cdots \rightarrow M_l \rightarrow M$. In this case, we write $M \sigma \rightarrow M_l$ and refer to $M_l$ as a marking reachable from $M$ and $\sigma$ is the firing sequence. We write $R(\mathcal{N}, M)$ for the set of all markings reachable from $M$. The initial marking of the system is represented by an initial marking $M_0$. We will write $(\mathcal{N}, M_0)$ for a Petri net with its initial marking $M_0$.

Suppose that we have a sequence $\sigma$ of $(\mathcal{N}, M_0)$, then the Parikh vector $\theta : T \rightarrow \mathbb{N}^n$ is a map which assigns to every sequence $\sigma$ a vector $\theta(\sigma)$ in which each element represents the number of firings of each transition in $\sigma$. In other words, for $\theta(\sigma) : T \rightarrow \mathbb{N}$, $\theta(t)(i)$ is the number of occurrence of $t \in T$ within the sequence $\sigma$. Sometimes, we also write $\theta(t)$ to represent the number of the occurrences of $t$ in $\sigma$.

The set of sequences of transitions resulting in reachable markings is called the language of the Petri net and is denoted by $L(\mathcal{N}, M_0)$, i.e., $L(\mathcal{N}, M_0) = \{ \sigma \mid \exists M M_0 \sigma \rightarrow M \}$. Suppose that a destination marking $M$ is reachable from $M_0$ in a Petri net $\mathcal{N}$ through a sequence $\sigma$, we can then find $M$ using the following state equation:

$$M = M_0 + AX \geq \hat{0}$$

(1)

where $A$ is the incidence matrix of $\mathcal{N}$, and $x \in \mathbb{N}^n$ is an $n$-dimensional column vector with $x = (x_1, \ldots, x_n)$ and $x_i = \theta(t_i, \sigma)$ for $t_i \in T$. Then, for any sequence $\sigma \in L(\mathcal{N}, M_0)$, there exists $x = \theta(\sigma)$ satisfying (1). The converse is not always true. In some cases, e.g., acyclic Petri nets, the converse holds too.

Definition 1. (Tsujii and Murata (1993)) Let $v = (a_1, \ldots, a_\nu)$ be a solution of the state equation for a Petri net $(\mathcal{N}, M_0)$ with a destination marking $M$. Then, the firing count subnet with respect to $v$ is the subnet $\mathcal{N}_v$ where each transition $t_i$ in $\mathcal{N}_v$ is such that $a_i > 0$ together with its input and output places and its connecting arcs. $M_0v$ and $M_v$ denote the restrictions of $M_0$ and $M$ to places in $\mathcal{N}_v$.

Lemma 1. (Al-Ajeli and Parker (2018)) Suppose that $v$ is an $n \times 1$ column vector and $M$ is a reachable marking in a Petri net $\mathcal{N}$ such that $M = M + Av \geq \hat{0}$. Considering that $\mathcal{N}_v$ (see Definition 1) is cycle-free, then there exists a sequence $\sigma \in L(\mathcal{N}_v, M_0)$ such that $M_v \sigma \rightarrow M_v$ and $\theta(\sigma) = v$, where $M_v$ and $M_v'$ are restrictions of $M$ and $M'$ to places of $\mathcal{N}_v$. In addition, $\sigma$ can fire under $M$ resulting in $M'$ such that $M \sigma \rightarrow M'$.

Now, suppose that we have a Petri net $(\mathcal{N}, M_0)$, then the association of a label $e \in \Sigma$, where $\Sigma$ represents a set of labels (alphabet), to transitions in $\mathcal{N}$ is called a labelling function. This function is defined as $\lambda : T \rightarrow \Sigma \cup \{\epsilon\}$, i.e., $\lambda(t) = e$ or $\lambda(t) = \epsilon$ for $t \in T$. Also, this labelling function can be extended to the Kleene closure of $\Sigma$ by $\lambda : T^* \rightarrow \Sigma^*$, where for each sequence of transitions $\sigma$ and transition $t$, $\lambda(\sigma) = \lambda(\sigma)\lambda(t)$. A labelled Petri net is defined as a four tuple $(\mathcal{N}, M_0, \Sigma, \lambda)$ in which we associate to each label $e \in \Sigma$
a set of transitions \( \tau(e) \).

\[
\tau(e) = \{ t \mid t \in T, e = \lambda(i) \} \tag{2}
\]

### 2.2 Integer Fourier-Motzkin elimination method

The elimination of a variable from a set of inequalities \( I := Ax \leq b \), where \( A \in \mathbb{R}^{m \times n} \), \( b \in \mathbb{R}^m \) and \( x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n \) can be achieved by Fourier-Motzkin elimination (FME) method (Dantzig (1972); Duffin (1974)). The variables are eliminated one by one as explained as follows. It is sufficient to process the elimination of one variable, as the same procedure can be repeatedly applied to eliminate the required number of variables. Also, for the sake of simplicity, all entries in the last column of \( I \) are assumed to be 0, +1 or -1. Assuming that \( x_n \) is to be eliminated, \( I \) can be rewritten as shown in (3):

\[
P^i : \quad a_i^j x_j \leq b_i, \quad i = 1, \ldots, m_1
\]

\[
I^- : \quad a_i^j x_n - x_j \leq b_i, \quad j = m_1 + 1, \ldots, m_2
\]

\[
I^+ : \quad a_i^j x_n + x_j \leq b_i, \quad k = m_2 + 1, \ldots, m
\]

where \( x' = \{x_1, x_2, \ldots, x_{n-1}\} \), i.e. the same set of variables without \( x_n \). Also \( P^i, I^- \) and \( I^+ \) are sets of inequalities in \( I \) which have zero, negative and positive coefficients of \( x_n \). If \( I^+ \) is empty, all inequalities in \( I^- \) can simply be deleted. Likewise, if \( I^- \) is empty, then all inequalities in \( I^+ \) can be discarded. Assume that \( l = \max (a_i^j - b_j, j = m_1 + 1, \ldots, m_2) \) and \( u = \min (b_k - a_k^j, k = m_2 + 1, \ldots, m) \). Since the last two lines of (3) are equivalent to \( I \leq x_n \leq u \), then the variable \( x_n \) can be eliminated. This yields the reduced set \( R \) in (4) with no \( x_n \) as an equivalent to (3):

\[
a_i^j x_j \leq b_i, \quad i = 1, \ldots, m_1
\]

\[
a_i^j x_j - b_j \leq b_k - a_k^j x_j, \quad j = m_1 + 1, \ldots, m_2.
\]

\[
k = m_2 + 1, \ldots, m
\]

Theorem 1. (Duffin (1974)) Assume that the variables \( x_{k+1}, \ldots, x_n \) have been eliminated in order by using the FME method described above from a set of linear inequalities \( I \). This results in the reduced set \( R \). Then \( a_1, \ldots, a_n \) is a solution of \( R \) iff there exist values \( a_{k+1}, \ldots, a_n \) such that \( a_1, \ldots, a_k, a_{k+1}, \ldots, a_n \) is a solution of \( I \).

This theorem represents an important result for the purpose of fault diagnosis, as will be clear in the following sections. An extension of this result to a set of inequalities having integer-valued variables has been reported in Pugh (1991); Williams (1976). This extension, named Integer FME (IFME), is to ensure that for any integer solution in \( R \), there exists an integer solution in \( I \). In this paper, we have chosen the method presented in Pugh (1991), which better meets our need as it is somewhat simpler and more efficient.

### 3 Problem statement

In this section, a description of the problem of fault diagnosis in DES modelled by labelled Petri nets is given based on the formulation adopted by Cabasino et al. (2011) and Fanti et al. (2013). Consider a labelled Petri net \((\mathcal{N}, M_0, \Sigma, \lambda)\), as defined in Section 2.1. Suppose that the set of transitions \( T \) in \( \mathcal{N} \) is partitioned into two sets: observable transitions \( T_o \) and unobservable transitions \( T_u \). We further assume that faults are unobservable transitions, i.e. \( T_f \subseteq T_u \), in which \( T_f \) is the set of transitions which are modelling occurrences of faults. The set \( T_o \) may have other transitions which model no fault, i.e. they model normal events.

Consider also the projection function \( \pi : T \to T_o \cup \{ e \} \) that maps unobservable transitions to the empty string \( e \), i.e. \( \pi(t) = e \) for \( t \in T_o \), while \( \pi(t) = t \) for \( t \in T_o \). The projection function \( \pi \) can be extended to the Kleene-closure of \( T \) by \( \pi : T^* \to (T_o \cup \{ e \})^* \), where for each sequence of transitions \( \sigma \in T^* \) and each transition \( t, \pi(\sigma t) = \pi(\sigma) \pi(t) \).

We assume \( \pi(e) = e \) and that \( \pi(e t) = \pi(t e) = e \) for each \( t \in T_o \). Moreover, the inverse projection function is defined as \( \pi^{-1} : T_o^* \to \bigcup_{\sigma \in \Sigma^*} \{ t \in T ; \forall \sigma \in \Sigma^*, t \to \sigma \} \). A legal sequence \( s \in T_o^* \) is such that \( \pi^{-1}(s) \neq \emptyset \).

Let \( \omega \in \Sigma^* \) denote an observed sequence of events (labels), where \( \omega = \lambda(s) \) and \( s = \pi(\sigma) \) for a given sequence \( \sigma \in T^* \). To simplify the presentation of this paper, we only consider one type of fault \( T_f = \{ t_1, t_2, \ldots, t_k \} \); the extension to multiple types is straightforward. In particular, to create a set of inequalities for a given fault type, the transitions representing faults in the other fault types are considered as normal unobservable transitions. Since it is not required to uniquely identify occurrences of every fault of \( T_f \), a firing of any transition \( t \in T_f \) implies that a fault has occurred. We suppose that the labels captured by \( \omega \) are the only information we receive when a sequence of observable transitions fires. A diagnoser (as formally defined in the following sections) uses such information to identify if a fault has occurred or may have occurred.

In this paper, the problem of fault diagnosis is addressed with the assumption that different transitions could share the same label, taking into account that these transitions might be simultaneously enabled.

### 4 The IFME method for fault diagnosis in LPN

The main results obtained in this paper are covered in this section. In order to formulate the IFME-based solution, we first introduce some of necessary definitions and notation.

#### 4.1 Definitions and notations

The IFME-based approach for fault diagnosis essentially relies on using inequalities. The enabling conditions of Petri nets can be formed as a set of inequalities. Besides, the presence and absence of faults can be expressed in the form of inequalities. Suppose that transition \( t_i \in T \) is a fault transition. Then \( t_i \) does not appear in a firing sequence \( \sigma \) if and only if \( c_{t_i} := \#(t_i, \sigma) = 0 \) holds. Also, the occurrence of \( t_i \) in \( \sigma \) can be trivially written as \( c_e := \#(t_i, \sigma) > 0 \), i.e., the negation of \( c \). In addition, we can represent a set of faults as inequalities by extending the formulation above. Recall that \( T_f = \{ t_1, t_2, \ldots, t_k \} \) is a fault type; we associate two inequalities \( c_e := \sum_{t_i \in T_f} \#(t_i, \sigma) > 0 \) and \( c := \sum_{t_i \in T_f} \#(t_i, \sigma) \leq 0 \). Then, no fault of \( T_f \) appearing in \( \sigma \) implies that \( c \) holds. In contrast, a fault of \( T_f \) appears in \( \sigma \) implies that \( c \) does not hold. Next, two definitions are introduced for use in determining the set \( \lambda(\omega) \) described below.
Definition 2. Suppose that $e$ is an inequality of the form $a_1x_1 + \cdots + a_nx_n \leq b$ in the variables set $x = (x_1, \ldots, x_n), x_i \in \mathbb{N}$ and $a_1, \ldots, a_n, b \in \mathbb{Z}$. Consider the values $a_1, \ldots, a_n$ assigned to $x_1, \ldots, x_n$, respectively. Supposing that $v = (a_1, \ldots, a_n)$, then the notation $v \models e$ means that $v$ satisfies the inequality $e$ if and only if $a_1a_1 + \cdots + a_na_n \leq b$ is true.

Definition 3. The diagnosis labelling function: a diagnosis labelling function $D : T_+^* \times 2^T_+ \rightarrow \{N, F, FN\}$ is a mapping that associates to each sequence of observable transitions $s$ with respect to the fault type $T_f$ (expressed by $c$), one of the following diagnosis labels:

- $D(s, T_f) = N$ if $\forall \sigma \in L(\mathcal{N}, M_0)$ such that $\pi(\sigma) = s$, $\#(\sigma) \models c$ holds.
- $D(s, T_f) = F$ if $\forall \sigma \in L(\mathcal{N}, M_0)$ such that $\pi(\sigma) = s$, $\#(\sigma) \models \neg c$ holds.
- $D(s, T_f) = FN$ if there exists two sequences $\sigma_1, \sigma_2 \in L(\mathcal{N}, M_0)$ such that $\pi(\sigma_1) = \pi(\sigma_2) = s$, but $\#(\sigma_1) \models c$ and $\#(\sigma_2) \models \neg c$ holds.

Two sets of sequences are defined in the following. The first set characterises the set of sequences in the language of $\mathcal{N}$ corresponding to an observed sequence of events $\omega$ as shown below:

$$\Gamma(\omega) = \{\sigma \in L(\mathcal{N}, M_0) \mid s = \pi(\sigma), \omega = \lambda(s)\}$$

(5)

The second set consists of a number of pairs associated with a given sequence of observed events. Each pair captures the form (observed sequence, diagnosis label) expressed in the following definition:

Definition 4. Suppose that $\mathcal{N}, M_0, \Sigma, \lambda$ is a labelled Petri net. Given an observed sequence $\omega \in \Sigma^*$, we define a set of pairs associated with $\omega$ with respect to the fault type $T_f$ as:

$$X(\omega) = \{(s, l) \mid \exists \sigma \in \Gamma(\omega), s = \pi(\sigma), l = D(s, T_f)\}$$

(6)

Note that the set $X(\omega) \neq \emptyset$ because $\omega$ corresponds to a firing sequence. In the following, the definition of diagnoser is extended inspired by definitions presented in Cabasino et al. (2011) and Fanti et al. (2013).

Definition 5. A diagnoser is a function $\Delta : \Sigma^* \times 2^T_+ \rightarrow$ \{NoFault, Faulty, Uncertain\} that associates with each observed sequence $\omega \in \Sigma^*$ with respect to the fault type $T_f$ one of the following diagnosis states:

- $\Delta(\omega, T_f) = NoFault$ if $\forall \sigma \in \Gamma(\omega), \#(\sigma) \models c$ holds. This state indicates that there is no sequence having the same labels as $\omega$ containing a fault transition in $T_f$, i.e. no fault has occurred.
- $\Delta(\omega, T_f) = Faulty$ if $\forall \sigma \in \Gamma(\omega), \#(\sigma) \models \neg c$ holds. This state holds for all sequences having the same labels as $\omega$ containing a fault transition in $T_f$.
- $\Delta(\omega, T_f) = Uncertain$ if there exists two sequences $\sigma_1, \sigma_2 \in \Gamma(\omega)$ such that $\pi(\sigma_1) \models c$ and $\pi(\sigma_2) \models \neg c$ hold. This case, the behaviour of the system is ambiguous because both NoFault and Faulty states are possible during the observed sequence.

Example 1. Consider the labelled Petri net depicted in Fig. 1. In this net, the initial marking is $M_0 = [100000000000]$. In the figure, the set of observable transitions are depicted by solid rectangles, while empty rectangles represent unobservable transitions. The labelling function $\lambda$ yields $\tau(e) = \{t_3, t_4, t_5, t_6, t_{11}, t_{13}\}$, $\tau(a) = \{t_1\}$, $\tau(b) = \{t_2, t_7\}$, $\tau(c) = \{t_8, t_{10}, t_{14}\}$ and $\tau(d) = \{t_9, t_{12}\}$. Moreover, there is one fault type having two fault transitions $t_8$ and $t_{11}$ denoted by $f_1$ and $f_2$, respectively as shown in the figure. Thus, we have one constraint $\mathbf{c} := x_8 + x_{11} \leq 0$ and its negation $\neg \mathbf{c} := x_8 + x_{11} > 0$ (also written as $\neg \mathbf{c} := -x_8 - x_{11} < -1$). Note that in this Petri net, two transitions sharing the same fault type and the same fault. One constraint $\mathbf{c} := x_8 + x_{11} \leq 0$ and its negation $\neg \mathbf{c} := x_8 + x_{11} > 0$ (also written as $\neg \mathbf{c} := -x_8 - x_{11} < -1$). Note that in this Petri net, two transitions sharing the same fault type and the same fault.
quence $\sigma \in L(\mathcal{N}, M_0)$ such that $\sigma = \sigma_1 t_1 \ldots \sigma_{t_h}$. The set of inequalities in $\mathcal{S}$ can also be rewritten as:

$$
\mathcal{S}' = \left\{ \begin{array}{l}
-A_u \cdot \#(\sigma_1) + Pre(\cdot, t_1) \leq M_0 \\
-A_u \cdot \#(\sigma_1) + \#(\sigma_2) - A \cdot u_1 + Pre(\cdot, t_2) \leq M_0 \\
\vdots \\
-A_u \sum_{1 \leq i < h} \#(\sigma_i) - A \sum_{1 \leq i < h} u_i + Pre(\cdot, t_h) \leq M_0 (h)
\end{array} \right. 
$$

where each subset $\mathcal{S}_i', i = 1, \ldots, h$, of inequalities in $\mathcal{S}'$, e.g. $\mathcal{S}_i' = A_u \cdot \#(\sigma_i) + Pre(\cdot, t_i)$, can be represented by the following general form:

$$
I := (-A \cdot x) + y \leq M_0
$$

given a sequence of transitions $\sigma_1 t_1 \ldots \sigma_{t_h}$, where $y = Pre(\cdot, t_i)$ and $x = \#(\sigma_1 \ldots \sigma_i)$. If we assume that the sequence $\sigma_1 t_1 \ldots \sigma_i$ is enabled at $M_0$, then the transition $t_i$ is enabled if (7) holds.

4.2 Identification of the legal sequences

Given the set of inequalities $I$ as defined in Section 4.1 in the sets of variables $x$ and $y$. Then, assume that the IFME is applied to $I$ to eliminate the variables corresponding to the unobservable transitions resulting in the set of inequalities $I'$. We present the following proposition to characterise legal sequences (sequences of observable transitions). In other words, this proposition can be applied to decide whether a sequence of observable transitions has at least one corresponding sequence in a labelled Petri net.

**Proposition 2.** Suppose that $(\mathcal{N}, M_0, \Sigma, \lambda)$ is a labelled Petri net having no cycle of unobservable transitions. Also, assume that $I$ is the set of inequalities of (7) in the sets of variables $x$ and $y$. The set of inequalities $I'$ is as defined above. Then, for any given sequence of observable transitions $s = t_1 \ldots t_h$, there exists a corresponding sequence $\sigma = \sigma_1 t_1 \ldots \sigma_{t_h}$ in $\mathcal{N}$ such that $M_0 \xrightarrow{\sigma} M_1 \xrightarrow{\sigma_2} \ldots M_0$ if and only if there exists a vector $v' = (\alpha_1, \ldots, \alpha_k, Pre(p_{t_1}, t_1), \ldots, Pre(p_{t_h}, t_h))$ of $I'$, where $\alpha = \#(t_i, s), \alpha = t_1 \ldots t_{i-1}$ and $k = |T_o|$.

**Proof.** Necessity: If there exists $\sigma$ such that $\Pi(\sigma) = s$, then there exists $v = \#(\sigma)$ such that $v \models I$ by the enabling condition. As a result, there exists a corresponding $v'$ such that $v' \models I'$ by Theorem 1.

**Sufficiency:** If there exists $v' \models I'$, there exists a corresponding sequence in $\mathcal{N}$. We prove the base case on the length of $s$, denoted by $|s|$ as follows:

**Base case:** Assume that $|s| = 1$. If $(\alpha_1, \ldots, \alpha_k, Pre(p_{t_1}, t_1), \ldots, Pre(p_{t_h}, t_h)) \models I'$, where $\alpha_0 = 0$ for $1 \leq i \leq k$, then there exists a solution $v = (\alpha_1, \ldots, \alpha_k, \ldots, \alpha, Pre(p_{t_1}, t_1), \ldots, Pre(p_{t_h}, t_h))$ of $I$ by Theorem 1. Assume that $v = (\alpha_1, \ldots, \alpha_k)$, then the sub-net $\mathcal{N}_v$ has only unobservable transitions. Since $\mathcal{N}_v$ is cycle free by the assumption, there exists a sequence $\sigma_1 \in T_o^c$ such that $M_0 \xrightarrow{\sigma_1} M_1$ and $\#(\sigma_1) = v$ by Lemma 1. As a result, we have a sequence $\sigma_1 t_1$ such that $M_0 \xrightarrow{\sigma_1} M_1$ for $s = t_1$. This proves the case.

**Induction step:** Suppose that the result holds for all $s$ with $|s| < h$ (Induction hypothesis). Then, we prove that the result holds for $|s| = h$. Hence, for $s' = t_1 \ldots t_{h-1}$ there exists a sequence $\sigma' = \sigma_1 t_1 \ldots \sigma_{t_{h-1}}$ such that $M_0 \xrightarrow{\sigma_1} M_1 \xrightarrow{\sigma_2} \ldots \xrightarrow{\sigma_{t_{h-1}}} M_{h-1}$. If we have $s = s' \cdot t_h$ such that $(\alpha_1, \ldots, \alpha_k, Pre(p_{t_1}, t_h), \ldots, Pre(p_{t_h}, t_h)) \models I'$, then there exists a solution $v = (\alpha_1, \ldots, \alpha_k, \alpha_{t_{h+1}}, \ldots, \alpha, Pre(p_{t_1}, t_h), \ldots, Pre(p_{t_h}, t_h)) \models I$ by Theorem 1. Assume that $v = (\alpha_1, \ldots, \alpha_k, \alpha_{t_{h+1}}, \ldots, \alpha_n)$ and $z = v - \#(\sigma')$. Then, $M = M_{h-1} + az \geq 0$. Since the sub-net $\mathcal{N}_s$ has only unobservable transitions and it is cycle free, then there exists a sequence $\sigma_i$ such that $M_{h-1} \xrightarrow{\sigma_i} M$ with $\#(\sigma_i) = z$. Further, $v = (\alpha_1, \ldots, \alpha_k, \alpha_{t_{h+1}}, \ldots, \alpha, Pre(p_{t_1}, t_h), \ldots, Pre(p_{t_h}, t_h)) \models I$, then $M \models M_h$. Consequently, there exists a sequence $\sigma = \sigma_1 t_1 \ldots \sigma_{t_h}$ in $\mathcal{N}$ such that $s = t_1 \ldots t_h$. This also proves this case.

**Proposition 2** gives a complete procedure to identify the legal sequences. Identification of these sequences is necessary to determine the diagnosis states.

4.3 Computing the Diagnosis States

Suppose that the set of fault transitions in $\mathcal{N}$ is $T_f \subseteq T_u$ and all faults are of the same type. We can further suppose that $I$ is as defined in (7) in variables $x$ and $y$. Then, assume that the IFME is applied to the sets $I \cup \{c\}$. In order to compute the diagnosis state, we first create two sets $I \cup \{c\}$. Then, applying the IFME method to the sets $I \cup \{c\}$ and $I \cup \{e\}$ respectively yields the reduced sets $R$ and $R'$ created by eliminating every variable corresponding to a transition in the set $T_u$. In the following, we present the results that capture the details of computing a diagnosis state upon observing a sequence of events $\omega$.

**Theorem 2.** Suppose that $(\mathcal{N}, M_0, \Sigma, \lambda)$ is a labelled Petri net having no cycle of unobservable transitions. Also, assume that the set of inequalities $I$ is as defined in (7). The sets of inequalities $R$ and $R'$ plus the inequalities $c$ and $\neg c$ are described above. Then, for any given sequence of observable transitions $s = s' = (\pi(\sigma)) \in I$ and $T_o$, there exists a sequence $\sigma 

D(s, T_f) = \begin{cases} 
N & \text{iff } v' \neq R' \\
F & \text{iff } v' \neq R \\
FN & \text{iff } v' = R \\
\text{Impossible} & \text{iff } v' \neq R \land v' \neq R'
\end{cases}

**Proof.** Case i) $D(s, T_f) = N$: by contradiction, assume that $v' \neq R'$, but $D(s, T_f)$ is not $N$. If $v' \neq R'$, then does not exist a corresponding solution of $v'$ in $I \cup \{e\}$ by Theorem 1. But $v'$ has a corresponding solution, say $v$, in $I$ because it is coming from a sequence in $L(\mathcal{N}, M_0)$, see Section 2.1. Thus, $v' \neq c$, i.e. $v \in c$. As a result, $\forall \sigma \in L(\mathcal{N}, M_0)$ such that $\Pi(\sigma') = s$, $\Pi(\sigma) \models c$ holds. Hence $D(s, T_f) = N$, see Definition 3. This contradicts the assumption. The converse is also true.
Algorithm 1: build the diagnoser (offline step).

Input: A labelled Petri net \((\mathcal{N}, M_0, \Sigma, \lambda)\), a set of unobservable transitions \(T_u\), a single fault type \(T_f\).

Output: The pair \((R, R')\) plus the set \(I'\).

1: Let \(I \leftarrow Ax + \text{Pre}(e, t) \leq M_0\)
2: Let \(e = \sum_{j \in T_f} x_j \leq 0\), \(-e = \sum_{j \in T_f} -x_j \leq 1\)
3: \(I' = I\)
4: \(R = I \cup \{e\}\)
5: \(R' = I \cup \{-e\}\)
6: for all \(t_j \in T_u\) do
7: \(F' \leftarrow \text{IFME method}(I', x_j)\)
8: \(R = \text{IFME method}(R, x_j)\)
9: \(R' = \text{IFME method}(R', x_j)\)
10: end for

Case ii) \(D(s, T_f) = F\): Using a similar argument in the proof of Case i by replacing \(R'\) with \(R\), we can prove this case.

Case iii) \(D(s, T_f) = FN\): If \(v' \models R\), then there exists a corresponding solution in \(v' = I \cup \{e\}\) by Theorem 1. Hence, there exists a sequence in \(L(N, M_0)\) which satisfies \(e\). Likewise, we can prove that if \(v' \models R'\), there exists another sequence satisfying \(-e\). Since there are two sequences having the same \(s\), but one of them satisfies \(e\) and the other satisfies \(-e\), then we have \(D(s, T_f) = FN\), see Definition 3. The converse is also true.

Case iv) Impossible: It is a contradictory statement to have \(v'\), which corresponds to an observed sequence, does not satisfy \(e\) and \(-e\) at the same time. The converse is also true and this completes the proof.

Corollary 1. Assume that \((\mathcal{N}, M_0, \Sigma, \lambda)\) is a labelled Petri net. Then, for any given sequence of observed events \(\omega \in \Sigma^*\), considering that the set \(X(\omega)\) is such that each \((s, l) \in X(\omega)\) is legal, \(\Delta(\omega, T_f)\) is determined as follows:

\[
\Delta(\omega, T_f) = \begin{cases} 
\text{NoFault} & \text{if } \forall(s, l) \in X(\omega), l = N \\
\text{Faulty} & \text{if } \forall(s, l) \in X(\omega), l = F \\
\text{Uncertain} & \text{Otherwise} 
\end{cases}
\]

Proof. A direct proof.

4.4 Fault diagnosis algorithms

In this section, the algorithms developed for fault diagnosis in labelled Petri nets are described. In Algorithm 1, steps 7-9 recursively invoke the IFME procedure (explained previously in Section 2.2) with two parameters. The first parameter represents the set of inequalities and the second one is the variable to be eliminated from the set. The output of Algorithm 1 consists of sets of inequalities \(I', R\) and \(R'\).

The input of Algorithm 2 are the fault type \(T_f\) and the transitions \(\tau(e) \forall e \in \Sigma\). In addition to sets of inequalities \(I', R\) and \(R'\). The output of the algorithm is a diagnosis state from \{NoFault, Faulty, Uncertain\} (see Definition 5). This algorithm starts by initialising \(\omega'\) and \(X(\omega')\). Then, in step 2 in particular, the algorithm enters into a loop to estimate the diagnosis state. In step 3, the algorithm waits until a new event \(e\) is observed and then adds it to the previous sequence \(\omega'\), creating the sequence \(\omega\). From step 5 to step 21, the algorithm builds the set \(X(\omega)\). First, the set of all sequences \(s \in T^*\) corresponding to \(\omega\) in \(\mathcal{N}\) is generated in steps 6-8. The variables \(x_1, \ldots, x_k, y_1, \ldots, y_m\) are computed and their values are allocated to the vector \(v'\) (step 9). Then, each generated sequence is checked to determine whether it has a corresponding sequence in the Petri net (step 10), see Proposition 2. The function \(D(s, T_f)\) is computed in steps 11-17 by applying Theorem 2. Steps 22-28 determine the diagnosis state \(\Delta(\omega, T_f)\) based on Corollary 1.

Algorithm 2: fault diagnosis (online step).

Input: A single fault type \(T_f\); \(\tau(e) \forall e \in \Sigma\) and the sets \(R, R'\) of the fault type \(T_f\), then the online step requires in the worst case \(O(\mid X(\omega)\mid \cdot \mid \tau(e)\mid \cdot m_F)\) to decide the diagnosis.
I start by extending the set of inequalities

\[ x_j + y_j \leq 1 \quad (1) \]

The sets of inequalities

<table>
<thead>
<tr>
<th>( I )</th>
<th>( I' \sim \text{IFME}(I) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_j + y_j \leq 1 )</td>
<td>( x_j + y_j \leq 1 )</td>
</tr>
<tr>
<td>( -x_j + x_j - x_j + y_j \leq 0 )</td>
<td>( -x_j + x_j - x_j + y_j \leq 0 )</td>
</tr>
<tr>
<td>( -x_j + x_j - x_j + y_j \leq 0 )</td>
<td>( -x_j + x_j - x_j + y_j \leq 0 )</td>
</tr>
<tr>
<td>( -x_j + x_j - x_j + y_j \leq 0 )</td>
<td>( -x_j + x_j - x_j + y_j \leq 0 )</td>
</tr>
<tr>
<td>( -x_j + x_j - x_j + y_j \leq 0 )</td>
<td>( -x_j + x_j - x_j + y_j \leq 0 )</td>
</tr>
</tbody>
</table>

Table 1

Now, suppose that we observe the sequence \( \omega = ab \). Two potential sequences \( s_1 = t_1 t_2 \) and \( s_2 = t_1 t_7 \) could correspond \( ab \). The vector \( v' \) can be computed for \( s_1 \) and \( s_2 \) as follows. Assume that \( s_1 = t_1 t_2 \) and \( s_2 = t_1 t_7 \). In case of \( s_1 \), we obtain \( \#(1, s'_1) = 1 \) and \( \#(1, s'_1) = 0 \); \( \forall t_i \in \{2, 7, 8, 9, 10, 12, 14\} \); also \( \text{Pre}(p_1, t_1) = 1 \) and \( \text{Pre}(p_1, t_2) = 0 \); \( \forall j = 2, \ldots, 12 \). For the sequence \( s_2 \), we obtain \( \#(1, s'_2) = 1 \) and \( \#(1, s'_2) = 0 \); \( \forall t_i \in \{2, 7, 8, 9, 10, 12, 14\} \); also \( \text{Pre}(p_1, t_1) = 1 \) and \( \text{Pre}(p_1, t_2) = 0 \); \( \forall j = 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12 \). Hence, the vectors \( v_1' = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \)
and \( v_2' = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \) are determined for \( s_1 \) and \( s_2 \), respectively. Since \( v_1' \not\parallel I' \), then \( s_1 \) is a legal sequence, but \( s_2 \) is not (see Proposition 2). Thus, we ignore \( s_2 \) and check \( v_1' \not\parallel R \); we find that \( v_1' \not\parallel R \) and \( v_1' \not\parallel R' \). This implies that \( D(s_1, T_J) = N \) (see Theorem 2). Based on this, the set \( X(ab) = \{(t_1 t_2, N) \} \). Since \( X(ab) \) contains one sequence with diagnosis label \( N \), then we have NoFault diagnosis state (see Corollary 1).

5 Conclusion

We have presented a new approach for fault diagnosis under partial observation in labelled Petri net models of DES. This approach adopts the IFME method to build the diagnos

erifier offline. In particular, this paper addresses the most general case of fault diagnosis in Petri nets in which another source of non-determinism originates from the fact that different transitions could share the same label and these transitions could be indistinguishable. As a result, part of computational effort is required online to handle this case. By observing a sequence of events (labels), a set of sequences of transitions corresponding to these observed sequences is generated. Then, using the diagnoser this set is analysed to make diagnosis decisions. Since the diagnoser is no longer represented as an automaton, the IFME-based approach can be used in both finite and infinite systems. Furthermore, this current representation of the diagnoser makes the computational complexity of our approach heavily relies on the number of unobservable transitions and not state space size.

A future direction of research can investigate the diagnosis of more complex forms and other types of faults. In addition, decentralised and distributed diagnosis, where many local diagnosers could monitor the state of the system will be taken into account.

References


