Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-7 (2015) 118–123

# Structural Fault Detection in Join Free Timed Continuous Petri Nets

**Manuel Navarro-Gutiérrez * Antonio Ramírez-Treviño * David Gómez-Gutiérrez * Javier Ruiz-León ***

* *CINVESTAV–IPN, Unidad Guadalajara, Av. del Bosque 1145, colonia el Bajío, Zapopan , 45019, Jalisco, México.*
*email:* {mnavarro, art, dgomez, jruiz}@gdl.cinvestav.mx

**Abstract:** This work is concerned with fault detection in a class of timed continuous Petri nets ($TCPN$) under Infinite Server Semantics. Faults are deviations from the nominal system behavior due to undesired changes of the transition firing rates or place token losses. These fault types are represented by adding transitions, named fault transitions, to the $TCPN$ model. This work presents a novel characterization of fault detectability in $TCPN$ based on input-output observability results applied to faults. This paper also proposes the design of a diagnoser for detecting fault occurrences in $TCPN$ which is based on a sliding mode observer.



## 1. INTRODUCTION

Nowadays, the systems are becoming large and complex. Their availability and reliability are major concerns for both academic and industry communities, since with these two characteristics the production costs are reduced and the safety for the human operators, the environment and the system itself are increased.

In modern systems it is mandatory that the control includes a fault tolerance stage in order to confine the system evolution to a safety one by detecting these faults and making a decision to keep working until the system can be repaired and recovers the normal behaviour.

There exist several works dealing with the problem of fault detection, isolation and fault recovery in the case of continuous-time [Edwards et al., 2000] and discrete-time systems as well as in discrete event systems [Ramirez-Trevino et al., 2012], [Ramirez-Trevino et al., 2007], [Mahulea et al., 2012]. The main idea is to solve this problem by using the available system information (provided by sensors), that is, partial observation of the state.

In [Ramirez-Trevino et al., 2007] fault diagnosis of discrete event systems is studied using interpreted Petri nets ($IPN$). The autors model the system including normal and faulty behaviour and assume that events and states are partially observed, then they define the diagnosability for $IPN$ and characterize this property, providing polynomial algorithms for checking it. In [Prock, 1991] a technique for online fault detection using Petri nets based on monitoring the number of tokens residing into P-semiflows is proposed.

In [Lefebvre, 2014] faults in discrete event systems are modeled with partially observed Petri nets measuring the marking and the events to perform state estimation, fault diagnosis and fault prediction. In our work, similar considerations are taken to study faults in TCPN. We partially measure the marking of the system to estimate the whole marking and to detect and isolate faults.

Due to the state explosion of discrete event systems with a large number of reachable states, the study of the faults presented in the system becomes computationally prohibitive. In order to overcome the state explosion problem, the Petri net community developed the Continuous Petri Nets (*CPN*) [Silva and Recalde, 2005], [David and Alla, 1987], [Silva and Recalde, 2002], a relaxation of the Petri nets, where the marking and the firing of transitions become continuous. This makes possible to study the fault diagnosis using linear programming problems [Mahulea et al., 2012], [Mahulea et al., 2013], [Cabasino et al., 2010]. Some advantages are discussed in [Mahulea et al., 2012] where the *fluidization* reduces the computational complexity of the fault analysis in discrete event systems in which discrete approaches are not feasible in practice.

Considering that many works are using *Timed Continuous Petri Nets (TCPN)* for modeling [Ross-León et al., 2010], [Julvez and Boel, 2010], [Tolba et al., 2005] and controlling [Ross-León et al., 2010], [Mahulea et al., 2008], [Kara et al., 2009] discrete event systems, this paper presents a study of fault diagnosis for them, particularly for $TCPN$ under infinite server semantics.

There are some fault detectors that are constructed using a model-based technique or an observer-based scheme, to detect the fault event. In this work, an observer-based fault detector is presented taking advantage of the large work on observability in $TCPN$ that has been done [Aguayo-Lara et al., 2011], [Aguayo-Lara et al., 2014], [Mahulea et al., 2010]. This observer-based fault detector must be able to *detect* when a fault has occurred. Furthermore, it must be capable of *isolating* this fault, i.e., to know where the fault took place. This work presents conditions to achieve both properties.

This document is organized as follows. In Section II basic concepts on Petri nets are presented. The detectability and diagnosability of faults in $TCPN$ is defined and characterized in Section III. Section IV shows how to construct an observer-based fault detector for a diagnosable $TCPN$ and an example is provided. Finally, Section V presents some concluding remarks.


Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2015.06.482

## 2. BASIC DEFINITIONS

This section briefly presents the basic concepts related with *timed continuous Petri Nets*. An interested reader can review [Cassandras and Lafortune, 2006], [Desel and Esparza, 1995], [Jiménez et al., 2004] for further information. Also, it defines the concepts related to fault detection and diagnosability.

### 2.1 Petri Nets concepts

*Definition 1.* A Continuous Petri Net (*CPN*) system is a pair $(N, \mathbf{m_0})$, where $N = (P, T, \mathbf{Pre}, \mathbf{Post})$ is a Petri Net structure (PN) and $\mathbf{m_0} \in \{\mathbb{R}^+ \cup 0\}^{|P|}$ is the initial marking. $P = \{p_1, \ldots, p_n\}$ and $T = \{t_1, \ldots, t_k\}$ are finite sets of elements named places and transitions, respectively. $\mathbf{Pre}, \mathbf{Post} \in \{\mathbb{N} \cup 0\}^{|P|\times|T|}$ are the Pre and Post incidence matrices, respectively, where $\mathbf{Pre}[i, j]$, ($\mathbf{Post}[i, j]$) represents the weights of the arcs going from $p_i$ to $t_j$ (from $t_j$ to $p_i$). The Incidence matrix denoted by $\mathbf{C}$ is computed as $\mathbf{C} = \mathbf{Post} - \mathbf{Pre}$.

We say that $N$ is Join Free (JF) if $\forall t \in T$, $|^\bullet t| = 1$.

Each place $p_i$ has a marking denoted by $m_i \in \{\mathbb{R}^+ \cup 0\}$. The set $\bullet t_i = \{p_j | \mathbf{Pre}[j, i] > 0\}$, ($t_i \bullet = \{p_j | \mathbf{Post}[j, i] > 0\}$) is the preset (postset) of $t_i$. A transition $t_j \in T$ is enabled at marking $\mathbf{m}$ iff $\forall p_i \in \bullet t_j$, $m_i > 0$. Its enabling degree is:

$$enab(t_j, \mathbf{m}) = \min_{p\in\bullet t_j} \frac{m(p)}{\mathbf{Pre}[p, t_j]} \tag{1}$$

The firing of a transition $t_j$ leads to a new marking, in this case it is computed by:

$$\mathbf{m} = \mathbf{m_0} + \mathbf{C}\sigma \tag{2}$$

where $\sigma \in \{\mathbb{R}^+ \cup 0\}^{|T|}$ is the firing count vector, i.e. $\sigma_j$ is the cumulative amount of firing $t_j$ in the sequence $\sigma$. This last equation is named the *CPN* state equation.

### 2.2 Timed Continuous Petri Nets

*Definition 2.* A Timed *CPN* is the tuple $TCPN = (N, \boldsymbol{\lambda}, \mathbf{m_0})$, where N is a Petri Net structure, $\boldsymbol{\lambda} : T \to \{\mathbb{R}^+\}^{|T|}$ is a function that associates a maximum firing rate to each transition, and $\mathbf{m_0}$ is the initial marking of the net N.

The state equation of a *TCPN* is

$$\dot{\mathbf{m}}(\tau) = \mathbf{C}\mathbf{f}(\tau) \tag{3}$$
$$\text{where } \mathbf{f}(\tau) = \dot{\sigma}(\tau).$$

This relaxation of the discrete model considers infinite server semantics, thus the flow $f_j$ through a transition $t_j$ is computed as follows

$$f_j = \lambda_j \cdot \min_{p\in\bullet t_j} \frac{m(p)}{\mathbf{Pre}[p, t_j]} \tag{4}$$

where $\lambda_j$ represents the maximum firing rate of transition $t_j$. Then, a *TCPN* with infinite server semantics can be represented as a switched linear system (*SLS*), which is composed by a family of linear systems switching among them due to the change of the marking in the net of the form:

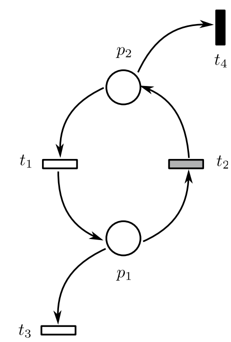


Fig. 1. Petri net with regular and fault transitions

$$\dot{\boldsymbol{m}} = A_k\boldsymbol{m} + B\boldsymbol{u}; \boldsymbol{y} = S\boldsymbol{m} \tag{5}$$

each linear system is represented as $\Sigma_k(A_k, B, S)$ where $B$ and $S$ are the input and output matrices of the $TCPN$.

### 2.3 Introducing Faults to $TCPN$ models

Among other models, $TCPN$ nets are used to model biological, manufacturing, traffic and thermal systems ([Cabasino et al., 2010], [Ross-León et al., 2010], [Julvez and Boel, 2010], [Desirena-Lopez et al., 2014]). They are focused on representing the normal system behaviour. In order to represent fault events, fault transitions are added to the $TCPN$ normal behavior model, that is, transitions representing fault events must be added to the system $TCPN$.

Faults are deviations of the system behavior from its nominal behavior. $TCPN$ captures faults as transitions. This work considers two fault types, one due to transition Firing Rate Disturbances ($FRD$ faults) and the other due to Production Reject, i.e. goods that are discarded ($PR$ faults), represented by sink transitions.

A fault in the $TCPN$ occurs when a firing transition rate is different from its required rate (for sink transitions representing faults the required value is zero and for other transitions representing faults is its nominal value). When no fault is present, the $TCPN$ exhibits a normal behavior. Consequently, in a $TCPN$ with faults we do not know the actual firing rate of fault transitions, this may result in an undesired $TCPN$ behavior.

Fig. 1 shows a $TCPN$ with $P = \{p_1, p_2\}, T = \{t_1, t_2, t_3, t_4\}$, where the grey transition $t_2 \in FRD$ and black transition $t_4 \in PR$.

As it is mentioned above, in order to increase the sytem safety, a fault diagnosis stage must be included in the system, this work addresses the fault location and isolation in $TCPN$. The fault detection characterization is based on the invariant zeros of the system, and the fault isolation is based on the $TCPN$ observer proposed in [Edwards et al., 2000].

### 2.4 Observability in TCPN

Since an observer-based fault detector is used in this paper, some useful results on observability will be recalled from [Aguayo-Lara et al., 2014] and [Mahulea et al., 2013].