12th IFAC/IEEE Workshop on Discrete Event Systems
Cachan, France. May 14-16, 2014

Testing experiments on unbounded systems: synchronizing sequences using Petri nets

Marco Pocci *,** **Isabel Demongodin** * **Norbert Giambiasi** *
Alessandro Giua *,**

* *Aix-Marseille University, France; (e-mail:{marco.pocci,isabel.demongodin,norbert.giambiasi, alessandro.giua}@lsis.org).*
** *Department of Electrical and Electronic Engineering, University of Cagliari, Italy (e-mail: {marco.pocci,giua}@diee.unica.it)*

Abstract: Determining the state of a system when one does not know its current initial state is a very important problem in many practical applications as checking communication protocols, part orienteers, digital circuit reset etc. Synchronizing sequences have been proposed on 60's to solve the problem on systems modeled by finite state machines.
This paper presents a first investigation of the synchronizing problem on unbounded systems, in particular synchronized Petri nets are taken into account.
The proposed approach suffers from the fact that no finite space representation can exhaustively answer to the reachability problem. This problem is shown to be solved for some particular semantics.

1. INTRODUCTION

Testing problems have assumed an important role in the area of discrete event systems due to the increasing need for performance monitoring and verification of complex man-made systems. Several testing problems have been defined: see [Lee and Yannakakis, 1996] for a comprehensive survey.

In this work we focus in particular on the *synchronization problem*. It consists in finding an input sequence that drives a system to a known state having no (or at best partial) information on its current state and without observing the system's output. Such an input is called a *synchronizing sequence* (SS). Interesting and practical applications in this setting can be found in robotics [Ananichev and Volkov, 2003, Natarajan, 1986], bio-computing [Benenson et al., 2001, 2003], network theory [Kari, 2002], theory of codes [Jürgensen, 2008] and testing synchronous circuits with no reset [Cho et al., 1993].

Typical models used for testing are *input/output automata* such as Mealy machines [Moore, 1956]. Recently, however, in a series of papers [Pocci et al., 2014, 2013] we have investigated the problem of determining a SS in the setting of Petri nets (PNs). In particular, we have shown how several approaches developed for automata can be easily applied to the class of bounded deterministic synchronized Petri nets using the *reachability graph* (RG) of a net. Such a graph is an automaton whose behavior is equivalent to that of the net and whose states are vectors in $\mathbb{N}^m$ representing reachable markings. Furthermore we have shown that for special classes of nets a SS can be computed without exploring the complete reachability set but simply analyzing the net structure.

The objective of this paper is to further extend our investigation to the case of *unbounded* PNs, i.e., nets whose reachability set is infinite. Note that, to the best of our knowledge, the synchronizing problem for unbounded models (automata or PNs) has never been investigated before.

The behavior of an unbounded PN can be approximated with a finite *coverability graph* (CG) [Karp and Miller, 1969]. Such a graph is an automaton where each state is a vector in $(\mathbb{N} \cup \{\omega\})^m$ representing a set of markings. An ω component denotes a place whose token content may be arbitrarily large. Unfortunately, as pointed out by Devillers and Van Begin [2006], synchronized PNs have a non-necessarily monotonic evolution. This is why, as explained by David and Alla [2004], for such nets no algorithmic CG construction has been given yet.

This paper has two distinct contributions. First, we propose a CG construction; second, we extend our techniques for the bounded case [Pocci et al., 2014] to this more general setting. However, the ω symbol, used to keep the graph finite, entails loss of information in terms of reachable markings and of firing sequences. That is why, in the case of unbounded nets, the graph can only give precise information about the marking of bounded places and we consider a weaker notion of SS as a sequence that yields a marking where only the token content of bounded places must be exactly known.

This concept of partial synchronization of the state has been suggested in the context of sequential machines by Cheng and Agrawal [1992], where if a SS is prohibitively long or non existent, a selected subset of all flip-flops could be electrically reset.

978-3-902823-61-8/2014 © IFAC

10.3182/20140514-3-FR-4046.00104

A direct application of the classic approach to the CG gives a set of sequences —called *potentially synchronizing sequences*— that are not necessarily fireable from all makings, but that can be fireable from a subset of it, denoted *validating marking set*. In this work we first show how to construct those sequences and then determine the validating marking set.

The paper is organized as follows. Section 2 provides background on Petri nets formalisms. Next, section 3 better states the problem. In section 4 an algorithmic construction of the coverability set of unbounded synchronized Petri nets is provided. Section 5 presents our approach for SS construction. Finally, in section 6 conclusions are drawn and future works presented.

2. PETRI NET FORMALISMS

This section recalls the Petri nets (PNs) used in the paper. For more details on PNs the reader is referred to [David and Alla, 2004].

A PN is a structure $N = (P, T, Pre, Post)$, where P is the set of m places, T is the set of q transitions, $Pre : P \times T \to \mathbb{N}$ and $Post : P \times T \to \mathbb{N}$ are the pre and post incidence functions that specify arcs from places to transitions and viz.

A *marking* is a vector $M : P \to \mathbb{N}$ that assigns to each place a nonnegative integer number of tokens; the marking of a place p is denoted $M(p)$. A marked net, i.e. a net N with an initial marking M_0, is denoted $\langle N, M_0 \rangle$.

A transition t is enabled at M iff $M \geq Pre(\cdot, t)$ and may fire yielding the marking $M' = M + Post(\cdot, t) - Pre(\cdot, t)$. The set of transitions enabled at M is denoted $\mathcal{E}(M)$.

The *preset* and *postset* of a place p are respectively denoted ${}^\bullet p$ and $p^\bullet$. This is extended to a set of places P' as follows: ${}^\bullet P' = \{t : \forall p \in P',\ t \in {}^\bullet p\}$ and $P'^\bullet = \{t : \forall p \in P',\ t \in p^\bullet\}$.

The notation $M[\sigma\rangle$ is used to denote that the sequence of transitions $\sigma = t_1 \dots t_k$ is enabled at M; moreover $M[\sigma\rangle M'$ denotes the fact that the firing of σ from M yields to M'.

Given a sequence $\sigma = t_1 t_2 \dots$, we call $\pi : T^* \to \mathbb{N}^n$ the function that associates to σ a vector $y \in \mathbb{N}^n$, named the *firing vector* of σ. In particular, $y = \pi(\sigma)$ is such that $y(t) = k$ if the transition t is contained k times in σ.

T^* denotes the set of finite-length sequences that can be generated by concatenating arbitrary elements of T allowing the use of the same element multiple times. Here, $*$ is the kleene star.

A marking M is said to be *reachable* in $\langle N, M_0 \rangle$ if there exists a firing sequence σ such that $M_0[\sigma\rangle M$.

The set of all markings reachable from M_0 defines the *reachability set* of $\langle N, M_0 \rangle$ and is denoted $R(N, M_0)$.

A place is bounded if there exists $k > 0$ s.t. $\forall M \in R(N, M_0)$ $M(p) \leq k$. A marked PN $\langle N, M_0 \rangle$ is bounded if all places are bounded. If such is not the case, the marked PN is unbounded.

A *synchronized PN* [David and Alla, 2004] is a structure $\langle N, M_0, E, f \rangle$ such that: i) $\langle N, M_0 \rangle$ is a marked net; ii) E is an alphabet of input events; iii) $f : T \to E$ is a labeling function that associates with each transition t an input event $f(t)$. The set T_e of transitions associated with the input event e is defined as follows: $T_e = \{t \mid t \in T, f(t) = e\}$.

Let $V = \{v_1, v_2, \dots v_k\}$. V^* denotes the set of finite-length sequences that can be generated by concatenating arbitrary elements of V allowing the use of the same element multiple times. Here, $*$ is the kleene star.

The evolution of a synchronized net is driven by an input sequence as follows. At a marking M, a transition $t \in T$ fires only if:

(1) it is enabled, i.e., $t \in \mathcal{E}(M)$;
(2) the event $e = f(t)$ occurs.

On the contrary, the occurrence of an event associated with a transition $t \notin \mathcal{E}(M)$ does not produce any firing. Equivalently to PNs, $M[w|\sigma\rangle M'$ denotes that the occurrence of w at M yields marking M' by the firing of sequence σ.

Enabled transitions associated with event e are denoted $\mathcal{E}_e(M) = \{t : t \in T_e \bigcap \mathcal{E}(M)\}$.

A *totally synchronized PN* is net such that there is a one-to-one mapping between transitions and input events.

In the rest of the paper, the reader will only deal with the class of synchronized PNs that also satisfy the following structural restriction, that is common in the literature to ensure the determinism of the model:

$$\nexists p \ s.t. \qquad t, t' \in p^\bullet \ and \ f(t) = f(t'). \qquad (1)$$

When an event occurs in a deterministic net, all enabled transitions receptive to that event can simultaneously fire. Thus an input sequence $w = e_1 e_2 \cdots e_k \in E^*$ drives a deterministic net through the sequence of markings M_0, M_1, M_2, $\cdots$, M_k where M_0 is the initial marking and

$$M_{i+1} = M_i + \sum_{t \in T_{e_{i+1}} \bigcap \mathcal{E}(M_i)} (Post(\cdot, t) - Pre(\cdot, t)).$$

A standard analysis technique for unbounded PNs is based on the so-called *coverability graph* (CG), that provides an approximated finite description of the infinite reachability set. Unfortunately, no algorithmic construction for such a graph exists in the synchronized PN framework. Let us first extend the notion of marking as follows.

Definition 1. **(ω–marking)** *Let $\mathbb{N}_\omega = \mathbb{N} \bigcup \{\omega\}$. An ω–marking of a PN N with m places is a column vector $M_\omega \in \mathbb{N}_\omega^m$, whose components may either be an integer number or be equal to ω.* ■

For every unbounded place $\exists M_\omega \in \mathcal{G}$ s.t. $M(p) = \omega$.

When the content of a place may grow indefinitely, its marking is replaced by the ω-symbol. This occurs whenever an increasing sequence is found.

Definition 2. **(Increasing input sequence)** *Consider a marked synchronized PN $\langle N, M_0, E, f \rangle$. An input sequence $w \in E^*$ is called* increasing *at marking $M_1 \in R(N, M_0)$ if:*

$$\begin{cases} M_1[w|\sigma\rangle M_2[w|\sigma\rangle M_3[w|\sigma\rangle \cdots \\ M_i \lneqq M_{i+1} \ \forall i = 1, 2, \dots \end{cases}$$ ■

In other words, an increasing sequence can fire infinitely often starting from M_1, always producing the same firing of transitions and leading to a greater marking.