Expert Systems with Applications 42 (2015) 519–526

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Avoiding state explosion in a class of Petri nets

Latif Salum

Dokuz Eylul University, Muhendislik Fak. – End. Bol., Izmir, Turkey

ARTICLE INFO

Article history:
Available online 13 August 2014

Keywords:
Petri nets
State explosion problem
Reachability
Superposition

ABSTRACT

This paper introduces leveled Petri nets (PNs), and proposes a novel PN analysis tool, the superposition chain (SC), to avoid state explosion. It also introduces underlying tools—superposition and the leveled token game—to tackle the *P* vs *NP* problem, a well known problem in CS/AI community. The leveled token game, defined over a leveled PN, generates the SC of the PN. The leveling is based on the transitions such that a transition and all its input places are in the same level, and that there is no causality among transitions in a level, while transitions across levels indicate causality. The enabling rule is extended by superposition and firing history. Superposition of markings is defined by a set ${}^{\veebar}M$ of places p marked in superposition, and denotes that each p in ${}^{\veebar}M$ is marked individually, yet it is uncertain if all p in ${}^{\veebar}M$ are marked together. In other words, superposition loses which p in ${}^{\veebar}M$ is marked by conflicting transitions, which are revealed by the transition firing history. The firing history of $p \in {}^{\veebar}M$ is also defined by a set, $h(p)$, and denotes transition firings participated in $p \in {}^{\veebar}M$, yet does not enumerate their firing sequences to avoid the state explosion. Then, the compound firing history defined over ${}^{\veebar}M$, $h({}^{\veebar}M)$, is used to reveal all conflicting transitions participated in ${}^{\veebar}M$. Hence, ${}^{\veebar}M$ is not coverable as a whole, if there are conflicting firings in $h({}^{\veebar}M)$, which is used for the transition enabling. Consequently, the SC, generated by the leveled token game, specifies the PN behavior, as a reachability tree, generated by the (conventional) token game, also specifies a PN behavior.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

The reachability tree, constructed by the token game over a Petri net (PN), is one of the PN analysis tools. It specifies the PN behavior and verifies its behavioral properties, e.g., reachability and coverability (Murata, 1989). However, it suffers from the state explosion problem, thus cannot be used extensively. This paper proposes the superposition chain (SC) to specify the PN behavior, and to verify its behavioral properties by *avoiding* the state explosion. The paper also introduces a new class of PNs to facilitate the PN analysis, namely *leveled* PNs, and shows that almost every PN *structure* can be leveled. The PN under study is then called the leveled acyclic-special net, which is a safe PN in which places assume at most one input transition.

The motivation of the paper is not only to tackle the state explosion problem, but also to introduce such tools as superposition and the leveled token game to address a well known problem in CS/AI community, namely the *P* vs *NP* problem. In this regard, the paper acts as the first part of an upcoming paper (Salum) that tackles the reachability problem of acyclic safe PNs, which is *NP*-complete. The reader may also refer to Heiner, 1998, Notomi and Murata, 1992, Notomi and Murata, 1994, Shen, Chung, Chen, and Guo, 2013, Va, 1992, Valmari, 1993 and Valmari, 1998 in particular for a broad discussion and tools to avoid the state explosion problem in PNs.

A superposition chain is constructed by the leveled token game defined over a leveled PN of a system to be analyzed. The PN is leveled based on its transitions such that a transition and all its input places are in the same level, and that there is no causality among transitions in a level, while transitions across levels indicate causality. The transition enabling rule for this game is then extended by superposition and transition firing history proposed to avoid the state explosion.

Superposition, exhibited by quantum systems, is a combination of transition firings, hence of their consequent markings (states of the system). It is facilitated by some operators; $\dot{\vee}$, $\vee$, and $\veebar$. The $\dot{\vee}$-superposition means that *exactly one* marking in the superposition is coverable individually. The $\vee$-superposition means that *every* marking in the superposition is coverable together, while the $\veebar$-superposition means that *each* marking in the superposition is coverable individually, yet it is *uncertain* if some markings are coverable together. In other words, the $\veebar$-superposition, denoted by a set ${}^{\veebar}M$ of places p marked in superposition, *encloses* the semantics of $\dot{\vee}$ and $\vee$, thus *loses* which p in ${}^{\veebar}M$ is marked by conflicting transitions. Consequently, each $p \in {}^{\veebar}M$ is marked individually, yet it is *uncertain* if all $p \in {}^{\veebar}M$ are marked together. This uncertainty is eradicated by the transition firing history defined.

E-mail address: latif.salum@deu.edu.tr

http://dx.doi.org/10.1016/j.eswa.2014.07.037
0957-4174/© 2014 Elsevier Ltd. All rights reserved.

The firing history of a marked place, i.e., $h(p)$ for $p \in {}^{\veebar}M$, denotes transition firings participated in $p \in {}^{\veebar}M$. That is, each transition in $h(p)$ is live in some firing sequence that leads to $p \in {}^{\veebar}M$. However, $h(p)$ does not enumerate the firing sequences to avoid the state explosion, i.e., it indicates *what*, but *not how*. These sequences can be determined when necessary. Then, the compound firing history defined over ${}^{\veebar}M$, $h({}^{\veebar}M)$, is used to reveal all conflicting transitions participated in ${}^{\veebar}M$. Therefore, ${}^{\veebar}M$ is not coverable, if there are conflicting transitions in $h({}^{\veebar}M)$, which is used for the extended enabling rule in the leveled token game. That is, a transition is enabled if there are no conflicting transitions in the compound firing history defined over its input places marked in superposition.

The SC is constructed by the leveled token game that exploits certain features of the PN structure. Its construction process is thus called the *scan* of the PN. In other words, the PN (model) is verified by scanning its structure. The SC (the system) then evolves (is presumed to evolve) from current superposition of markings (of states) in a level, denoted by ${}^{\veebar}M_l$, to the consequent superposition ${}^{\veebar}M_{l^+}$, by firing each transition in superposition enabled in ${}^{\veebar}M_l$ based on the compound firing history defined over its input places marked in ${}^{\veebar}M_l$.

2. Basic Definitions

2.1. Petri nets

This section introduces basics of Petri nets (PNs), while Sections 2.2 and 2.3 address definitions related to the superposition chain proposed.

Definition 2.1. A PN (model) is a tuple, $PN = (P, T, F, w, M^0)$, in which:

- $P = \{p_1, p_2, \ldots, p_m\}$ is a set of places,
- $T = \{t_1, t_2, \ldots, t_n\}$ is a set of transitions such that $P \cap T = \emptyset$,
- $F \subseteq (P \times T) \cup (T \times P)$ is a flow relation,
- $w : F \longrightarrow \{1, 2, \ldots\}$ is a weight function,
- $M^0/M^k/M : P \longrightarrow \{0, 1, 2, \ldots\}$ is an initial/kth/arbitrary marking.
  - $(P, T, F, w) = N$ denotes a PN structure, and (N, M^0) denotes its model.
  - ${}^{\bullet}t = \{p \in P \mid (p, t) \in F\}$ defines a pre-set (input places) of t.
  - $t^{\bullet} = \{p \in P \mid (t, p) \in F\}$ defines a post-set (output places) of t.
  - ${}^{\bullet}p = \{t \in T \mid (t, p) \in F\}$ and $p^{\bullet} = \{t \in T \mid (p, t) \in F\}$ is defined similarly.

Definition 2.2 (*Safe/Special PNs*). A (1-)safe PN is a tuple $PN = (P, T, F, M^0)$ over Definition 2.1 such that each place is marked by at most *one* token, i.e., $M : P \longrightarrow \{0, 1\}$, while a special PN ensures that each place has at most *one* input transition, i.e., $|{}^{\bullet}p| = \{0, 1\}\ \forall p \in P$.

Remark 2.1. *PN is safe* iff *M is a set*. *A marking M is defined by a multi-set in a k-safe PN, where each place is marked by at most k tokens.*

Remark 2.2. The function w is redundant in a safe PN.

Places, transitions and markings in a PN model have various interpretations, and are also represented graphically (Murata, 1989). Their interpretations in Definitions 2.1 and 2.2 correspond to, respectively, conditions, events and states of a real world system, which are depicted, respectively, by circles, squares, and balls resided in circles, called tokens. F denotes a flow relation in a *PN*, which is depicted by directed arcs connecting places and transitions only, i.e., a connection among transitions (or places) is not allowed. For example, each place in Fig. 1, i.e., p_1 and p_2, is marked by one token, i.e., $p_1 \in M$ and $p_2 \in M$. The system state (the marking $M = \{p_1, p_2\}$) then indicates that the preconditions of the events (the input places of the transitions) hold (are marked).

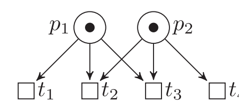

Fig. 1. (Symmetric) confusion.

Definition 2.3 (*The Enabling and Firing Rule*). A transition t_j is *enabled* in a marking M^k if ${}^{\bullet}t_j \subseteq M^k$, i.e., each of its input places is marked by a token in M^k (the preconditions of a system event hold). If t_j is enabled, it can *fire* (the corresponding event can occur). Then, the tokens are *removed* from ${}^{\bullet}t_j$ and the new ones are *created* in $t_j^{\bullet}$ (the post-conditions hold). This transition firing results in a one-step transition in the system, i.e., yields the consequent marking (system state), denoted by $M^k \overset{t_j}{\mapsto} M^n$, where $M^n = \left(M^k \cup t_j^{\bullet}\right) - {}^{\bullet}t_j$. Similarly, $M^0 \overset{\sigma}{\mapsto} M$ denotes that M is reached from M^0 by (firing) σ, where $\sigma = (t_{j_1}, t_{j_2}, \ldots, t_{j_k})$ is a transition firing sequence to reach M from M^0. Then, $M = \left(M^0 \cup t_{j_1}^{\bullet} \cup t_{j_2}^{\bullet} \cup \cdots \cup t_{j_k}^{\bullet}\right) - ({}^{\bullet}t_{j_1} \cup {}^{\bullet}t_{j_2} \cup \cdots \cup {}^{\bullet}t_{j_k})$.

Definition 2.4 (*Execution of a Safe/Special PN*). A *PN* is executed by the *token game*, played from M^0, by the enabling and firing rule until no $t_j \in T$ is enabled. This game/execution then generates a *reachability tree* $\mathcal{T}$, which contains all reachable markings (states) from M^0, denoted by the set $R(M^0)$.

Definition 2.5 (*Reachability and Coverability (Murata, 1989)*). The *reachability problem* is the problem of finding if $M \in R(M^0)$ for a given M. A marking M^c is said to be *coverable* if $M^c(p) \leqslant M(p)\ \forall p \in P$ ($M^c \subseteq M$ for $M \in R(M^0)$ in safe PNs).

A reachability tree $\mathcal{T}$ specifies a *PN* behavior, i.e., a system behavior, as a system under study corresponds to a *PN* (model). However, the $\mathcal{T}$ construction results in a state/M^k explosion. A node of $\mathcal{T}$ denotes a reachable marking $M \in R(M^0)$ (state of the system), and the arc connecting to the node is labeled by the transition firing to reach the marking. The transitions in $\mathcal{T}$ (*PN*) have such firing semantics as concurrency, conflict, confusion, and order (causality), which are discussed by Smith, 1996 in detail. When *concurrent* transitions are enabled, the occurrence of one does not disable the others. When *conflicting* transitions are enabled, the occurrence of one disables the others. *Confusion* is a mixture of concurrent and conflicting transitions, while t_u firing leads to marking of some input place of t_v, if there is *causality* from t_u to t_v. That is, the t_u occurrence *affects* the t_v enabling. For example, t_1 and t_4 in Figs. 1 and 3(a) are concurrent, while t_1, t_2, and t_3 are conflicting in Fig. 1. Further, t_1, t_2, t_3, and t_4 in Fig. 1 are in a (symmetric) confusion, while there is causality (order) from t_3 to t_8 in Fig. 2(c).

Definition 2.6. $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ is a family of sets of conflicting transitions, i.e., of conflicts, where $C_r = p_{(r)}^{\bullet}$ for $|p_{(r)}^{\bullet}| > 1$.

Example 2.1. $C_1 = p_{(1)}^{\bullet} = p_2^{\bullet} = \{t_2, t_3, t_4\}$ and $C_2 = \{t_1, t_2, t_3\}$ in Fig. 1.

Download English Version:

https://daneshyari.com/en/article/382955

Download Persian Version:

https://daneshyari.com/article/382955

Daneshyari.com