Acta Cybernetica 14 (2000) 455-459.

A Note on Decidability of Reachability for Conditional Petri Nets

Ferucio Laurenţiu ŢIPLEA * Cristina BĂDĂRĂU *

Abstract

The aim of this note is to prove that the reachability problem for Petri nets controlled by finite automata, in the sense of [5], is decidable.

1 Introduction and preliminaries

In [5] a new restriction on the transition rule of Petri nets has been introduced by associating to each transition t a language L_t from a family $\mathcal{L}$ of languages. Petri nets obtained in this way have been called *$\mathcal{L}$-conditional Petri Nets* ($\mathcal{L}$-*cPN*, for short). In an $\mathcal{L}$-*cPN* γ, a sequence w of transitions is a transition sequence of γ if it is a transition sequence in the classical sense and additionally $w_1 \in L_t$ for any decomposition $w = w_1 t w_2$. In other words, the transition t is conditioned by the transition sequence previously applied.

It has been proved in [6] that the reachability problem for $\mathcal{L}$-*cPN* in the case that $\mathcal{L}$ contains the Dyck language and is closed under inverse homomorphisms and letter-disjoint shuffle product, is undecidable. The families of context-free, context-sensitive, recursive, recursively enumerable languages, and all the families of L-type Petri net languages satisfy the conditions above, but this is not the case of the family of regular languages; the reachability problem for $\mathcal{L}_3$-*cPN*, where $\mathcal{L}_3$ is the family of regular languages, remained open. In this paper we give a positive answer to this problem.

The set of non-negative integers is denoted by $\mathbf{N}$. For an alphabet V (that is, a nonempty finite set), V^* denotes the free monoid generated by V under the operation of concatenation and λ denotes the unity of V^*. The elements of V^* are called *words* over V. A *language* over V is any subset of V^*. Given a word $w \in V^*$, $|w|$ denotes the length of w.

A *finite deterministic automaton* is a 5-tuple $A = (Q, V, \delta, q_0, Q_f)$, where Q is the set of *states*, V is the set of *input symbols*, $q_0 \in Q$ is the *initial state*, $Q_f \subseteq Q$ is the set of *final states* and δ is a function from $Q \times V$ into Q. The *language accepted* by A is defined by $L(A) = \{w \in V^* | \delta(q_0, w) \in Q_f\}$ (the extension of δ to

*Faculty of Computer Science,"Al. I. Cuza" University of Iaşi, 6600 Iaşi, Romania, e-mail: fltiplea@infoiasi.ro

$V^* \times Q$ is defined as usual). The family of languages accepted by finite deterministic automata, called *regular languages*, is denoted by $\mathcal{L}_3$.

A (finite) *Petri net* (with infinite capacities), abbreviated PN, is a 4-tuple $\Sigma = (S, T, F, W)$, where S and T are two finite non-empty sets (of *places* and *transitions*, respectively), $S \cap T = \emptyset$, $F \subseteq (S \times T) \cup (T \times S)$ is the *flow relation* and $W : (S \times T) \cup (T \times S) \rightarrow \mathbf{N}$ is the *weight function* of Σ verifying $W(x, y) = 0$ iff $(x, y) \notin F$. A *marking* of a PN Σ is a function $M : S \rightarrow \mathbf{N}$. A *marked PN*, abbreviated mPN, is a pair $\gamma = (\Sigma, M_0)$, where Σ is a PN and M_0, the *initial marking* of γ, is a marking of Σ.

The behaviour of the net γ is given by the so-called *transition rule*, which consists of:

(a) the *enabling rule*: a transition t is *enabled* at a marking M (in γ), abbreviated $M[t\rangle_\gamma$, iff $W(s, t) \leq M(s)$, for any place s;

(b) the *computing rule*: if $M[t\rangle_\gamma$ then t may *occur* yielding a new marking M', abbreviated $M[t\rangle_\gamma M'$, defined by $M'(s) = M(s) - W(s, t) + W(t, s)$, for any place s.

The transition rule is extended usually to sequences of transitions by $M[\lambda\rangle_\gamma M$, and $M[wt\rangle_\gamma M'$ whenever there is a marking M'' such that $M[w\rangle_\gamma M''$ and $M''[t\rangle_\gamma M'$, where M and M' are markings of γ, $w \in T^*$ and $t \in T$.

Let $\gamma = (\Sigma, M_0)$ be a marked Petri net. A word $w \in T^*$ is called a *transition sequence* of γ if there exists a marking M of γ such that $M_0[w\rangle_\gamma M$. Moreover, the marking M is called *reachable* in γ.

Let $\mathcal{L}$ be an arbitrary family of languages. An *$\mathcal{L}$-conditional Petri net*, abbreviated $\mathcal{L}$-cPN, is a pair $\gamma = (\Sigma, \varphi)$ where Σ is a PN and φ, the *$\mathcal{L}$-conditioning function* of γ, is a function from T into $\mathcal{P}(T^*) \cap \mathcal{L}$. *Marked conditional Petri nets* are defined as marked Petri nets by changing "Σ" into "Σ, φ".

The *c-transition rule* of a conditional net γ consists of:

(c) the *c-enabling rule*: let M be a marking of γ and $u \in T^*$; the transition t is enabled at (M, u) (in γ), abbreviated $(M, u)[t\rangle_{\gamma,c}$, iff $W(s, t) \leq M(s)$ for any place s, and $u \in \varphi(t)$;

(d) the *c-computing rule*: if $(M, u)[t\rangle_{\gamma,c}$, then t may *occur* yielding a pair (M', v), abbreviated $(M, u)[t\rangle_{\gamma,c}(M', v)$, where $M[t\rangle_\Sigma M'$ and $v = ut$.

As for Petri nets, it can be extended to sequences of transitions.

Let $\gamma = (\Sigma, \varphi, M_0)$ be a marked conditional Petri net. A word $w \in T^*$ is called a *transition c-sequence* of γ if there exists a marking M of γ such that $(M_0, \lambda)[w\rangle_{\gamma,c}(M, w)$. Moreover, the marking M is called *c-reachable* in γ.

2 The main result

The *reachability problem* for Petri nets asks whether, given a net γ and a marking M of γ, M is reachable in γ. The *submarking reachability problem* for Petri nets asks whether, given a net γ, a subset S' of places and a marking M of γ, there exists M' reachable in γ such that $M|_{S'} = M'|_{S'}$. It is well-known that these two problems are equivalent [1] ([4]) and decidable ([3]).

The reachability problem for conditional Petri nets can be defined in a similar way: given an $\mathcal{L}$-conditional net γ and a marking M of γ, is M c-reachable in γ? As we have already mentioned in the first section, for $\mathcal{L}$ being the family of context-free languages (context-sensitive, etc.) the reachability problem is undecidable, and the question is whether this problem is decidable for the case $\mathcal{L} = \mathcal{L}_3$. In what follows we shall give a positive answer to this problem by reducing it to the submarking reachability problem for Petri nets.

Let $\gamma = (\Sigma, \varphi, M_0)$ be an $\mathcal{L}_3$-cPN. We may assume, without loss of generality, that at least a transition of γ is c-enabled at M_0 (otherwise, a marking M is c-reachable in γ iff $M = M_0$). Consider $T = \{t_1, \ldots, t_n\}$, $n \geq 1$, and let $A_i = (Q_i, T, \delta_i, q_0^i, Q_f^i)$ be a finite deterministic automaton accepting the regular language $\varphi(t_i)$, for all i, $1 \leq i \leq n$. We may assume that

- $Q_i \cap Q_j = \emptyset$, for all $i \neq j$, and
- $(S \cup T) \cap \bigcup_{i=1}^{n} Q_i = \emptyset$,

and let $S_i = \{s_q | q \in Q_i\}$, for all i.

We transform now the net Σ into a new net Σ' by adding to the set S all the sets S_i and replacing each transition t_i by some "labelled copies" as follows:

- for each sequence of states $q_1, q_1' \in Q_1, \ldots, q_n, q_n' \in Q_n$ such that $q_i \in Q_f^i$ and $\delta_1(q_1, t_i) = q_1', \ldots, \delta_n(q_n, t_i) = q_n'$, consider a transition $t^i_{q_1,q_1',\ldots,q_n,q_n'}$ which will be connected to places as follows:
  - $t^i_{q_1,q_1',\ldots,q_n,q_n'}$ is connected to places in S as t_i is;
  - for any $1 \leq j \leq n$,

$$W(s_{q_j}, t^i_{q_1,q_1',\ldots,q_n,q_n'}) = W(t^i_{q_1,q_1',\ldots,q_n,q_n'}, s_{q_j'}) = 1.$$

Let M_0' be the marking given by

- $M_0'(s) = M_0(s)$, for all $s \in S$;
- $M_0'(s_{q_0^i}) = 1$, for all $1 \leq i \leq n$;
- $M_0'(s_q) = 0$, for all states $q \in \bigcup_{i=1}^{n} S_i - \{q_0^i | 1 \leq i \leq n\}$,

and let $\gamma' = (\Sigma', M_0')$ be the mPN such obtained (we have to remark that the set T' is non-empty because of the hypothesis). Consider next the homomorphism $h : (T')^* \rightarrow T^*$ given by

$$h(t^i_{q_1,q_1',\ldots,q_n,q_n'}) = t_i,$$

[1] A *decision problem* is a function $A : \mathcal{I} \rightarrow \{0, 1\}$, where $\mathcal{I}$ is a countable set whose elements are called *instances* of A. A decision problem A is *reducible* to a decision problem B if any instance i of A can be transformed into an instance j of B such that $A(i) = 1$ iff $B(j) = 1$. The problems A and B are *equivalent* if each of them can be reduce to the other one.

for any transition $t^i_{q_1,q'_1,\ldots,q_n,q'_n}$ defined as above (the net Σ' together with the homomorphism h is pictorially represented in Figure 2.1: the places are represented by circles, transitions by boxes, the flow relation by arcs, and the numbers $W(f)$ will label the arcs f whenever $W(f) > 1$. The values of h are inserted into the boxes representing transitions).

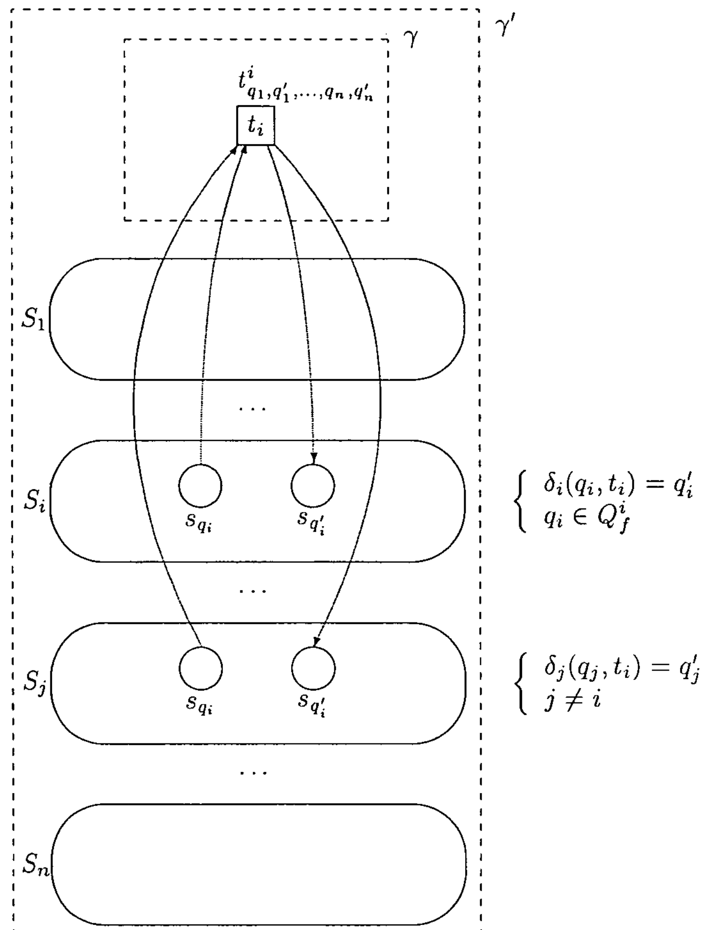

Figure 2.1

It is clear that for any $w \in T^*$ and marking M of γ, $(M_0, \lambda)[w\rangle_\gamma(M, w)$ iff there is $w' \in (T')^*$ and a marking M' of γ' such that $h(w') = w$, $M'_0[w'\rangle_{\gamma'} M'$, and $M = M'|_S$. This shows us that a marking M is reachable in γ iff there is a marking M' reachable in γ' such that $M'|_S = M$. That is, the reachability problem for $\mathcal{L}_3$-cPN can be reduced to the submarking reachability problem for Petri nets, and because this problem in decidable for Petri nets we obtain the next result.

Theorem 2.1 *The reachable problem for $\mathcal{L}_3$-cPN is decidable.*

We close this note by the remark that the reachability problem for Petri nets controlled by finite automata, in the sense of Burkhard ([1], [2]), is undecidable. Our approach to control Petri nets by finite automata ([5]) seams to be more adequate because the reachability problem is decidable and, on the other hand, the power of Petri nets is subtle increased (see [6]).

References

[1] H.D. Burkhard. *Ordered Firing in Petri Nets*, Journal of Information Processing and Cybernetics EIK 17, 1981, 71 – 86.

[2] H.D. Burkhard. *What Gives Petri Nets More Computational Power*, Preprint 45, Sektion Mathematik, Humboldt-Universität zu Berlin, 1982.

[3] E.W. Mayr. *An Algorithm for the General Petri Net Reachability Problem*, Proceedings of the 13rd Annual ACM STOC, 1981, 238–246.

[4] J.L. Peterson. *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, 1981.

[5] F.L. Ţiplea, T. Jucan, C. Masalagiu. *Conditional Petri net languages*, Journal of Information Processing and Cybernetics EIK 27, 1991, 55 – 66.

[6] F.L. Ţiplea. *On Conditional Grammars and Conditional Petri Nets*, in Mathematical Aspects of Natural and Formal Languages (Gh. Păun, ed.), World Scientific Series in Computer Science vol. 43, Singapore, 1994, 431–456.

Received May, 1999