Reachability in contextual nets *

Józef Winkowski
Instytut Podstaw Informatyki PAN
Ordona 21, 01-237 Warszawa, Poland
wink@ipipan.waw.pl

October 11, 2004

Abstract. Contextual nets, or Petri nets with read arcs, are models of concurrent systems with context dependent actions. The problem of reachability in such nets consists in finding a sequence of transitions that leads from the initial marking of a given contextual net to a given goal marking. The solution to this problem that is presented in this paper consists in constructing a finite complete prefix of the unfolding of the given contextual net, that is a finite prefix in which all the markings that are reachable from the initial marking are present, and in searching in each branch of this prefix for the goal marking by solving an appropriate linear programming problem.

Keywords: contextual net, Petri net with read arcs, contextual occurrence net, branching process, unfolding, configuration, history, cut, marking, state, complete prefix, linear programming.

1. Motivation and introduction

The problem of reachability in a system consists in finding a sequence of actions that leads from the initial state of this system to a given goal state.

For systems which can be represented by standard safe Petri nets this problem can be solved by constructing a finite complete prefix of the unfolding of the system net, as proposed by McMillan in [McM 93], and by searching in the branches of such a prefix for an appropriate configuration of events with the aid of linear programming method, as proposed by Esparza in [Espa 93].

The unfolding of the system net is an occurrence net, that is an acyclic Petri net without backward branching at places, that contains information on the possible partial and complete runs of the system and its net (cf. [Eng 91]). Its transitions represent events of executing transitions of the system net. Its places represent tokens that belong to the initial marking of the system net or are produced due to executions of transitions of the system net.

A finite complete prefix of the unfolding is a finite initial fragment of this unfolding that contains all necessary information on reachable markings.

*This work has been supported by the Institute of Computer Science of the Polish Academy of Sciences. The paper is available under the address http://www.ipipan.waw.pl/~wink/winkowski.htm and in Fundamenta Informaticae 51 (2002) 235-250.

A configuration of events of a branch of a finite complete prefix of the unfolding that leads to the given goal marking can be found due to the fact that its characteristic function is a solution of a linear programming problem that is determined by the structure of the respective branch.

In the present paper we adapt the existing results in order to develop a method of solving the problem of reachability for systems which can be represented by contextual nets.

Contextual nets are models of concurrent systems with context dependent actions (cf. [MR 95] and [VSY 98]). Formally, they are Petri nets with extra arcs from places to transitions, called read arcs. The read arcs to a transition make executions of this transition dependent on presence of tokens in the connected places, but not consuming such tokens. The tokens that must be present in places connected to a transition by read arcs play the role of a context that is necessary in order to execute the transition, but that is not affected by the possible execution. A token may belong to contexts of many transitions without preventing such transitions from concurrent execution. Safety and unfoldings of contextual nets can be defined by adapting the corresponding definitions for standard Petri nets.

The problem of reachability in contextual nets can be reduced to the problem of reachability in standard Petri nets. This can be done by simulating a context of a transition as a self loop. However, such a reduction is not satisfactory since it leads to unfoldings with a lot of irrelevant branching that results from the necessity of taking into account the inessential orders of accessing contexts. Consequently, the problem should be solved in the framework of the model of contextual nets rather than by reducing it to the problem of reachability in standard Petri nets.

An attempt of extending the method of constructing a finite complete prefix of the unfolding of a contextual net is described in [VSY 98]. Unfortunately, the solution presented there applies only to a very particular subclass of contextual nets.

The results we present in this paper apply to all finite safe contextual nets and they include both the problem of finite complete prefix of unfolding and the problem of checking branches of such a prefix for existence of a marking. They could be obtained due to the choice of precedence relations in occurrence nets according to the principle "x precedes y iff x necessarily ends before y starts".

The concept of contextual nets is a variant of the concept introduced by [MR 95]. Processes of contextual nets are described by adapting the ideas presented in [Eng 91], [MR 95], [BCM 98], and [VSY 98].

The paper is organized as follows. In section 2 we define contextual nets. In sections 3 and 4 we define processes of safe contextual nets. In section 5 we describe how processes of finite safe contextual nets, can be represented by finite complete prefixes. In section 6 we describe the method of reducing the problem of reachability of a marking in a branch of a finite complete prefix of the unfolding of a contextual net to a linear programming problem.

2. Contextual nets

We start with some preliminaries.

For a relation $R \subseteq X \times Y$ we often write $(x,y) \in R$ as xRy, we define the inverse as the relation $R^{-1} = \{(y,x) : xRy\}$, and for $A \subseteq X$ and $B \subseteq Y$ we define AR and RB as the sets $\{y : aRy \text{ for some } a \in A\}$ and $\{x : xRb \text{ for some } b \in B\}$, respectively, and we write $\{a\}R$ and $R\{b\}$ as aR and Rb, respectively. In particular, for a subset W of a set X with a partial order $\leq$ we have $\leq W = \{x \in X : x \leq w \text{ for some } w \in W\}$ and $W \leq = \{x \in X : w \leq x \text{ for some } w \in W\}$. For relations $R \subseteq X \times Y$ and $S \subseteq Y \times Z$ we define the composition of R and S as the relation $RS = \{(x,z) : xRy \text{ and } ySz \text{ for some } y \in Y\}$. In particular, for mappings $f : X \to Y$ and $g : Y \to Z$ we define the composition of f and g as the mapping $fg : x \mapsto g(f(x))$. For a relation $R \subseteq X \times X$ by R^+ and R^* we denote respectively the transitive and the reflexive and transitive closure of R. For a relation

$R \subseteq X \times X$ and a mapping $f : X \to Y$ we define the image of R under f as the relation $f(R) = \{(f(x), f(x')) \in Y \times Y : xRx'\}$. For a set X, by $|X|$ we denote the cardinality of X.

Contextual nets, or contextual nets of [MR 95] with positive contexts, or Petri nets with read arcs in the sense of [VSY 98], can be defined as follows.

2.1. Definition. A *contextual net* (or briefly, a *net*) is a tuple $N = (P, T, F, C, I)$, where

(1) P is a set of *state elements* (*places*),
(2) T is a set of *transition elements* (*transitions*,) such that $P \cap T = \emptyset$,
(3) $F \subseteq P \times T \cup T \times P$ is a *flow relation* such that $Ft \neq \emptyset$ and $tF \neq \emptyset$ for all $t \in T$,
(4) $C \subseteq T \times P$ is a *context relation* such that $C \cap F = \emptyset$ and $C^{-1} \cap F = \emptyset$,
(5) $I \subseteq P$ is an *initial state* (*initial marking*). □

We denote by U the set $P \cup T$, and we use subscripts, U_N, P_N, T_N, F_N, C_N, I_N, when necessary.

Each state element $p \in P$ represents a place in which some objects, called *tokens* may appear.

Each multiset s of places, that is a function $s : P \to \{0, 1, 2, ...\}$, represents a collection of tokens, $s(p)$ tokens in each place $p \in P$, called a *state* or a *marking* of N. In particular, I represents an initial collection, one token in each place $p \in I$ and no tokens in each place $p \in P - I$.

Each transition element $t \in T$ represents a transition that consumes a collection of tokens, one token from each place of the set Ft, and produces a collection of tokens, one token in each place of the set tF, in the presence of a collection of tokens, one token in each place of the set Ct, the latter collection playing the role of a context that must be present when t is executed, but is not affected by the execution of t. Such a transition element is said to be *enabled* in a state s if the collections corresponding to Ft and Ct are contained in the collection represented by s, and then it transforms s to $s' = (s - Ft) + tF$.

Finally, the net N is said to be *safe* if $s(p) \leq 1$ for all places $p \in P$ and all states s that are *reachable* from the initial state I in the sense that there is a finite sequence $I = s_0, s_1, ..., s_k = s$ of states and a finite sequence $t_1, ..., t_k$ of transitions such that t_{i+1} is enabled in s_i and it transforms s_i to s_{i+1}.

2.2. Example. The graph in figure 1 represents the contextual net $N = (P, T, F, C, I)$ with $P = \{p, q, p', q'\}$, $T = \{v, w, v', w'\}$, $F = \{(p, v), (v, p'), (p', v'), (v', p), (q, w), (w, q'), (q', w'), (w', q)\}$, $C = \{(p, w), (q, v)\}$, $I = \{p, q\}$. □

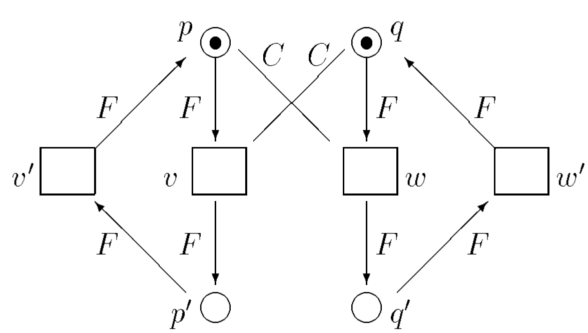

Figure 1

3. Occurrence nets

A process of a contextual net can be defined as an occurrence contextual net as defined below with a suitable mapping to the considered contextual net.

3.1. Definition. A *contextual occurrence net* (or briefly, an *occurrence net*) is a net $N = (P, T, F, C, I)$ with the following properties:

(1) for each $p \in P$ there exists at most one $t \in T$ such that tFp,
(2) $(F \cup FC)^*$, the reflexive and transitive closure of the relation $F \cup FC$, is a partial order $\leq$, called the *precedence relation*, such that each element of $U = P \cup T$ has only a finite number of predecessors with respect to $\leq$, (a *finitary* partial order),
(3) $(\leq \cup C^{-1}F)^*$, the reflexive and transitive closure of the relation $\leq \cup C^{-1}F$, is a quasi-order $\preceq$, called the *strong precedence relation*, such that, for each element of U, the restriction of $\preceq$ to the set of predecessors of this element with respect to $\leq$ is a partial order,
(4) the relation $\sharp$, where $u \sharp u'$ if $t \leq u$ and $t' \leq u'$ for $t, t' \in T$ such that $t \neq t'$ and pFt and pFt' for a some $p \in P$ (the *conflict relation*), is irreflexive, that is such that $u \sharp u'$ implies $u \neq u'$.
(5) I is min, the set of those elements of P that are minimal with respect to $\leq$. □

We use subscripts, $\leq_N$, $\preceq_N$, $\sharp_N$, min_N, when necessary.

Our definition of a contextual occurrence net is essentially as that in [BCM 98]. It is similar in spirit also to the definition of an occurrence net given in [VSY 98]. It differs from the latter in defining the precedence relation $\leq$ as the reflexive and transitive closure of the relation $F \cup FC$ rather than the reflexive and transitive closure of the relation $F \cup C$. Such a definition, which corresponds to the definition of the causal dependency relation in [BCM 98], allows us to guarantee the important property stated in the proposition 3.9 below.

A contextual occurrence net N as defined represents a process that may branch, each branch corresponding to one of a number of possible runs of the process. Each state element $p \in P$ represents an occurrence of a fact. Each transition element $t \in T$ represents an occurrence of an action, (an *event*). For each event $t \in T$, the sets Ct, Ft, tF represent respectively the context in which t occurs and the state elements t consumes and produces. The precedence relation $\leq$ describes how state elements and events follow one another. In particular, each event that consumes a state element follows this element, each state element produced by an event follows this event, and each event that has in its context a state element produced by an event follows this event (but not the state element itself since such an element is not consumed). Note that, according to (3) of 2.1, only state elements can be minimal w.r. to $\leq$.

The strong precedence relation $\preceq$ allows to impose on each run of the represented process the requirement that there is no consumption of a state element before completing all the events having this element in the context. The union of the conflict relation $\sharp$ and the restriction of $\preceq$ to nonidentical elements corresponds to the asymmetric conflict relation of [BCM 98].

3.2. Example. An occurrence net with state elements a, b, c, d, e, f, g, h and events $\alpha, \beta, \gamma, \delta, \varepsilon, \varphi$, where $aF\alpha$, αFc, $bF\beta$, βFd, $cF\gamma$, γFe, $dF\delta$, δFf, $eF\varepsilon$, εFg, $fF\varphi$, φFh, $aC\beta$, $aC\varphi$, $bC\alpha$, $bC\varepsilon$, is shown in figure 2. This net has two branches: one corresponding to the execution of $\alpha, \gamma, \varepsilon$, and the other corresponding to the execution of β, δ, φ. □

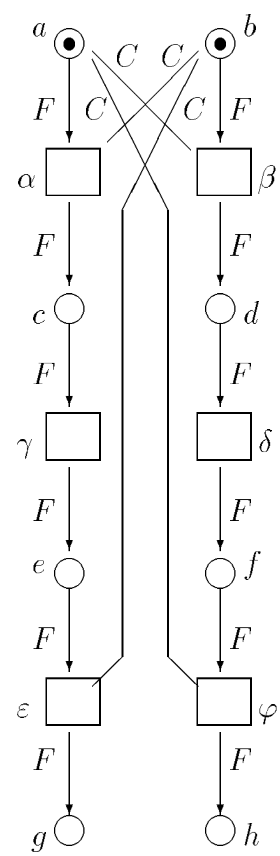

Figure 2

3.3. Definition. A *configuration* of an occurrence net $N = (P, T, F, C, I)$ is a set $V \subseteq T$ of events of N such that:

(1) $(\leq V) \cap T \subseteq V$ (V is lower closed w.r. to the the restriction of the precedence relation $\leq$ to T),
(2) V does not contain events t and t' such that $t \sharp t'$ (V is conflict-free),
(3) the restriction of the strong precedence relation $\preceq$ to the set V is a partial order (V is acyclic w.r. to the strong precedence relation $\preceq$). $\square$

Given an occurrence net N and its configuration V as defined, the set $\overline{V} = I \cup V \cup VF$ is called the *closure* of V. The restriction of N to $\overline{V}$ is an occurrence net, called the *history* corresponding to V and written as $hist(V)$. The restriction of the strong precedence relation $\preceq$ to $\overline{V}$ is a partial order, and it coincides with $\preceq_{hist(V)}$, the strong precedence relation of $hist(V)$. Each maximal antichain of $\overline{V}$ with respect to this partial order is called a *cut* of N. Such a cut is said to be *proper* if it does not contain events. In particular, if the set of those elements of $\overline{V}$ that are maximal with respect to $\preceq_{hist(V)}$ is a maximal antichain of $\overline{V}$ with respect to $\preceq_{hist(V)}$ then it is a proper cut, called the *resulting cut* of V and written as $cut(V)$.

Given a cut Z, it follows from the definitions that the set of events belonging to $\leq Z$ is a configuration. We write such a configuration as $conf(Z)$ and say that Z is *reachable* if $conf(Z)$ is finite. It is clear that $conf(cut(V)) = V$ for each configuration V that has the resulting cut, $cut(V)$.

Configurations of N are ordered by the relation: $V \sqsubseteq V'$ iff $V \subseteq V'$ and $(\preceq_{hist(V')} V) \cap T \subseteq V$, that is iff each event that is a predecessor of an event of V with respect to the strong precedence relation of $hist(V')$ belongs to V. Moreover, according to 3.1, for each event $t \in T$

there exists at least one configuration containing t, namely the configuration $(\leq t) \cap T$, and this configuration is finite and minimal in the set of configurations which contain t.

Cuts of N are ordered by the relation: $Z \Rightarrow Z'$ iff $conf(Z) \sqsubseteq conf(Z')$ and each $z \in Z$ has in Z' an upper bound z' with respect to $\preceq_{hist(conf(Z'))}$.

As the precedence relation $\leq$ is finitary, N has the least cut, namely min_N, the set of those state elements that are minimal with respect to $\leq$.

We say that N is *well bounded* if it has the greatest cut Z and coincides with $hist(conf(Z))$ for this cut. If N is well bounded then $\preceq$ is a partial order and the greatest cut is max_N, the set of maximal elements of N with respect to this order.

Finally, we say that N is *finite* if the set of state elements and events of N is finite.

3.4. Example. The sets $\emptyset$, $\{\alpha\}$, $\{\beta\}$, $\{\alpha,\gamma\}$, $\{\beta,\delta\}$, $\{\alpha,\gamma,\varepsilon\}$, $\{\beta,\delta,\varphi\}$ are configurations of the occurrence net in figure 2. We have
$\overline{\emptyset} = cut(\emptyset) = \{a, b\}$,
$\overline{\{\alpha\}} = \{a, b, c, \alpha\}$, $cut(\{\alpha\}) = \{b, c\}$,
$\overline{\{\beta\}} = \{a, b, d, \beta\}$, $cut(\{\beta\}) = \{a, d\}$,
$\overline{\{\alpha,\gamma\}} = \{a, b, c, e, \alpha, \gamma\}$, $cut(\{\alpha,\gamma\}) = \{b, e\}$,
$\overline{\{\beta,\delta\}} = \{a, b, d, f, \beta, \delta\}$, $cut(\{\beta,\delta\}) = \{a, f\}$,
$\overline{\{\alpha,\gamma,\varepsilon\}} = \{a, b, c, e, g, \alpha, \gamma, \varepsilon\}$, $cut(\{\alpha,\gamma,\varepsilon\}) = \{b, g\}$,
$\overline{\{\beta,\delta,\varphi\}} = \{a, b, d, f, h, \beta, \delta, \varphi\}$, $cut(\{\beta,\delta,\varphi\}) = \{a, h\}$,
$\emptyset \sqsubseteq \{\alpha\} \sqsubseteq \{\alpha,\gamma\} \sqsubseteq \{\alpha\gamma,\varepsilon\}$,
$\emptyset \sqsubseteq \{\beta\} \sqsubseteq \{\beta,\delta\} \sqsubseteq \{\beta,\delta,\varphi\}$.
Moreover, the sets $\{b,\alpha\}$, $\{b,\gamma\}$, $\{b,\varepsilon\}$, $\{a,\beta\}$, $\{a,\delta\}$, $\{a,\varphi\}$ are unproper cuts, and we have
$\{a,b\} \Rightarrow \{b,\alpha\} \Rightarrow \{b,c\} \Rightarrow \{b,\gamma\} \Rightarrow \{b,e\} \Rightarrow \{b,\varepsilon\} \Rightarrow \{b,g\}$,
$\{a,b\} \Rightarrow \{a,\beta\} \Rightarrow \{a,d\} \Rightarrow \{a,\delta\} \Rightarrow \{a,f\} \Rightarrow \{a,\varphi\} \Rightarrow \{a,h\}$. $\square$

The following propositions allow us to consider configurations as partial runs of the respective process and cuts as potential stages of process development.

Let $N = (P, T, F, C, I)$ be an occurrence net.

3.5. Proposition. Given a configuration V of N, and an event $v \in V$, each element of the set $Fv \cup Cv$ belongs to I or to $v'F$ with $v' \in V$ such that $v' \preceq v$ and $v' \neq v$, and it does not belong to Fv'' for any $v'' \in V$ such that $v'' \preceq v$ and $v'' \neq v$. $\square$

Proof outline: The first part of the proposition follows from the fact that for each $p \in Fv \cup Cv$ such that p does not belong to I there exists $v' \in T$ such that $p \in v'F$ and, consequently, $v' \leq v$, that is $v' \in V$.

For the second part it suffices to notice that the conditions $p \in Fv$ and $v'' \neq v$ exclude $p \in Fv''$, and that the conditions $p \in Cv$ and $p \in Fv''$ imply $v \preceq v''$. $\square$

3.6. Proposition. Given an event $t \in T$, and a state element $p \in Ct$, such a state element belongs to each cut of N which contains t. $\square$

Proof outline: Suppose that Z is a cut that contains t and that V is a configuration such that Z is a maximal antichain of $hist(V)$ with respect to $\preceq' = \preceq_{hist(V)}$. Then p is a state element of $hist(V)$ since either $p \in I$ or $p \in uF$ for some $u \in T$, where $u \in V$ due to $p \in Ct$. There is no $z \in Z$ such that $z \neq p$ and $z \preceq' p$ since otherwise there would be $z \preceq t$ in spite of $z, t \in Z$. There is no $z \in Z$ such that $z \neq p$ and $p \preceq' z$ since otherwise there would be $q \in Z$ such that $q \neq p$ and $p \preceq' q$ and this would imply the existence of $v \in V$ such that $q \in vF$ and, consequently, $t \preceq' v \preceq' q$ in spite of $q, t \in Z$. Hence $p \in Z$. $\square$

3.7. Proposition. Given a cut Z of N, the configuration $V = conf(Z)$, and an event $t \in T$, if $Ft \cup Ct \subseteq Z$ then:

(1) $V' = V \cup \{t\}$ is a configuration of N,
(2) $V \sqsubseteq V'$,
(3) $Z' = (Z - Ft) \cup tF$ is a cut of N,
(4) $conf(Z') = V'$. $\square$

Proof outline: For (1), suppose that $u \leq t$ for some $u \in T$. Then either $u = t$ and, consequently, $u \in V'$, or $u \leq p$ for some $p \in Ft \cup Ct \subseteq Z$ and, consequently, $u \in V'$. Moreover, there is no $v \in V$ such that $t \preceq v$ since then there would be $p \preceq q$ for some $p \in Ft$ and $q \in Z$ such that $v \preceq q$, and this would contradict $p, q \in Z$.

For (2), it suffices to notice that $v \preceq' t$ for $v \in T$ and $v \neq t$ and $\preceq' = \preceq_{hist(V')}$ implies $v \preceq p$ for some $p \in Ft \cup Ct$ or $v \in Z$ and, consequently, $v \in V$, as required. (3) and (4) follow from the fact that Z' is the set of maximal elements of $hist(V')$. $\square$

3.8. Proposition. For each reachable cut Z of N there exists a finite sequence $t_1, ..., t_n$ of events of N and a finite chain $min_N = Z_0 \Rightarrow Z_1 \Rightarrow ... \Rightarrow Z_n = Z$ of cuts such that each t_i is enabled at Z_{i-1} and it has the result Z_i in the sense that $Ft_i \cup Ct_i \subseteq Z_{i-1}$ and $conf(Z_i) = conf(Z_{i-1}) \cup \{t_i\}$. $\square$

Proof outline: The proposition follows by induction on the number of events in $conf(Z)$ taking into account 3.7. $\square$

3.9. Proposition. Given a configuration V of N that has the resulting cut, $cut(V)$, the restriction of N to the set of those $u \in (cut(V) \leq)$ for which the context of each event $t \in (cut(V) \leq)$ such that $t \leq u$ is contained in $(cut(V) \leq)$ is an occurrence net, written as $tail_N(V)$. Moreover, for each configuration V' of N such that $V \sqsubseteq V'$, the set $V' - V$ is a configuration of $tail_N(V)$. $\square$

Proof outline: The fact that $tail_N(V)$ is an occurrence net is a direct consequence of the properties of N. In order to prove that for $V \sqsubseteq' V'$ the set $V' - V$ is a configuration of $tail_N(V)$ it suffices to notice that all the state elements that are in contexts of events of $tail_N(V)$ must be in the set $(cut(V) \leq)$ and hence the conditions that u is an event of $tail_N(V)$ and that $u \leq v$ imply $u \in V' - V$. $\square$

4. Processes of contextual nets

In general, processes of contextual nets can be defined as contextual occurrence nets with suitable mappings to the considered contextual nets. Such mappings, called in the sequel morphisms, can be defined as follows.

4.1. Definition. A *morphism* from a net N to a net N' is a triple $m : N \to N'$, where m is a mapping from U_N to $U_{N'}$ such that

(1) $m(P_N) \subseteq P_{N'}$ and $m(T_N) \subseteq T_{N'}$,
(2) $m(F_N t) = F_{N'} m(t)$ and $m(tF_N) = m(t)F_{N'}$ and $m(C_N t) = C_{N'} m(t)$ for all $t \in T$,
(3) the restriction of m to $\{t\} \cup F_N t \cup C_N t \cup tF_N$ a bijection from $\{t\} \cup F_N t \cup C_N t \cup tF_N$ to $\{m(t)\} \cup F_{N'} m(t) \cup C_{N'} m(t) \cup m(t)F_{N'}$ for all $t \in T_N$,
(4) the restriction of m to I_N is a bijection from I_N to $I_{N'}$. $\square$

4.2. Definition. Given a contextual net N_0, a *concrete branching process* (or briefly, a *concrete process*) of N_0 is a tuple $M = (P, T, F, C, I, m)$, where $onet(M) = (P, T, F, C, I)$

is an occurrence net, and $mor(M) = m$ is a net morphism such that, for all $t', t'' \in T$, the relations $Ft' = Ft''$ and $Ct' = Ct''$ and $m(t') = m(t'')$ imply $t' = t''$. $\square$

We use subscripts, U_M, P_M, T_M, F_M, C_M, I_M, $\leq_M$, $\sharp_M$, $\preceq_M$, m_M, when necessary, and we apply to M the terminology introduced for occurrence nets. By *configurations* (resp.: *histories*, *cuts*) of M we mean configurations (resp.: histories, cuts) of $onet(M)$. By min_M (resp.: max_M) we denote $min_{onet(M)}$ (resp.: $max_{onet(M)}$). For each cut Z of M by $tail_M(Z)$ we mean $tail_{onet(M)}(Z)$. We say that M is *well bounded* (resp.: *finite*) if such is $onet(M)$.

The condition imposed on the morphism m is exactly as in [Eng 91].

Note that each contextual net has a concrete process, namely the process that consists of the restriction of this net to its initial state and of the embedding of this this restriction into the net.

4.3. Example. The structure $M = (P, T, F, C, I, m)$, where (P, T, F, C, I) is the occurrence net in figure 2 and m is the morphism from this net to the contextual net in figure 1 that is defined as follows is a concrete process of the net in figure 1: $m(a) = m(e) = p$, $m(b) = m(f) = q$, $m(c) = m(g) = p'$, $m(d) = m(h) = q'$, $m(\alpha) = m(\varepsilon) = v$, $m(\beta) = m(\varphi) = w$, $m(\gamma) = v'$, $m(\delta) = w'$. $\square$

4.4. Definition. Given a contextual net N_0, a *process morphism* (or briefly, a *morphism*) from a concrete process M of N_0 to a concrete process M' of N_0 is a net morphism $h : onet(M) \to onet(M')$ such that $hm_{M'} = m_M$. $\square$

4.5. Definition. Given a contextual net N_0, an *abstract branching process* (or briefly, an *abstract process*, or a *process*) of N_0 is an isomorphism class of concrete processes of N_0. $\square$

The occurrence net with anonymous labelled elements in figure 3 can be regarded as an abstract process of the net in figure 1.

Let N_0 be a contextual net.

4.6. Definition. Given two abstract processes μ and μ' of N_0, the process μ is said to *approximate* the process μ', written as $\mu \trianglelefteq_{N_0} \mu'$, if there exists an injective morphism from a concrete process $M \in \mu$ to a concrete process $M' \in \mu'$. $\square$

4.7. Theorem. The relation $\trianglelefteq_{N_0}$ is a partial order. $\square$

Proof outline: The proof is essentially as in [Eng 91] with the representation of each abstract process μ of N_0 by the unique canonical member M of μ in which each $x \in U_M = P_M \cup T_M$ coincides with its code defined recursively by the formula
$cod(x) = (m_M(x), cod(F_M x), cod(C_M x))$. $\square$

4.8. Theorem. The set of abstract processes of N_0 with the partial order $\trianglelefteq_{N_0}$ is a complete lattice. $\square$

Proof outline: The proof is as in [Eng 91].

By $Processes(N_0)$ we denote the lattice of abstract processes of N_0 with the partial order $\trianglelefteq_{N_0}$.

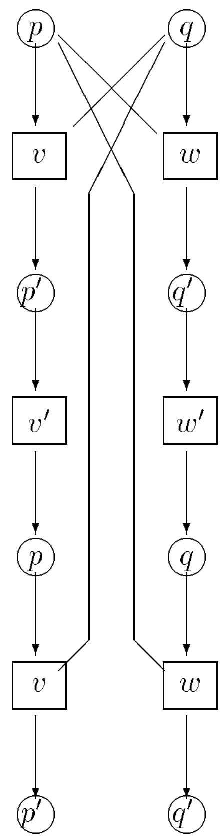

Figure 3

4.9. Definition. The *unfolding* (or the *behaviour*) of N_0 is the gretest element of the lattice $Processes(N_0)$, written as $unf(N_0)$. Each concrete process in the isomorphism class $unf(N_0)$ is called a *concrete unfolding* of N_0. $\Box$

In the rest of the paper we restrict ourselves to safe contextual nets.

The following proposition is a simple consequence of 3.8.

4.10. Proposition. A contextual net N_0 is *safe* iff each concrete process $M = (P, T, F, C, I, m)$ of N_0 enjoys the following property:
for all $p', p'' \in P$ such that $p' \neq p''$, the equality $m(p') = m(p'')$ implies that there is no cut of M containing both p' and p''. $\Box$

4.11. Definition. Given a cut Z of the underlying occurrence net of a concrete process M of a safe contextual net N_0, the *state* or the *marking* of M at Z, $state_M(Z)$, is defined as follows:

$$state_M(Z) = \{mor_M(z) : z \in Z\}. \Box$$

From 3.8 and 4.10 we obtain that this definition is consistent with the standard definition of reachable markings.

4.12. Proposition. Given a contextual net N_0, a set $s \subseteq P_{N_0}$ is a reachable state of N_0 iff there exists a reachable cut Z of a concrete process M of N_0 such that $s = state_M(Z)$. □

5. Unfoldings of finite safe contextual nets

The unfolding of a safe contextual net contains information about all reachable states of this net. In this section we show that if a net is also finite then this information is contained in a finite prefix of the unfolding, and we adapt the known algorithm of McMillan of constructing such a prefix (cf. [McM 93]). Due to the specific definitions of the relations of precedence and strong precedence in contextual occurrence nets, these results do not need any particular restrictions as in [VSY 98] of the class of nets.

Let N_0 be a finite safe contextual net.

5.1. Definition. Given a concrete unfolding M of N_0, a configuration V of M is said to be *prime* if it has a unique event t that is maximal w.r. to the restriction to V of the strong precedence relation $\preceq_M$ of M. The set of prime configurations with the unique maximal event t is denoted by $[t]$. □

The following proposition is a direct consequence of definitions.

5.2. Proposition. Each prime configuration of the underlying occurrence net of a concrete unfolding of N_0 is finite. □

5.3. Definition. A prime configuration of a concrete unfolding of N_0 is said to be a *cut-off configuration* of M if it contains two different prime subconfigurations V' and V'' such that $V' \sqsubseteq V''$ and $state_M(cut(V')) = state_M(cut(V''))$. □

5.4. Definition. An event t of a concrete unfolding of N_0 is said to be a *cut-off event* (resp.: an *informative event*) if each prime configuration $V \in [t]$ is a cut-off configuration (resp.: there exists a prime $V \in [t]$ that is not a cut-off configuration). □

5.5. Example. The net in figure 4 has a concrete unfolding whose prefix is shown in figure 5. The restriction of this prefix that corresponds to the events $\gamma, \varphi, \varepsilon$ is a complete prefix of the unfolding. Note that ε is an informative event since $[\varepsilon] = \{\{\gamma, \varepsilon\}, \{\gamma, \varphi, \varepsilon\}\}$, and $\{\gamma, \varphi, \varepsilon\}$ is a prime configuration in $[\varepsilon]$ which has not any prime proper subconfiguration with the resulting marking $\{p', q'\}$. On the other hand, ψ is a cut-off event since each prime subconfiguration in $[\psi] = \{\{\gamma, \varepsilon, \psi\}, \{\gamma, \varphi, \varepsilon, \psi\}\}$ has a prime subconfiguration with the same resulting marking. □

5.6. Theorem. Given a concrete unfolding M of N_0, the set of informative events of M, written as $inf(M)$, is finite. Moreover, the restriction of M to the set $I \cup inf(M) \cup inf(M)F_M$, written as $B(M)$, is a *prefix* of M in the sense that it approximates M. □

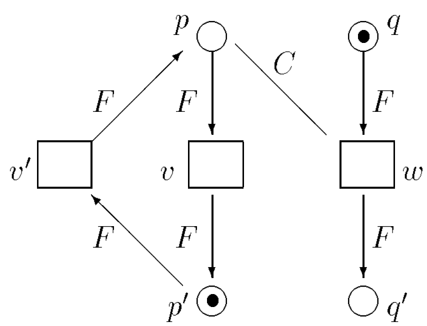

Figure 4

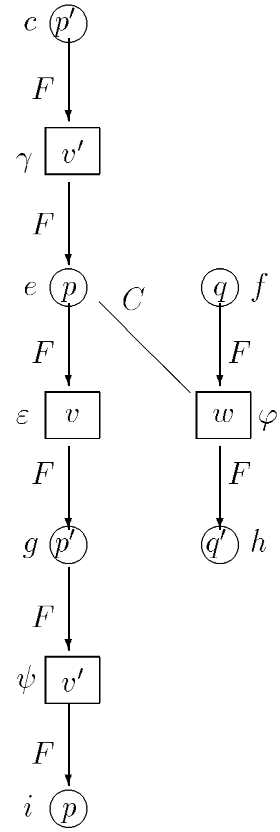

Figure 5

Proof outline: Suppose that $inf(M)$ is infinite. Then the set of prime configurations that are not cut-off ones, denoted by $ncc(M)$, is infinite as well. As each configuration in this set has in this set only a finite number of direct successors, there exists an infinite chain

$$V_1 \sqsubseteq V_2 \sqsubseteq \ldots$$

such that each V_i is in $ncc(M)$ and it is different from V_{i+1}. As N_0 is finite, the set of states of M is finite. Consequently, there exists a state s such that $s = state_M(cut(V_i)) = state_M(cut(V_j))$ for some $i < j$. However, this implies that V_j is a cut-off configuration, which contradicts $V_j \in ncc(M)$.

The second part of the theorem follows from the fact that if a prime configuration in $[t]$ does not contain two different subconfigurations with the same state at the resulting cuts then the same holds true for each event t' such that $t' \leq t$. □

5.7. Theorem. Given a concrete unfolding M of N_0, the prefix $B(M)$ of M is *complete* in the sense that for each reachable state s of N_0 there exists a configuration V of $B(M)$ such that $state_{B(M)}(cut(V)) = s$. □

Proof outline: If V is a finite configuration of M such that $state_M(cut(V)) = s$ and V does not contain a cut-off event then V is a configuration of $B(M)$, as required. Otherwise V contains a cut-off event t and a prime cut-off configuration $V' \in [t]$ such that $V' \sqsubseteq V$ and $state_M(cut(V')) = state_M(cut(V''))$ for some V'' such that $V'' \sqsubseteq V'$ and $V'' \neq V'$. As $tail_M(cut(V'))$ is isomorphic to $tail_M(cut(V''))$, there exists a configuration W that contains less events than V such that $state_M(cut(W)) = s$. Consequently, by replacing V by W and iterating this procedure, we come to a configuration in $B(M)$, as required. □

6. Examining the internal structure of configurations

The construction of a finite complete prefix of the unfolding of a finite safe contextual net results in a concrete process of the net with a finite set of maximal configurations. Consequently, in order to find out how to reach a state with given contents of certain places it suffices to investigate the subconfigurations of the maximal configurations of the finite complete prefix. In this section we show that it is possible with a modified version of the method of Esparza of representing the problem as a problem of linear programming (cf. [Espa 93]).

Let N_0 be a finite safe contextual net.

By modifying the ideas of [Espa 93] one can obtain the following results.

6.1. Proposition. Given a concrete unfolding M of N_0, its configuration V, and a nonempty family $(V_k : k \in K)$ of subconfigurations of V, the set $\bigcup(V_k : k \in K)$ is a configuration of M and it is the least upper bound of the family $(V_k : k \in K)$ with respect to the partial order $\sqsubseteq$. $\square$

6.2. Proposition. Given a concrete unfolding M of N_0, its configuration V, and two disjoint sets S^+ and S^- of state elements of N_0, if the set of subconfigurations V' of V such that $S^+ \subseteq state_M(cut(V'))$ and $S^- \cap state_M(cut(V')) = \emptyset$, written as $Subconf(V, S^+, S^-)$, is nonempty, then it contains the greatest member, that is the greatest subconfiguration V' of V such that $S^+ \subseteq state_M(cut(V'))$ and $S^- \cap state_M(cut(V')) = \emptyset$. $\square$

6.3. Theorem. Given a concrete unfolding M of N_0, its configuration V, and two disjoint sets S^+ and S^- of state elements of N_0, if the set $Subconf(V, S^+, S^-)$ is nonempty, W is the greatest subconfiguration of V that belongs to this set, and $X : V \to \{0,1\}$ is the characteristic function of W as a subset of V, that is $X(v) = 1$ for $v \in W$ and $X(v) = 0$ for $v \in V - W$, then the values of this function constitute a solution of the linear programming problem of finding maximum of $\Sigma(X(v) : v \in V)$ such that the following conditions are satisfied:

(1) for every $v \in V$: $0 \leq X(v) \leq 1$,
(2) for every $p \in FV \cap VF$: $X(pF) \leq X(Fp)$,

where pF denotes the unique $u \in V$ such that pFu, and Fp denotes the unique $v \in V$ such that vFp,

(3) for every $s \in S^+$: $\Sigma(Y(p) : p \in H(s)) = 1$,

where

$$Y(p) = \begin{cases} 1 & \text{if } p \in min_M - FV \\ 1 - X(pF) & \text{if } p \in min_M \cap FV \\ X(Fp) & \text{if } p \in VF - FV \\ X(Fp) - X(pF) & \text{if } p \in FV \cap VF \end{cases}$$

and where $H(s)$ is the set of state elements p of $min_M \cup VF$ with $(mor(M))(p) = s$,

(4) for every $s \in S^-$: $\Sigma(Y(p) : p \in H(s)) = 0$,
(5) for every $u, v \in V$ such that uFp and pCv for some p: $X(v) \leq X(u)$,
(6) for every $u, v \in V$ such that pCu and pFv for some p: $X(v) \leq X(u)$. $\square$

Proof outline: The inequations (1) follow from the fact that X is a characteristic function. The inequations (2), (5), (6) follow from the fact that W is a subconfiguration of V. The equations (3) and (4) reflect the fact that the resulting cut of W has a state element p with

$(mor(M))(p) = s'$ whenever $s' \in S^+$ and it has not any state element q with $(mor(M))(q) = s''$ whenever $s'' \in S^-$. Finally, the maximality of $\Sigma(X(v) : v \in V)$ implies that W is the greatest member of $Subconf(V, S^+, S^-)$. □

6.4. Theorem. Given a concrete unfolding M of N_0, its configuration V, and two disjoint sets S^+ and S^- of state elements of N_0, if the linear programming problem as in 6.3 has no solution then the set $Subconf(V, S^+, S^-)$ is empty. Otherwise this problem has a unique solution and this solution is the set of values of the characteristic function of the greatest member of $Subconf(V, S^+, S^-)$. □

Proof outline: It suffices to prove that the existence of a solution of the considered linear programming problem implies the existence of a solution consisting of integers, and then to prove that the subset of V it defines is the greatest member of $Subconf(V, S^+, S^-)$.

The proof can be carried out by properly modifying the line of [Espa 93].

First of all, if X is a solution of the considered linear programming problem and each component $X(v)$ is replaced by the least integer not less than $X(v)$, denoted by $X'(v)$, then X' and the corresponding Y' satisfy (1) - (6) of 6.3. In fact, only (3) of 6.3 is not trivial.

For a proof of (3) of 6.3 it suffices to notice that each set $H(s)$ with $s \in S^+$ is contained in a chain $p_0 \preceq v_1 \preceq p_1 \preceq ... \preceq v_n \preceq p_n$, and to consider

$$\Sigma(Y(p) : p \in H(s)) = (1 - X(v_1)) + (X(v_1) - X(v_2)) + ... + (X(v_{n-1}) - X(v_n) + X(v_n) = 1.$$

Due to (2), (5), and (6), there exists i such that $X(v_j) > 0$ for $1 \leq j \leq i$ and $X(v_j) = 0$ for $i+1 \leq j \leq n$. Moreover, $(1 - X(v_1)) \geq 0$, $(X(v_1) - X(v_2)) \geq 0$,..., $(X(v_{i-1}) - X(v_i)) \geq 0$, $X(v_i) \geq 0$, and $Y(p_i) > 0$. Consequently, $p_i \in H(s)$. Thus $X'(v_j) = 1$ for $1 \leq j \leq i$ and $X'(v_j) = 0$ for $i+1 \leq j \leq n$. This implies $Y'(p_i) = 1$ and $Y'(p_j) = 0$ for $i+1 \leq j \leq n$. Hence $\Sigma(Y'(p) : p \in H(s)) = 1$ as required.

The fact that X' satisfies (1) - (6) of 6.3 and $\Sigma(X'(v) : v \in V) \geq \Sigma(X(v) : v \in V)$ implies that there must be $X' = X$. This means that if a solution of the considered linear programming problem exists, it must consist of integers, it must be unique, and it must be the characteristic function of a subset W of V.

Finally, from (1) - (6) it follows easily that W is a subconfiguration of V and from the maximality of $\Sigma(X(v) : v \in V)$ it follows that W is the greatest member of $Subconf(V, S^+, S^-)$. □

6.5. Example. Consider the configuration $V = \{\alpha, \gamma, \varepsilon\}$ of the concrete process of the net in figure 1 that corresponds to the occurrence net in figure 2 and to the abstract process in figure 3. Suppose that $S^+ = \{p'\}$ and $S^- = \emptyset$. The greatest configuration in $Subconf(V, S^+, S^-)$ can be obtained by finding $X(\alpha)$, $X(\gamma)$, $X(\varepsilon)$ such that $X(\alpha) + X(\gamma) + X(\varepsilon)$ is maximal provided that the following conditions are satisfied:

$$0 \leq X(\alpha), X(\gamma), X(\varepsilon) \leq 1,$$

$$X(\varepsilon) \leq X(\gamma) \leq X(\alpha),$$

$$Y(c) = X(\alpha) - X(\gamma) = 1.$$

It is easy to see that this takes place for $X(\alpha) = 1$ and $X(\gamma) = X(\varepsilon) = 0$. □

7. Conclusions

We have been considering the problem of reachability of a goal state of a contextual net from the initial state of this net. We have described how to solve this problem for finite safe contextual nets by adapting the methods elaborated for standard Petri nets. In particular, we have shown how to construct a finite complete prefix of the unfolding of a net, and how to reduce the problem of searching for a given state in a given branch of this prefix to a linear programming problem. Thus we have shown a way allowing one to exploit the independence of transitions of finite safe contextual nets in order to reduce the complexity of the procedure of solving the problem of reachability in such nets.

Acknowledgements: The author is grateful to Dr. Edward Ochmański and to the anonymous referee for helpful remarks.

References

[BCM 98] Baldan, P., Corradini, A., Montanari, U., *An Event Structure Semantics for P/T Contextual Nets: Asymmetric Event Structures*, in Proc. of FoSSaCS'98, The First International Conference on Foundations of Software Science and Computation Structures, Lisbon, 1998, Nivat., M. (Ed.), Springer LNCS 1378 (1998) 63-80

[Eng 91] Engelfriet, J., *Branching Processes of Petri Nets*, Acta Informatica 28 (1991) 575-591

[Espa 93] Esparza, J., *Model Checking Using Net Unfoldings*, in Proc. of TAPSOFT'93, Springer LNCS 668 (1993) 613-628

[McM 93] McMillan, K., L., *Using unfoldings to avoid the state explosion problem in the verification of asynchronous circuits*, in Proc. of the 4th Workshop on Computer Aided Verification, Montreal 1992, G. v. Bochmann and D. K. Probst (Eds.), Springer LNCS 663 (1993) 164-174

[MR 95] Montanari, U., Rossi, F., *Contextual Nets*, Acta Informatica 32 (1995) 545-596

[VSY 98] Vogler, W., Semenov, A., Yakovlev, A., *Unfolding and Finite Prefix for Nets with Read Arcs*, Proc. of CONCUR'98, Nice, September 1998, Davide Sangiorgi and Robert de Simone (Eds.), Springer LNCS 1466 (1998) 501-516