Deriving Unbounded Petri Nets from Formal Languages

Philippe Darondeau

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Deriving unbounded Petri nets from formal languages

Philippe Darondeau

N° 3365

Février 1998

THÈME 1

Rapport de recherche

ISSN 0249-6399

Deriving unbounded Petri nets from formal languages

Philippe Darondeau

Thème 1 — Réseaux et systèmes
Projet Paragraphe

Rapport de recherche n3365 — Février 1998 — 23 pages

Abstract: We propose decision procedures based on regions for two problems on pure unbounded Petri nets with injective labelling. One problem is to construct nets from incomplete specifications, given by pairs of regular languages that impose respectively upper and lower bounds on their expected behaviours. The second problem is to derive equivalent nets from deterministic pushdown automata, thus exhibiting their hidden concurrency.

Key-words: Petri nets, regular languages, context-free languages, pushdown automata, semi-linear sets

(Résumé : tsvp)

* Email: Philippe.Darondeau@irisa.fr

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu, 35042 RENNES Cedex (France)
Téléphone : 02 99 84 71 00 - International : +33 2 99 84 71 00
Télécopie : 02 99 84 71 71 - International : +33 2 99 84 71 71

Synthèse de réseaux de Petri non bornés à partir de langages

Résumé : Nous donnons des procédures de décision fondées sur les régions pour deux problèmes de synthèse relatifs aux réseaux de Petri non bornés à étiquetage injectif. Le premier problème est de construire un réseau à partir de spécifications incomplètes, données par deux langages réguliers qui bornent supérieurement et inférieurement le comportement attendu du réseau. Le second problème est d'exhiber le parallélisme caché d'un automate à pile déterministe en construisant un réseau de Petri équivalent.

Mots-clé : réseaux de Petri, langages réguliers, langages hors contexte, automates à pile, ensembles semi-linéaires

1 Introduction

Regions of labelled graphs have been introduced in [ER90a] and ER90b where they served among other to characterize graphs *isomorphic* to marking graphs of elementary net systems. A region maps states to $\{0,1\}$ in such a way that changes of values are uniform on all arcs with a common label, hence it traces the values of a potential place of a net compatible with the considered graph. The Petri-regions of [Muk92], which take integer values, play the same role w.r.t. Petri nets. The *synthesis problem* for nets then reduces to search for *admissible* subsets of regions, distinguishing between all states and justifying restraints on actions at each state [DR96]. Synthesis algorithms deciding whether a given graph holds an *admissible* subset of regions have actually been studied and implemented for elementary nets [CKLY95] and for bounded Petri nets [Ca97].

The tool SYNET described in [Ca97] allows as well to construct bounded nets from finite automata *up to language equivalence*. The bounded regions of regular languages have just been studied in [BBD95] to provide a linear algebraic solution to this relaxed synthesis problem. The present paper extends over this earlier work in several respects. First, the boundedness constraint on nets, quite natural when constructing nets from finite transition systems up to graph isomorphism, but artificial when deriving nets from languages, seen as service specifications, is lifted. Second, complete specifications are replaced by incomplete specifications, namely pairs of regular languages expressing safety and liveness requirements. Third, the net synthesis problem is solved for deterministic context-free languages. The techniques used to obtain the result might work as well for other classes of languages with semi-linear commutative images.

The rest of the paper is organized as follows. Section 2 defines regions in languages and states variant characterizations of net languages in terms of regions. Section 3 solves the synthesis problem for nets from pairs of regular languages. Section 4 solves the synthesis problem for nets from deterministic context-free languages. A short conclusion comments on the limitations and possible continuation of this work.

2 Regions of a language

Let E be a finite set of *events*. A *language* over E is a subset of E^* where $(E^*, \cdot, \varepsilon)$ is the free monoid generated by E. A sequence of events $w \in E^*$ is a *word*. When $w = u \cdot v$, the word u is a *left factor* of the word w (notation: $u \leq w$). A language L is *prefix-closed* if it includes all the left factors of the words it contains (in formulas: $pr(L) \subseteq L$ letting $pr(L) = \{u \in E^* \mid (\exists w \in L)\ u \leq w\}$). A prefix-closed language L over E may be identified with the deterministic (but generally incomplete) automaton $(L, E, T, \varepsilon, L)$, where the states of the automaton are the words of L, the initial state is the empty word, all states are accepting, and T is the set of labelled transitions $u \xrightarrow{e} v$ such that $u, v \in L$, $e \in E$, and $v = ue$. Through this identification, the concept of integer valued regions of a transition system accounted for in [Muk92], [DS93], or [BDPV96], may be adapted as follows to prefix-closed languages over E.

Definition 2.1 (Regions) *A* region *of L is a pair of maps (σ, η), with $\sigma : L \to I\!N$ and $\eta : E \to Z\!\!Z$, such that $w = uv \Rightarrow \sigma(w) = \sigma(u) + \eta(v)$ for all $w \in L$ and $u, v \in E^*$, letting $\eta : E^* \to Z\!\!Z$ be the unique morphism of monoids that extends the map $\eta : E \to Z\!\!Z$ (in formulas: $\eta(uv) = \eta(u) + \eta(v)$ where $u, v \in E^*$ and E is identified with the set of words with unit length). Let $\mathcal{R}(L)$ denote the set of regions of language L.*

Fact 2.2 *$L \subseteq L' \Rightarrow \mathcal{R}(L') \subseteq \mathcal{R}(L)$.*

A region (σ, η) of L is entirely determined by the map η from the value $\sigma(\varepsilon)$, or more generally from $\sigma(w)$ for some $w \in L$. In case when L occurs to be the set of behaviours of a pure Petri net, each place of the net gives rise to and may actually be seen as a region of L. Let us recall the definition of Petri nets.

Definition 2.3 (Petri nets) *A* Petri net *is a triple $N = (P, E, F)$ where P and E are disjoint sets of* places *and* events, *and $F : (P \times E) \cup (E \times P) \to I\!N$. A* marking *of N is a map $M : P \to I\!N$. An event e has* concession *at M if and only if $(\forall p \in P)\quad F(p, e) \leq M(p)$. An event e with concession at M may* fire. *This results in the transition $M[\,e > M'$ such that $(\forall p \in P)$ $M'(p) = M(p) - F(p, e) + F(e, p)$. A Petri net is* pure *if $(\forall p \in P)$ $(\forall e \in E)$ $F(p, e) \times F(e, p) = 0$. The behaviours of an* initialized *net $N = (P, E, F, M_{init})$*

with initial marking M_{init} *are the sequences of events in the inductively defined set* $\mathcal{B}(N) = \{\varepsilon\} \cup \{ev \mid e \in E \ \wedge \ (M_{init} \,|\, e > M_e) \ \wedge \ v \in \mathcal{B}(P, E, F, M_e)\}$. *The set* $\mathcal{B}(N)$ *is also called the* language *of* N.

Definition 2.4 (Subnet) *A* subnet *of an initialized net* $N = (P, E, F, M_{init})$ *is an initialized net* $N' = (P', E, F', M'_{init})$ *induced as a restriction of* N *on some subset of places* $P' \subseteq P$ *(thus*M'_{init} *is the restriction of* M_{init} *and* $F' = F \cap ((P' \times E) \cup (E \times P')))$.

Fact 2.5 $\mathcal{B}(N) \subseteq \mathcal{B}(N')$ *for every subnet* N' *of* N.

Each place p of an initialized pure Petri net $N = (P, E, F, M_{init})$ determines a unique region (σ, η) of $\mathcal{B}(N)$, such that $\sigma(\varepsilon) = M_{init}(p)$ and $\eta(e) = F(e,p) - F(p,e)$ for every event $e \in E$. Conversely, each region (σ, η) of $\mathcal{B}(N)$ determines uniquely an *auxiliary* place p which may be added to N without affecting its behaviours, such that $M_{init}(p) = \sigma(\varepsilon)$ and for every event e, $F(e,p) - F(p,e) = \eta(e)$ and $F(p,e) \times F(e,p) = 0$. More widely, an infinite Petri net $\mathcal{N}(L) = (P, E, F, M_{init})$ may be constructed in this way from any non-empty prefix-closed language L, with a set of places P connected similarly with the regions of L. The inclusion relation $L \subseteq \mathcal{B}(\mathcal{N}(L))$ is then always satisfied by definition of regions. The converse inclusion is generally invalid. For instance, assuming that $L = \mathcal{B}(\mathcal{N}(L))$ for $L = a^*(b+\varepsilon)$ leads to a contradiction as follows. As the sequence b is a behaviour of $\mathcal{N}(L)$ and ba is not, there must exist a region (σ, η) of L such that $\sigma(b) \geq 0$ and $\sigma(ba) < 0$, whence $\eta(a) < 0$. From another source, all sequences a^k are behaviours of $\mathcal{N}(L)$, hence $\sigma(\varepsilon) + k \times \eta(a)$ must be non negative for all k. As $\sigma(\varepsilon)$ is finite, this is clearly impossible. Generalizing on this example, one can state the following proposition, the proof of which is straightforward.

Proposition 2.6 (Characterizing net languages) *For non-empty prefix-closed languages* L *over set of events* E*, the following are equivalent:*
i) *L is the language of some initialized pure Petri net,*
ii) $L = \mathcal{B}(\mathcal{N}(L))$,
iii) $\mathcal{B}(\mathcal{N}(L)) \subseteq L$,
iv) $(\forall u \in L)(\forall e \in E) \quad ue \notin L \ \Rightarrow \ \eta(e) < 0 \ \wedge \ \sigma(\varepsilon) + \eta(u) + \eta(e) < 0$
for some region $(\sigma, \eta) \in \mathcal{R}(L)$.

When these assertions are valid, $L = \mathcal{B}(N)$ *for any subnet* N *of* $\mathcal{N}(L)$ *assembled from a subset of regions* $\mathcal{R}'(L) \subseteq \mathcal{R}(L)$ *large enough to witness for the validity of every instance of assertion (iv).*

This characterization accounts for languages of possibly infinite nets, i.e. nets with infinite sets of places. But not every language of a net is the language of a finite net! For instance, the infinite net $N = (P, \{a, b\}, F, M_{init})$ with countable set of places $P = \mathbb{N}_+$ such that $F(a, p) = 1$ $F(p, b) = p$ and $M_{init}(p) = p(p-1)/2$ for all p has no finite equivalent. This may be proved from the fact that $a^{n(n+1)/2}\, b^n \in \mathcal{B}(N)$ for every natural number n, whereas at the same time $a^{n(n+1)/2}\, b^{n+1} \notin \mathcal{B}(N)$. While we are moderately interested in infinite nets, we consider here as first class citizens the finite initialized nets which have infinite *reachability sets* (sets of markings reached by finite behaviours of the initialized net). This is a significant divergence from the standpoint adopted in [BBD95], where scope was restricted to bounded nets (nets whose marking graphs are finite). The move from finite transition systems to infinite transition systems has an impact on the type of regions that induce places of the synthesized nets $\mathcal{N}(L)$. Work reported in [BBD95] was based upon the linear algebraic properties of the set of bounded regions of a regular language, where $(\sigma, \eta) \in \mathcal{R}(L)$ is *bounded* if $\{\sigma(w) \mid w \in L\}$ is upper bounded (in $\mathbb{N}$), or equivalently, if $\{\eta(w) \mid w \in L\}$ is upper bounded (in $\mathbb{Z}$). It should be emphasized that $\{\eta(w) \mid w \in L\}$ is lower bounded for any region of L. This characteristic property of regions, of little help when focus is on bounded regions, is the cornerstone of the developments presented in this paper. Things may be described as follows. Given a prefix-closed language L over E, let $\eta : E \to \mathbb{Z}$ be any map such that $\{\eta(w) \mid w \in L\}$ has a minimum, let z, necessarily non-positive since $\eta(\varepsilon) = 0$. The fixed map η gives rise to a collection of regions $(\sigma, \eta) \in \mathcal{R}(L)$, each of which corresponds with some integer $K \geq 0$ such that $\sigma(\varepsilon) = K - z$. Among these regions, the least one ($K = 0$) is the most significant since, for every event e such that $\eta(e) < 0$, it is the most likely to bar the exits $ue \notin L$ (where $u \in L$) by producing negative values $\sigma(ue) = \sigma(\varepsilon) + \eta(ue) = (\eta(u) - z) + (K + \eta(e))$. This motivates the following definition.

Definition 2.7 (Abstract regions) *Given a non-empty prefix-closed language* L *over* E*, a map* $\eta : E \to \mathbb{Z}$ *such that* $\{\eta(w) \mid w \in L\}$ *has a minimum is* *called an* abstract region *of* L. *Let* $R(L)$ *be the set of abstract regions of* L. *Given a word* $u \in L$ *and an event* $e \in E$ *such that* $ue \notin L$, *a region* $\eta \in R(L)$ *is said to* bar the exit ue *if* $\eta(u) - min\{\eta(w) \mid w \in L\} + \eta(e) < 0$.

Definition 2.8 (Admissible sets of regions) *A subset* R *of* $R(L)$ *is* admissible *with respect to* L *if every exit* $ue \notin L$ *is barred by some region* $\eta \in R(L)$.

The following is a straightforward adaptation of Prop. 2.6.

Proposition 2.9 (Variant characterization) *A non-empty prefix-closed language* L *over set of events* E *is the language of a (finite) pure Petri net if and only if* $R(L)$ *includes some (finite)* L*-admissible subset. In addition, for any* L*-admissible subset* R *of* $R(L)$, $L = \mathcal{B}(N)$ *letting* $N = (P, E, F, M_{init})$ *be the pure net with places* $p_\eta \in P$ *constructed from corresponding regions* $\eta \in R$ *such that* $M_{init}(p_\eta) = -min\{\eta(w) \mid w \in L\}$ *and* $F(e, p_\eta) - F(p_\eta, e) = \eta(e)$ *for every event* $e \in E$.

In the sequel, the term *region* is always used with the meaning of *abstract region*. In the rest of the section, we glance at the algebraic structure of $R(L)$, the family of regions of L. A region $\eta \in R(L)$ may be identified with a vector in the finite dimensional module $\mathbb{Z} < E >$ and it may therefore be written as a formal sum $\sum_{e \in E} \eta(e) \times e$, where $\eta(e)$ is the e-component of the vector. For instance $2a - 3b$, i.e. $< 2, -3 >$ in vector form, represents the region $\eta(a) = 2$ and $\eta(b) = -3$.

Definition 2.10 (Equivalent regions) *Two regions* $\eta_1, \eta_2 : E \to \mathbb{Z}$ *are* equivalent *if* $(\exists k_1, k_2 \in \mathbb{N})$ $(\forall e \in E)$ $k_1 \times \eta_1(e) = k_2 \times \eta_2(e)$.

Fact 2.11 *Two equivalent regions of* L *bar exactly the same exits* $ue \notin L$.

So, the direction of a region-vector is all that matters. For this reason, it is quite convenient to extend regions to rational values, keeping in mind that multiplication by an adequate natural number is left implicit.

Definition 2.12 (Rational regions) *A* rational region *of* L *is a map* $\eta : E \to \mathbb{Q}$ *such that* $\{\eta(w) \mid w \in L\}$ *has a minimum. Let* $RR(L)$ *denote the set of these maps. A subset of* $RR(L)$ *is* admissible *w.r.t.* L *if every exit* $ue \notin L$ *is barred by some region in this set, where* η bars ue *(notation:* $\eta \ominus ue$*) if* $\eta(u) - min\{\eta(w) \mid w \in L\} + \eta(e) < 0$.

By abuse of notations, the rational regions of L may be identified with vectors in the vector-space $\mathbb{Q} < E >$, and they may be represented by formal sums $\sum_{e \in E} \eta(e) \times e$, where $\eta(e) \in \mathbb{Q}$. For instance, $2a - 3b$ and $(2/3) \times a - b$ denote equivalent rational regions. This convention of representation may also be applied to the *Parikh-images* of the words $u \in E^*$, let $\Psi(u) \in \mathbb{Q} < E >$, where the e-component of the vector counts the occurrences of e in u (thus $\Psi(u)(e) \in \mathbb{N}$). For instance, $\Psi(ababa) = 3a + 2b$. One may then observe that $\eta(u) =< \eta, \Psi(u) >$ for any rational region $\eta : E \to \mathbb{Q}$ and for any word $u \in E^*$, where $< \cdot, \cdot >$ is scalar product. Thus, when η is an unknown region, a set of constraints $\{\eta(u_i) \bowtie_i q_i \mid 1 \leq i \leq n\}$, where $u_1, \ldots, u_n$ are given words, $\bowtie_i \in \{<, \leq, =, \geq, >\}$, and $q_i \in \mathbb{Q}$, may be interpreted as a finite system of equalities and inequalities to be solved in the finite dimensional vector space $\mathbb{Q} < E >$. Deciding the feasability of such a system and computing a solution when it is feasible takes time polynomial in the size of the system (see e.g. [Sch86]). Returning to the algebraic structure of $R(L)$, one may state an obvious proposition.

Proposition 2.13 *The rational regions of L form a cone in $\mathbb{Q} < E >$.*

One cannot say more about the structure of regions without setting specific assumptions upon languages.

3 Deriving unbounded nets from safety and liveness assertions

The goal of the section is to supply a decision of feasability and a solution to the following problem.

Problem 3.1 *Given regular languages $\underline{L}$ and $\overline{L}$ on a finite set of events E, construct a finite initialized pure net $N = (P, E, F, M_{init})$ such that $\underline{L} \subseteq \mathcal{B}(N) \subseteq \overline{L}$.*

This problem is a relaxed version of the following, solved in [BBD95].

Problem 3.2 *Given a regular language L on a finite set of events E, construct an initialized pure net $N = (P, E, F, M_{init})$, where P is a finite set, such that $L = \mathcal{B}(N)$ and the reachability set of N is finite.*

The new motivations under Problem 3.1 are to get rid of the boundedness constraint previously imposed on nets, and to allow constructing nets from *incomplete specifications*. Notice that the usual distinction between state assertions and behavioural assertions fades away for Petri nets, for their marking graphs are deterministic (and even co-deterministic). Therefore, in a net specification $\underline{L} \subseteq \mathcal{B}(N) \subseteq \overline{L}$, the assertions $\mathcal{B}(N) \subseteq \overline{L}$ and $\underline{L} \subseteq \mathcal{B}(N)$ may be seen respectively as a *safety* assertion and a *liveness* assertion. The specification is complete if $\underline{L} = \overline{L}$, otherwise it is incomplete. In any case, one may assume that $\underline{L}$ and $\overline{L}$ are prefix-closed and that $\underline{L} \subseteq \overline{L}$. These assumptions do not affect the decision problem because the operation of closure by left factors and the relation of inclusion are recursive on regular languages. When the specification is incomplete, there may exist zero, one or several languages of nets $L_i = \mathcal{B}(N_i)$ such that $\underline{L} \subseteq L_i \subseteq \overline{L}$. When the specification is complete, i.e. when $\underline{L} = L = \overline{L}$, there exists at most one, namely L. But even in that case, Problem 3.2 and Problem 3.1 are not equivalent. For instance, every finite pure net N such that $\mathcal{B}(N) = a^* + a\,(a^*)\,b\,(a^*)$ has an infinite reachability set, as was observed in [BBD95]. Still, the concepts of branches and loops of regular expressions put forward in [BBD95] will be helpful for solving Problem 3.1. Recall that a *regular expression* over E is an expression in the B.N.F. syntax $L ::= \varepsilon \mid e \mid L + L \mid L \cdot L \mid L^*$, where $e \in E$. An *iterated sub-expression* of L is an expression I such that I^* appears in L.

Definition 3.3 (Branches and loops) *The* branches *of a regular expression L are the words of the language* $\mathrm{br}(L)$ *derived from L by changing every iterated sub-expression of L to ε. The branches of the iterated sub-expressions of L are called* loops *of L, and their set is denoted by* $\mathrm{lp}(L)$.

The branches respectively the loops of a regular expression form actually two *finite* sets. From now on, let $Reg(E^*)$ denote ambiguously the set of the regular expressions over E, and the set of regular languages which they define. Using this ambiguity, we take the liberty to talk of the branches and loops of a regular language without any explicit reference to its expression. Although branches and loops do depend on the chosen regular expression, the indeterminacy left in this way bears no consequence. Next proposition follows immediately from Def. 2.12 and Def. 3.3.

Proposition 3.4 *Let $L \in Reg(E^*)$. A map $\eta : E \to \mathbb{Q}$ is a rational region of L if and only if the finite set $\{\eta(w) \mid w \in \mathrm{lp}(L)\}$ contains no negative number. The minimum of the set $\{\eta(w) \mid w \in L\}$ is then equal to the minimum of the finite set $\{\eta(w) \mid w \in \mathrm{br}(L)\}$.*

In particular, $RR(L) = RR(pr(L))$ for $L \in Reg(E^*)$, as $lp(L) = lp(pr(L))$. We will now propose a decision procedure for Problem 3.2 in the extended case of Petri nets with (possibly) infinite reachability sets. Solving this simpler form of Problem 3.1 is not pointless, for the solution we produce is direct, while on the other hand the solution we shall propose later on for Problem 3.1 relies on the decision of the covering problem for VASSs.

Proposition 3.5 *Given a non-empty prefix-closed language $L \in Reg(E^*)$, one may decide whether $L = \mathcal{B}(N)$ for some finite initialized pure net N.*

Proof: For every $e \in E$, let $L_e = (L \cdot e \cap C(L)) \backslash e$, where $C(\cdot)$ is the complement w.r.t. E^* and $\cdot \backslash e$ is the right quotient by e (hence L_e is regular). From Prop. 2.9 and Def. 2.12, the problem $L = \mathcal{B}(N)$ is solvable if and only if exists for every $e \in E$ a finite set of rational regions $R_e \subseteq RR(L)$ such that $(\forall u \in L_e)(\exists \eta \in R_e)\ \eta \ominus ue$. Since E is finite, the proposition obtains if one can decide upon the existence of an adequate set of regions R_x for a fixed event $x \in E$.

Let $A = (Q, E, T, q_{init}, Q_F)$ be a finite deterministic automaton recognizing L_x, with set of final states Q_F such that some state in Q_F can be reached from any state in Q. One may transform A into an equivalent *tree-like* automaton $A' = (Q', E, T', q'_{init}, Q'_F)$ with components as follows. Let Q' be the set of pairs $(q, u) \in Q \times pr(L_x)$ such that $q_{init} \xrightarrow{u} q$ is a run of A and any two states visited in this run are different. Let T' be the set of transitions $(q, u) \xrightarrow{e} (q', u')$ such that $q \xrightarrow{e} q' \in T$ and $u' = ue$ or u' is a left factor of u. Finally let $q'_{init} = (q_{init}, \varepsilon)$ and $Q'_F = \{(q, u) \in Q' \mid q \in Q_F\}$. A' is a finite deterministic automaton recognizing L_x .

Let us comment on the specific structure of this automaton. The directed graph formed by the *forward* transitions $(q,u) \xrightarrow{e} (q',ue)$ is a tree rooted at q'_{init}, let $\mathcal{T}$. The states $q' \in Q'$ are in bijective correspondence with the paths originated from the root of this tree. Hence the tree $\mathcal{T}$ spans the directed graph (Q',E,T'). The remaining transitions in $T' \setminus \mathcal{T}$, of the form $(q,u'v) \xrightarrow{e} (q',u')$, are chords directed towards the root of $\mathcal{T}$. Hence, each *backward* transition $(q,u'v) \xrightarrow{e} (q',u')$ determines a circuit formed of the forward transitions from (q',u') to $(q,u'v)$ in $\mathcal{T}$ and this backward transition.

Resuming the main course of the proof, one may decompose L_x into a disjoint union $\cup L_{f,B}$ where $f \in Q'_F$, $B \in \mathcal{P}(T' \setminus \mathcal{T})$, and $L_{f,B}$ is the language recognized by the runs of A' that start from q'_{init}, end in f, and pass through all and only the backwards transitions in B (plus the forward transitions on the elementary path from q'_{init} to f). Since regular languages are closed under intersection and morphisms, all the languages $L_{f,B}$ are regular (but some may be empty). As the considered decomposition of L_x is finite, the proposition obtains if one can decide for a fixed state $f \in Q'_F$ and for a fixed set of backwards transitions B upon the existence of a finite set $R_{f,B} \subseteq RR(L)$ such that $(\forall u \in L_{f,B})\,(\exists \eta \in R_{f,B})\;\; \eta \ominus ux$.

In the sequel, we set out $t = (\partial^0(t) \xrightarrow{e(t)} \partial^1(t))$ for every transition $t \in T'$. Let $t_1 \dots t_m$ be the sequence of forward transitions from q'_{init} to f and let $B = \{t_{m+1} \dots t_{m+n}\}$. For $1 \le k \le n$, let $t'_{k,1} \dots t'_{k,n_k}$ be the sequence of forward transitions from $\partial^1(t_{m+k})$ to $\partial^0(t_{m+k})$. Define $u_0 = e(t_1) \dots e(t_m)$, and for $1 \le k \le n$, define $u_k = e(t'_{k,1}) \dots e(t'_{k,n_k})\, e(t_{m+k})$. As $L_{f,B} \subseteq L_x$ and $L_x \subseteq L$, $u_0 \in L$ and u_k is a loop of L for $1 \le k \le n$. It follows from Prop. 3.4 that $\eta(u_1) \ge 0, \dots, \eta(u_n) \ge 0$ for every rational region $\eta \in RR(L)$.

We prove now that if exists a finite set of regions $\{\eta_1, \dots, \eta_N\} \subseteq RR(L)$ such that $\forall u \in L_{f,B} \exists j \le N\; \eta_j \ominus ux$, then throwing away all regions η_j such that $\eta_j(u_k) \ne 0$ for some k (where $1 \le k \le n$) does not harm this condition. So, suppose $\eta_j \ominus ux$ for some $u \in L_{f,B}$ and for some $j \le N$ such that $\eta_j(u_k) > 0$ for some k (where $1 \le k \le n$). We will show that $\eta_i \ominus ux$ for some $i \ne j$ (with $1 \le i \le N$). The main thing is to observe that the Parikh image of $L_{f,B}$ is either the empty set or the linear set of vectors:

$$\{ \sum_{k=0}^{n} \Psi(u_k) \;+\; \sum_{k=1}^{n} c_k \times \Psi(u_k) \mid c_k \in I\!N \}$$

Now let $r = \eta_j(u) - min\{\eta_j(w) \mid w \in br(L)\} + \eta_j(x)$. From Def. 2.12 and Prop. 3.4, $\eta_j \ominus ux$ entails $r < 0$. Choose a positive integer c such that $r + c \times \eta_j(u_k) > 0$ and a word $u' \in L_{f,B}$ such that $\Psi(u') = \Psi(u) + c \times \Psi(u_k)$. The existence of u' is guaranteed by the specific structure of the automaton A'. Since the relation $\eta_j \ominus u'x$ cannot hold, there must exist $i \neq j$ (with $1 \leq i \leq N$) such that $\eta_i \ominus u'x$. From Def. 2.12 and Prop. 3.4, this means that $\eta_i(u) + c \times \eta_i(u_k) - min\{\eta_i(w) \mid w \in br(L)\} + \eta_i(x) < 0$. Recalling that u_k is a loop of L, $\eta_i(u_k)$ is non-negative, hence $\eta_i(u) - min\{\eta_i(w) \mid w \in br(L)\} + \eta_i(x) < 0$, that is $\eta_i \ominus ux$.

The proof is close to its end. Since it suffices to consider regions $\eta \in RR(L)$ satisfying $\eta(u_k) = 0$ for all k $(1 \leq k \leq n)$, constructing a finite set $R_{f,B}$ of regions of L such that $(\forall u \in L_{f,B})\,(\exists \eta \in R_{f,B})\ \eta \ominus ux$ reduces to constructing a map $\eta : E \to \mathbb{Q}$ satisfying the conditions $\eta(w) \geq 0$ for $w \in lp(L)$, $\eta(u_k) = 0$ for $1 \leq k \leq n$, and $\eta(u_0) - min\{\eta(w) \mid w \in br(L)\} + \eta(x) < 0$. Now $\varepsilon \in br(L)$ because L is prefix-closed, and the last condition amounts to the conjunction of the inequalities $\eta(u_0) - \eta(v) + \eta(x) < 0$ for $v \in br(L)$. Since $lp(L)$ and $br(L)$ are finite sets, we are left with a classical problem of linear programming in the rational. Deciding upon the feasability of such problems and computing solutions when exist takes time polynomial in the size of the linear system. Hence the proof is complete. ∎

Let us now tackle Problem 3.1. Although Prop. 3.5 dealt with a particular case of this problem, the decision method based on Prop. 2.9 which we have proposed does not extend to the general case. A different route will be followed, to the cost of an increased complexity. The decision result comes from three facts, established below:
i) $\mathcal{B}(\mathcal{N}(\underline{L}))$ is the least net-language that includes $\underline{L}$;
ii) one may construct from $\underline{L}$ a *finite* subnet N of $\mathcal{N}(\underline{L})$ such that $\mathcal{B}(\mathcal{N}(\underline{L})) = \mathcal{B}(N)$;
iii) the relation $\mathcal{B}(N) \subseteq \overline{L}$ is recursive in N and $\overline{L}$.
It demands few efforts to establish (i). On the one hand, $L \subseteq \mathcal{B}(\mathcal{N}(L))$ for every language L. On the other hand, if $L \subseteq \mathcal{B}(N)$ then each place of N may be identified with a region of $\mathcal{B}(N)$ and therefore with a region of L (from Fact. 2.2), showing that N is a subnet of $\mathcal{N}(L)$, and thence $\mathcal{B}(\mathcal{N}(L)) \subseteq \mathcal{B}(N)$ (from Fact. 2.5). The justification for (iii) relies chiefly on the decision of the

covering problem for vector addition systems with states (VASSs). We recall below the definition of VASSs given in [RY86].

Definition 3.6 (VASS) *A k-dimensional* VASS *is a 5-tuple* $(\vec{v_0}, A, Q, q_0, \delta)$*, where* $\vec{v_0}$ *is a vector in* $\mathbb{N}^k$ *(the* start vector*),* A *is a finite set of vectors in* $\mathbb{Z}^k$ *(the* addition set*),* Q *is a finite set of states,* $q_0 \in Q$ *is the* initial state*, and* $\delta \subseteq Q \times A \times Q$ *is the transition relation. A* configuration *of a VASS is a pair* $(q, \vec{v})$*, where* $q \in Q$ *and* $\vec{v} \in \mathbb{N}^k$*. A configuration* $(q', \vec{v'})$ follows $(q, \vec{v})$ *(notation:* $(q, \vec{v}) \rightsquigarrow (q', \vec{v'})$*) if* $(q, \vec{v'} - \vec{v}, q') \in \delta$*. The* reachability set *of a VASS is the set of configurations* $(q, \vec{v})$ *such that* $(q_0, \vec{v_0}) \rightsquigarrow^* (q, \vec{v})$*.*

We recall also that the *covering problem* for VASSs is the question, given a VASS and a configuration $(q, \vec{v})$, whether exists in the reachability set of the VASS a configuration $(q', \vec{v'})$ such that $q = q'$ and $\vec{v} \leq \vec{v'}$. This problem reduces to the covering problem for vector addition systems, hence it is decidable [Ra78]. Bounds of complexity are stated in [RY86].

Ad (iii). The relation $\mathcal{B}(N) \subseteq L$ is satisfied if and only if $\mathcal{B}(N) \cap C(L) = \emptyset$. Let (Q, E, T, q_0, q_f) be a finite automaton with initial state q_0 and final state q_f recognizing $C(L)$. From this automaton and the net $N = (P, E, F, M_0)$, construct a VASS $(\vec{v_0}, A, Q, q_0, \delta)$ as follows: the dimension of the VASS is the number of places in P; $\vec{v_0}$ is equal to M_0; A is the set of vectors represented by the maps $\vec{v}_e = F(e, \cdot) - F(\cdot, e)$ for $e \in E$; finally, let $(q, \vec{v}_e, q') \in \delta$ if and only if $(q, e, q') \in T$. Deciding whether $\mathcal{B}(N) \cap C(L) \neq \emptyset$ amounts to deciding whether exists in the reachability set of the VASS a configuration covering $(q_f, \vec{0})$. This establishes (iii).

Showing the decidability of Problem 3.1 reduces now to proving (ii), which is an original contribution of this paper. Given a nonempty and prefix-closed language $L \in Reg(E^*)$, let $\mathcal{N}(L) = (P, E, F, M_{init})$ be the pure net synthesized from L. Thus $P = \mathcal{R}(L)$ and for every place $p = (\sigma, \eta)$ in this set, $M_{init}(p) = \sigma(\varepsilon)$ and $F(e, p) - F(p, e) = \eta(e)$ for every $e \in E$. We will show that P includes a *finite* subset of places which is *complete* in the following sense.

Definition 3.7 *Given a net* $N = (P, E, F, M_{init})$*, a subset of places* $P' \subseteq P$ *is* complete *if the following holds for every reachable marking* M *and for every* $e \in E$*:* $(\exists p \in P)\,[\, M(p) - F(p, e) < 0 \,] \;\Rightarrow\; (\exists p \in P')\,[\, M(p) - F(p, e) < 0 \,]$.

Owing to the connections between the sets $\mathcal{R}(L)$, $R(L)$, and $RR(L)$ respectively defined in Def. 2.1, Def. 2.7, and Def. 2.12, $\mathcal{R}(L)$ includes a finite complete subset if and only if $RR(L)$ includes a finite *complete* subset in the following sense.

Definition 3.8 *A set of rational regions $R \subseteq RR(L)$ is* complete *w.r.t. L if, for every $u \in E^*$ such that $\eta(v) - min\{\eta(w) \mid w \in L\} \geq 0$ for all $v \leq u$ and for all $\eta \in RR(L)$, the following is satisfied for every $e \in E$: if $\eta(ue) - min\{\eta(w) \mid w \in L\} < 0$ for some $\eta \in RR(L)$ then the same holds for some $\eta \in R$.*

The set $RR(L)$ is always complete but generally not admissible: the complete subsets and the admissible subsets of $RR(L)$ coincide if and only if L is the language of a net. Nevertheless, whenever R is a complete subset of $RR(L)$, $\mathcal{B}(\mathcal{N}(L)) = \mathcal{B}(N)$ for any net N derived from integer multiples $k \times \eta$ of the rational regions $\eta \in R$. Therefore, (ii) follows from the next proposition.

Proposition 3.9 *For any nonempty prefix-closed language $L \in Reg(E^*)$, the set $RR(L)$ includes a finite complete subset, recursively computable from L.*

Proof: Let $br(L) = \{v_1, \ldots, v_n\}$ and $lp(L) = \{u_1, \ldots, u_K\}$, where by convention $K = 0$ if L is free of loops (see Def. 3.3). From Prop. 3.4, a map $\eta : E \to \mathbb{Q}$ is a rational region of L if and only if $< \eta, \Psi(u_k) > \geq 0$ for all k (where $1 \leq k \leq K$). Therefore, $RR(L)$ is a polyhedral cone. By the Farkas-Minkowski-Weyl theorem (see [Sch86] p.87), this cone is finally generated, hence $RR(L) = \{\sum_{i=1}^{m} q_i \times \eta_i \mid q_i \in \mathbb{Q}_+\}$ for some finite set of maps $\{\eta_1, \ldots, \eta_m\}$, computable from $lp(L)$. From Prop. 3.4, when η is a rational region of L, the minimum of the set $\{\eta(w) \mid w \in L\}$ is always reached at some $w \in br(L)$. Therefore, the cone $RR(L)$ may be covered by a finite family of smaller cones $RR(L,j) = \{\eta \in RR(L) \mid (\forall\, k \leq n)\ \eta(v_j) \leq \eta(v_k)\}$. The proposition will follow if we can show that each small cone $RR(L,j)$ contains a finite subset of regions $R(L,j)$ such that, for every $u \in \mathcal{B}(\mathcal{N}(L))$ and for every $e \in E$, the relation $\eta(ue) < min\{\eta(w) \mid w \in L\}$ holds for some $\eta \in RR(L,j)$ if and only if it holds for some $\eta \in R(L,j)$.

At this stage, consider a fixed $j \in \{1, \ldots, n\}$. By construction, $RR(L,j)$ is the set of regions expressed as linear combinations $\sum_{i=1}^{m} q_i \times \eta_i$ of the generators of $RR(L)$ with coefficients $q_i \in \mathbb{Q}_+$ such that, for all $k \in \{1, \ldots, n\}$, the following holds: $\sum_{i=1}^{m} q_i \times (\eta_i(v_j) - \eta_i(v_k)) \leq 0$. Let $\nabla \subseteq \mathbb{Q}^m$ be the set of vectors $(q_1, \ldots, q_m)$ satisfying this relation for all k. By the Farkas-Minkowski-Weyl theorem, ∇ is a finitely generated cone. Hence $\nabla = \{ \sum_{p=1}^{P} r_p \times \vec{x_p} \mid r_p \in \mathbb{Q}_+ \}$, where $\{\vec{x_1}, \ldots, \vec{x_P}\}$ is a finite set of vectors with components $\vec{x_p}(i)$ in $\mathbb{Q}_+$. Let $R(L,j) = \{\eta'_1, \ldots, \eta'_P\}$, where $\eta'_p = \sum_{i=1}^{m} \vec{x_p}(i) \times \eta_i$ for $1 \leq p \leq P$. Thus $RR(L,j)$ is the set of regions expressed as linear combinations $\sum_{p=1}^{P} r_p \times \eta'_p$ with coefficients r_p in $\mathbb{Q}_+$. Consider now a region $\eta \in RR(L,j)$ such that $\eta(ue) < min\{\eta(w) \mid w \in L\}$ for some $u \in \mathcal{B}(\mathcal{N}(L))$ and for some $e \in E$. Then $\eta(v_j) = min\{\eta(w) \mid w \in L\}$ and $\eta(ue) - \eta(v_j) < 0$. Let $\eta = \sum_{p=1}^{P} r_p \times \eta'_p$, where $r_p \in \mathbb{Q}_+$ for $1 \leq p \leq P$. Necessarily, $\eta'_p(ue) - \eta'_p(v_j) < 0$ for some p. Since $\eta'_p \in RR(L,j)$, and by the assumption on j, $\eta'_p(v_j) = min\{\eta'_p(w) \mid w \in L\}$, hence $\eta'_p(ue) < min\{\eta'_p(w) \mid w \in L\}$, and the proof is complete since $\eta'_p \in R(L,j)$. ∎

4 Deriving unbounded nets from dpda's

Climbing one step in Chomsky's hierarchy, we now address ourselves to construct nets with sets of behaviours specified by *deterministic context-free languages.* Namely, we face the following problem.

Problem 4.1 *Given a deterministic pushdown automaton A accepting a language $L \subseteq E^*$, construct an initialized pure Petri net $N = (P, E, F, M_{init})$ with a finite set of places P, such that $\mathcal{B}(N) = pr\,(L)$.*

Thus, the objective is to transform a sequential machine with central store (the pushdown store) into a non-sequential machine with distributed store (the places of the net). Although parallelization is quite independent of verification, the techniques we propose for solving Prob. 4.1 rely on semi-linear sets like the techniques of verification of pushdown systems proposed in [BH96]. Before tackling Prob. 4.1, let us recall the definition of pushdown automata and semi-linear sets and some of their properties. More information may be found e.g. in [Har78].

Definition 4.2 *A* pushdown automaton *(*pda *for short) is a 7-tuple $A = (Q, E, G, \delta, q_0, g_0, Q_F)$, where Q is a finite nonempty set of states, $q_0 \in Q$ is the* initial state, *$Q_F \subseteq Q$ is the subset of the* final states, *G is a finite nonempty set of* pushdown symbols, *$g_0 \in G$ is the* initial symbol *on the pushdown store, and δ is a function mapping $Q \times (E \cup \{\varepsilon\}) \times G$ to the family of finite subsets of $Q \times G^*$. A* configuration *of A is a triple (q, w, γ), where $q \in Q$ is the current state of control, $w \in E^*$ is the unread part of the word ...w fed into the automaton, and $\gamma \in G^*$ is the contents of the pushdown store (read from bottom to top). Let $\vdash$ be the binary relation on configurations such that $(q, ew, \gamma g) \vdash (q', w, \gamma\gamma')$ whenever $(q', \gamma') \in \delta(q, e, g)$ (for any $q, q' \in Q$, any $e \in (E \cup \{\varepsilon\})$, any $w \in E^*$, any $\gamma, \gamma' \in G^*$, and any $g \in G$). The* language *$\mathcal{T}(A)$ accepted by the pushdown automaton A is the set of the words $w \in E^*$ such that $(q_0, w, g_0) \vdash^* (q, \varepsilon, \gamma)$ for some $q \in Q_F$ and for some $\gamma \in G^*$, where $\vdash^*$ is the reflexive and transitive closure of $\vdash$. The automaton A is* deterministic *(it is a* dpda*) if the set $\delta(q, e, g) \cup \delta(q, e, \varepsilon)$ contains at most one element for each $e \in E$. A language $L \subseteq E^*$ is* deterministic context-free *if $L = \mathcal{T}(A)$ for some dpda A.*

Theorem 4.3 *Let $L \subseteq E^*$ be a deterministic context-free language, then its complement $C(L) = E^* - L$ is a deterministic context-free language, and a dpda accepting $C(L)$ may be constructed from a dpda accepting L.*

The family of languages accepted by pda's concides with the *context-free* languages, and it contains strictly the deterministic context-free languages. It is important to note that one may always construct from a pda a context-free grammar generating the language accepted by this pda. The construction is given in [Har78] (see the proofs of theorems 5.3.2 and 5.4.3). Thanks to this fact and owing to Parikh's theorem (recalled hereafter), one may always construct from a pda A with input alphabet E a regular language $L \in Reg(E^*)$ with Parikh-image $\Psi(L)$ equal to $\Psi(\mathcal{T}(A))$ (the Parikh-image of L is $\Psi(L) = \{\Psi(w) \mid w \in L\}$). This is the stepping stone for synthesizing nets from pushdown systems. In the sequel, we let $\mathbb{N} < E >$ denote the subset of vectors $\vec{x} \in \mathbb{Q} < E >$ such that all their components $\vec{x}(e)$ are in $\mathbb{N}$.

Definition 4.4 (Semi-linear sets) *A* linear *subset of $\mathbb{N} < E >$ is a set of vectors $\{\vec{x_0} + n_1 \vec{x_1} + \ldots + n_m \vec{x_m} \mid n_j \in \mathbb{N}\}$, that is a set of linear combinations of a finite family of generators $\vec{x_j} \in \mathbb{N} < E >$, with non-negative integer coefficients. A* semi-linear *set is a finite union of linear subsets.*

Theorem 4.5 (Parikh's theorem) *The Parikh-image of a context-free language is a semi-linear set.*

Theorem 4.6 *The Parikh-image of a language $L \subseteq E^*$ is a semi-linear set if and only if $\Psi(L) = \Psi(R)$ for some regular language $R \in Reg(E^*)$.*

Given a context-free grammar G generating language L, one may indeed construct from G a regular expression R such that $\Psi(L) = \Psi(R)$ (see section 6.9 in [Har78]). An important consequence is that we obtain therefrom a practical characterization of the rational regions of a context-free language.

Proposition 4.7 *Let $L \subseteq E^*$ be a nonempty context-free language (possibly not prefix-closed), and let $R \subseteq E^*$ be a regular language (possibly not prefix-closed), such that $\Psi(pr(L)) = \Psi(R)$. A map $\eta : E \to \mathbb{Q}$ is a rational region of $pr(L)$ if and only if it is a rational region of $pr(R)$.*

Proof: By Def. 2.12, $\eta : E \to \mathbb{Q}$ is a rational region of $pr(L)$ if and only if $\{\eta(w) \mid w \in pr(L)\}$ has a minimum. Since $\eta(w) =< \eta, \Psi(w) >$ for any $w \in E^*$, η is a rational region of $pr(L)$ if and only if $\{< \eta, \Psi(w) > \mid w \in R\}$ has a minimum. Because R is regular, this amounts to require that $\eta(w)$ be nonnegative for every loop w of R (see Def. 3.3). Since R and $pr(R)$ have similar loops, the proposition obtains. ∎

Combining the above parts, one can construct from any pushdown automaton a finite system of linear inequations characterizing the set of rational regions of the associated context-free language. Given a pda A with input alphabet E, accepting the language $L = \mathcal{T}(A)$, the steps are the following. Derive from A a context-free grammar G generating L. Check from G that L is nonempty. Construct from G a context-free grammar G' generating $pr(L)$. Construct from G' a regular language R such that $\Psi(pr(L)) = \Psi(R)$. Compute the finite set of loops of R, let $lp(L) = \{u_1, \ldots, u_K\}$, where by convention $K = 0$ if L is free of loops. Set out the linear inequations $< \eta, \Psi(u_k) > \geq 0$ for $1 \leq k \leq K$. The solutions $\eta : E \to \mathbb{Q}$ of the system are the rational regions of the context-free language $pr(L)$.

On this ground, one might think that Prob. 4.1 can be solved or shown unfeasible for unrestricted pda's, but the solution we have in mind applies only to deterministic pda's. The reason is that it depends on the crucial property of deterministic context-free languages to be closed under the *max* operation (see below), which is not true for general context-free languages. The following closure properties of (deterministic) context-free languages will be used (see sections 11.2 and 11.3 of [Har78]).

Proposition 4.8 *The (deterministic) context-free languages are closed under right product and under right quotient by regular languages: if L is (deterministic) context-free and R is regular, then $LR = \{uv \mid u \in L \wedge v \in R\}$ and $LR^{-1} = \{u \mid (\exists v \in R)(uv \in L)\}$ are (deterministic) context-free. In the deterministic case, dpda's for LR and LR^{-1} may be constructed from a dpda accepting L.*

Proposition 4.9 *If $L \subseteq E^*$ is a deterministic context-free language, then $max(L) = \{u \in L \mid u < v \Rightarrow v \notin L\}$ is deterministic context-free, and a dpda accepting $max(L)$ may be constructed from a dpda accepting L.*

Proof (left as exercise 7 in section 11.3 of [Har78]) :

For $L, R \subseteq E^*$, define $div(L,R) = \{u \mid uR \subseteq L\}$. If L is deterministic context-free and R is regular then $div(L,R)$ is deterministic context-free, because $div(L,R) = C(\,C(L)\,R^{-1}\,)$.

For $L \subseteq E^*$, define $min(L) = \{u \in L \mid v < u \Rightarrow v \notin L\}$. If L is deterministic context-free then $min(L)$ is deterministic context-free. A dpda accepting $min(L)$ may be constructed from a dpda accepting L by overloading the definition of δ with $\delta(q,e,g) = \emptyset$ for every $q \in Q_F$.

Similarly, if L is deterministic context-free then $pr(L)$ is deterministic context-free, because $pr(L) = L(E^*)^{-1}$.

Finally observe that $max(L) = div\,(min\,(C(\,pr(L))), E)$, and that all operations on the right-hand side may be performed directly on dpda's. ∎

Prop. 4.9 would be false for general context-free languages. Even worse, when L is context-free, $\Psi(max(L))$ may not be semi-linear. A counter-example is shown below.

Example 4.10 *Define context-free languages as follows on a five letter alphabet:* $A = \{a^n b\, c^m \mid n \neq m\}$, $B = b\, c^*$, $C = \{c^n b\, c^m \mid n \neq m\}$, $D = a^* B^* B B\, b\, d$, $E = A B^* B\, b\, d\, e + a^* B^* b\, C B^* b\, d\, e$, *and* $L = D + E$. *Then* $max(L) = E + F$, *where* $F = \{a^n (b\, c^n)^m b\, d \mid n \geq 0 \wedge m \geq 2\}$. *Assume that* $\Psi(max(L))$ *is semi-linear, then* $\Psi(F)$ *is semi-linear. Since* $\Psi(F)$ *is the set of the integer vectors of the form* $(n, m+1, n \times m, 1, 0)$, *it follows that multiplication may be defined in Presburger's arithmetic. Due to this contradiction,* $\Psi(max(L))$ *is not semi-linear.*

The reasons why Prop. 4.9 is essential to this work may now be clarified.

Fact 4.11 *Given* $L \subseteq E^*$, $u \in pr(L)$ *and* $e \in E$, $ue \notin pr(L)$ *if and only if* $ue \in max\,(pr(L)\, e)$. *If* L *is deterministic context-free, a dpda accepting* $max\,(pr(L)\, e)$ *may be constructed from a dpda accepting* L.

Fact 4.11 may be used to reduce Prob. 4.1 to a more tractable problem. We now describe the reduction.

From Prop. 2.9, $pr(L)$ is the language of a pure Petri net with a finite set of places if and only if, for every $e \in E$, there exists a finite subset of regions $R_e \subseteq R\,(pr(L))$ such that every exit $ue \in max\,(pr(L)\, e)$ is barred by some $\eta \in R_e$. A finite net N such that $\mathcal{B}(N) = pr(L)$ may then be constructed from the set of places $\cup_e R_e$.

Assuming that L is context-free, let $R \in Reg\,(E^*)$ be a regular language such that $\Psi(R) = \Psi(pr(L))$. Let $lp(R) = \{u_1, \ldots, u_K\}$, where by convention $K = 0$ if R is free of loops, and let $br(R) = \{v_1, \ldots, v_n\}$, where $n \geq 1$. Thus, the rational regions of the language $pr(L)$ are all maps $\eta : E \to \mathbb{Q}$ such that $< \eta, \Psi(u_k) > \geq 0$ for all k $(1 \leq k \leq K)$.

For $\eta \in RR\,(pr(L))$, one may compute $min\{\eta(w) \mid w \in pr(L)\}$ from relations $min\{\eta(w) \mid w \in pr(L)\} = min\{\eta(w) \mid w \in R\} = min\{\eta(w) \mid w \in br(R)\}$. These relations show in particular that a rational region $\eta \in RR\,(pr(L))$ bars an exit $ue \in max\,(pr(L)\, e)$ if and only if $\eta(ue) < \eta(v_j)$ for all j $(1 \leq j \leq n)$.

Summing up, Prob. 4.1 can be solved for L if and only if, for all $e \in E$, there exists a finite set R_e of maps $\eta : E \to \mathbb{Q}$, satisfying $\eta(u_k) \geq 0$ for all k, such that $\eta(w) < \eta(v_j)$ for all j for some η whenever $w \in max\,(pr(L)\, e)$.

Assume now that L is deterministic context-free. Choose $e \in E$ and let $L' = (max\,(pr(L)\,e))\;e^{-1}$. Then L' is deterministic context-free. By theorems 4.5 and 4.6, $\Psi(L') = \Psi(R')$ for some regular language $R' \in Reg(E^*)$, recursively computable from L' hence from L. So $\Psi(max\,(pr(L)\,e)) = \Psi(R'e)$, and for every $w \in max\,(pr(L)\,e)$ and $\eta : E \to \mathbb{Q}$, $\eta(w) = \eta(u) + \eta(e)$ for some $u \in R'$. By definition of rational regions, $\Psi(L') = \Psi(R')$ entails $RR\,(L') = RR\,(R')$. Observing relations $RR\,(pr(L)) = RR\,(pr(L)\,e\,)$, $RR\,(pr(L)\,e\,) \subseteq RR\,(max\,(\,pr(L)\,e\,))$ (by Fact 2.2), and $RR\,(max\,(\,pr(L)\,e\,)) = RR\,(L')$, it follows that $\eta(u_k) \geq 0$ for all k entails $\eta \in RR\,(R')$.

On the whole, Prob. 4.1 reduces to a finite number of instances of the following (one instance for each $e \in E$).

Problem 4.12 *Given $e \in E$, $R' \in Reg(E^*)$, and finite sets of words $\{v_1, \ldots, v_n\}$ and $\{u_1, \ldots, u_K\}$ such that $(\forall k)(\eta(u_k) \geq 0) \Rightarrow \eta \in RR\,(R')$ for every map $\eta : E \to \mathbb{Q}$, decide whether exists and compute a finite set H of maps $\eta : E \to \mathbb{Q}$ such that:*
i) $(\forall \eta \in H)(\forall k) \quad \eta(u_k) \geq 0$, *and*
ii) $(\forall w \in R')(\exists \eta \in H)(\forall j) \quad \eta(w) + \eta(e) < \eta(v_j)$.

Proposition 4.13 *Problem 4.12 is recursively solvable in R'.*

Proof: As $RR\,(R')$ depends solely upon $\Psi(R')$, and $\eta(w) =< \eta, \Psi(w) >$ for $w \in R'$, we are free to replace R' by another regular language, following the rules $\Psi(XY) = \Psi(YX)$, $\Psi((X \cup Y)^*) = \Psi(X^*Y^*)$, and $\Psi((X^*Y)^*) = \Psi(\{\varepsilon\} \cup X^*Y^*Y)$ (see [ABB97]). We can therefore assume that $R' = R_1 + \ldots + R_m$ is a finite sum of languages $R_l = z_l\,(X_l)^*$ such that, for all l, $z_l \in E^*$ and X_l is a *finite* language over E. Under this assumption, solving Prob. 4.12 amounts to solving all the instances of this problem for $R' = R_l$ when l ranges from 1 to m. Actually, the relation $(\forall k)(\eta(u_k) \geq 0) \Rightarrow \eta \in RR\,(R_l)$ holds for all l (from Fact 2.2 and the hypotheses on $RR\,(R')$). Therefore, in the rest of the proof, we shall suppose that $R' = R_l = w_0\,\{w_1, \ldots, w_p\}^*$.

We claim that any minimal solution of Prob. 4.12 for $R' = w_0\,\{w_1, \ldots, w_p\}^*$, where solutions are compared w.r.t. the inclusion of sets, is a singleton set $H = \{\eta\}$ such that $\eta(w_l) = 0$ for $1 \leq l \leq p$. Let us establish this claim. Consider a solution $H = \{\eta_1, \ldots, \eta_t\}$ such that $t > 1$. We proceed by case analysis with regard to η_1. Suppose $\eta_1(w_l) = 0$ for every $l \geq 1$, hence $\eta_1(w) = \eta_1(w_0)$ for every $w \in R'$. Then either $\eta_1(w_0) + \eta_1(e) \geq \eta_1(v_j)$ for some j and $H - \{\eta_1\}$ is a solution, or $\eta_1(w_0) + \eta_1(e) < \eta_1(v_j)$ for all j and the singleton set $\{\eta_1\}$ is a solution. In the converse case, we may assume w.l.o.g. that $\eta_1(w_1) \neq 0$. As a solution of Prob. 4.12, H satisfies condition (i), and from the hypotheses on $RR(R')$, $\eta_1 \in RR(R')$. As w_1 is a loop of R' and by Prop. 3.4, $\eta_1(w_1) \geq 0$, hence $\eta_1(w_1) > 0$. $H - \{\eta_1\}$ is then a solution of Prob. 4.12. To show this, consider any $w \in R'$ such that $\eta_1(w) + \eta_1(e) < \eta_1(v_j)$ for all j. Let $w = w_0\,(w_1)^{n_1} \ldots (w_p)^{n_p}$. Choose $h \in \mathbb{N}$ such that $h \times \eta_1(w_1) + \eta_1(w) + \eta_1(e) \geq \eta_1(v_j)$ for some j. Thus, $\eta_1(w') + \eta_1(e) \geq \eta_1(v_j)$ for $w' = w_0\,(w_1)^h\,(w_1)^{n_1} \ldots (w_p)^{n_p}$. As $w' \in R'$ and by condition (ii), there exists $\eta_l \in H$ such that $\eta_l(w') + \eta_l(e) < \eta_l(v_j)$ for all j (hence $l \neq 1$). Now, seeing that w_1 is a loop of R', $\eta_l(w_1) \geq 0$ by Prop. 3.4, hence $\eta_l(w) \leq \eta_l(w')$ and $\eta_l(w) + \eta_l(e) < \eta_l(v_j)$ for all j. We have thus shown that $H - \{\eta_1\}$ is a solution of Prob. 4.12. The claim that any minimal solution is a singleton set $H = \{\eta\}$ such that $\eta(w_l) = 0$ for $1 \leq l \leq p$ follows by induction.

We are left with a decision problem in e, w_0, $\{w_1, \ldots, w_p\}$, $\{u_1, \ldots, u_K\}$ and $\{v_1, \ldots, v_n\}$, viz. deciding whether exists and computing a map $\eta : E \to \mathbb{Q}$ such that:
i) $(\forall k)\;\; \eta(u_k) \geq 0$,
ii) $(\forall j)\;\; \eta(w_0) + \eta(e) < \eta(v_j)$, and
iii) $(\forall l \geq 1)\; \eta(w_l) \leq 0$.
Note that every map η satisfying (i) is a rational region of $R' = w_0\,\{w_1, \ldots, w_p\}^*$ and hence satisfies $\eta(w_l) \geq 0$ for all l (as w_l is a loop of R' and by Prop. 3.4). Condition (iii) is therefore equivalent to the condition $(\forall l \geq 1)\;\; \eta(w_l) = 0$. Now, conditions (i) and (iii) determine a polyhedral cone $\mathcal{C}$ in the vector space $\mathbb{Q} < E >$. This cone may be covered by a finite family of smaller cones $\mathcal{C}_j$ $(1 \leq j \leq n)$, defined as $\{\eta \in \mathcal{C} \mid (\forall\, k \leq n)\; \eta(v_k) - \eta(v_j) \geq 0\,\}$. The proposition will obtain if one can decide for a fixed $j \leq n$ whether $\eta(w_0) + \eta(e) < \eta(v_j)$ for some $\eta \in \mathcal{C}_j$. Let j be fixed. By the Farkas-Minkowski-Weyl theorem, the polyhedral cone $\mathcal{C}_j$ is finally generated, and it is equal to $\{\, \sum_{i=1}^m\; q_i \times \eta_i \mid q_i \in \mathbb{Q}_+\}$ for some finite set of maps $\{\eta_1, \ldots, \eta_m\}$, computable from the data j, $\{u_1, \ldots, u_K\}$, $\{v_1, \ldots, v_n\}$, and $\{w_1, \ldots, w_p\}$. Clearly, $\eta(w_0) + \eta(e) - \eta(v_j) < 0$ for some $\eta \in \mathcal{C}_j$ if and only if $\eta_i(w_0) + \eta_i(e) - \eta_i(v_j) < 0$ for some $i \leq m$. Thus the proof is

complete. ∎

The net result which has been established in this section is the following.

Theorem 4.14 *Given a deterministic context-free language L , one may decide whether exists and then construct a finite pure Petri net N such that $\mathcal{B}(N) = pr\,(L)$.*

5 Conclusion

Let us briefly indicate possible continuations of this work. Considering pure Petri nets exclusively is a clear limitation. Our experience of bounded nets synthesis suggests that getting an extension to arbitrary Petri nets demands few adaptations, similar to those proposed in [BD96]. From another side, we note that the technical development of section 4 is not specific for context-free languages and may be applied to any class of languages with semi-linear commutative images. This invites us to try synthesizing nets from languages of MSCs (see [MR97]), which was suggested to us by B. Caillaud. The main difficulty is to delimit first an adequate class of "deterministic" MSCs.

References

[ABB97] Autebert, J.M., Berstel, J., and Boasson, L., *Context-Free Languages and Pushdown Automata*. In vol. 1 of "Handbook of Formal Languages", G. Rozenberg and A. Salomaa eds., Springer-Verlag (1997) 111–174

[BBD95] Badouel, E., Bernardinello, L. and Darondeau, Ph., *Polynomial algorithms for the synthesis of bounded nets*, Proceedings Caap 95, Lecture Notes in Computer Science 915 (1995) 647–679.

[BD96] Badouel, E. and Darondeau, Ph., *On the Synthesis of General Petri Nets*, Inria Research Report no 3025 (1996).

[BDPV96] Bernardinello, L., De Michelis, G., Petruni, K., and Vigna, S., *On the Synchronic Structure of Transitions Systems*. In "Structures in Concurrency Theory", J. Desel ed., Springer-Verlag (1996) 11–31.

[BH96] BOUAJJANI, A., and HABERMEHL, P., *Constrained Properties, Semilinear Systems, and Petri Nets.* Proceedings of Concur 95, Lecture Notes in Computer Science 1119 (1995) 481–497.

[Ca97] CAILLAUD, B., SYNET : *un outil de synthèse de réseaux de Petri bornés, applications* Inria Research Report no 3155 (1997).

[CKLY95] CORTADELLA, J., KISHINEVSKY, M., LAVAGNO, L., AND YAKOVLEV, A., *Synthesizing Petri Nets from State-Based Models.* Proceedings of ICCAD'95 (1995) 164–171.

[DR96] DESEL, J., AND REISIG, W., *The Synthesis Problem of Petri Nets.* Acta Informatica, vol. 33 (1996) 297–315.

[DS93] DROSTE, M., and SHORTT, R.M., *Petri Nets and Automata with Concurrency Relations – an Adjunction.* in "Semantics of Programming Languages and Model Theory", M. Droste and Y. Gurevich eds(1993) 69–87.

[ER90a] EHRENFEUCHT, A., and ROZENBERG, G., *Partial (Set) 2-Structures;* Part I: *Basic Notions and the Representation Problem.* Acta Informatica, vol. 27 (1990) 315–342.

[ER90b] EHRENFEUCHT, A., and ROZENBERG, G., *Partial (Set) 2-Structures;* Part II: *State Spaces of Concurrent Systems.* Acta Informatica, vol. 27 (1990) 343–368.

[Har78] HARRISSON, M., *Introduction to Formal Language Theory.* Addison-Wesley (1978).

[Muk92] MUKUND, M., *Petri Nets and Step Transition Systems.* International Journal of Foundations of Computer Science, vol. 3 no. 4 (1992) 443–478.

[MR97] MAUW, S. and RENIERS, M.A., *High-level Message Sequence Charts.* Proceedings of the Eighth SDL Forum, North-Holland (1997).

[Ra78] RACKOFF, C., *The Covering and Boundedness Problems for Vector Addition Systems.* Theoretical Computer Science, vol. 6 (1978) 223–231.

[RY86] ROSIER, L.E., and YEN, H.C., *A Multiparameter Analysis of the Boundedness Problem for Vector Addition Systems.* Journal of Computer and System Science, vol. 32 (1986) 105–135.

[Sch86] SCHRIJVER, A., *Theory of Linear and Integer Programming.* John Wiley (1986).

Unit´e de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unit´e de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unit´e de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unit´e de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unit´e de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
http://www.inria.fr
ISSN 0249-6399