RAIRO
INFORMATIQUE THÉORIQUE

MATTHIAS JANTZEN

On the hierarchy of Petri net languages

RAIRO – Informatique théorique, tome 13, nº 1 (1979), p. 19-30.
<http://www.numdam.org/item?id=ITA_1979__13_1_19_0>

© AFCET, 1979, tous droits réservés.

L'accès aux archives de la revue « RAIRO – Informatique théorique » implique l'accord avec les conditions générales d'utilisation (http://www.numdam.org/legal.php). Toute utilisation commerciale ou impression systématique est constitutive d'une infraction pénale. Toute copie ou impression de ce fichier doit contenir la présente mention de copyright.

NUMDAM
Article numérisé dans le cadre du programme
Numérisation de documents anciens mathématiques
http://www.numdam.org/

R.A.I.R.O. Informatique théorique/Theoretical Informatics
(vol. 13, n° 1, 1979, p. 19 à 30)

ON THE HIERARCHY OF PETRI NET LANGUAGES (*)

by Matthias JANTZEN [1]

Communicated by W. BRAUER

Abstract. – *We prove* $\mathscr{M}_\cap(D_1'^*) \subsetneq \hat{\mathscr{M}}(D_1'^*)$, *where* $D_1'^*$ *is the one-sided Dyck language, and discuss some old and new results concerning Petri net languages. The above result shows that Petri nets without* λ*-labeled transitions are less powerful than general nets as regards their firing sequences since the class* $\mathscr{L}_0^\lambda$ *of general Petri net languages (Hack [13]) is identical with* $\hat{\mathscr{M}}_\cap(D_1'^*)$, *and the class* $\mathscr{C}\mathscr{S}\mathscr{S}$ *of computation sequence sets (Peterson [21]) equals* $\mathscr{M}_\cap(D_1'^*)$.

INTRODUCTION

The reader is supposed to be familiar with the notion of Petri nets and with formal language theory. For exact definitions of Petri net languages, *see* Hack [13] and Peterson [21]. AFL theory, *see* Ginsburg [8], is used extensively.

For readers who like to read this note without going too much into details some informal explanation of abbreviations follows:

$\mathscr{L}_0^\lambda$ denotes the family of languages each of which is a set of firing sequences leading some arbitrary labeled Petri net from a start marking to a final marking;

$\mathscr{L}_0$ denotes the family of languages each of which is a set of firing sequences leading some arbitrary but λ-free labeled Petri net from a start marking to a different final marking;

$\mathscr{C}\mathscr{S}\mathscr{S}$ is defined like $\mathscr{L}_0$ but without the restriction that the final marking is different from the start marking;

$\mathscr{L}^\lambda$ denotes the family of languages each of which is a set of firing sequences leading some arbitrary labeled Petri net from a start marking to some other marking;

$\mathscr{L}$ is defined like $\mathscr{L}^\lambda$ without using λ-labels.

(*) Received March 1978.
[1] Fachbereich Informatik, Universität Hamburg, Hamburg.

© Bordas-Dunod.

$\mathcal{S}_z$ denotes the family of Szilard languages (Salomaa [24]) which are also known as derivation languages of context-free grammars (Penttonen [22]) or associate languages (Moriya [19]).

Note: Szilard languages do not contain the empty word λ ! $\mathcal{M}(\mathcal{L})$ [$\hat{\mathcal{M}}(\mathcal{L})$, $\mathcal{U}(\mathcal{L})$, $\hat{\mathcal{U}}(\mathcal{L})$ resp.] denotes the least trio (least full trio, least semi-AFL, least full semi-AFL resp.) containing $\mathcal{L}$.

For $\mathcal{O}$ being $\mathcal{M}$ ($\hat{\mathcal{M}}$, $\mathcal{U}$, $\hat{\mathcal{U}}$ resp.) $\mathcal{O}_\cap(\mathcal{L})$ denotes the least intersection-closed family containing $\mathcal{L}$ and closed under the operations which define $\mathcal{O}$.

$\mathcal{R}$ (resp. $\mathcal{RE}$) denotes the family of regular (resp. recursively enumerable) sets.

The shuffle operation on languages L_1 and L_2 is defined by:

$$\text{Shuf}(L_1, L_2) := \{ w = x_1 y_1 \ldots x_n y_n \mid x_1 x_2 \ldots x_n \in L_1, y_1 y_2 \ldots y_n \in L_2 \}.$$

The operation perm (L) denotes the commutative closure of the language L.

For families of languages $\mathcal{L}_1$, $\mathcal{L}_2$ we use the following notations

$$\mathcal{L}_1 \vee \mathcal{L}_2 := \{ L \mid L = L_1 \cup L_2 \text{ for some } L_1 \in \mathcal{L}_1, L_2 \in \mathcal{L}_2 \},$$

$$\mathcal{L}_1 \wedge \mathcal{L}_2 := \{ L \mid L = L_1 \cap L_2 \text{ for some } L_1 \in \mathcal{L}_1, L_2 \in \mathcal{L}_2 \},$$

$$\text{Shuf}(\mathcal{L}_1, \mathcal{L}_2) := \{ L \mid L = \text{Shuf}(L_1, L_2), L_1 \in \mathcal{L}_1, L_2 \in \mathcal{L}_2 \},$$

$$\bigwedge \mathcal{L} := \{ L \mid \text{there exists } n \geqq 1, L_1, \ldots, L_n \in \mathcal{L} \text{ such that } L = L_1 \cap L_2 \cap \ldots \cap L_n \}.$$

$$\mathcal{H}(\mathcal{L}) := \{ L \mid L = h(L') \text{ for some nonerasing homomorphism } h \text{ and some } L' \in \mathcal{L} \},$$

$$\hat{\mathcal{H}}(\mathcal{L}) := \{ L \mid L = h(L') \text{ for some arbitrary homomorphism } h \text{ and some } L' \in \mathcal{L} \}.$$

$$\mathcal{H}^{-1}(\mathcal{L}) := \{ L \mid L = h^{-1}(L') \text{ for some homomorphism } h \text{ and some } L' \in \mathcal{L} \}.$$

$$\text{perm}(\mathcal{L}) := \{ L \mid L = \text{perm}(L') \text{ for some } L' \in \mathcal{L}. \}.$$

SOME SIMPLE FACTS ON PETRI NETS

A number of proofs have been published to exhibit several closure properties for Petri net languages. The proofs can be found in Höpner [14], Hack [13] and Peterson [21]. We summarize the results in proposition 1:

PROPOSITION 1: *$\mathcal{CSS}$ and $\mathcal{L}_0^\lambda$ are closed with respect to union, concatenation, intersection, shuffle, substitution by λ-free regular sets, inverse homomorphism and* *limited erasing. $\mathscr{C}\mathscr{S}\mathscr{S}$ and $\mathscr{L}_0^\lambda$ contain all the regular sets, whereas $\mathscr{L}_0$ contains only the λ-free regular sets.*

Of course these operations are not independent from each other.

The characterization $\mathscr{L}_0^\lambda = \mathscr{H}(\mathscr{S}_z \wedge \mathscr{R})$ is more or less folklore because of the obvious connections between Petri net languages and derivation languages of matrix grammars. *See* Nash [20], van Leeuwen [18], Crespi-Reghizzi and Mandrioli [4, 6], Höpner [14], Salomaa [24], and many others cited there.

The equality $\mathscr{L}_0 = \mathscr{H}(\mathscr{S}_z \wedge \mathscr{R})$ has been proven by Crespi-Reghizzi and Mandrioli [6] though it is not explicitly stated there.

Using the equations above, proposition 1 and AFL theory we can characterize the Petri net languages in the following way:

PROPOSITION 2:

$$\mathscr{L}_0 = \mathscr{M}(\mathscr{S}_z) = \mathscr{U}(\mathscr{S}_z) = \mathscr{H}(\mathscr{H}^{-1}(\mathscr{S}_z) \wedge \mathscr{R}),$$
$$\mathscr{L}_0^\lambda = \hat{\mathscr{M}}(\mathscr{S}_z) = \hat{\mathscr{U}}(\mathscr{S}_z) = \hat{\mathscr{H}}(\mathscr{H}^{-1}(\mathscr{S}_z) \wedge \mathscr{R}),$$
$$\mathscr{C}\mathscr{S}\mathscr{S} = \mathscr{M}(\mathscr{S}_z \vee \{\{\lambda\}\}).$$

This characterization, as we whall see, is not optimal, since the family $\mathscr{S}\mathscr{Z}$ which generates $\mathscr{L}_0$, $\mathscr{L}_0^\lambda$ and $\mathscr{C}\mathscr{S}\mathscr{S}$ via *a-transductions can be replaced by a smaller family.*

It is easy to see that each Szilard language $L \in \mathscr{S}_z$ is a finite intersection of one-counter languages. A first hint in this direction has been given by Brauer [3], and in [6] it has been shown that certain Petri net languages can be written as finite intersections of deterministic context-free languages. We state this as:

PROPOSITION 3: *If $L \in \mathscr{S}_z$, then there exist $n \geq 1$ and deterministic one-conter languages $K_1, \ldots, K_n \in \mathscr{M}(D_1'^*)$ such that $L = K_1 \cap \ldots \cap K_n$ holds.*

Proof: The proof is obvious: each K_1 is a language accepted by an automaton which counts the number of occurences of the nonterminal A_i in the sentential form of the derivation in progress.

If the context-free grammar has m nonterminals then at most m one-counter languages are needed. Moreover, if the number of occurences of the nonterminal A_i within each sentential form of a terminating derivation is bounded by some constant, then the corresponding language K_i is a regular set. This shows that the integer n in proposition 3 can be chosen equal to the number of unbounded nonterminals of the grammar generating L.

Note: This does not mean that n equals the number of simultaneously unbounded nonterminals of that grammar There are examples where no nonterminal is bounded but only one at a time may occur arbitrarily often.

THE HIERARCHY

To obtain a simple and obvious characterization for Petri net languages we define a special kind of k-counter language which is the k-fold shuffle of the one-counter Dyck language.

DEFINITION: Let C_1^i denote the semi-Dyck language over the pair of brackets $\{a_i, \overline{a}_i\}$.

Then C_k is recursively defined by:

$$C_1 := C_1^1,$$
$$C_k := \mathrm{Shuf}(C_{k-1}, C_1^k).$$

Using AFL theory we easily show:

THEOREM 1:
$$\mathscr{L}_0^\lambda = \hat{\mathscr{M}}(\{C_i \mid i \geqq 1\}) = \hat{\mathscr{M}}_\cap(D_1'^*) = \hat{\mathscr{U}}_\cap(D_1'^*),$$
$$\mathscr{C}\mathscr{S}\mathscr{S} = \mathscr{M}(\{C_i \mid i \geqq 1\}) = \mathscr{M}_\cap(D_1'^*) = \mathscr{U}_\cap(D_1'^*).$$

Proof: Since $\mathscr{L}_0^\lambda = \hat{\mathscr{H}}(\mathscr{C}\mathscr{S}\mathscr{S}) = \hat{\mathscr{H}}(\mathscr{L}_0)$ (*see* proposition 2 and the definitions) we only have to show

$$\mathscr{C}\mathscr{S}\mathscr{S} = \mathscr{M}(\{C_i \mid i \geqq 1\}) = \mathscr{M}_\cap(D_1'^*).$$

The equality $\mathscr{M}_\cap(D_1'^*) = \mathscr{U}_\cap(D_1'^*)$ [resp. $\hat{\mathscr{M}}_\cap(D_1'^*) = \hat{\mathscr{U}}_\cap(D_1'^*)$] follows from proposition 1 and AFL theory.

Since

$$\mathscr{M}_\cap(\mathscr{M}(D_1'^*)) = \mathscr{M}(\wedge\mathscr{M}(D_1'^*)) = \mathscr{M}(\wedge\mathscr{M}(D_1'^*))$$

(*see* Ginsburg [8], prop. 3.6.1) and $\mathscr{S}_z \subseteq \wedge\mathscr{M}(D_1'^*)$ (by prop. 3) we get

$$\mathscr{M}(\mathscr{S}_z) \subseteq \mathscr{M}(\wedge\mathscr{M}(D_1'^*)) = \mathscr{M}_\Psi(\mathscr{M}(D_1'^*)) = \mathscr{M}(D_1'^*)$$

thus by proposition 2:

$$\mathscr{L}_0 \subseteq \mathscr{M}_\cap(D_1'^*) \quad \text{and} \quad \mathscr{C}\mathscr{S}\mathscr{S} \subseteq \mathscr{M}_\cap(D_1'^*).$$

Since $\mathscr{C}\mathscr{S}\mathscr{S}$ contains the language $D_1'^*$ (*see* [13, 17]) and is closed with respect to λ-free a-transductions (*see* prop. 1 and 2) we get:

$$\mathscr{C}\mathscr{S}\mathscr{S} = \mathscr{M}_\cap(D_1'^*).$$

To verify $\mathscr{M}(\{C_i \mid i \geqq 1\}) = \mathscr{M}_\cap(D_1'^*)$ we first observe that for each $k \geqq 1$ the language C_k is a member of $\mathscr{M}_\cap(D_1'^*)$ since this family contains $C_1 = D_1'^*$ and is closed with respect to shuffle.

Note: A trio is intersection-closed if and only if it is closed with respect to shuffle (exercice 5.5.6 in [8] or corollary 3 in [7]).

Thus we have $\mathcal{M}(\{C_i \mid i \geqq 1\}) \subseteq \mathcal{M}_\cap(D_1'^*)$.

Now suppose

$$L \in \mathcal{M}_\cap(D_1'^*) = \mathcal{M}(\bigwedge \mathcal{M}(D_1'^*)),$$

then by definition of $\bigwedge \mathcal{L}$ there exists $k \geqq 1$ such that

$$L \in \mathcal{M}(\mathcal{M}(C_1^1) \wedge \ldots \wedge \mathcal{M}(C_1^k)).$$

Using proposition 5.1.1 and theorem 5.5.1(*d*) in [8] we get

$$\mathcal{H}(\mathcal{M}(C_{k-1}) \wedge \mathcal{M}(C_1^k)) = \mathcal{H}(\mathcal{U}(C_{k-1}) \wedge \mathcal{U}(C_1^k))$$
$$= \mathcal{U}(\mathrm{Shuf}(C_{k-1}, C_1^k)) = \mathcal{U}(C_k) = \mathcal{M}(C_k).$$

By induction we obtain

$$\mathcal{H}(\mathcal{M}(C_1) \wedge \ldots \wedge \mathcal{M}(C_1^k)) =$$

$$\mathcal{H}(\mathcal{H}(\mathcal{M}(C_1^1) \wedge \ldots \wedge \mathcal{M}(C_1^{k-1})) \wedge \mathcal{M}(C_1^k)) = \mathcal{H}(\mathcal{M}(C_{k-1}) \wedge \mathcal{M}(C_1^k)) = \mathcal{M}(C_k).$$

Thus we have shown $L \in \mathcal{M}(C_k)$ which proves $\mathcal{M}(\{C_i \mid i \geqq 1\}) = \mathcal{M}_\cap(D_1'^*)$, and the proof of theorem 1 is finished.

Theorem 1 gives a similar characterization for $\mathcal{L}_0^\lambda$ as theorem 5.6 in [13]. Whereas Hack uses $D_1'^*$ and the regular sets as basis and the operations homomorphism, shuffle and intersection, we use $D_1'^*$ as basis and the following operations: homomorphism, inverse homomorphism, intersection with regular sets and either shuffle or intersection.

Using ideas of Greibach [10] one can show that for each $k \geqq 1$ the language

$$L_k := \{a_1^{n_1} \ldots a_k^{n_k} b a_k^{n_k} \ldots a_1^{n_1} \mid n_i \geqq 0\}$$

is not a member of the family $\mathcal{M}(C_{k-1})$ (*see* example 4.5.2 in [8]).

But obviously $L_k \in \mathcal{M}(C_k)$, thus there exists an infinite hierarchy of families of Petri net languages

$$\mathcal{M}(C_1) \subsetneqq \mathcal{M}(C_2) \subsetneqq \ldots \subsetneqq \mathcal{M}(C_k) \subsetneqq \mathcal{M}(C_{k+1}) \subsetneqq \ldots$$

Since $\mathcal{M}_\cap(D_1'^*) = \bigcup_{i \geqq 1} \mathcal{M}(C_i)$ (by the definition of $\bigwedge \mathcal{L}$ and previous results) we apply theorem 5.1.2 in Ginsburg [8] which shows that $\mathcal{M}(D_1'^*) = \mathcal{CSS}$ is not a principal semi-AFL.

REMARK: With the method of counting the number of reachable configurations Peterson [21] proved that PAL: $=\{ww^R \mid w \in \{0, 1\}^*\}$ is not a member of $\mathscr{C}\mathscr{S}\mathscr{S}$.

Now if the reachability problem for Petri nets is decidable as announced by Tenney and Sacerdote [23]:

(i) PAL is not a member of $\mathscr{L}_0^\lambda$;

(ii) $\mathscr{C}\mathscr{S}\mathscr{S}$ is not closed with respect to Kleene star.

Proof: Suppose PAL $\in \mathscr{L}_0^\lambda$ then $\mathscr{L}_0^\lambda = \mathscr{R}\mathscr{E}$, since $\mathscr{R}\mathscr{E}$ is the least intersection-closed full semi-AFL containing PAL (*see* [1]).

But this would contradict the result of Tenney and Sacerdote.

Suppose $\mathscr{C}\mathscr{S}\mathscr{S}$ to be star-closed, then $\mathscr{L}_0^\lambda$ would be star-closed too and thus a full AFL. But then again $\mathscr{L}_0^\lambda = \mathscr{R}\mathscr{E}$ would yield the contradiction since $\mathscr{R}\mathscr{E}$ is the least intersection closed full AFL containing the language $\{a^n b^n \mid n \geqq 0\}$ which is in $\mathscr{L}_0^\lambda$ (*see* [1]).

Unfortunately there is no direct proof of (i) or (ii) which does not use the result of Tenney and Sacerdote.

Note: Theorem 9.8 in [13], stating that the language $Q_0 = (D_1'^* \cdot \{0\})^* \cdot D_1'^*$ is not a member of $\mathscr{L}_0^\lambda$, is based on an incorrect proof as observed by Valk [26]!

THE NONCLOSURE OF $\mathscr{C}\mathscr{S}\mathscr{S}$ UNDER ERASING

There are two problems which are to be solved:

PROBLEM 1: Does or does not hold

$$\hat{\mathscr{M}}_\cap(D_1'^*) = \mathscr{M}_\cap(D_1'^*)\ ?$$

PROBLEM 2: Does or does not hold

$$\hat{\mathscr{M}}(C_k) = \hat{\mathscr{M}}(C_{k+1})\ ?$$

Before we solve the first one, let us shortly discuss the second one.

Of course $\hat{\mathscr{M}}(C_1) \nsubseteq \hat{\mathscr{M}}(C_2)$ since C_2 is not context-free and $\hat{\mathscr{M}}(C_1)$ contains only context-free languages. We will even see that $\hat{\mathscr{M}}(C_2)$ contains a language BIN such that ψ(BIN) is not a semilinear set (ψ denotes the usual Parikh mapping). It can be shown that $\hat{\mathscr{M}}(C_k) = \hat{\mathscr{M}}(C_{k+1})$ implies $\hat{\mathscr{M}}_\cap(C_1) = \hat{\mathscr{M}}(C_k)$, thus the family $\mathscr{L}_0^\lambda$ would be a principal semi-AFL which would be surprising. I conjecture that $\hat{\mathscr{M}}(C_k) \nsubseteq \hat{\mathscr{M}}(C_{k+1})$ holds for each $k \geqq 1$.

Compare this conjecture with results by Latteux [17] who has shown that $\hat{\mathscr{M}}_\cap(D_1^*) = \hat{\mathscr{M}}(\{O_n \mid n \geqq 1\})$ is not principal. The language O_n is defined similar to our language C_n by:

$$O_1 := \mathrm{perm}(\{a_1 \bar{a}_1\}^*) = D_1^*,$$

which is the two-sided Dyck language, and

$$O_n := \mathrm{Shuff}(O_{n-1}, \mathrm{perm}(\{a_n \bar{a}_n\}^*)).$$

To solve problem 1 we define the language BIN which will be the counterexample to show the desired inequality:

DEFINITION:

$$\mathrm{BIN} := \{ wa^k \mid w \in \{0, 1\}^*, 0 \leqq k \leqq n(w) \},$$

where $n(w)$ denotes the integer represented by w as a binary number. Convention: $n(\lambda) := 0$.

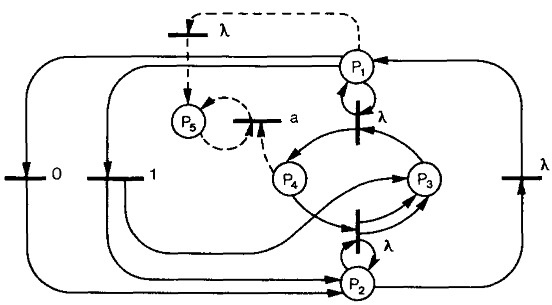
We first prove :

THEOREM 2:

$$\mathrm{BIN} \in \hat{\mathscr{M}}(C_2).$$

Proof: Let N be the following Petri net (*fig.*) including the place p_5, the dotted arcs and the transition labeled with the symbol "a".

Let N′ be the net N without the dotted lines.



We will verify that Petri net N accepts the language BIN, i.e. each firing sequence beginning with the start marking (1, 0, 0, 0, 0) spells out a word from BIN and conversely each element of BIN can be accepted in that way.

Let $|p_i|$ denote the number of tokens at place p_i. By induction we first prove a basic property of the net N':

FACT: After $w \in \{0, 1\}^*$ has been accepted by the net N' starting with the marking (1, 0, 0, 0) then $|p_3| + |p_4| \leqq n(w)$ holds true for the marking which has been reached.

Basic step: For $w \in \{0\}^*$ trivially $|p_3| + |p_4| = 0 = n(w)$.

For $w \in \{0\}^* \cdot \{1\}$ obviously $|p_3| + |p_4| = 1 = n(w)$.

Induction step: Assume the fact to be true for all $w \in \{0, 1\}^*$ of length m and suppose the net N' has already accepted such a word w. Then either p_2 or p_1 has one token. In order to accept a word $w' \in \{0, 1\}^*$ of length $m+1$ we have to reach a situation where p_1 has the token. This can be done using the λ-transitions. Suppose the situation reached so far is described by the marking $(1, 0, x, y)$. By our assumption $x + y \leqq n(w)$ holds true.

Now two cases are of interest:

Case 1: We use the transition labeled with "0". This means we accept $w' = w0$. In this case, not using one of the λ-transitions, we directly reach the marking $(0, 1, x, y)$. Still leaving the token on p_2 we can only reach a marking $(0, 1, x', y')$ where

$$0 \leqq y' \leqq y \qquad \text{and} \qquad x' = 2(y - y') + x.$$

Now we can shift the token from p_2 to p_1 and then we may reach some marking $(1, 0, x'', y'')$ where

$$x'' = x' - z \qquad \text{and} \qquad y'' = y' + z$$

for some $0 \leqq z \leqq x'$. Thus

$$x'' + y'' = x' + y' = 2y - 2y' + x + y' = 2y + x - y' \leqq 2y + x.$$

Since $x + y \leqq n(w)$ implies $y \leqq n(w)$ we get $2y + x \leqq 2n(w)$. Thus finally

$$x'' + y'' \leqq 2n(w) = n(w0) = n(w')$$

This proves the induction step restricted to case 1.

Case 2: Suppose we use the transition labeled with "1". This means we accept $w' = w1$. Then $n(w') = 2n(w) + 1$ and the same considerations as in case 1 show that in this case $|p_3| + |p_4| = x'' + y'' + 1$, so that $|p_3| + |p_4| \leqq n(w')$. Therefore we have proved the fact for all works $w \in \{0, 1\}^*$.

Now, looking at the net N we can easily verify that the transition labeled with "a" can be used at most $|p_4|$ times, thus at most $n(w)$ times if w has been accepted and p_5 has got the token from p_1. This shows that each word accepted by the net N is in BIN.

Conversely, we have to show that each word in BIN can be accepted by the net. This is easily seen in the following way: First of all each word $w \in \{0, 1\}^*$ can be accepted by the net. Moreover, if each λ-transition is used as often as possible until w has been accepted and p_1 has one token, then $|p_4| = n(w)$. Of course the transition labeled with "a" may now be used k times, where $0 \leqq k \leqq n(w)$ is arbitrary.

This shows that the net N accepts exactly the language BIN without using final markings. Of course we could add some more λ-transitions to clear all places if we liked.

Since the net has only the two unbounded places p_3 and p_4 we have the result $\mathrm{BIN} \in \hat{\mathcal{M}}(C_2)$.

The language BIN is similar to a language used by Greibach [11] to show that linear-time is more powerful than real-time recognition by multicounter machines. We now show $\mathrm{BIN} \notin \mathcal{M}_\cap(D_1'^*)$. The proof uses Dedekind's idea of distributing more than n pieces into less than n boxes.

THEOREM 3:

$$\mathrm{BIN} \notin \mathcal{M}_\cap(D_1'^*).$$

Proof: Assume $\mathrm{BIN} \in \mathcal{M}_\cap(D_1'^*)$, then there exists a net N with k places which accepts BIN not using λ-transitions. We will derive a contradition.

Let m be the maximal number of tokens which can be added to the net in firing one transition. Let m_0 be the total number of tokens in the net at the beginning. Then after n steps, each step being the firing of one transition, there are at most $m_0 + n \cdot m$ tokens in the net. Distributing up to that many tokens over the k places of the net yields at most

$$\sum_{i=0}^{m_0+n\cdot m} \binom{i+k-1}{k-1} = \binom{m_0+n\cdot m+k}{k} \leqq (m_0+n\cdot m+1)^k,$$

different markings which are reachable within n steps!

Note: $\binom{i+k-1}{k-1}$ equals the number of different possibilities to distribute exactly i indistinguishable objects into k different boxes.

Of course the upper bound obtained above is quite bad, on the other hand it is good enough for our purpose.

Now, there are 2^n different words $w \in \{0, 1\}^*$ of length n. Each word represents an integer $n(w)$, where $0 \leqq n(w) \leqq 2^n - 1$. Let $w_0, w_1, \ldots, w_{2^n-1}$ be the ordering of all words of length n such that $n(w_i)$ equals i for $i = 0, 1, \ldots, 2^n - 1$.

For each word w_i there must exist at least one marking M_i of the net which is reachable while accepting w_i and from which it is possible to accept a^i, since the word $w_i a^i$ is in BIN. We shall see that all these markings $M_0, \ldots, M_{2^n-1}$ must be different. But this then is a contradiction, because there are at most $(m_0 + n \cdot m)^k$ different markings reachable within n steps, which for n big enough is strictly less than 2^n.

Now suppose for some $i \neq j$ we would have $M_i = M_j$. Then we could reach this marking accepting the word $w_{\min(i,j)}$, and starting with this marking we could accept the word $a^{\max(i,j)}$, thus we could accept the word $w_{\min(i,j)}\, a^{\max(i,j)}$ which is not a member of BIN. The contradiction is met and we have shown that no Petri net without λ-labeled transitions can accept the language BIN.

COROLLARY 1:

$$\mathcal{M}_{\cap}(D_1'^{*}) \subsetneqq \hat{\mathcal{M}}_{\cap}(D_1'^{*}) \qquad \textit{and} \qquad \mathcal{CSS} \subsetneqq \mathcal{L}_0^{\lambda}.$$

Proof: Trivial, using theorem 2, theorem 3 and the propositions.

COROLLARY 2:

$$\mathcal{L} \subsetneqq \mathcal{L}^{\lambda}.$$

Proof: Since BIN is in $\mathcal{L}^{\lambda}$ and the proof of theorem 3 works for nets with or without final markings.

REMARK: When writing this note, I have been told that Greibach [12] has shown $\mathcal{CSS} = \mathcal{M}_{\cap}(D_1'^{*}) \subsetneqq \hat{\mathcal{M}}_{\cap}(D_1'^{*})$ independently.

Vidal Naquet [27] has proved corollary 2 using a different method which was not applicable for nets with final markings.

Corollary 1 solves the open problem of Hack [13] whether λ-labels can be eliminated in arbitrary Petri nets.

The well known language $L_{St} := \{ a^n b^m \mid 1 \leqq n,\ 1 \leqq m \leqq 2^n \}$, the Parikh image of which is not a semi-linear set (Stotzkij [25]) now simply can be shown to be a member of $\hat{\mathcal{M}}(C_2)$ since

$$L_{St} = h(\mathrm{BIN} \cap \{1\}^{+} \{a\}^{*}) \cdot \{b\},$$

where h is the coding defined by $h(1) := a$ and $h(a) := b$.

Surprisingly enough it can be shown that this language can be accepted by a certain net without λ-labeled transitions. We state this as:

PROPOSITION 4:

$$L_{St} \in \mathcal{M}(C_3).$$

The proof can be found in [16].

Careful inspection of the net for this language L_{St} which in fact is a modified version of the net for BIN shows that the Parikh image of the set of all reachable markings is not a semi-linear set.

Using results of van Leeuwen [18] we see that Petri nets with three unbounded places are strictly more powerful than vector addition systems of dimension 3. This follows since van Leeuwen [18], theorem 6.4, has proved that for each vector addition system of dimension 3 the Parikh image of the set of reachable points is a semi-linear set.

Looking at the proof of theorem 3 one can check that the method used here doesn't work if the language under consideration is bounded, i. e. if $L \subseteq \{w_1\}^* \ldots \{w_m\}^*$ for a fixed collection of words $w_1, \ldots, w_m$. In this case there are at most $D(n; \lg(w_1), \ldots, \lg(w_m))$ different words of length n, where the "denumerant" $D(n; a_1, \ldots, a_m)$ equals the number of different points $x := (x_1, \ldots, x_m)$ for which

$$a_1 \cdot x_1 + a_2 \cdot x_2 + \ldots + a_m \cdot x_m = n \quad \text{holds true.}$$

Using results of Bell [2] it can be shown that for all $n \geqq 1$ $D(n; a_1, \ldots, a_m) \leqq c \, . \, n^{m-1}$ for some appropriate constant c depending only on $a_1, \ldots, a_m$.

Thus the number of words of a certain length n and the number of different markings reachable within n steps both are bounded by some polynomial in n.

These suggestions give rise to the following:

Conjecture: Each bounded language $L \in \hat{\mathscr{M}}_\cap(D_1'^*)$ is in fact a member of $\mathscr{M}_\cap(D_1'^*)$.

REFERENCES

1. B. S. Baker and R. V. Book, *Reversal-Bounded Multipushdown Machines*, J. Comp. Syst. Sc., Vol. 8, 1974, pp. 315-332.
2. E. T. Bell, *Interpolated Denumerants and Lambert Series*, Amer. J. Math., Vol. 65, 1943, pp. 382-386.
3. W. Brauer, *On Grammatical Complexity of Context-Free Languages*, M.F.C.S. Proceedings of Symposium and Summerschool, High Tatras, 1973, pp. 191-196.
4. S. Crespi-Reghizzi and D. Mandrioli, *Petri Nets and Commutative Grammars*, Technical Report 74-5, Istituto Elettronica del Politecnico di Milano, 1974.
5. S. Crespi-Reghizzi and D. Mandrioli, *A Decidability Theorem for a Class of Vector-Addition Systems*, Information Processing Letters, Vol. 3, 1975, pp. 78-80.
6. S. Crespi-Reghizzi and D. Mandrioli, *Petri Nets and Szilard Languages*, Information and Control, Vol. 33, 1977, pp. 177-192.
7. S. Ginsburg and S. A. Greibach, *Principal AFL*, J. Comp. Syst. Sc., Vol. 4, 1970, pp. 308-338.
8. S. Ginsburg, *Algebraic and Automata-Theoretic Properties of Formal Languages*, North-Holland Publishing Company, 1975.
9. S. Ginsburg, J. Goldstine and S. A. Greibach, *Some Uniformly Erasable Families of Languages*, Theoretical Computer Science, Vol. 2, 1976, pp. 29-44.
10. S. A. Greibach, *An Infinite Hierarchy of Context-Free Languages*, J. Assoc. Computing Machinery, Vol. 16, 1969, pp. 91-106.
11. S. A. Greibach, *Remarks on the Complexity of Nondeterministic Counter Languages*, Theoretical Computer Science, Vol. 1, 1976, pp. 269-288.
12. S. A. Greibach, *Remarks on Blind and Partially Blind One-Way Multicounter Machines*, Submitted for Publication, 1978.

13. M. HACK, *Petri Net Languages*, Computation Structures Group Memo 124, Project MAC, M.I.T., 1975.
14. M. HÖPNER, *Über den Zusammenhang von Szilardsprachen und Matrixgrammatiken*, Technical Report IFI-HH-B-12/74, Univ. Hamburg, 1974.
15. M. HÖPNER and M. OPP, *About Three Equational Classes of Languages Built up by Shuffle Operations*, Lecture Notes in Computer Science, Springer, Vol. 45, 1976, pp. 337-344.
16. M. JANTZEN, *Eigenschaften von Petrinetzsprachen*, Research Report, Univ. Hamburg, 1978.
17. M. LATTEUX, *Cônes rationnels commutativement clos*, R.A.I.R.O., Informatique théorique, Vol. 11, 1977, pp. 29-51.
18. J. VAN LEEUWEN, *A Partial Solution to the Reachibility-Problem for Vector Addition Systems*, Proceedings of the 6th annual A.C.M. Symposium on Theory of Computing, 1974, pp. 303-309.
19. E. MORIYA, *Associate Languages and Derivational Complexity of Formal Grammars and Languages*, Information and Control, Vol. 22, 1973, pp. 139-162.
20. B. O. NASH, *Reachibility Problems in Vector-Addition Systems*, Amer. Math. Monthly, Vol. 80, 1973, pp. 292-295.
21. J. L. PETERSON, *Computation Sequence Sets*, J. Comp. Syst. Sc., Vol. 13, 1976, pp. 1-24.
22. M. PENTTONEN, *On Derivation Languages Corresponding to Context-Free Grammars*, Acta Informatica, Vol. 3, 1974, pp. 285-293.
23. G. S. SACERDOTE and R. L. TENNEY, *The Decidability of the Reachability Problem for Vector-Addition Systems*, Proceedings of the 9th annual A.C.M. Symposium on Theory of Computing, 1977, pp. 61-76.
24. A. SALOMAA, *Formal Languages*, Academic Press New York and London, 1973.
25. E. D. STOTZKIJ, *On Some Restrictions on Derivations in Phrase-Structure Grammars*, Akad. Nauk. S.S.S.R. Nauchno-Tekhn., Inform. Ser. 2, 1967, pp. 35-38 (in Russian).
26. R. VALK, *Self-Modifying Nets*, Technical Report IFI-HH-B-34/77, Univ. Hamburg, 1977.
27. G. VIDAL-NAQUET, *Méthodes pour les problèmes d'indécidabilité et de complexité sur les réseaux de Petri, in* Proceedings of the AFCET Workshop on Petri Nets, Paris, 1977, pp. 137-144.