Acta Cybernetica **19** (2010) 609–634.

Petri Net Controlled Grammars with a Bounded Number of Additional Places*

Jürgen Dassow† and Sherzod Turaev‡

Abstract

A context-free grammar and its derivations can be described by a Petri net, called a *context-free Petri net*, whose places and transitions correspond to the nonterminals and the production rules of the grammar, respectively, and tokens are separate instances of the nonterminals in a sentential form. Therefore, the control of the derivations in a context-free grammar can be implemented by adding some features to the associated cf Petri net. The addition of new places and new arcs from/to these new places to/from transitions of the net leads grammars controlled by *k-Petri nets*, i.e., Petri nets with additional k places. In the paper we investigate the generative power and give closure properties of the families of languages generated by such Petri net controlled grammars, in particular, we show that these families form an infinite hierarchy with respect to the numbers of additional places.

Keywords: grammars, grammars with regulated rewriting, Petri nets, Petri net controlled grammars

1 Introduction

It is well-known fact that context-free grammars are not able to cover all phenomena of natural and programming languages, and also with respect to other applications of sequential grammars they cannot describe all aspects. On the other hand, context-sensitive grammars are powerful enough but have bad features with respect to decidability problems which are undecidable or at least very hard. Therefore it is a natural idea to introduce grammars which use context-free rules and have a device which controls the application of the rules. The monograph [2] gives a summary of this approach.

*This paper is an extended version of the paper presented at the Second International Conference on Language and Automata Theory and Applications, March 13-19, 2008, Tarragona, Spain [3].

†Otto-von-Guericke-Universität Magdeburg, PSF 4120, D-39016 Magdeburg, Germany, E-mail: dassow@iws.cs.uni-magdeburg.de

‡Faculty of Computer Science and Information Technology, UPM, 43400 Serdang, Selangor, Malaysia, E-mail: sherzod@fsktm.upm.edu.my

A context-free grammar and its derivation process can be described by a Petri net where places correspond to nonterminals, transitions are the counterpart of the productions, the tokens reflect the occurrences of symbols in the sentential form, and there is a one-to-one correspondence between the application of (sequences of) rules and the firing of (sequence of) transitions (see, [1]). Therefore it is a natural idea to control the derivations in a context-free grammar by adding some features to the associated Petri net.

In [7] and [13] it has been shown that by adding some places and arcs which satisfy some structural requirements one can generate well-known families of languages as random context languages, vector languages and matrix languages. Thus the control by Petri nets can be considered as a unifying approach to different types of control (note that random context is a control by occurrence/non-occurrence of letters whereas matrices give a prescribed set of sequences in which the productions have to be applied). In this paper we add new places, called *counters*, and new arcs associated with the new places. Adding k places leads to a control by k-Petri nets. The aim of this paper is the study of properties of the family of languages which can be generated by context-free grammars with a control by k-Petri nets. We present results on the generative power and we give some closure properties.

The paper is organized as follows. In Section 2 we give some notions and definitions from the theories of formal languages and Petri nets needed in the sequel. Moreover we introduce the Petri net associated with a context-free grammar. In Section 3 we construct the new Petri net control mechanism and define the corresponding grammar. Furthermore, we give some examples. In Section 4 we show that context-free grammars with the simple control by one additional place can generate non-context-free languages. We also give relations to valence grammars and vector grammars. Furthermore, we show that we get an infinite hierarchy with respect to the numbers of additional places. In Section 5 we investigate the fundamental closure properties of the families of languages generated by k-Petri net controlled grammars.

2 Preliminaries

The reader is assumed to be familiar with basic notions of formal language theory and Petri net theory as, e.g. contained in [8, 2, 4, 5, 6, 9, 10, 11, 12].

2.1 Grammars

Let Σ be an *alphabet* which is a finite nonempty set of symbols. A *string* over the alphabet Σ is a finite sequence of symbols from Σ. The *empty* string is denoted by λ. The set of all strings over the alphabet Σ is denoted by Σ^*. A subset of Σ^* is called a *language*. The *length* of a string w, denoted by $|w|$, is the number of occurrences of symbols in w. The number of occurrences of a symbol a in a string w is denoted by $|w|_a$. For a subset Δ of Σ, the number of occurrences of symbols of Δ in a string $w \in \Sigma^*$ is denoted by $|w|_\Delta$.

The operation *shuffle* for languages $L_1, L_2 \subseteq \Sigma^*$ is defined by

$$\text{Shuf}(L_1, L_2) = \{u_1v_1u_2v_2 \cdots u_nv_n \mid u_1u_2 \cdots u_n \in L_1, v_1v_2 \cdots v_n \in L_2, \\ u_i, v_i \in \Sigma^*, 1 \leq i \leq n\}$$

and for $L \subseteq \Sigma^*$,

$$\begin{aligned} \text{Shuf}^1(L) &= L, \\ \text{Shuf}^k(L) &= \text{Shuf}(\text{Shuf}^{k-1}(L), L), k \geq 2, \\ \text{Shuf}^*(L) &= \bigcup_{k\geq 1} \text{Shuf}^k(L). \end{aligned}$$

A *context-free grammar* is a quadruple $G = (V, \Sigma, S, R)$ where V and Σ are the disjoint finite sets of *nonterminal* and *terminal* symbols, respectively, $S \in V$ is the *start* symbol and $R \subseteq V \times (V \cup \Sigma)^*$ is a finite set of *(production) rules*. Usually, a rule (A, x) is written as $A \to x$. A rule of the form $A \to \lambda$ is called an *erasing rule*. $x \in (V \cup \Sigma)^+$ *directly derives* $y \in (V \cup \Sigma)^*$, written as $x \Rightarrow y$, iff there is a rule $r = A \to \alpha \in R$ such that $x = x_1Ax_2$ and $y = x_1\alpha x_2$. The reflexive and transitive closure of $\Rightarrow$ is denoted by $\Rightarrow^*$. A derivation using the sequence of rules $\pi = r_1r_2 \cdots r_n$ is denoted by $\stackrel{\pi}{\Rightarrow}$ or $\xRightarrow{r_1r_2\cdots r_n}$. The *language* generated by G is defined by $L(G) = \{w \in \Sigma^* \mid S \Rightarrow^* w\}$. The family of context-free languages is denoted by **CF**.

A *vector grammar* is a quadruple $G = (V, \Sigma, S, M)$ where V, Σ, S are defined as for a context-free grammar, and M is a finite set of strings over a set of context-free rules called *matrices*. The language generated by the grammar G is defined by $L(G) = \{w \in \Sigma^* \mid S \stackrel{\pi}{\Rightarrow} w \text{ and } \pi \in \text{Shuf}^*(M)\}$.

An *additive valence grammar* is a quintuple $G = (V, \Sigma, S, R, v)$ where V, Σ, S, R are defined as for a context-free grammar and v is a mapping from R into the set $\mathbb{Z}$ of integers. The language generated by G consists of all strings $w \in \Sigma^*$ such that there is a derivation $S \xRightarrow{r_1r_2\cdots r_n} w$ where $\sum_{i=1}^{n} v(r_i) = 0$.

A *positive valence grammar* is a quintuple $G = (V, \Sigma, S, R, v)$ whose components are defined as for an additive valence grammar. The language generated by G consists of all strings $w \in \Sigma^*$ such that there is a derivation $S \xRightarrow{r_1r_2\cdots r_n} w$ where $\sum_{i=1}^{n} v(r_i) = 0$ and for any $1 \leq j < n$, $\sum_{i=1}^{j} v(r_i) \geq 0$.

The families of languages generated by vector, additive valence and positive valence grammars (with erasing rules) are denoted by **V**, **aV** and **pV** ($\mathbf{V}^\lambda$, $\mathbf{aV}^\lambda$ and $\mathbf{pV}^\lambda$), respectively.

2.2 Petri Nets

A *Petri net* (PN) is a construct $N = (P, T, F, \phi)$ where P and T are disjoint finite sets of *places* and *transitions*, respectively, $F \subseteq (P \times T) \cup (T \times P)$ is the set of *directed arcs*, $\phi : (P \times T) \cup (T \times P) \to \{0, 1, 2, \cdots\}$ is a *weight function*, where $\phi(x, y) = 0$ for all $(x, y) \in ((P \times T) \cup (T \times P)) - F$. A Petri net can be represented by a bipartite directed graph with the node set $P \cup T$ where places are drawn as *circles*, transitions as *boxes* and arcs as *arrows*. The arrow representing an arc $(x, y) \in F$ is labeled with $\phi(x, y)$; if $\phi(x, y) = 1$, the label is omitted.

A mapping $\mu : P \to \{0, 1, 2, \ldots\}$ is called a *marking*. For each place $p \in P$, $\mu(p)$ gives the number of *tokens* in p. Graphically, tokens are drawn as small solid *dots* inside circles. ${}^\bullet x = \{y \mid (y, x) \in F\}$ and $x^\bullet = \{y \mid (x, y) \in F\}$ are called *pre-* and *post-sets* of $x \in P \cup T$, respectively. For $X \subseteq P \cup T$, define ${}^\bullet X = \bigcup_{x \in X} {}^\bullet x$ and $X^\bullet = \bigcup_{x \in X} x^\bullet$. For $t \in T$ ($p \in P$), the elements of ${}^\bullet t$ (${}^\bullet p$) are called *input* places (transitions) and the elements of $t^\bullet$ ($p^\bullet$) are called *output* places (transitions) of the transition t (the place p).

A transition $t \in T$ is *enabled* by marking μ if and only if $\mu(p) \geq \phi(p, t)$ for all $p \in P$. In this case t can *occur* (*fire*). Its occurrence transforms the marking μ into the marking μ' defined for each place $p \in P$ by $\mu'(p) = \mu(p) - \phi(p, t) + \phi(t, p)$. We write $\mu \xrightarrow{t} \mu'$ to indicate that the firing of t in μ leads to μ'. A finite sequence $t_1 t_2 \cdots t_k$, $t_i \in T, 1 \leq i \leq k$, is called *an occurrence sequence* enabled at a marking μ and finished at a marking μ' if there are markings $\mu_1, \mu_2, \ldots, \mu_{k-1}$ such that $\mu \xrightarrow{t_1} \mu_1 \xrightarrow{t_2} \ldots \xrightarrow{t_{k-1}} \mu_{k-1} \xrightarrow{t_k} \mu'$. In short this sequence can be written as $\mu \xrightarrow{t_1 t_2 \cdots t_k} \mu'$ or $\mu \xrightarrow{\nu} \mu'$ where $\nu = t_1 t_2 \cdots t_k$.

A *marked* Petri net is a system $N = (P, T, F, \phi, \iota)$ where (P, T, F, ϕ) is a Petri net, ι is the *initial marking*. Let M be a set of markings, which will be called *final* markings. An occurrence sequence ν of transitions is called *successful* for M if it is enabled at the initial marking ι and finished at a final marking τ of M. If M is understood from the context, we say that ν is a successful occurrence sequence.

2.3 Context-Free Petri Nets

The construction of the following type of Petri nets is based on the idea of using similarity between the firing of a transition and the application of a production rule in a derivation in which places are nonterminals and tokens are different occurrences of nonterminals.

Definition 1. *A context-free Petri net (in short, a cf Petri net) with respect to a context-free grammar $G = (V, \Sigma, S, R)$ is a tuple $N = (P, T, F, \phi, \beta, \gamma, \iota)$ where*

- *(P, T, F, ϕ) is a Petri net;*
- *labeling functions $\beta : P \to V$ and $\gamma : T \to R$ are bijections;*
- *there is an arc from place p to transition t if and only if $\gamma(t) = A \to \alpha$ and $\beta(p) = A$. The weight of the arc (p, t) is 1;*
- *there is an arc from transition t to place p if and only if $\gamma(t) = A \to \alpha$ and $\beta(p) = x$ where $|\alpha|_x > 0$. The weight of the arc (t, p) is $|\alpha|_x$;*
- *the initial marking ι is defined by $\iota(\beta^{-1}(S)) = 1$ and $\iota(p) = 0$ for all $p \in P - \{\beta^{-1}(S)\}$.*

We also use the natural extension of the labeling function $\gamma : T^* \to R^*$, which is done in the usual manner.

Example 1. Let G_1 be a context-free grammar with the rules:

$$r_0 : S \to AB,\ r_1 : A \to aAb,\ r_2 : A \to ab,\ r_3 : B \to cB,\ r_4 : B \to c$$

(the other components of the grammar can be seen from these rules). Figure 1 illustrates a cf Petri net N with respect to the grammar G_1. Obviously,

$$L(G_1) = \{a^n b^n c^m \mid n, m \geq 1\}.$$

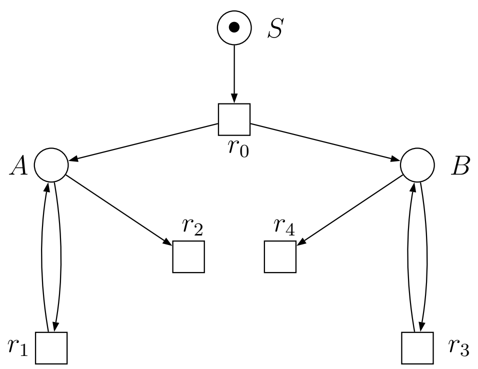

Figure 1: A cf Petri net N

The following proposition shows the similarity between terminal derivations in a context-free grammar and successful occurrences of transitions in the corresponding cf Petri net.

Proposition 1. *Let $N = (P, T, F, \phi, \iota, \beta, \gamma)$ be the cf Petri net with respect to a context-free grammar $G = (V, \Sigma, S, R)$. Then $S \xRightarrow{r_1 r_2 \cdots r_n} w, w \in \Sigma^*$, is a derivation in G iff $t_1 t_2 \cdots t_n$, $\iota \xrightarrow{t_1 t_2 \cdots t_n} \mu_n$, is an occurrence sequence of transitions in N such that $\gamma(t_1 t_2 \cdots t_n) = r_1 r_2 \cdots r_n$ and $\mu_n(p) = 0$ for all $p \in P$.*

Proof. Let $S \xRightarrow{r_1 r_2 \cdots r_n} w, w \in \Sigma^*$ be a derivation in the grammar G. By induction on the number $1 \leq k \leq n$ of derivation steps, we show that $t_1 t_2 \cdots t_n$ with $\gamma(t_1 t_2 \cdots t_n) = r_1 r_2 \cdots r_n$ is an occurrence sequence enabled at ι and finished at the marking μ_n where $\mu_n(p) = 0$ for all $p \in P$.

Let $k = 1$. $S \Rightarrow_{r_1} w_1$, i.e., the sentential form w_1 is obtained from S by the application of a rule $r_1 : S \to w_1 \in R$. Then the transition $t_1 = \gamma^{-1}(r_1)$ also occurs as its input place $\beta^{-1}(S)$ has a token, i.e., by definition, $\iota(\beta^{-1}(S)) = 1$. Let $\iota \xrightarrow{t_1} \mu_1$. Then for each $A \in V$, we have $\mu_1(p) = |w_1|_A$ where $p = \beta^{-1}(A)$.

Suppose $S \xRightarrow{r_1 r_2 \cdots r_m} w_m, w_m \in (V \cup \Sigma)^*$, $1 \leq m \leq k-1 < n$, and $t_1 t_2 \cdots t_m$ be an occurrence sequence of transitions of N such that $\gamma(t_1 t_2 \cdots t_m) = r_1 r_2 \cdots r_m$. Consider case $m = k$. Then the transition $t_k = \gamma^{-1}(r_k)$, $r_k : A \to \alpha \in R$, can fire since ${}^\bullet t_k = \{\beta^{-1}(A)\}$ and $\mu_k(\beta^{-1}(A)) = |w_k|_A > 0$. If $k = n$, then $\mu_n(p) = 0$ for all $p \in P$ as $w_n \in \Sigma^*$, i.e., $|w_k|_A = 0$ for all $A \in V$.

Let $\nu = t_1 t_2 \cdots t_n$ be an occurrence sequence of transitions of N enabled at ι and finished at μ_n where $\mu_n(p) = 0$ for all $p \in P$. By induction on the number $1 \leq k \leq n$ of occurrence steps we show that $S \xRightarrow{r_1 r_2 \cdots r_n} w, w \in \Sigma^*$, is a derivation in G where $r_1 r_2 \cdots r_n = \gamma(t_1 t_2 \cdots t_n)$.

For $k = 1$ we have $\iota \xrightarrow{t_1} \mu_1$. Then the rule $r_1 = \gamma^{-1}(t_1) : S \to \alpha \in R$ can also be applied and $S \Rightarrow_{r_1} w_1 = \alpha$. By definition, for each $A \in V$, $|w_1|_A = \mu_1(\beta^{-1}(A))$.

We suppose that for $1 \leq m \leq k-1 < n$, $S \xRightarrow{r_1 r_2 \cdots r_m} w_m \in (V \cup \Sigma)^*$ is a derivation in G where $r_1 r_2 \cdots r_m = \gamma(t_1 t_2 \cdots t_m)$. Then for each $A \in V$ and $1 \leq i \leq m$, $|w_i|_A = \mu_i(p)$ where $A = \beta(p)$. If $m = k$, the rule $r_k : A \to \alpha \in R$, $r_k = \gamma(t_k)$, can be applied since $|w_k|_A > 0$. For $k = n$, $\mu_n(p) = 0$ for all $p \in P$ and consequently, $|w_n|_A = \mu_n(\beta^{-1}(A)) = 0$ for all $A \in V$, i.e., $w_n \in \Sigma^*$. □

3 Petri Net Controlled Grammars and Examples

Now we define a *k-Petri net*, i.e., a cf Petri net with additional k places and additional arcs from/to these places to/from transitions of the net, the pre-sets and post-sets of the additional places are disjoint.

Definition 2. *Let $G = (V, \Sigma, S, R)$ be a context-free grammar with its corresponding cf Petri net $N = (P, T, F, \phi, \beta, \gamma, \iota)$. Let k be a positive integer and let $Q = \{q_1, q_2, \ldots, q_k\}$ be a set of new places called counters. A k-Petri net is a construct $N_k = (P \cup Q, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$ where*

- *$E = \{(t, q_i) \mid t \in T_1^i, 1 \leq i \leq k\} \cup \{(q_i, t) \mid t \in T_2^i, 1 \leq i \leq k\}$ such that $T_1^i \subset T$ and $T_2^i \subset T$, $1 \leq i \leq k$ where $T_l^i \cap T_l^j = \emptyset$ for $1 \leq l \leq 2$, $T_1^i \cap T_2^j = \emptyset$ for $1 \leq i < j \leq k$ and $T_1^i = \emptyset$ if and only if $T_2^i = \emptyset$ for any $1 \leq i \leq k$.*
- *the weight function $\varphi(x, y)$ is defined by $\varphi(x, y) = \phi(x, y)$ if $(x, y) \in F$ and $\varphi(x, y) = 1$ if $(x, y) \in E$,*
- *the labeling function $\zeta : (P \cup Q) \to V \cup \{\lambda\}$ is defined by $\zeta(p) = \beta(p)$ if $p \in P$ and $\zeta(p) = \lambda$ if $p \in Q$,*
- *the initial marking μ_0 is defined by $\mu_0(\beta^{-1}(S)) = 1$ and $\mu_0(p) = 0$ for all $p \in (P \cup Q) - \{\beta^{-1}(S)\}$,*
- *τ is the final marking where $\tau(p) = 0$ for all $p \in (P \cup Q)$.*

Definition 3. *A k-Petri net controlled grammar (in short, a k-PN controlled grammar) is a quintuple $G = (V, \Sigma, S, R, N_k)$ where V, Σ, S, R are defined as for a context-free grammar and N_k is a k-Petri net with respect to the context-free grammar (V, Σ, S, R).*

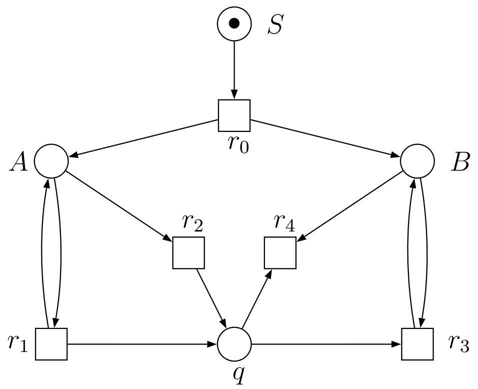

Figure 2: A 1-Petri net N_1

Definition 4. *The language generated by a k-Petri net controlled grammar G consists of all strings $w \in \Sigma^*$ such that there is a derivation*

$$S \xrightarrow{r_1 r_2 \cdots r_n} w \text{ where } t_1 t_2 \cdots t_n = \gamma^{-1}(r_1 r_2 \cdots r_n) \in T^*$$

is an occurrence sequence of the transitions of N_k enabled at the initial marking μ_0 and finished at the final marking τ.

We denote the family of languages generated by k-PN controlled grammars (with erasing rules) by $\mathbf{PN}_k$ ($\mathbf{PN}_k^{\lambda}$), $k \geq 1$. We also use bracket notation $\mathbf{PN}_k^{[\lambda]}$ in order to say that a statement holds in both cases: with and without erasing rules.

We give two examples which will be used in the sequel.

Example 2. Figure 2 illustrates a 1-Petri net N_1 which is constructed from the cf Petri net N in Figure 1 adding a single counter place q. Let $G_2 = (V, \Sigma, S, R, N_1)$ be the 1-PN controlled grammar where V, Σ, S, R are defined as for the grammar G_1 in Example 1. It is not difficult to see that $L(G_2) = \{a^n b^n c^n \mid n \geq 1\}$.

Example 3. Let G_3 be a 2-PN controlled grammar with the production rules:

$$\begin{array}{lll} r_0 : S \to A_1 B_1 A_2 B_2, & r_1 : A_1 \to a_1 A_1 b_1, & r_2 : A_1 \to a_1 b_1, \\ r_3 : B_1 \to c_1 B_1, & r_4 : B_1 \to c_1, & r_5 : A_2 \to a_2 A_2 b_2, \\ r_6 : A_2 \to a_2 b_2, & r_7 : B_2 \to c_2 B_2, & r_8 : B_2 \to c_2 \end{array}$$

and the corresponding 2-Petri net N_2 is given in Figure 3. Then it is easy to see that G_3 generates the language $L(G_3) = \{a_1^n b_1^n c_1^n a_2^m b_2^m c_2^m \mid n, m \geq 1\}$.

Lemma 1. *The language $L' = \{a_1^n b_1^n c_1^n a_2^m b_2^m c_2^m \mid n, m \geq 1\}$ cannot be generated by a 1-PN controlled grammar.*

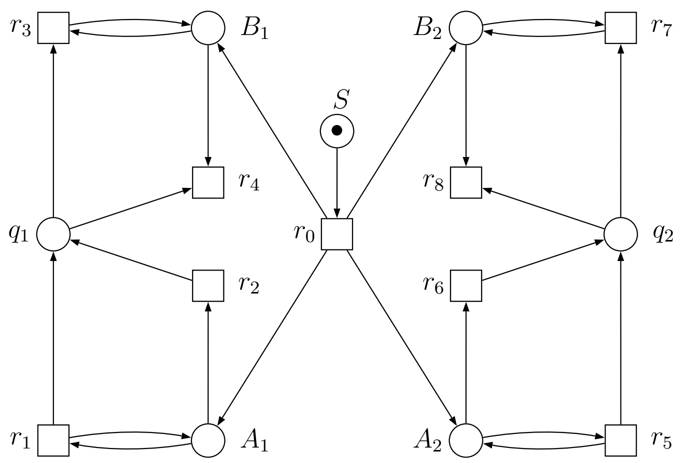

Figure 3: A 2-Petri net N_2

Proof. Suppose the contrary: there is a 1-Petri net controlled grammar $G = (V, \Sigma, S, R, N_1)$ where $\Sigma = \{a_1, b_1, c_1, a_2, b_2, c_2\}$ such that $L(G) = L'$. Let $w = a_1^n b_1^n c_1^n a_2^m b_2^m c_2^m$. Since the set V is finite, and if n and m are chosen sufficiently large, every derivation $S \Rightarrow^* w$ in G contains a subderivation of the form D: $A \Rightarrow^* xAy$ where $A \in V$ and $x, y \in \Sigma^*$ with $xy \neq \lambda$. As L' is infinite, there are words with enough large length obtained by iterating such a derivation D arbitrarily many times. Suppose

$$S \Rightarrow^* uAv \Rightarrow^* uxAyv \Rightarrow^* \cdots \Rightarrow^* ux^nAy^nv \Rightarrow^* w' \in \Sigma^* \tag{1}$$

is also a derivation in G. Then x^n and y^n are substrings of w'. By the structure of the words of L', x and y can be only powers of two symbols from $\Sigma \cup \{\lambda\}$. Therefore, in order to generate a word $w = a_1^n b_1^n c_1^n a_2^m b_2^m c_2^m \in L'$ for large n and m, we need at least three subderivations of the form

$$D_1 : A_1 \Rightarrow^* x_1A_1y_1, \tag{2}$$

$$D_2 : A_2 \Rightarrow^* x_2A_2y_2, \tag{3}$$

$$D_3 : A_3 \Rightarrow^* x_3A_3y_3 \tag{4}$$

where $x_1, x_2, x_3, y_1, y_2, y_3$ are powers of the symbols from Σ, i.e.,

$$x_i = \alpha_i^{k_i} \text{ and } y_i = \beta_i^{l_i} \text{ where } \alpha_i, \beta_i \in \Sigma \text{ and } k_i + l_i \geq 1, i = 1, 2, 3.$$

First, we assume that (1) has exactly three subderivations of the form (2)–(4). According to the production and consumption of tokens by the subderivations (2)–(4) the following cases can occur:

Case 1. One of the derivations (2)–(4) does not produce and consume any token. Without loss of generality we can assume that this derivation is (2). If

$$S \Rightarrow^* uA_1v \Rightarrow^* uwv \in L'$$

then for any $k > 1$ we apply (2) k times and get a string which is not in L', i.e.

$$S \Rightarrow^* uA_1v \Rightarrow^* ux_1A_1y_1v \Rightarrow^* ux_1^2A_1y_1^2v \Rightarrow^* ux_1^kA_1y_1^kv \Rightarrow^* ux_1^kwy_1^kv \notin L'$$

since (2) increases only the powers of at most two letters.

Case 2. One of the subderivations (2)–(4) produces tokens and another one consumes tokens. Without loss of generality we assume that (2) produces $p \geq 1$ tokens and (3) consumes $q \geq 1$ tokens.

Suppose

$$S \Rightarrow^* u_1A_1u_2A_2u_3 \Rightarrow^* u_1w_1u_2w_2u_3 \in L'.$$

Then the derivation

$$\begin{aligned} S &\Rightarrow^* u_1A_1u_2A_2u_3 \\ &\Rightarrow^* u_1x_1A_1y_1u_2A_2u_3 \Rightarrow^* u_1x_1^kA_1y_1^ku_2A_2u_3 \\ &\Rightarrow^* u_1x_1^kA_1y_1^ku_2x_2A_2y_2u_3 \Rightarrow^* u_1x_1^kA_1y_1^ku_2x_2^lA_2y_2^lu_3 \\ &\Rightarrow^* u_1x_1^kw_1y_1^ku_2x_2^lw_2y_2^lu_3 \end{aligned}$$

where $k, l \geq 1$, is also in G. It can be done by choosing the numbers k, l in such a way, that $kp - lq = 0$, thus we can choose k and l as $k = q$ and $l = p$ and still get a string $w' \in L'$. Now

- if $1 \leq |\{\alpha_1, \beta_1, \alpha_2, \beta_2\} \cap \{a_i, b_i, c_i\}| \leq 2$, $i = 1$ or $i = 2$ then $w' \notin L'$ as the powers of at most two symbols are increased;
- if $\{\alpha_1, \beta_1, \alpha_2, \beta_2\} \cap \{a_i, b_i, c_i\} \neq \emptyset$ for both $i = 1$ and $i = 2$ then $1 \leq |\{\alpha_1, \beta_1, \alpha_2, \beta_2\} \cap \{a_i, b_i, c_i\}| \leq 2$ for $i = 1$ or $i = 2$ and again $w' \notin L'$.

From the above it follows that $\{\alpha_1, \beta_1, \alpha_2, \beta_2\} = \{a_i, b_i, c_i, \lambda\}$ for $i = 1$ or $i = 2$. Without loss of generality we assume that $i = 1$. But from the subderivation (4) (which produces or consumes tokens) it follows that $\alpha_3, \beta_3 \notin \{a_1, b_1, c_1\}$ and at least one of them belongs to $\{a_2, b_2, c_2\}$. Again we get the contradiction since (4) can increase the powers of at most two symbols from $\{a_2, b_2, c_2\}$.

If the derivation has the form

$$S \Rightarrow^* u_1A_1u_4 \Rightarrow^* u_1u_2A_2u_3u_4 \Rightarrow^* u_1u_2wu_3u_4,$$

then one gets that $\{x_1, y_1, x_2, y_2\}$ contains only two elements from Σ and a contradiction follows as above.

Case 3. Two of the subderivations of (2)–(4) produce (consume) tokens and the other consumes (produces). Without loss of generality we assume that (2) and (3) produces p_1 and p_2 tokens, respectively and (4) consumes q tokens. If

$$S \Rightarrow^* u_1A_1u_2A_2u_3A_3u_4 \Rightarrow^* u_1w_1u_2w_2u_3w_3u_4 \in L',$$

then the derivation

$$\begin{aligned} S &\Rightarrow^* u_1A_1u_2A_2u_3A_3u_4 \\ &\Rightarrow^* u_1x_1A_1y_1u_2x_2A_2y_2u_3x_3A_3y_3u_4 \\ &\Rightarrow^* u_1x_1^{k_1}A_1y_1^{k_1}u_2x_2^{k_2}A_2y_2^{k_2}u_3x_3^{l}A_3y_3^{l}u_4 \\ &\Rightarrow^* u_1x_1^{k_1}w_1y_1^{k_1}u_2x_2^{k_2}w_2y_2^{k_2}u_3x_3^{l}w_3y_3^{l}u_4 = w' \qquad (5) \end{aligned}$$

is also in G. By the definition of the final marking, we have $k_1p_1 + k_2p_2 - lq = 0$. For instance, if we choose k_1, k_2, l as $k_1 = p_2q$, $k_2 = p_1q$ and $l = 2p_1p_2$, this equality holds. By structure of a derivation there are two possibilities:

$$\{\alpha_1, \beta_1, \alpha_2, \beta_2, \alpha_3, \beta_3\} = \{a_1, b_1, c_1, a_2, b_2, c_2, \lambda\} \qquad (6)$$

or

$$\{\alpha_1, \beta_1, \alpha_2, \beta_2, \alpha_3, \beta_3\} = \{a_i, b_i, c_i, \lambda\} \text{ where } i = 1 \text{ or } i = 2. \qquad (7)$$

Consider (6), here we only have the case $\alpha_1 = a_1$, $\beta_1 = b_1$, $\alpha_2 = c_1$, $\beta_2 = a_2$, $\alpha_3 = b_2$ and $\beta_3 = c_2$. It follows that the powers of all symbols of w' are the same. But from (5), by continuing the derivation, we get a string which is not in L':

$$\begin{aligned} S &\Rightarrow^* u_1x_1^{k_1}A_1y_1^{k_1}u_2x_2^{k_2}A_2y_2^{k_2}u_3x_3^{l}A_3y_3^{l}u_4 \\ &\Rightarrow^* u_1x_1^{k_1}w_1y_1^{k_1}u_2x_2^{k_2}w_2y_2^{k_2}u_3x_3^{l}A_3y_3^{l}u_4 \\ &\Rightarrow^* u_1x_1^{k_1}w_1y_1^{k_1}u_2x_2^{k_2}w_2y_2^{k_2}u_3x_3^{2l}A_3y_3^{2l}u_4 \\ &\Rightarrow^* u_1x_1^{k_1}w_1y_1^{k_1}u_2x_2^{2k_2}w_2y_2^{2k_2}u_3x_3^{3l}w_3y_3^{3l}u_4 \notin L' \end{aligned}$$

where the powers of four symbols are increased.

Now consider (7). Let $i = 1$. From Case 2, we can conclude that one of the following three cases is possible:

$$\begin{array}{llll} (a) & \{\alpha_1, \beta_1\} = \{a_1, b_1\}, & \{\alpha_2, \beta_2\} = \{\lambda\}, & \{\alpha_3, \beta_3\} = \{c_1, \lambda\}, \\ (b) & \{\alpha_1, \beta_1\} = \{\lambda\}, & \{\alpha_2, \beta_2\} = \{a_1, b_1\}, & \{\alpha_3, \beta_3\} = \{c_1, \lambda\}, \\ (c) & \{\alpha_1, \beta_1\} = \{a_1, \lambda\}, & \{\alpha_2, \beta_2\} = \{b_1, \lambda\}, & \{\alpha_3, \beta_3\} = \{c_1, \lambda\}. \end{array}$$

Cases (a) and (b) are similar to *Case 2.* If we choose $k_1 = 3p_2l$, $k_2 = 2p_1l$ and $q = 5p_1p_2$ in case (c), we again get different powers for symbols a_1, b_1, c_1, i.e., $w' \notin L'$.

Next, we analyze the general case: let the derivation (1) have $n \geq 4$ subderivations of the form $D_i : A_i \to x_iA_iy_i$ where $A_i \in V$, $x_i = \alpha_i^{l_i}$ and $y_i = \beta_i^{l'_i}$, $\alpha_i, \beta_i \in \Sigma$,

$l_i + l'_i \geq 1$, $1 \leq i \leq n$. Without loss of generality we can assume that for some $1 \leq s \leq n-1$, the derivations D_i, $1 \leq i \leq s$, produce p_i tokens and the derivations D_j, $s+1 \leq j \leq n$, consume q_j tokens. If

$$S \Rightarrow^* u_1A_1u_2A_2u_3 \cdots u_nA_nu_{n+1} \Rightarrow^* u_1w_1u_2w_2u_3 \cdots u_nw_nu_{n+1} = w \in L', \quad (8)$$

then by assumption,

$$\begin{aligned} S &\Rightarrow^* u_1A_1u_2A_2u_3 \cdots u_nA_nu_{n+1} \\ &\Rightarrow^* u_1x_1A_1y_1u_2x_2A_2y_2u_3 \cdots u_nx_nA_ny_nu_{n+1} \\ &\Rightarrow^* u_1x_1^{k_1}A_1y_1^{k_1}u_2x_2^{k_2}A_2y_2^{k_2}u_3 \cdots u_nx_n^{k_n}A_ny_n^{k_n}u_{n+1} \\ &\Rightarrow^* u_1x_1^{k_1}w_1y_1^{k_1}u_2x_2^{k_2}w_2y_2^{k_2}u_3 \cdots u_nx_n^{k_n}w_ny_n^{k_n}u_{n+1} = w' \in L'. \quad (9) \end{aligned}$$

According to the definition of the final marking, we have

$$\sum_{i=1}^{s} k_ip_i - \sum_{i=s+1}^{n} k_iq_i = 0.$$

and

$$\{\alpha_1, \beta_1, \alpha_2, \beta_2, \ldots, \alpha_n, \beta_n\} = \{a_1, b_1, c_1, a_2, b_2, c_2, \lambda\}.$$

If for some $1 \leq i \leq n$, $\alpha_i = c_1$ and $\beta_i = a_2$, then all symbols in w' have the same power. Then by continuing two subderivations one of which produces tokens and the other consumes, one increases the powers of at most four symbols, and get a string $w'' \notin L'$.

Let, for some $2 \leq i \leq n-2$,

$$\{\alpha_1, \beta_1, \alpha_2, \beta_2, \ldots, \alpha_i, \beta_i\} = \{a_1, b_1, c_1, \lambda\} \quad (10)$$

and

$$\{\alpha_{i+1}, \beta_{i+1}, \alpha_{i+2}, \beta_{i+2}, \ldots, \alpha_n, \beta_n\} = \{a_2, b_2, c_2, \lambda\}. \quad (11)$$

It follows that at least one of the subderivations which generate symbols in (10) (symbols in (11)) produces and another subderivation consumes tokens, since symbols a_i, b_i, c_i, $i = 1, 2$, have the same power. Then the tokens produced by a subderivation D_j, for some $1 \leq j \leq i$, can be consumed by a subderivation D_k, for some $i+1 \leq k \leq n$ as the both group of subderivations use the same counter, which result that the powers of at most two symbols from a_1, b_1, c_1 and a_2, b_2, c_2 are increased, i.e., a string $w' \notin L'$ is generated. In all cases, we get contradiction to our assumption $L' = L(G)$. □

4 Hierarchy Results

We start with a simple fact.

Lemma 2. $\mathbf{CF} \subsetneq \mathbf{PN}_1$.

Proof. It is clear that $\mathbf{CF} \subseteq \mathbf{PN}_1$ if we take $T_1 = T_2 = \emptyset$. From Example 2 it follows that $\mathbf{CF} \subsetneq \mathbf{PN}_1$. □

Now we present some relations to (positive) additive valence languages.

Lemma 3. $\mathbf{PN}_1^{[\lambda]} \subseteq \mathbf{pV}^{[\lambda]}$.

Proof. Let $G = (V, \Sigma, S, R, N_1)$ be a 1-PN controlled grammar (with or without erasing rules) where $N_1 = (P \cup \{q\}, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$ is a corresponding 1-Petri net with the counter q (with the notions of Definition 2). We define a positive valence grammar $G' = (V, \Sigma, S, R, v)$ where V, Σ, S, R are defined as for the grammar G and for each $r \in R$, the mapping v is defined by

$$v(r) = \begin{cases} 1 & \text{if } \gamma^{-1}(r) \in {}^\bullet q, \\ -1 & \text{if } \gamma^{-1}(r) \in q^\bullet, \\ 0 & \text{otherwise.} \end{cases}$$

Let $S \stackrel{\pi}{\Rightarrow} w, w \in \Sigma^*$, $\pi = r_1 r_2 \cdots r_k$, be a derivation in G. Then $\nu = t_1 t_2 \cdots t_k = \gamma^{-1}(r_1 r_2 \cdots r_k)$ is an occurrence sequence of transitions of N_1 enabled at the initial marking μ_0 and finished at the final marking τ, i.e.,

$$\mu_0 \xrightarrow{t_1} \mu_1 \xrightarrow{t_2} \cdots \xrightarrow{t_k} \mu_k = \tau$$

By definition, if $|\nu|_t > 0$ for some $t \in {}^\bullet q$ then there is a transition $t' \in q^\bullet$ such that $|\nu|_{t'} > 0$. Let

$$U_1 = \{t_{1,1}, t_{1,2}, \ldots, t_{1,k_1}\} \subseteq {}^\bullet q \text{ where } |\nu|_{t_{1,j}} > 0, 1 \leq j \leq k_1$$

and

$$U_2 = \{t_{2,1}, t_{2,2}, \ldots, t_{2,k_2}\} \subseteq q^\bullet \text{ where } |\nu|_{t_{2,j}} > 0, 1 \leq j \leq k_2.$$

Since $\mu_i(q) \geq 0$ for each occurrence step $1 \leq i \leq k$, we have $|\nu|_{U_1} \geq |\nu|_{U_2}$, consequently, $v(r_1) + v(r_2) + \ldots + v(r_j) \geq 0$ for any $1 \leq j < k$ and from $\mu_0(q) = \tau(q) = 0$, $\tau \in M$, it follows that

$$\sum_{t \in U_1} |\nu|_t - \sum_{t \in U_2} |\nu|_t \stackrel{\text{def}}{=} \sum_{i=1}^{k} v(r_i) = 0.$$

Hence, $L(G) \subseteq L(G')$.

Let $D : S \xrightarrow{r_1 r_2 \cdots r_k} w \in \Sigma^*$ be a derivation in G' where $v(r_1) + v(r_2) + \ldots + v(r_k) = 0$ and $v(r_1) + v(r_2) + \ldots + v(r_j) \geq 0$ for any $1 \leq j < k$. By construction of G', D is also a derivation in (V, Σ, S, R).

According to the bijection $\gamma : T \to R$, there is an occurrence sequence $\nu = t_1 t_2 \cdots t_k$, $\mu \xrightarrow{t_1} \mu_1 \xrightarrow{t_2} \cdots \xrightarrow{t_k} \mu_k$, in N_1 such that $\nu = \gamma^{-1}(r_1 r_2 \cdots r_k)$.

$\mu = \mu_0$ since D starts from S, i.e., $\mu_0(\beta^{-1}(S)) = 1$ and $\mu_0(\beta^{-1}(x)) = 0$ for all $x \in (V \cup \Sigma) - \{S\}$ as well as $\mu_0(q) = 0$.

Since $w \in \Sigma^*$, we have $\mu_k(\beta^{-1}(x)) = 0$ for all $x \in V$. From $\sum_{i=1}^{j} v(r_i) \geq 0$, it follows that $\mu_j(q) \geq 0$ for any $1 \leq j < k$.

$$\sum_{i=1}^{k} v(r_i) \stackrel{\text{def}}{=} \sum_{\gamma^{-1}(r) \in {}^\bullet q} v(r) + \sum_{\gamma^{-1}(r) \in q^\bullet} v(r) = 0$$

shows that $\mu_k(q) = 0$. Therefore $\mu_k = \tau$. Consequently, $L(G') \subseteq L(G)$. □

Lemma 4. $\mathbf{aV}^{[\lambda]} \subsetneq \mathbf{PN}_2^{[\lambda]}$.

Proof. Let $G = (V, \Sigma, S, R, v)$ be an additive valence grammar (with or without erasing rules). Without loss of generality we can assume that $v(r) \in \{1, 0, -1\}$ for each $r \in R$ (Lemma 2.1.10 in [2]).

For each rule $r : A \to \alpha \in R$, $v(r) \neq 0$ we add a nonterminal symbol A_r and a pair of rules $r' : A \to A_r$, $r'' : A_r \to \alpha$ and we set

$$V' = V \cup \{A_r \mid r : A \to \alpha \in R, v(r) \neq 0\},$$
$$R' = R \cup \{r' : A \to A_r, r'' : A_r \to \alpha \mid r : A \to \alpha \in R, v(r) \neq 0\}.$$

Let $N = (P, T, F, \phi, \beta, \gamma, \iota)$ be a cf Petri net with respect to the context-free grammar (V', Σ, S, R'). We construct a 2-Petri net $N_2 = (P \cup Q, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$ where $Q = \{q, q'\}$ and $E = F_1 \cup F_2$ with

$$F_1 = \{(t, q) \mid t = \gamma^{-1}(r), r \in R \text{ and } v(r) = 1\}$$
$$\cup \{(t', q') \mid t' = \gamma^{-1}(r'), r \in R \text{ and } v(r) = -1\},$$
$$F_2 = \{(q, t) \mid t = \gamma^{-1}(r), r \in R \text{ and } v(r) = -1\}$$
$$\cup \{(q', t') \mid t' = \gamma^{-1}(r'), r \in R \text{ and } v(r) = 1\}.$$

The rest components of N_2 are defined the same as those in the definition. Consider the 2-PN controlled grammar $G' = (V', \Sigma, S, R', N_2)$.

Let $D : S \stackrel{\pi}{\Rightarrow} w, w \in \Sigma^*$, $\pi = r_1 r_2 \cdots r_n$, be a derivation in G'. Then $\sigma = t_1 t_2 \cdots t_n = \gamma^{-1}(r_1 r_2 \cdots r_n)$ is an occurrence sequence enabled at the initial marking μ_0 and finished at the final marking τ. By construction,

$$\sum_{i=1}^{n} v(r_i) = \sum_{t \in {}^\bullet q} |\sigma|_t + \sum_{t \in q'^\bullet} |\sigma|_t - \sum_{t \in q^\bullet} |\sigma|_t - \sum_{t \in {}^\bullet q'} |\sigma|_t = 0$$

since

$$\sum_{t \in {}^\bullet q} |\sigma|_t = \sum_{t \in q^\bullet} |\sigma|_t = \sum_{i=1}^{n} \mu_i(q) \text{ and } \sum_{t \in {}^\bullet q'} |\sigma|_t = \sum_{t \in q'^\bullet} |\sigma|_t = \sum_{i=1}^{n} \mu_i(q').$$

It follows that D is also a derivation in G.

Let $D' : S \xrightarrow{r_1 r_2 \cdots r_n} w, w \in \Sigma^*$ be a derivation in G. For each $1 \leq k \leq n$,

(1) if $\sum_{i=1}^{k} v(r_i) > 0$, then for the rule r_k with $v(r_k) \in \{1, 0, -1\}$ in G choose the rule r_k in G';

(2) if $\sum_{i=1}^{k} v(r_i) < 0$, then for the rule r_k with $v(r_k) \neq 0$ in G choose the rules r'_k and r''_k in G'; if $v(r_k) = 0$ then choose r_k in G'.

(3) if $\sum_{i=1}^{k} v(r_i) = 0$, then for the rule r_k with $v(r_k) \in \{-1, 0\}$ in G choose the rule r_k in G'; if $v(r_k) = 1$, then choose r'_k, r''_k in G'.

Therefore D' is also a derivation in G'. The strict inclusion follows from the fact that

$$\{a_1^n b_1^n c_1^n a_2^m b_2^m c_2^m \mid n, m \geq 1\} \in \mathbf{PN}_2$$

cannot be generated by an additive valence grammar (Example 2.1.7 in [2]). □

The following lemma shows that, for any $n \geq 1$, an n-PN controlled grammar generates a vector language.

Lemma 5. *For* $n \geq 1$, $\mathbf{PN}_n^{[\lambda]} \subseteq \mathbf{V}^{[\lambda]}$.

Proof. Let $G = (V, \Sigma, S, R, N_n)$ be an n-PN controlled grammar (with or without erasing rules) with the n-Petri net $N_n = (P \cup Q, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$. Let $Q = \{q_1, q_2, \dots, q_n\}$ and

$$^\bullet q_k = \{t_{k,1,1}, t_{k,1,2}, \dots, t_{k,1,s(k)}\}$$

where $t_{k,1,i} = \gamma^{-1}(r_{k,1,i})$, $r_{k,1,i} : A_{k,1,i} \to w_{k,1,i}$, $1 \leq k \leq n$, $1 \leq i \leq s(k)$, and

$$q_k^\bullet = \{t_{k,2,1}, t_{k,2,2}, \dots, t_{k,2,l(k)}\}$$

where $t_{k,2,j} = \gamma^{-1}(r_{k,2,j})$, $r_{k,2,j} : A_{k,2,j} \to w_{k,2,j}$, $1 \leq k \leq n$, $1 \leq j \leq l(k)$.

Let

$$\beta(p_{k,1,i}) = A_{k,1,i},\ 1 \leq k \leq n,\ 1 \leq i \leq s(k)$$

and

$$\beta(p_{k,2,j}) = A_{k,2,j},\ 1 \leq k \leq n,\ 1 \leq j \leq l(k).$$

First, we construct a PN controlled grammar $G' = (V', \Sigma, S, R', N')$ in such a way that each counter place of N' has a single input transition and a single output transition, and we show that the grammars G and G' generate the same language. We set $V' = V \cup \{B_{k,i,j}, C_{k,j,i} \mid 1 \leq k \leq n, 1 \leq i \leq s(k), 1 \leq j \leq l(k)\}$ where $B_{k,i,j}$ and $C_{k,j,i}$, $1 \leq k \leq n$, $1 \leq i \leq s(k)$, $1 \leq j \leq l(k)$, are new nonterminals. R' consists of the following rules

$$\begin{aligned} R' = (R - \{&r_{k,1,i}, r_{k,2,j} \mid 1 \leq k \leq n,\ 1 \leq i \leq s(k),\ 1 \leq j \leq l(k)\}) \\ \cup \{&r'_{k,1,i,j} : A_{k,1,i} \to B_{k,i,j} \mid 1 \leq k \leq n, 1 \leq i \leq s(k), 1 \leq j \leq l(k)\} \\ \cup \{&r''_{k,1,i,j} : B_{k,i,j} \to w_{k,1,i} \mid 1 \leq k \leq n, 1 \leq i \leq s(k), 1 \leq j \leq l(k)\} \\ \cup \{&r'_{k,2,j,i} : A_{k,2,j} \to C_{k,j,i} \mid 1 \leq k \leq n, 1 \leq i \leq s(k), 1 \leq j \leq l(k)\} \\ \cup \{&r''_{k,2,j,i} : C_{k,j,i} \to w_{k,2,j} \mid 1 \leq k \leq n, 1 \leq i \leq s(k), 1 \leq j \leq l(k)\} \end{aligned}$$

and $N' = (P' \cup Q', T', F', \varphi', \zeta', \gamma', \mu_0', \tau')$ where the sets of places, transitions and arcs

$$\begin{aligned}
P' =& P \cup \{p_{k,1,i,j} \mid 1 \le k \le n,\, 1 \le i \le s(k),\, 1 \le j \le l(k)\} \\
& \cup \{p_{k,2,j,i} \mid 1 \le k \le n,\, 1 \le i \le s(k),\, 1 \le j \le l(k)\}, \\
Q' =& \{q_{k,i,j} \mid 1 \le k \le n,\, 1 \le i \le s(k), 1 \le j \le l(k)\}, \\
T' =& (T - \bigcup_{k=1}^{n} ({}^\bullet q_k \cup q_k^\bullet)) \\
& \cup \{t'_{k,1,i,j}, t''_{k,1,i,j} \mid 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k)\} \\
& \cup \{t'_{k,2,j,i}, t''_{k,2,j,i} \mid 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k)\},
\end{aligned}$$

$$\begin{aligned}
F' = (F \cup E - \bigcup_{k=1}^{n} (&\{(p_{k,1,i}, t_{k,1,i}), (t_{k,1,i}, q_k) \mid 1 \le i \le s(k)\} \\
& \cup \{(t_{k,1,i}, p) \mid p = \zeta^{-1}(x),\, |w_{k,1,i}|_x > 0, 1 \le i \le s(k)\} \\
& \cup \{(q_k, t_{k,2,j}), (p_{k,2,j}, t_{k,2,j}) \mid 1 \le j \le l(k)\} \\
& \cup \{(t_{k,2,j}, p) \mid p = \zeta^{-1}(x),\, |w_{k,2,j}|_x > 0, 1 \le j \le l(k)\})) \\
\cup \bigcup_{k=1}^{n} \bigcup_{i=1}^{s(k)} \bigcup_{j=1}^{l(k)} (&\{(p_{k,1,i}, t'_{k,1,i,j}), (t'_{k,1,i,j}, p_{k,1,i,j}), (p_{k,1,i,j}, t''_{k,1,i,j}), \\
& (t''_{k,1,i,j}, q_{k,i,j})\} \cup \{(t''_{k,1,i,j}, p) \mid p = \zeta^{-1}(x),\, |w_{k,1,i}|_x > 0\}) \\
\cup \bigcup_{k=1}^{n} \bigcup_{j=1}^{l(k)} \bigcup_{i=1}^{s(k)} (&\{(p_{k,2,j}, t'_{k,2,j,i}), (t'_{k,1,j,i}, p_{k,2,j,i}), (p_{k,2,j,i}, t''_{k,2,j,i}), \\
& (t''_{k,2,j,i}, q_{k,i,j})\} \cup \{(t''_{k,2,j,i}, p) \mid p = \zeta^{-1}(x),\, |w_{k,2,j}|_x > 0\}).
\end{aligned}$$

- The weight function is defined by

$$\varphi'(x,y) = \begin{cases} \varphi(x,y) & \text{if } (x,y) \in F, \\ \varphi(t_{k,1,i}, p) & \text{if } x = t_{k,1,i,j}, y = p = \zeta^{-1}(x), |w_{k,1,i}|_x > 0, \\ & \quad 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k), \\ \varphi(t_{k,2,j}, p) & \text{if } x = t_{k,2,j,i}, y = p = \zeta^{-1}(x), |w_{k,2,j}|_x > 0, \\ & \quad 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k), \\ 1 & \text{otherwise.} \end{cases}$$

- The labeling functions are defined by

$$\zeta'(p) = \begin{cases} \zeta(p) & \text{if } p \in P, \\ B_{k,i,j} & \text{if } p = p_{k,1,i,j}, 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k), \\ C_{k,j,i} & \text{if } p = p_{k,2,j,i}, 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k), \\ \lambda, & \text{if } p = q_{k,i,j}, 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k) \end{cases}$$

and

$$\gamma'(t) = \begin{cases} \gamma(t) & \text{if } t \in T, \\ r'_{k,1,i,j} & \text{if } t = t'_{k,1,i,j}, 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k), \\ r''_{k,1,i,j} & \text{if } t = t''_{k,1,i,j}, 1 \le k \le n, 1 \le i \le s(k), 1 \le j \le l(k), \\ r'_{k,2,j,i} & \text{if } t = t'_{k,2,j,i}, 1 \le k \le n, 1 \le j \le l(k), 1 \le i \le s(k), \\ r''_{k,2,j,i} & \text{if } t = t''_{k,2,j,i}, 1 \le k \le n, 1 \le j \le l(k), 1 \le i \le s(k). \end{cases}$$

- The initial marking is defined by $\mu'_0(\zeta^{-1}(S)) = 1$ and $\mu'_0(p) = 0$ for all $p \in P' \cup Q' - \{\zeta^{-1}(S)\}$.
- The final marking is defined by $\tau'(p) = 0$ for all $p \in P' \cup Q'$.

By the construction of N', an occurrence sequence of the form

$$\mu_1 \xrightarrow{t'_{k,1,i,j}} \mu_2 \xrightarrow{\sigma'} \mu_3 \xrightarrow{t''_{k,1,i,j}} \mu_4 \xrightarrow{\sigma''} \mu_5 \xrightarrow{t''_{k,2,j,i}} \mu_6 \xrightarrow{\sigma'''} \mu_7 \xrightarrow{t'_{k,2,j,i}} \mu_8 \tag{12}$$

where $\sigma', \sigma'', \sigma''' \in T'^*$ can be replaced by

$$\mu_1 \xrightarrow{t'_{k,1,i,j}} \mu_2 \xrightarrow{t''_{k,1,i,j}\cdot\sigma'} \mu_4 \xrightarrow{\sigma''} \mu_5 \xrightarrow{\sigma'''\cdot t''_{k,2,j,i}} \mu_7 \xrightarrow{t'_{k,2,j,i}} \mu_8. \tag{13}$$

Then, it is clear that (13) can be replaced in N_n by

$$\mu_1 \xrightarrow{t_{k,1,i}} \mu' \xrightarrow{\sigma'\cdot\sigma''\cdot\sigma'''} \mu'' \xrightarrow{t_{k,2,j}} \mu_8.$$

Conversely, an occurrence sequence of the form

$$\mu_1 \xrightarrow{t_{k,1,i}} \mu_2 \xrightarrow{\sigma} \mu_3 \xrightarrow{t_{k,2,j}} \mu_4$$

in N_n can be replaced in N' by

$$\mu_1 \xrightarrow{t'_{k,1,i,j}} \mu' \xrightarrow{t''_{k,1,i,j}} \mu_2 \xrightarrow{\sigma} \mu_3 \xrightarrow{t'_{k,2,j,i}} \mu'' \xrightarrow{t''_{k,2,j,i}} \mu_4.$$

Correspondingly, without loss of generality we can change the order of the application of rules of derivations in the grammars G and G'. Therefore, $L(G) = L(G')$.

Now we show that the grammar G' generates a vector language. By the construction of N', $|{}^\bullet q| = |q^\bullet| = 1$ for all $q \in Q'$.

We associate with each pair of rules $r_1, r_2 \in R'$ where $r_1 = \gamma'(t_1)$, $t_1 \in {}^\bullet q$ and $r_2 = \gamma'(t_2)$, $t_2 \in q^\bullet$, $q \in Q'$, the matrix $m = (r_1, r_2)$ and with each rule $r \in R' - \{r' = \gamma'(t') \mid t' \in {}^\bullet Q' \cup Q'^\bullet\}$, the matrix $m = (r)$. We consider a vector grammar $G'' = (V', \Sigma, S, M)$ where M is the set of all matrices constructed above.

Let $S \overset{\pi}{\Rightarrow} w, w \in \Sigma^*$, $\pi = r_1 r_2 \cdots r_n$, is a derivation in G' where $\iota \xrightarrow{\nu} \tau$ with $\nu = t_1 t_2 \cdots t_n = \gamma'^{-1}(\pi)$.

Let ${}^\bullet q = \{t\}$ and $q^\bullet = \{t'\}$ for some $q \in Q'$. If t in ν, i.e., $|\nu|_t > 0$ then t' is also in ν and $|t_1 t_2 \cdots t_k|_t \ge |t_1 t_2 \cdots t_k|_{t'}$ for each $1 \le k \le n$, moreover, by the definition of the final marking, $|\nu|_t = |\nu|_{t'}$. By the bijection γ', $m = (r, r')$, $r = \gamma'(t), r' = \gamma'(t')$ is in π and $|r_1 r_2 \cdots r_k|_r \geq |r_1 r_2 \cdots r_k|_{r'}$ for each $1 \leq k \leq n$ as well as $|\pi|_r = |\pi|_{r'}$. Hence, $\pi \in \mathrm{Shuf}^*(M)$.

Let $S \stackrel{\pi}{\Rightarrow} w, w \in \Sigma^*$, $\pi = r_1 r_2 \cdots r_n \in \mathrm{Shuf}^*(M)$, be a derivation in G'' then again by the bijection γ', $\nu = t_1 t_2 \cdots t_n = \gamma^{-1}(r_1 r_2 \cdots r_n)$ is an occurrence sequence of transitions of N': $\mu_0 \stackrel{\nu}{\rightarrow} \mu_n$. Since the derivation π starts from S (i.e., S is the only symbol at the starting sentential form), $\mu_0(\beta^{-1}(S)) = 1$ and $\mu_0(p) = 0$ for all $p \in P - \{\beta^{-1}(S)\}$. It follows that $\mu_0 = \mu'_0$. On the other hand, from $w \in \Sigma^*$, it follows that $\mu_n(\beta^{-1}(x)) = 0$ for all $x \in V$. From $\pi \in \mathrm{Shuf}^*(M)$, if the rules r, r' of a matrix $m = (r, r')$ in π then $|r_1 r_2 \cdots r_k|_r \geq |r_1 r_2 \cdots r_k|_{r'}$ for each $1 \leq k \leq n$ and $|\pi|_r = |\pi|_{r'}$. By the bijection γ, $|t_1 t_2 \cdots t_k|_t \geq |t_1 t_2 \cdots t_k|_{t'}$ for each $1 \leq k \leq n$ where $t = \gamma^{-1}(r)$, $\gamma^{-1}(r')$ and $|\nu|_t = |\nu|_{t'}$. It follows that $\mu_n(q) = 0$ for all $q \in Q'$. Hence, $\mu_n = \tau'$. □

Theorem 1. *For $k \geq 1$, $\mathbf{PN}_k^{[\lambda]} \subsetneq \mathbf{PN}_{k+1}^{[\lambda]}$.*

Proof. We first prove that $\mathbf{PN}_1^{[\lambda]} \subseteq \mathbf{PN}_2^{[\lambda]}$.

Let $G = (V, \Sigma, S, R, N_1)$ be a 1-PN controlled grammar (with or without erasing rules) where $N_1 = (P \cup \{q\}, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$ 1-PN with the counter place q. Let

$$^\bullet q = \{t_{1,1}, t_{1,2}, \ldots, t_{1,k_1}\}, k_1 \geq 1 \text{ and } q^\bullet = \{t_{2,1}, t_{2,2}, \ldots, t_{2,k_2}\}, k_2 \geq 1$$

where $t_{i,j} = \gamma^{-1}(r_{i,j})$, $r_{i,j} : A_{i,j} \to w_{i,j}$, $1 \leq i \leq 2$, $1 \leq j \leq k_i$ and by definition $^\bullet q \cap q^\bullet = \emptyset$. Let $p_{i,j} = \zeta^{-1}(A_{i,j})$, $1 \leq i \leq 2$, $1 \leq j \leq k_i$.

We set

$$V' = V \cup \{B_{i,j} \mid 1 \leq i \leq 2, 1 \leq j \leq k_i\}$$

where $B_{i,j}$, $1 \leq i \leq 2$, $1 \leq j \leq k_i$, are new nonterminal symbols, introduced for each transition $t_{i,j}$.

For each rule $r_{i,j} : A_{i,j} \to w_{i,j}$, $1 \leq i \leq 2$, $1 \leq j \leq k_i$, we add the new rules $r'_{i,j} : A_{i,j} \to B_{i,j}$, $r''_{i,j} : B_{i,j} \to w_{i,j}$. Let R' be the set of all rules of R and all rules constructed above, i.e.,

$$\begin{aligned} R' = R &\cup \{r'_{1,j} : A_{1,j} \to B_{1,j} \mid \gamma^{-1}(A_{1,j} \to w_{1,j}) \in {}^\bullet q, 1 \leq j \leq k_1\} \\ &\cup \{r''_{1,j} : B_{1,j} \to w_{1,j} \mid \gamma^{-1}(A_{1,j} \to w_{1,j}) \in {}^\bullet q, 1 \leq j \leq k_1\} \\ &\cup \{r'_{2,j} : A_{2,j} \to B_{2,j} \mid \gamma^{-1}(A_{2,j} \to w_{2,j}) \in q^\bullet, 1 \leq j \leq k_2\} \\ &\cup \{r''_{2,j} : B_{2,j} \to w_{2,j} \mid \gamma^{-1}(A_{2,j} \to w_{2,j}) \in q^\bullet, 1 \leq j \leq k_2\}. \end{aligned}$$

We construct a 2-PN controlled grammar $G' = (V', \Sigma, S, R', N_2)$ where V' and R' are defined above and $N_2 = (P', T', F', \varphi', \zeta', \gamma', \mu'_0, \tau')$ is constructed as follows:

$$
\begin{aligned}
P' &= P \cup \{p'_{i,j} \mid 1 \le i \le 2,\, 1 \le j \le k_i\} \cup \{q, q'\},\\
T' &= T \cup \{t'_{i,j}, t''_{i,j} \mid 1 \le i \le 2,\, 1 \le j \le k_i\},\\
F' &= F \cup \bigcup_{i=1}^{2} \bigcup_{j=1}^{k_i} (\{(p_{i,j}, t'_{i,j}), (t'_{i,j}, p'_{i,j}), (p'_{i,j}, t''_{i,j})\}\\
&\quad \cup \{(t''_{i,j}, p) \mid p = \zeta^{-1}(x),\, |w_{i,j}|_x > 0\})\\
&\quad \cup \{(t''_{1,j}, q') \mid 1 \le j \le k_1\}\\
&\quad \cup \{(q', t''_{2,j}) \mid 1 \le j \le k_2\}.
\end{aligned}
$$

For the weight function we set

$$
\varphi'(x, y) = \begin{cases} \varphi(x, y) & \text{if } (x, y) \in F,\\ \varphi(t_{i,j}, p) & \text{if } x = t''_{i,j}, y = p = \zeta^{-1}(x), |w_{i,j}|_x > 0,\\ & \quad 1 \le i \le 2, 1 \le j \le k_i,\\ 1 & \text{otherwise.} \end{cases}
$$

The initial and final markings are defined by $\mu'_0(\zeta'^{-1}(S)) = 1$, $\mu'_0(p) = 0$ for all $p \in P' - \{\zeta'^{-1}(S)\}$ and $\tau'(p) = 0$ for all $p \in P'$.

The inclusion $L(G) \subseteq L(G')$ is obvious, which directly follows from the construction of G'.

Let $S \stackrel{\pi}{\Rightarrow} w, w \in \Sigma^*$, $\pi = r_1 r_2 \cdots r_n$, be a derivation in G' with the occurrence sequence $\nu = t_1 t_2 \cdots t_n = \zeta'^{-1}(\pi)$ of transitions of N_2 enabled at the initial marking μ'_0 and finished at the final marking τ'. It is clear that for some $1 \le i \le 2$, $1 \le j \le k_i$, if a rule $r'_{i,j} : A_{i,j} \to B_{i,j}$ in π, i.e., $|\pi|_{r'_{i,j}} > 0$, then the rule $r''_{i,j} : B_{i,j} \to w_{i,j}$ is also in π, i.e., $|\pi|_{r''_{i,j}} > 0$, moreover, $|\pi|_{r'_{i,j}} = |\pi|_{r''_{i,j}}$. Without loss of generality we can assume that a rule $r''_{i,j}$ is the next to a rule $r'_{i,j}$ in π (as to the nonterminal $B_{i,j}$ only the rule $r''_{i,j}$ is applicable and we can change the order in which the derivation π is used). Then we can replace any derivation steps of the form $x_1 A_{i,j} x_2 \Rightarrow_{r'_{i,j}} x_1 B_{i,j} x_2 \Rightarrow_{r''_{i,j}} x_1 w_{i,j} x_2$ by $x_1 A_{i,j} x_2 \Rightarrow_{r_{i,j}} x_1 w_{i,j} x_2$.

Accordingly, the occurrence sequence $t'_{i,j} t''_{i,j}$, $\mu \xrightarrow{t'_{i,j}} \mu' \xrightarrow{t''_{i,j}} \mu''$, is replaced by $t_{i,j}$, $\mu \xrightarrow{t_{i,j}} \mu''$, where $t_{i,j} = \gamma'^{-1}(r_{i,j})$, $t'_{i,j} = \gamma'^{-1}(r'_{i,j})$ and $t''_{i,j} = \gamma'^{-1}(r''_{i,j})$, $1 \le i \le 2$, $1 \le j \le k_i$. Clearly, $L(G') \subseteq L(G)$.

Let us consider the general case $k \ge 1$. Let $G = (V, \Sigma, S, R, N_k)$ be a k-Petri net controlled grammar where $N_k = (P \cup Q, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$ is a k-Petri net with $Q = \{q_1, q_2, \ldots, q_k\}$. We can repeat the arguments of the proof for $k = 1$ considering q_k instead of q and adding the new counter place q_{k+1}.

For $k \ge 1$, let the language L_k be defined by

$$
L_k = \{\prod_{i=1}^{k} a_i^{n_i} b_i^{n_i} c_i^{n_i} \mid n_i \ge 1, 1 \le i \le k\}.
$$

Then we can show analogously to Example 3 and Lemma 1 that, for $k \geq 1$,

$$L_{k+1} \in \mathbf{PN}_{k+1} \text{ and } L_{k+1} \notin \mathbf{PN}_k.$$

Thus the inclusions are strict. □

5 Closure Properties

We define the following binary form for k-PN controlled grammars, which will be used in some of the next proofs.

Definition 5. *A k-Petri net controlled grammar $G = (V, \Sigma, S, R, N_k)$ is said to be in a binary form if for each rule $A \to \alpha \in R$, the length of α is not greater than 2, i.e., $|\alpha| \leq 2$.*

Lemma 6 (Binary Form)**.** *For each k-Petri net controlled grammar there exists an equivalent k-Petri net controlled grammar in the binary form.*

Proof. Let $G = (V, \Sigma, S, R, N_k)$ be a k-Petri net controlled grammar with $N_k = (P \cup Q, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$.

We denote by $R^{>2}$ the set of all rules of the form $A \to \alpha \in R$ where $|\alpha| > 2$.

For each rule $r = A \to x_1 x_2 \cdots x_n \in R^{>2}$, $x_1, x_2, \ldots, x_n \in V \cup \Sigma$ we set

$$V_r = \{B_1, B_2, \ldots, B_{n-2}\}$$

and

$$R_r = \{A \to x_1 B_1, B_1 \to x_2 B_2, \ldots, B_{n-2} \to x_{n-1} x_n\}$$

where B_i, $1 \leq i \leq n-2$, are new nonterminal symbols, $V_r \cap V_{r'} = \emptyset$ for all $r, r' \in R$, $r \neq r'$, and $V_r \cap V = \emptyset$ for all $r \in R$. Let

$$V' = V \cup \bigcup_{r \in R^{>2}} V_r \text{ and } R' = (R \cup \bigcup_{r \in R^{>2}} R_r) - R^{>2}.$$

We define the context-free grammar $G' = (V', \Sigma, S, R')$ and construct a k-Petri net $N'_k = (P', T', F', \varphi', \zeta', \gamma', \mu'_0, \tau')$ with respect to G' such that

(1) for $A \to \alpha \in R$, $|\alpha| \leq 2$,

$$\gamma^{-1}(A \to \alpha) \in {}^\bullet q \cup q^\bullet \text{ iff } \gamma'^{-1}(A \to \alpha) \in {}^\bullet q' \cup q'^\bullet,$$

(2) for $A \to \alpha \in R$, $|\alpha| > 2$,

$$\gamma^{-1}(A \to \alpha) \in {}^\bullet q \text{ iff } \gamma'^{-1}(B_{n-2} \to x_{n-1} x_n) \in {}^\bullet q', \tag{14}$$

$$\gamma^{-1}(A \to \alpha) \in q^\bullet \text{ iff } \gamma'^{-1}(A \to x_1 B_1) \in q'^\bullet \tag{15}$$

where $\alpha = x_1 x_2 \cdots x_n$, $x_i \in V \cup \Sigma$, $1 \leq i \leq n$.

Let $D : S \xRightarrow{r_1 r_2 \cdots r_k} w, w \in \Sigma^*$ be a derivation in the grammar G. Then $t_1 t_2 \cdots t_k = \gamma^{-1}(r_1 r_2 \cdots r_k)$ is a successful occurrence sequence of transitions in N_k. We construct a derivation D' in the grammar G' from D as follows.

If for some $1 \leq m \leq k$, $r_m : A \to x_1 x_2 \cdots x_n \in R^{>2}$ then we replace the derivation step

$$y_1 A y_2 \underset{r_m}{\Longrightarrow} y_1 x_1 x_2 \cdots x_n y_2$$

by the derivation steps

$$y_1 A y_2 \underset{r_1'}{\Longrightarrow} y_1 x_1 B_1 y_2 \underset{r_2'}{\Longrightarrow} y_1 x_1 x_2 B_2 y_2 \underset{r_3'}{\Longrightarrow} \cdots \underset{r_{n-2}'}{\Longrightarrow} y_1 x_1 x_2 \cdots x_n y_2$$

where $r_i' \in R_{r_m}$, $1 \leq i \leq n-2$. Correspondingly, $\mu_m \xrightarrow{t_m} \mu_{m+1}$ is replaced by

$$\mu_m \xrightarrow{t_1' t_2' \cdots t_{n-2}'} \mu_{m+1}$$

where $t_i' = \gamma'^{-1}(r_i')$, $1 \leq i \leq n-2$. By (14)–(15), the number of tokens produced and consumed by the transitions $t_1', t_2', \ldots, t_{n-2}'$ and the transition t_m are the same. Then D' is a derivation in G', which generates the same word as D does, i.e., $L(G) \subseteq L(G')$.

Inverse inclusion can also be shown using the similar arguments. □

Lemma 7 (Union)**.** *The family of languages* $\mathbf{PN}_k^{[\lambda]}$, $k \geq 1$ *is closed under union.*

Proof. Let $G_1 = (V_1, \Sigma_1, S_1, R_1, N_{k,1})$ and $G_2 = (V_2, \Sigma_2, S_2, R_2, N_{k,2})$ be two k-PN controlled grammars where $N_{k,i} = (P_i \cup Q_i, T_i, F_i \cup E_i, \varphi_i, \zeta_i, \gamma_i, \mu_i, \tau_i)$, $i = 1, 2$ (with the notions of Definition 2). We assume (without loss of generality) that $V_1 \cap V_2 = \emptyset$. We construct the k-PN controlled grammar

$$G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, S, R_1 \cup R_2 \cup \{S \to S_1, S \to S_2\}, N_k)$$

where $N_k = (P, T, F, \varphi, \zeta, \gamma, \mu_0, \tau)$ is defined by

- the set of places: $P = P_1 \cup P_2 \cup Q_1 \cup \{q\}$ where q is a new place;
- the set of transitions: $T = T_1 \cup T_2 \cup \{t_{01}, t_{02}\}$ where t_{01} and t_{02} are new transitions;
- the set of arcs:

$$\begin{aligned} F = F_1 \cup F_2 \cup E_1 \cup \{(q, t_{0i}), (t_{0i}, p_{0i}) \mid i = 1, 2\} \\ \cup \{(t, q_{1i}) \mid (t, q_{2i}) \in E_2, 1 \leq i \leq k\} \\ \cup \{(q_{1i}, t) \mid (q_{2i}, t) \in E_2, 1 \leq i \leq k\} \end{aligned}$$

  where p_{0i} are the places labeled by S_i, i.e., $\zeta_i(p_{0i}) = S_i$, $i = 1, 2$;

- the weight function:

$$\varphi(x,y) = \begin{cases} \varphi_i(x,y) & \text{if } (x,y) \in F_i, i = 1,2, \\ 1 & \text{otherwise;} \end{cases}$$

- the labeling function ζ is defined by

$$\zeta(p) = \begin{cases} \zeta_1(p) & \text{if } p \in P_1 \cup Q_1, \\ \zeta_2(p) & \text{if } p \in P_2 \\ S & \text{if } p = q; \end{cases}$$

- the labeling function γ is defined by

$$\gamma(t) = \begin{cases} \gamma_i(t) & \text{if } t \in T_i, i = 1,2, \\ S \to S_i & \text{if } t = t_{0i}, i = 1,2; \end{cases}$$

- the initial marking:

$$\mu_0(p) = \begin{cases} 1 & \text{if } p = q, \\ 0 & \text{otherwise;} \end{cases}$$

- the final marking: $\tau(p) = 0$ for all $p \in P$.

By the construction of N_k any occurrence of its transitions can start by firing of t_{01} or t_{02} then transitions of T_1 or transitions of T_2 can occur, correspondingly we start a derivation with the rule $S \to S_1$ or $S \to S_2$ then we can use rules of R_1 or R_2.

A string w is in $L(G)$ if and only if there is a derivation $S \Rightarrow S_i \Rightarrow^* w \in L(G_i)$, $i = 1,2$. On the other hand, we can initialize any derivation $S_i \Rightarrow^* w \in L(G_i)$ with the rule $S \to S_i$, $i = 1,2$, i.e., $w \in L(G)$. □

Lemma 8 (Concatenation)**.** *The family of languages* $\mathbf{PN}_k$*,* $k \geq 1$ *is not closed under concatenation.*

Proof. Let L_k and L'_k be two languages, with the same structure but disjoint alphabets, given at the end of the proof of Theorem 1. Then $L_k, L'_k \in \mathbf{PN}_k$ and $L_k \cdot L'_k \notin \mathbf{PN}_k$. □

The next lemma shows that the concatenation of two languages generated by k- and m-PN controlled grammars, $k, m \geq 1$, can be generated by a $(k+m)$-PN controlled grammar.

Lemma 9. *For* $L_1 \in \mathbf{PN}_k^{[\lambda]}$*,* $k \geq 1$ *and* $L_2 \in \mathbf{PN}_m^{[\lambda]}$*,* $m \geq 1$*,*

$$L_1 \cdot L_2 \in \mathbf{PN}_{k+m}^{[\lambda]}.$$

Proof. Let $G_1 = (V_1, \Sigma, S_1, R_1, N_k)$ where $N_k = (P_1, T_1, F_1, \varphi_1, \zeta_1, \gamma_1, \mu_1, \tau_1)$ and $G_2 = (V_2, \Sigma, S_2, R_2, N_m)$ where $N_m = (P_2, T_2, F_2, \varphi_2, \zeta_2, \gamma_2, \mu_2, \tau_2)$ be, respectively, k-Petri net and m-Petri net controlled grammars such that $L(G_1) = L_1$ and $L(G_2) = L_2$. Without loss of generality we assume that $V_1 \cap V_2 = \emptyset$. We set $V = V_1 \cup V_2 \cup \{S\}$ where S is a new nonterminal and

$$R = R_1 \cup R_2 \cup \{S \to S_1 S_2\}.$$

We define a $(k+m)$-PN controlled grammar $G = (V, \Sigma, S, R, N_{k+m})$ with $N_{k+m} = (P, T, F, \varphi, \zeta, \gamma, \mu_0, \tau)$ where

- $P = P_1 \cup P_2 \cup \{p_0\}$ where p_0 is a new place;
- $T = T_1 \cup T_2 \cup \{t_0\}$ where t_0 is a new transition;
- $F = F_1 \cup F_2 \cup \{(p_0, t_0), (t_0, p_1), (t_0, p_2)\}$ where $\zeta_i(p_i) = S_i$, $i = 1, 2$;
- the weight function φ is defined by

$$\varphi(x, y) = \begin{cases} \varphi_i(x, y) & \text{if } (x, y) \in F_i, i = 1, 2, \\ 1 & \text{otherwise;} \end{cases}$$

- the labeling function ζ is defined by

$$\zeta(p) = \begin{cases} \zeta_i(p) & \text{if } p \in P_i, i = 1, 2, \\ S & \text{if } p = p_0; \end{cases}$$

- the labeling function γ is defined by

$$\gamma(t) = \begin{cases} \gamma_i(t) & \text{if } t \in T_i, i = 1, 2, \\ S \to S_1 S_2 & \text{if } t = t_0; \end{cases}$$

- the initial marking:

$$\mu_0(p) = \begin{cases} 1 & \text{if } p = p_0, \\ 0 & \text{otherwise;} \end{cases}$$

- the final marking: $\tau(p) = 0$ for all $p \in P$.

It is not difficult to see that $L(G) = L(G_1)L(G_2)$. □

Lemma 10 (Substitution)**.** *The family of languages* $\mathbf{PN}_k$*,* $k \geq 1$ *is closed under substitution by context-free languages.*

Proof. Let $G = (V, \Sigma, S, R, N_k)$ be a k-PN controlled grammar with k-Petri net $N_k = (P \cup Q, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$. We consider a substitution $s : \Sigma^* \to 2^{\Delta^*}$ with $s(a) \in$ **CF** for each $a \in \Sigma$. Let $G_a = (V_a, \Sigma_a, S_a, R_a)$ be a context-free grammar for $s(a)$, $a \in \Sigma$. We can assume that $V \cap V_a = \emptyset$ for any $a \in \Sigma$ and $V_a \cap V_b = \emptyset$ for any $a, b \in \Sigma$, $a \neq b$.

Let $N_a = (P_a, T_a, F_a, \phi_a, \beta_a, \gamma_a, \iota_a)$ be a cf Petri net with respect to the grammar $G_a, a \in \Sigma$. We define the k-PN controlled grammar

$$G' = (V \cup \Sigma \cup \bigcup_{a \in \Sigma} V_a, \Delta, S, R' \cup \bigcup_{a \in \Sigma} R_a, N'_k)$$

where R' is the set of rules obtained by replacing each occurrence of $a \in \Sigma$ by S_a in R and N'_k is defined by

$$N'_k = (P \cup Q \cup P_\Sigma \cup \bigcup_{a \in \Sigma} P_a, T \cup \bigcup_{a \in \Sigma} T_a, F \cup F_\Sigma \cup \bigcup_{a \in \Sigma} F_a, \varphi', \zeta', \gamma', \mu'_0, \tau')$$

where

- $P_\Sigma = \{p_a \mid a \in \Sigma\}$ is the set of new places;
- $F_\Sigma = \{(t, p_a) \mid \gamma(t) = A \to \alpha, |\alpha|_a > 0, a \in \Sigma\}$ is the set of new arcs;
- the weight function φ' is defined by

$$\varphi'(x, y) = \begin{cases} \varphi(x, y) & \text{if } (x, y) \in F, \\ \phi_a(x, y) & \text{if } (x, y) \in F_a, a \in \Sigma, \\ |\alpha|_a, & \text{if } x = t, y = p_a, (t, p_a) \in F_\Sigma, a \in \Sigma; \end{cases}$$

- the labeling function ζ' is defined by

$$\zeta'(p) = \begin{cases} \zeta(p) & \text{if } p \in (P \cup Q), \\ \beta_a(p) & \text{if } p \in P_a, a \in \Sigma, \\ S_a & \text{if } p = p_a \in P_\Sigma, a \in \Sigma; \end{cases}$$

- the labeling function γ' is defined by

$$\gamma'(t) = \begin{cases} \gamma(t) & \text{if } t \in T, \\ \gamma_a(t) & \text{if } t \in T_a, a \in \Sigma; \end{cases}$$

- the initial marking:

$$\mu'_0(p) = \begin{cases} 1 & \text{if } p = \zeta'^{-1}(S), \\ 0 & \text{otherwise}; \end{cases}$$

- the final marking: $\tau'(p) = 0$ for all $p \in P'$;

Obviously, $L(G') \in \mathbf{PN}_k$. □

Lemma 11 (Mirror Image)**.** *The family of languages* $\mathbf{PN}_k$*,* $k \geq 1$ *is closed under mirror image.*

Proof. Let $G = (V, \Sigma, S, R, N_k)$ be a k-PN controlled grammar. Let

$$R^- = \{A \to x_n \cdots x_2 x_1 \mid A \to x_1 x_2 \cdots x_n \in R\}.$$

The context-free grammar (V, Σ, S, R) and its reversal (V, Σ, S, R^-) have the same corresponding cf Petri net $N = (P, T, F, \phi, \beta, \gamma, \iota)$ as N does not preserve the order of the positions of the output places for each transition. Thus we can also use the k-Petri net N_k as a control mechanism for the grammar (V, Σ, S, R^-), i.e. we define $G^- = (V, \Sigma, S, R^-, N_k)$. Clearly, $L(G^-) \in \mathbf{PN}_k$. □

Lemma 12 (Intersection with Regular Languages)**.** *The family of languages* $\mathbf{PN}_k$*,* $k \geq 1$ *is closed under intersection with regular languages.*

Proof. We use the arguments and notions of the proof of Lemma 1.3.5 in [2]. Let $G = (V, \Sigma, S, R, N_k)$ be a k-Petri net controlled grammar with a k-Petri net $N_k = (P \cup Q, T, F \cup E, \varphi, \zeta, \gamma, \mu_0, \tau)$ (with the notions of Definition 2). Without loss of generality we can assume that G is in a binary form.

Let $\mathcal{A} = (K, \Sigma, s_0, \delta, H)$ be a deterministic finite automaton. We set

$$V' = \{[s, x, s'] \mid s, s' \in K,\ x \in V \cup \Sigma\}.$$

For each rule $r \in R$ we construct the set $R(r)$ in the following way

1. If $r = A \to x_1 x_2,\ x_1, x_2 \in V \cup \Sigma$ then
$$R(r) = \{[s, A, s'] \to [s, x_1, s'][s', x_2, s''] \mid s, s', s'' \in K\}.$$
2. If $r = A \to x,\ x \in V \cup \Sigma$ then
$$R(r) = \{[s, A, s'] \to [s, x, s'] \mid s, s' \in K\}.$$

Further we define the set of rules

$$R_\Sigma = \{[s, a, s'] \to a \mid s' = \delta(s, a), s, s' \in K, a \in \Sigma\}.$$

Let

$$R' = \bigcup_{r \in R} R(r) \cup R_\Sigma.$$

We define the context-free grammar $G_s = (V', \Sigma, [s_0, S, s], R')$ for each $s \in H$. Let $N_s = (P_s, T_s, F_s, \phi_s, \beta_s, \gamma_s, \iota_s)$ be a cf Petri net with respect to the grammar G_s where

$$\begin{aligned}
P_s &= \{[s, p, s'] \mid s, s' \in K, p \in P\},\\
T_s &= \{[s, t, s'] \mid s, s' \in K, p \in P\},\\
F_s &= \{([s_1, x, s_2], [s_1', y, s_2']) \mid s_1, s_2, s_1', s_2' \in K, (x, y) \in F\}.
\end{aligned}$$

The weight function ϕ_s is defined by $\phi([s_1, x, s_2], [s'_1, y, s'_2]) = \phi(x, y)$ where $s_1, s_2, s'_1, s'_2 \in K, (x, y) \in F$.

The functions $\beta_s : P_s \to V'$ and $\gamma_s : T_s \to R'$ are bijections, and

$$\iota_s(\beta_s^{-1}([s_0, S, s])) = 1 \text{ and } \iota_s(p) = 0 \text{ for all } P_s - \{\beta_s^{-1}([s_0, S, s])\}.$$

We set

$$F_Q^- = \{((s, t, s'), q) \mid s, s' \in K, q \in Q \wedge t \in {}^\bullet q\}$$

and

$$F_Q^+ = \{(q, (s, t, s')) \mid s, s' \in K, q \in Q \wedge t \in q^\bullet\}.$$

We construct the k-Petri net

$$N_{k,s} = (P_s \cup Q, T_s, F_s \cup F_Q^- \cup F_Q^+, \varphi_s, \zeta_s, \gamma_s, \mu_s, \tau_s)$$

from N_s where

- the weight function φ_s is defined by

$$\varphi_s([s_1, x, s_2], [s'_1, y, s'_2]) = \varphi(x, y), s_1, s'_1, s_2, s'_2 \in K \text{ and } (x, y) \in F \cup E,$$

- the labeling function ζ_s is defined by

$$\zeta_s([s_1, p, s_2]) = \begin{cases} \beta_s([s_1, p, s_2]) & \text{if } [s_1, p, s_2] \in P_s, \\ \lambda & \text{if } [s_1, p, s_2] \in Q, \end{cases}$$

- the initial marking μ_s is defined by $\mu_s(\beta_s^{-1}([s_0, S, s])) = 1$ and $\mu_s(p) = 0$ for all $(P_s \cup Q) - \{\beta_s^{-1}([s_0, S, s])\}$,

- the final marking τ_s is defined by $\tau_s(p) = 0$ for all $p \in P_s \cup Q$,

and define the k-PN controlled grammar $G'_s = (V', \Sigma, (s_0, S, s), R', N_{k,s})$. Then one can see that $L(G) \cap L(A) = \bigcup_{s \in H} L(G'_s)$. □

The results of the previous lemmas are summarized in the following theorem:

Theorem 2. *The family of languages* $\mathbf{PN}_k$*,* $k \geq 1$*, is closed under union, substitution, mirror image, intersection with regular languages and it is not closed under concatenation.*

References

[1] Crespi-Reghizzi, S. and Mandrioli, D. Petri nets and commutative grammars. Internal Report 74-5, Laboraterio di Calcolatori, Instituto di Elettrotecnica ed Elettromca del Politecnico di Milano, Italy, 1974.

[2] Dassow, J. and Păun, Gh. *Regulated rewriting in formal language theory.* Springer-Verlag, Berlin, 1989.

[3] Dassow, J. and Turaev, S. k-Petri net controlled grammars. In Martín-Vide, C., Otto, F., and Fernau, H., editors, *Language and Automata Theory and Applications. Second International Conference, LATA 2008. Revised Papers*, volume 5196 of *LNCS*, pages 209–220. Springer, 2008.

[4] Esparza, J. Petri nets, commutative context-free grammars, and basic parallel processes. *Fundam. Inf.*, 31(1):13–25, 1997.

[5] Hack, M. *Decidablity questions for Petri nets.* PhD thesis, Dept. of Electrical Engineering, MIT, 1975.

[6] Hopcroft, J.E. and Ullman, J.D. *Introduction to automata theory, languages, and computation.* Addison-Wesley Longman Publishing Co., Inc., 1990.

[7] Marek, V. and Češka, M. Petri nets and random-context grammars. In *Proc. of the 35th Spring Conference: Modelling and Simulation of Systems*, pages 145–152, MARQ Ostrava, Hardec nad Moravicí, 2001.

[8] Martín-Vide, C., Mitrana, V., and Păun, Gh., editors. *Formal languages and applications.* Springer-Verlag, Berlin, 2004.

[9] Murata, T. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.

[10] Petri, C.A. *Kommunication mit Automaten.* PhD thesis, University of Bonn, 1962.

[11] Reisig, W. and Rozenberg, G., editors. *Lectures on Petri Nets I: Basic Models*, volume 1491 of *LNCS*, Berlin, 1998. Springer.

[12] Starke, P.H. *Petri-Netze.* Deutscher Verlag der Wissenschaften, 1980.

[13] Turaev, S. Petri net controlled grammars. In *Third Doctoral Workshop on Mathematical and Engineering Methods in Computer Science, MEMICS 2007*, pages 233–240, Znojmo, Czechia, 2007.

Received 26th January 2009