Regular string-token Petri nets

D. K. Shirley Gloria¹*, S. Devi² and K. Nirmala³

Abstract
String-Token Petri Net can generate Regular Language has been proved here. Also, it has been proved that Regular Language originated by String-Token Petri Net are closed with respect to union and concatenation.

Keywords
String-Token Petri Net (σ), Regular Language (µ), Regular Grammar (ρ), production rule (PR).

AMS Subject Classification
68Q45.

1Department of Mathematics, Dr. Ambedkar Government Arts College, Chennai - 600039, India.
2Department of Mathematics, Government Arts And Science College, Chennai - 600131, India.
3Department of Mathematics, SPM Institute of Science and Technology, Kattankulathur -603203, India.

*Corresponding author: ¹ shirleygloria1976@gmail.com; ² devi.snowdrop@gmail.com; ³gk.nimi@gmail.com

1. Introduction

Carl Adam Petri’s dissertation which was submitted in the year 1962 had brought the notion of Petri Net’s [7].
One of the modifications made on Petri net is colored Petri nets. Later on, in the year 2004, σ was introduced ( [5] and [1]). Using σ, its properties were studied ([2] and [3]). One of its application was derived [4].
In this paper, we introduce a new class of σ, called regular σ. Its properties like union, concatenation are obtained.

2. Basic definitions, examples and a theorem of σ

Definition 2.1. ρ definition can be seen from [6].

Definition 2.2. For every µ, ∃ a ρ ⇒ L = L(G). Converse is also true [6].

Definition 2.3. Definition of evolution regulations can be seen from [3] and [2].

Definition 2.4. σ definition is given in [3].

Definition 2.5. Behaviour of transitions of σ can also be seen from [3].

Example 2.6. A σ originating µ ‘L(N₁)’ is exhibited in figure 1 where L(N₁) = {(ab)^n a : n ≥ 0} is a µ.

In figure 1, N₁ = (P₁, T₁, V₁, F₁, R₁(t), M₁) where P₁ = \{p₁, p₂\}, T₁ = \{t₁, t₂\}, V₁ = \{S₁, a, b\}, R₁(t) = \{t₁ : S₁ → abS₁, t₂ : S₁ → a\}, M₁ = (S₁, ε). After a sequence of firing of transitions of N₁, we obtain a language L(N₁) = {(ab)^n a : n ≥ 0}.

N₁ :

\[ t₁ : S₁ → abS₁ \]

\[ p₁ \]

\[ t₂ : S₁ → a \]

\[ p₂ \]

Figure 1

Example 2.7. A σ originating the µ ‘L(N₂)’ is exhibited in figure 2 where L(N₂) = \{a(ab)^n : n ≥ 1\} is a µ.

In figure 2, N₂ = (P₂, T₂, V₂, F₂, R₂(t), M₂) where P₂ = \{p₃, p₄, p₅\}, with T₂ = \{t₃, t₄, t₅\}, V₂ = \{S₂, S₃, a, b\}, R₂(t) = \{t₃ : S₂ → S₃ab, t₄ : S₃ → S₃ab, t₅ : S₃ → a\}, M₂ = (S₂, ε, ε). After a sequence of firing of transitions of N₂, we obtain a language L₂ = \{a(ab)^n : n ≥ 1\}. 


Theorem 2.8. If L is a μ, then ∃ a σ ‘N’ ∈ L = L(N).

Proof. Let L be a μ originated by a ρ, G = (V,T,S,P) with PR’s of the form A → xB, A → xA → yD, D → zD, B → x, D → y where x, y, z ∈ T* and A, B, D ∈ V.

Erect a σ, N = (P1,T1,V1,F1,R1(t),M1) as follows: Let V1 = V ∪ T be a finite set of alphabets. Let T1 be a finite set of transitions and each t1 ∈ T1 be a tag of the PR’s of P.

**Categorize the PR’s of P as**

1. **T-regulations** (calling it as terminating regulations).
2. **NT-regulations** (calling it as non-terminating regulations).

**Name the PR’s of the form A → xB, A → xA → yD, D → zD as NT-regulations and all other regulations as T-regulations. Among NT-regulations, regulations like D → zD will have a loop structure as it can be applied any number of times. Name these kind of NT-regulations as LNT-regulations. Name other regulations which will not give rise to loop structure as WLN-regulations. Here, regulations like A → xB, A → yD will not generate loops. All T-regulations will lead to terminate (T-regulations like B → x, D → y ).**

Since S is the beginning character of G, erect a place with S as a token in it and name this place as pS (see fig 3). Now, group all S - PR’s of P (Here, A → xB, A → xD, A → x are known as A - PR’s of P). Among all these S-PR’s, group all NT-regulations. These NT-regulations of S- PR’s are known as SNT-regulations and other regulations of S- PR’s as ST-regulations. Let t_{S_{LNT}1} be the tag of the first S_{LNT}-regulation. Its input and output place be pS, t_{S_{LNT}1} can fire any number of times (see figure 4). (If S- PR’s of P have no LNT-regulations, then avoid this erection). If there is another S_{LNT}-regulation, then keeping pS as the input and output place, attach t_{S_{LNT}2} to it (see figure 4) (if there is no second S_{LNT}-regulation, then avoid this erection). Suppose there are more than two S_{LNT}-regulations, a similar way of erection can be done (that is, for all S_{LNT}-regulations, keep pS as the input and output place for the corresponding transitions). Presume for simpleness, there are only two S_{LNT}-regulations.

Let t_{S_{LNT}1} be the tag of the first S_{WLNT}-regulation. Then its input place be pS and let p_{SW1} be the output place of t_{S_{LNT}1} (see fig 5). Similarly, let t_{S_{LNT}2} be the tag of the second S_{WLNT}-regulation. Then its input place be pS and let p_{SW2} be the output place of t_{S_{LNT}2} (see fig 6). Likewise, for the remaining S_{WLNT}-regulation, erect places like p_{SW3}, p_{SW4}, ... with input transitions t_{S_{LNT}3}, t_{S_{LNT}4}, ... If there are no second, third,... S_{WLNT}-regulations, avoid this erection. Presume for simpleness, there are only two S_{WLNT}-regulations. Now, group all ST-regulations. Let t_{ST1} be the tag of first ST-regulation. Its input place be pS and denote its output place as p_{ST1}. If there is a second ST-regulation, let t_{ST2} be the tag of second ST-regulation. Its input place be pS and name its output place as p_{ST2} (See fig. ??). Similar erection can be done for any number of ST-regulations. For simpleness, we presume that there are only two ST-regulations. If there is no ST-regulation, then avoid this erection. Since S is a beginning non-terminal in G, there will be at least one terminal regulation in G. Suppose there are two ST-regulations, then...
\( p_{ST1}, p_{ST2} \) would have terminal strings, so name them as final places.

Now, find the leftmost non-terminal that appears in the string on \( p_{SW1} \). Let it be \( B \). Now, group all \( B \)-PR’s of \( P \). Since \( G \) is a \( \rho \), there will be at least one \( B \)-PR. Among all these \( B \)-PR’s, group all \( NT \)-regulations. If there are \( LNT \)-regulations in \( B \)-PR’s, erect \( t_{BLNT1} \) with \( p_{SW1} \) as its input and output place. For simpleness, presume that there is only one \( BLNT \)-regulation. (If there is no \( BLNT \)-regulation, avoid this erection) (see fig 7). Now, group all \( BLNT \)-regulation. Let it be \( t_{BLNT1} \) with \( p_{SW1} \) as its input and \( p_{SW2} \) as its output place (Similar to \( SWLNT \)-regulations). For simpleness, presume that there is only one \( BWLNT \)-regulation (If there is no \( BWLNT \)-regulation, avoid this erection) (see fig 8). Now, group all \( BWLNT \)-regulations. ie, If \( L \) is a \( \mu \) then \( L = L(N) \).

**Example 2.9.** Consider example 2.6. Let us see the erection of \( N_1 \). \( RG \) for \( L(N_1) \) is \( G = (\{S_1\}, \{(a,b)\}, S_1, P) \) where \( P = \{S_1 \rightarrow abS_1, S_1 \rightarrow a\} \). Among the \( PR \)’s of \( P \), \( S_1 \rightarrow a \) is a \( T \)-regulation and \( S_1 \rightarrow abS_1 \) is a \( NT \)-regulation. Let \( V_1 = \{S_1\} \cup \{a,b\} = \{S_1,a,b\} \). Since \( S_1 \) is the beginning character, erect a place \( p_S \) with \( S_1 \) as the token (see figure 10).

Now, group \( NT \)-regulation of \( S_1 \). We have only one \( NT \)-regulation. \( S_1 \rightarrow abS_1 \) is the \( NT \)-regulation of \( S_1 \). It is \( LNT \)-regulation, since it originates a loop. So, name it as \( S_1LNT \)-regulation. Let \( t_{S_1LNT1} \) be the tag of \( S_1LNT \)-regulation. Its input and output place is \( p_S \) (see figure 11). Now, group \( T \)-regulation of \( S_1 \). There is only one \( T \)-regulation of \( S_1 \), namely \( S_1 \rightarrow a \). Name it as \( S_1T \)-regulation and let \( t_{S_1T1} \) be the tag of \( S_1 \rightarrow a \).
3. Closure Properties

In this section, results on closure properties like union, concatenation are derived.

**Theorem 3.1.** The clan of regular \( \sigma \) language is closed under union.

**Proof.** Let \( N_1 = (P_1, T_1, V_1, F_1, R_1(t), M_1) \) be a \( \sigma \) originates a \( \mu \) ‘\( L_1 \)’ and \( N_2 = (P_2, T_2, V_2, F_2, R_2(t), M_2) \) be a \( \sigma \) originates a \( \mu \) ‘\( L_2 \)’. Now, a \( \sigma, N = (P_1 \cup P_2 \cup \{ p \}, T_1 \cup T_2 \cup \{ \alpha, \beta \}, V_1 \cup V_2 \cup \{ \sigma \}, F_1 \cup F_2 \cup \{ \} \) arcs from \( p \) to \( \alpha \), \( p \) to \( \beta \), \( \alpha \) to initial state of \( N_1, \beta \) to initial state of \( N_2 \}, R_1(t) \cup R_2(t) \cup \{ \alpha : S \rightarrow S_1, \beta : S \rightarrow S_2 \}, M_3 \) can be erected to originate \( L_1 \cup L_2 \). In this erection, remove the initial tokens from \( L_1 \) and \( L_2 \), put \( S \) on the place \( p \) and connect \( p \) and \( N_1 \) by \( \alpha \), also connect \( p \) and \( N_2 \) by \( \beta \). It is exhibited in figure 13.

The same can be extended to any number of \( \mu \). That is, if \( L_1, L_2, L_3, ... , L_n \) are \( \mu \)’s, then \( L_1 \cup L_2 \cup ... \cup L_n \) is also a \( \mu \) (erection of \( \sigma \) for \( L_1 \cup L_2 \cup ... \cup L_n \) is similar to that of \( N \) in figure 13).

Thus, it is yielded that the clan of regular \( \sigma \) Languages is closed under union. \( \Box \)

**Example 3.2.** A \( \sigma \) originating the \( \mu \) ‘\( L_1 \)’ is exhibited in figure 1 where \( L_1 = \{(ab)^n a : n \geq 0\} \). A \( \sigma \) originating \( \mu \) ‘\( L_2 \)’ is exhibited in figure 2 where \( L_2 = \{(a(ab))^n : n \geq 1\} \). From figure 14, it can be seen that a \( \sigma \) ‘\( N_3 \)’ originates \( L_1 \cup L_2 \).

Hence, it is concluded that \( L(N_3) = L(N_1) \cup L(N_2) \). Now, it has been verified that \( \mu \) ‘\( L(N_3) \)’ which is originated by \( \sigma \), \( N_1 \) and \( N_2 \) is closed under union. That is, \( \sigma \) ‘\( N_3 \)’ originates \( L_1 \cup L_2 \).

**Theorem 3.3.** The clan of regular \( \sigma \) Languages is closed under concatenation.

**Proof.** Let \( N_1 = (P_1, T_1, V_1, F_1, R_1(t), M_1) \) be a \( \sigma \) originates a \( \mu \) ‘\( L_1 \)’ and \( N_2 = (P_2, T_2, V_2, F_2, R_2(t), M_2) \) be a \( \sigma \) originates a \( \mu \) ‘\( L_2 \)’. Now, a \( \sigma, N = (P_1 \cup P_2 \cup T_1 \cup T_2 \cup \{ \alpha \}, V_1 \cup V_2 \cup \{ \} \), \( F_1 \cup F_2 \cup \{ \} \) a place with terminal string of \( N_1 \) to \( \alpha \), \( \alpha \) to the initial place of \( N_2 \}, R_1(t) \cup R_2(t) \cup \{ \alpha : S \rightarrow S_1, \beta : S \rightarrow S_2 \}, M_3 \) can be erected to generate \( L_2 \cdot L_1 \). It is exhibited in figure 15.

From figure 15, it can be seen that on the place with terminal string (say \( f_1 \)), strings of \( L_1 \) will be deposited after all the sequence of firing of transitions of \( N_1 \). So, \( L_1 \) can be obtained as a string that is deposited on \( f_1 \). Connect \( f_1 \) and initial place of \( N_2 \) by a new transition \( \alpha : S \rightarrow S_2 \). Remove initial token from \( N_2 \). When \( \alpha \) fires, \( S_2 \cdot L_1 \) will be deposited on the initial place of \( N_2 \). After all the sequence of firing of transitions of \( N_2 \), \( L_2 \cdot L_1 \) will be obtained as a string that is
deposited on $f_2$ (a place with terminal string on $N_2$). Similarly $L_1, L_2$ is obtained by taking $N_2$ first and then $N_1$. Also it can be seen that $L_1, L_2 \neq L_2, L_1$.

The same can be extended to any number of $\mu$. That is, if $L_1, L_2, L_3, \ldots, L_n$ are $\mu$’s, then $L_1, L_2, L_3, \ldots, L_n$ is also a $\mu$ (erection of $\sigma$ for $L_1, L_2, L_3, \ldots, L_n$ is similar to that of $N$ in figure 15).

Thus, it is yielded that the clan of regular $\sigma$ Languages is closed under concatenation.

Example 3.4. A $\sigma$ originating the $\mu$ ‘$L_1$’ is exhibited in figure 1 where $L_1 = \{ (ab)^n a : n \geq 0 \}$. A $\sigma$ originating $\mu$ ‘$L_2$’ is exhibited in figure 2 where $L_2 = \{ a(ab)^n : n \geq 1 \}$. Now, it can be seen from figure 16 that $\mu$ which is originated by $\sigma$’s $L_1$ and $L_2$ is closed under concatenation. That is, $\sigma$ ‘$N_4$’ originates $L_2, L_1$. In figure 16, $S_2$ will be removed from figure 2 and $t_a : \in \rightarrow S_2$ is taken as a transition between $p_2$ and $p_3$.

4. Conclusion

It can be concluded that every $\mu$ can be originated by $\sigma$. Also regular $\sigma$ is closed with respect to union and concatenation.

References