Journal of Analysis and Applications
Vol. 22 (2024), No.1, pp.15-30
ISSN: 0972-5954

URL : www.sasip.net

Semi-binary operations on β-languages

A. Jain*, S. Jain and G.C. Petalcorin, Jr.

Abstract. β-languages of order n have been introduced by the authors of [9]. The authors of [5] made a study of various closure properties of β-languages. In this paper, we first show that the family of β-languages of order n is not closed under intersection, difference and complementation. We then further introduce the notion of semi-binary operation on a non-empty set G with respect to its non-empty subject H using the binary operation on H. Finally, we show that the operations of intersection and difference are semi-binary operations on the class L of β-languages of order n with respect to class R of regular languages which is a non-empty subset of L and is closed under these operations.

AMS Subject Classification (2020): 68Q45

Keywords: β-languages, β-grammar, context-free grammar, context-free langauges (CFL)

1. Introduction

The authors of [9] introduced the concept of β-grammar and β-languages of order n and proved their equivalence with the semi-deterministic push-down automata (SDPDA) languages of order n in [6, 9]. The class of β-languages of order n lies between non-deterministic context-free languages and deterministic context-free languages. Since the class of deterministic context-free languages contains all regular languages, therefore, the class of β-languages of order n also contains all regular languages.

*Corresponding author

Again, the class of β-languages (or SDPDA languages) of order n includes the syntax of most programming languages including the mechanics of the parser in a typical compiler. Since each use of a production rule introduces exactly one terminal, including the null symbol "λ" into a sentential form, therefore, a string of length "k" has a derivation of at most "$(n+1)k$" steps using β-grammar of order n.

Further, the authors of [5] have shown that the class of β-languages of order n is closed under union, concatenation and star-closure operations. In this paper, we show that the family L of β-languages of order n is not closed under intersection, difference and complementation and therefore intersection and difference are not binary operations on L. Also, we know that the class R of regular languages is closed under intersection and difference.

Motivated by this, we introduce the notion of *semi-binary operations* on a non-empty set G with respect to its non-empty subset H using the binary operation on H. We show that the operations of intersection and difference are semi-binary operations on the class L of β-languages of order n with respect to its non-empty subset class R of regular languages.

2. Preliminaries

In this section, we present some definitions available in the literature:

Definition 2.1 [14].

(i) A finite non-empty set Σ is called an **"alphabet"**.

(ii) A **"string"** is a finite sequence of symbols from the alphabet.

(iii) The **"concatenation"** of two strings "u" and "v" is the string obtained by appending the symbols of "v" to the right end of "u".

(iv) The **"length"** of string w denoted by $|w|$ is the number of symbols in the string.

(v) An **"empty string"** is a string with no symbol in it. It is denoted by λ and $|\lambda| = 0$.

(vi) If Σ is any alphabet, then "Σ^k"$(k \geq 0)$ denotes the set of all strings of length k with symbols from Σ.

(vii) The set of all strings over an alphabet Σ is denoted by Σ^*, i.e.

$$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \cdots .$$

(viii) The set of all non-empty strings from the alphabet Σ is denoted by Σ^+ and is given by

$$\Sigma^+ = \Sigma^* - \{\lambda\} = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \cdots$$

(ix) A **"language"** L over an alphabet Σ is defined as a subset of Σ^*.

(x) A string in a language L is called a **"sentence"** of L.

(xi) The **"union"**, **"intersection"** and **"difference"** of two languages are defined in the set theoretic way.

(xii) The **"complement"** of a language L over an alphabet Σ is defined as $\bar{L} = \Sigma^* - L$.

(xiii) The **"concatenation"** of two languages L_1 and L_2 is the set of all strings obtained by concatenating a string of L_1 with a string of L_2, i.e.

$$L_1 L_2 = \{uv | u \in L_1 \text{ and } v \in L_2\}.$$

(xiv) The **"star-closure"** of a language L is defined as

$$L^* = L^0 \cup L^1 \cup L^2 \cup \cdots .$$

Also, the **"positive-closure"** of a language L is given by

$$L^+ = L^1 \cup L^2 \cup \cdots .$$

(xv) A **"grammar"** G is defined as a quadruple

$$G = (V, T, S, P),$$

where V is a finite set of objects called **"variables"**, T is a finite set of objects called **"terminal symbols"** with $V \cap T = \phi$, $S \in V$ is a special symbol called the **"start"** symbol, P is a finite set of **"productions"** of the form $x \to y$ where $x \in (V \cup T)^+$ and $y \in (V \cup T)^*$.

(xvi) We say that the string $w = uxv$ **"derives"** the string $z = uyv$ if the string z is obtained from w by applying the production $x \to y$ to w. This is written as $w \Rightarrow z$. If

$$w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n,$$

then we say that w_1 derives w_n and write $w_1 \Rightarrow^* w_n$.

(xvii) Let $G = (V, T, S, P)$ be a grammar. Then the **"language"** $L(G)$ generated by G is given by

$$L(G) = \{w \in T^* | S \Rightarrow^* w\}.$$

(xviii) If $w \in L(G)$, then the sequence

$$S \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n \Rightarrow w.$$

is a **"derivation"** of the sentence w. The strings $S, w_1, w_2, \cdots, w_n$ which contain variables as well as terminals are called **"sentential forms"** of the derivation.

(xix) A grammar $G = (V, T, S, P)$ is said to be **"right-linear" (resp. left-linear)** if all productions in G are of the form

$$A \to xB \text{ (resp.} A \to Bx),$$

or

$$A \to x,$$

where $A, B \in V$ and $x \in T^*$. A **"regular grammar"** is one that is either right linear or left linear.

Definition 2.2 [12]. A semi-deterministic virtual finite automaton (SD-VFA) of order (s, t) is a finite automaton that can make atmost "s"$(s \geq 1)$ transitions on receiving a real input and atmost "t"$(t \geq 0)$ transitions on virtual input (or no input). (Zero transition means the automaton remains in the same state).

Formally, we may define semi-deterministic virtual finite automaton (SD-VFA) of order (s, t) as follows:

Definition 2.3 [12]. A semi-deterministic virtual finite automaton (SD-VFA) of order (s, t) consists of

1. A finite set of states (including the dead state) often denoted by Q.
2. A finite set of input symbols including the empty string symbol ϵ. This is often denoted by $\Sigma \bigcup \{\epsilon\}$. Σ is called real alphabet.
3. A transition function $\delta_{(s,t)}$ that takes as arguments a state and an input symbol. On real input symbol i.e. if the symbol is a member of real alphabet Σ, $\delta_{(s,t)}$ returns a set of atmost "s" states while on virtual input ϵ, the transition function returns a set of atmost "t" states.
4. A start state S which is one of the states in Q.

5. A set of final or accepting states F. The set F is a subset of Q. Dead state is never an accepting state and it makes a transition to itself on every possible input symbol.

We can also denote an SDVFA of order (s,t) by a "five tuple" notation:

$$V = (Q, \Sigma \bigcup \{\epsilon\}, \delta_{(s,t)}, q_0, F)$$

where V is the name of the SDVFA, Q is the set of states, $\Sigma \bigcup \{\epsilon\}$ is the set of input symbols, $\delta_{(s,t)}$ is the transition function, q_0 is the start state and F is the set of accepting states.

Definition 2.4 [9]. A context-free grammar $G = (V, T, S, P)$ is said to be a **"β-grammar of order n"** $(n \geq 1)$ if all productions in P are of the form $A \to ax$ where $a \in T \cup \{\lambda\}$ and $x \in V^*$ and any pair (A, a) occurs at most "n'' times in P. A β-grammar of order n is denoted by $\beta(n)$.

Definition 2.5 [9]. The language generated by a β-grammar of order n is called a **"β-language of order n"**.

Definition 2.6 [9]. A "semi-deterministic pushdown automata(SDPDA) of order n" is a PDA that can make atmost "n"$(n \geq 1)$ transitions corresponding to a given input symbol (real or virtual input λ) and stack top symbol.

We now formally define a "semi-deterministic pushdown automata (SDPDA) of order n as follows:

Definition 2.7 [9]. A "semi-deterministic pushdown automata(SDPDA) of order n" consists of

1. A finite set of states denoted by Q.

2. A finite set of input symbols including the empty string symbol λ. This is denoted by $\Sigma U\{\lambda\}$. Σ is called real input alphabet.

3. A finite set of stack symbols denoted by $\Gamma \cup \{\lambda\}$ where Γ is the real stack alphabet.

4. A transition function δ, that takes as arguments a state, an input symbol and stack top symbol and returns atmost n pairs of the form (q, x) where q is the next state of the control unit and x is a string of stack symbols which is put on the top of the stack in place of the single stack symbol there before with left most symbol of the string to be placed highest on the stack.

5. An initial start state $q_0 \in Q$.

6. A stack start symbol $Z_0 \in \Gamma$.

7. A set of final states $F \subseteq Q$.

Note that δ is defined in such a way that it needs a stack symbol and no move is possible if the stack is empty.

We can also denote an SDPDA of order n by a "sep-tuple" notation as

$$M(n) = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F),$$

where $M(n)$ is the name of the SDPDA of order n, Q is the finite set of internal states of the control unit, Σ is the set of real input symbols, Γ is the set of stack symbols, δ is the transition function from $Q \times (\Sigma U\{\lambda\}) \times \Gamma$ to finite subset of $Q \times \Gamma^*$ of order atmost n, $q_0 \in Q$ is the initial state of the control unit, $Z_0 \in \Gamma$ is the stack start symbol and $F \subseteq Q$ is the set of the final states.

3. Intersection, difference and complement of β-languages of order n

In this section, we show that the the family of β-languages of order n is not closed under intersection, difference and complementation.

Theorem 3.1. *The class of β-languages of order n is not closed under intersection.*

Proof. Consider the two languages

$$L_1 = \{0^n 1^n 2^i | n \geq 1,\ i \geq 1\},$$

and

$$L_2 = \{0^i 1^n 2^n | n \geq 1,\ i \geq 1\}.$$

The languages l_1 and L_2 are both β-languages of order 2 as shown below:

A β-grammar of order 2 for languages L_1 is given by:

$$S \to AB;$$

$$A \to 0AC;$$

$$C \to 1;$$

$$A \to 0C;$$

$$B \to 2B;$$

$$B \to 2.$$

A β-grammar of order 2 for language L_2 is given by

$$S \to AB;$$

$$A \to 0A;$$

$$A \to 0;$$

$$B \to 1BC;$$

$$B \to 1C;$$

$$C \to 2.$$

Now, $L_1 \cap L_2$ is given by

$$L_1 \cap L_2 = \{0^n 1^n 2^n | n \geq 1\}.$$

Then using the pumping lemma, it can be shown that $L_1 \cap L_2$ is not a context-free language and hence not a β-language of order 2.

Thus, the family of β-languages of order n is not closed under intersection. □

Theorem 3.2. *The family of β-grammar of order n is not closed under complementation.*

Proof. Let L_1 and L_2 be two β-languages of order n. Assume on the contrary that the family of β-languages of order n is closed under complementation.

Consider

$$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}} \tag{1}$$

Then the right hand side of (1) would be a β-languages of order n. This gives a contradiction in view of Theorem 3.1.

Hence, the family of β-languages of order n is not closed under complementation. □

Theorem 3.3. *The difference of two β-languages of order n is not necessarily a β-language of order n.*

Proof. Assume on the contrary that if L_1 and L_2 are two β-languages of order n , then $L_1 - L_2$ is also a β-language of order n.

Now, we make an observation that if Σ is an alphabet having n elements, then the star-closure Σ^* is a β-language of order n.

For example, if $\Sigma = \{a, b\}$, then Σ^* is a β-language of order 2. A β-grammar $G = (V, T, P, S)$ (where $V = \{S, A\}$ and $T = \Sigma = \{a, b\}$) of order 2 for Σ^* is given by the following production rules:

$$S \to A;$$

$$A \to aA;$$

$$A \to bA;$$

$$A \to a;$$

$$A \to b;$$

$$A \to \lambda.$$

The assumption in the preceding paragraph implies that $\Sigma^* - L$ is always a β-language of order n whenever L is a β-language of order n.

However,

$$\Sigma^* - L = \overline{L} \quad \text{when } L \text{ is a proper subset of } \Sigma^*.$$

Thus we get $\bar{L}$ is a β-language of order n whenever L is a β-language of order n. A contradiction in view of Theorem 3.2.

Hence the difference of two β-languages of order n is not necessarily a β-language of order n. □

4. Semi-binary operations on β-languages of order n

In the preceding section, we have shown that the operations of intersection and difference are not binary operations on the class of β-languages of order n.

In this section, we first introduce the notion of semi-binary operation on a non-empty set G with respect to its non-empty subset H using the binary operation on H. We then show that the operations of intersection and difference are semi-binary operations on the class of L of β-language of order n with respect to its non-empty subset class R of regular languages which is closed under these operations.

We now define a *semi-binary operation on a non-empty* G *with respect to its non-empty subset* H as follows:

Definition 4.1. Let G be a non-empty set. Let H be a non-empty subset of G. A **semi-binary** operation $*$ on G with respect to H is a function on $G \times H$ satisfying the following properties:

(i) $g \times h \in G$ for all $g \in G,\ h \in H$; and

(ii) $h \times h \in H$ for all $h \in H$, i.e. "$*$" is a binary operation on H.

The set G is called a *semi-binary H-Superset* with respect to operation "$*$".

Example 4.2. Let Q be the set of all positive rational numbers and Q^+ be the set of all positive rational numbers. Let "/" denote the usual division operation on Q. Then "/" is not a binary operation on Q since "$a/0$" does not belong to Q for $a \in Q$. Also, "/" is a binary operation on Q^+. Therefore, "/" is a semi-binary operation on Q with respect to Q^+.

Again, we know that the family of regular languages is closed under intersection and difference. We now first prove that the intersection of a β-language of order n with a regular language is again a β-language of order n.

Theorem 4.3. *If L is a β-language of order n and R is a regular language, then $L \cap R$ is a β-language of order n.*

Proof. We shall use the semi-deterministic pushdown automaton (SDPDA) of order n representation of β-language of order n [6, 9] and semi-deterministic virtual finite automaton (SDVFA) or order (1,0) representation for regular language [10, 12].

Let

$$M(n) = (Q_M, \Sigma, \Gamma, \delta_M, q_M, Z_0, F_M)$$

be an SDPDA of order n that accepts the given β-language L by final state.

Let $N = (Q_N, \Sigma, \delta_N, q_N, F_N)$ be an SDVFA of order (1,0) for the given regular language R.

We construct a new SDPDA say M' of order n for $L \cap R$ by running M and N in parallel as follows:

$$M' = (Q_M \times Q_N, \Sigma, \Gamma, \delta, (q_M, q_N), Z_0, F_M \times F_N),$$

where $(\delta((q,p), a, X)$ is defined to be the set of all pairs of the form $((r,t),\gamma)$ such that

(i) $t = \hat{\delta_N}(p, a)$, and

(ii) pair (r, γ) is in $\delta_M(q, a, X)$.

In other words, the SDPDA M' makes the same move as the SDPDA M does but also carry along the state of the SDVFA N of order (1,0) in the second component of its state.

Also, M' simulates moves of M' on virtual input λ without changing the state of SDVFA N. When M' makes a move on input symbol "a", M simulates that move and also simulates $N's$ change of state on input a.

It is an easy induction on the number of moves made by the SDPDAs M and M' that

$$(q_M, w, Z_0) \vdash^* (q, \lambda, \gamma)$$

iff

$$((q_M, q_N), w, Z_0) \vdash^*_{M'} ((q, p), \lambda, \gamma),$$

where $p = \hat{\delta}(q_N, w)$.

Since (q, p) is an accepting state of M' iff q is an accepting state of M and p is an accepting state of N, we conclude that M' accepts w iff both M and N accepts w i.e. $w \in L \cap R$.

Hence $L \cap R$ is a β-language of order n. □

Theorem 4.4. *If L is a β-language of order n and R is a regular language, then $L - R$ is a β-language of order n.*

Proof. Since $L - R = L \cap \overline{R}$ and we know that regular languages are closed under complementation, therefore, using Theorem 4.3., we get $L - R$ is a β-language of order n. □

Theorem 4.5. *The operations of intersection and difference are semi-binary operations on the lass of L of β-languages of order n with respect to class R of regular languages.*

Proof. We know that the class of R of regular languages is closed under intersection and difference and also that R is a non-empty subset class of the class L of β-languages of order n. Then from Definition 4.1., Theorems 4.3 and 4.4, we get the result. □

5. Conclusion

In this paper, we have shown that the family of β-languages of order n is not closed under intersection, difference and complementation. We have further introduced the notion of semi-binary operation on a non-empty set G with respect to its non-empty subject H using the binary operation on H.

Finally, we have also shown that the operations of intersection and difference are semi-binary operations on the class L of β-languages of order n with respect to class R of regular languages which is a non-empty subset of L.

Acknowledgment. The authors would like to express their sincere gratitude to the referees for their valuable suggestions and comments which improved the paper.

References

[1] A.V. Aho and J.D. Ullman, The Theory of Parsing, Translation and Computing, Vol. 1, Englewood Cliffs. N.J.: Prentice Hall, 1972.

[2] M.A. Harriosn, Introduction to Formal languages Theory, Addison Wesley, Reading, Mass., 1978.

[3] J.E. Hopcroft, R. Motwani and J.D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education, Addison-Wesley, Reading, Mass., 2004.

[4] T.W. Hungerford, Algebra, Springer Private Ltd., 2005.

[5] A. Jain, S. Leeratanavalee, S. Jain and G.C. Petalcorin, Jr., *Semigroup and monoid structures of β-languages*, J. Algeb. and Applied Mathematics, 21 (2023), 111-121.

[6] A. Jain, S. Jain, H. Ghazwani and G.C. Petalcorin, Jr., *Formulation of β-language from the semi-deterministic pushdown automata(SDPDA) of order n*, J. Algeb. and Applied Mathematics, 21 (2023), 61-78.

[7] A. Jain, S. Jain and G.C. Petalcorin, Jr., *Construction of α-language from the language of a QDPDA of order* "n", J. Anal. and Appl., 20 (2022), 135-150.

[8] A. Jain, K.P. Shum, G.C. Petalcorin, Jr. and K.-S. Lee, *α-grammar and quasi-deterministic pushdown automata (QDPDA) of order "n"*, J. Algeb. and Applied Mathematics, 18 (2020), 99-114.

[9] A. Jain, G.C. Petalcorin, Jr. and K.-S. Lee: *Semi-deterministic push-down automata (SDPDA) of order "n" and β-languages*, J. Algeb. and Applied Mathematics, 14 (2016), 27-40.

[10] A. Jain, *Semi-deterministic Virtual Finite Automaton (SDVFA) of order (s,t) and Regular Grammar*, Ars Combinatoria, 115 (2014), 187-196.

[11] A. Jain,*Equivalence of Semi-deterministic Virtual Finite Automaton (SDVFA) with DFA, VDFA, NFA and ϵ-NFA*, Ars Combinatoria, 117 (2014), 363-373.

[12] A. Jain, *Semi-deterministic Virtual Finite Automaton (SDVFA) of order (s,t)*, J. Compt. Math. Optim., 5 (2009), 1-22.

[13] W. Kuich and S. Arto, Semirings, automata, languages, Springer Verlag, 1986.

[14] P. Linz, An Introduction to Formal languages and Automata, Narosa Publishing House, 2009.

[15] I. Petre and S. Arto, *Algebraic systems and pushdown automata*, In M. Droste, W. Kuich and H. Vogler (eds.), Handbook of Weighted Automata, Springer, Chapter 7, 2009, pp.257-289.

[16] G.E. Revesz, Introduction to Formal Languages, McGraw-Hill, 1983.

[17] A. Salomaa, Computations and Automata, in Encyclopedia of Mathematics and Its Applications, Cambridge University Press, Cambridge, 1985.

Department of Mathematics
Shanxi Normal University
P.R. China
E-mail: jainarihant@gmx.com

Department of Mathematics
Shanxi Normal University
P.R. China
E-mail: sapnajain@gmx.com

Department of Mathematics and Statistics
College of Science and Mathematics
MSU-Iligan Institute of Technology
Tibanga, Iligan City
Philippines
E-mail: gaudencio.petalcorin@g.msuiit.edu.ph

(*Received: October, 2023; Revised: March, 2024*)