Jurnal
Teknologi

Full Paper

VARIOUS REGULATIONS ON THE EXISTENCE OF *N*-TH ORDER LIMIT LANGUAGE

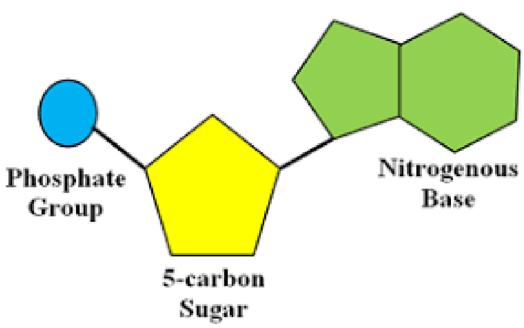
Siti Hajar Mohd Khairuddin, Muhammad Azrin Ahmad*, Noraziah Adzhar

Centre for Mathematical Sciences, Universiti Malaysia Pahang, 26300 Lebuhraya Tun Razak, UMP Gambang, Pahang, Malaysia

Article history
Received
22 October 2021
Received in revised form
23 January 2022
Accepted
26 January 2022
Published Online
20 April 2022

*Corresponding author
azrin@ump.edu.my

Graphical abstract



Abstract

In molecular biology, recombinant Deoxyribonucleic acid technology has ignited an increase in the interest of new researches. Moreover, the splicing system has generated enthusiasm in developing computational models collaborating with formal language theory. Formal language theory tends to be a natural structure for formalising and investigating DNA computing models from this viewpoint. The work of several researchers who added control structures to the splicing formalism, thus creating universal computation systems, has provided additional inspiration for the study of splicing systems. A splicing system is a conventional model of a set of dsDNA that undergoes the cutting and pasting process with the presence of restriction enzyme and ligase. Previously, an introduction of the *n*-th order limit language is presented and discussed. The properties and the characteristics of the *n*-th order limit language are developed and also explained by using examples and sort into a few cases. However, the regulation of the existence of the *n*-th order limit language is left unintended. In this paper, the factors that restrict the formation of the *n*-th order limit language are discussed. Several restrictions applied are the length of the rules are not equal and same rules applies on several crossing sites of the initial strings. In addition, some examples are given to show the restriction of the formation of *n*-order limit language.

Keywords: DNA, formal language theory, splicing system, splicing language, *n*-th order limit language

Abstrak

Dalam bidang biologi molekul, teknologi asid deoksiribonukleik rekombinan telah membangkitkan satu peningkatan minat terhadap penyelidikan baharu. Di samping itu, sistem hiris-cantum telah mencetuskan fenomena dalam pembangunan model pengkomputeran dengan menghubungkaitkan teori bahasa formal. Dari sudut pandangan ini, teori bahasa formal merupakan satu struktur semulajadi untuk pemformalan dan penyelidikan model komputasi DNA. Dengan usaha daripada beberapa penyelidik yang telah menambah struktur-struktur kawalan pada formalisme hiris-cantum, sistem-sistem komputasi yang lebih umum telah dapat dihasilkan. Hal ini telah berjaya memberikan inspirasi kepada penyelidikan sistem hiris-cantum. Sistem hiris-cantum adalah satu model konvensional kepada satu set dsDNA yang melalui proses potong dan tampal dengan kehadiran enzim pembatas dan ligase. Pengenalan tentang bahasa batas berperingkat ke-*n* ada dibentangkan dan dibincangkan sebelum ini. Sifat-sifat dan ciri-ciri bahasa batas berperingkat ke-*n* telah dibangunkan dan dihuraikan dengan menggunakan beberapa contoh dan telah dikategorikan kepada beberapa kes. Walau bagaimanapun, peraturan mengenai kewujudan bahasa batas berperingkat ke-*n* telah diabaikan secara tidak sengaja. Kertas kerja penyelidikan ini ada membincangkan faktor-faktor yang menghalang pembentukan bahasa batas berperingkat ke-*n*. Faktor-faktor itu adalah penggunaan peraturan yang tidak sama panjang dan penggunaan peraturan yang sama terhadap beberapa lintasan dalam jujukan awal. Selain itu, beberapa contoh juga diberikan untuk menunjukkan had dalam pembentukan bahasa batas berperingkat ke-*n*.

Kata kunci: DNA, teori bahasa formal, sistem hiris-cantum, bahasa hiris-cantum, bahasa batas berperingkat ke-*n*

1.0 INTRODUCTION

Deoxyribonucleic acid (DNA) is a molecule made up of double polynucleotide chains that coil together to form a double helix that contains coded information for the formation, operating, reproduction, and breeding of all organisms [1]. DNA is made up of nucleotides [2]. Covalent bonds are used to link nucleotides together in a chain. The structure of DNA is illustrated Figure 1.

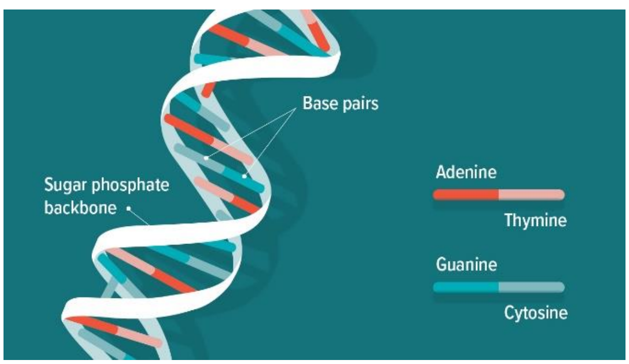


Figure 1 The Structure of DNA [1]

The Head splicing system rapidly grows and is significant to various fields such as information macromolecules and formal language theory [3]. The Head splicing system is also known as the deoxyribonucleic acid (DNA) splicing system [4]. Splicing system is the process of cut and paste double-stranded DNA (dsDNA) with the presentation of restriction enzymes is called the splicing system [6]. The DNA molecules from the splicing system operate as strings in formal language theory [5]. Several types of splicing systems have been introduced, such as in 1996, Paun and Pixton have introduced the splicing system, the Paun splicing system [7] and Pixton splicing system [8], respectively. A few years later, Goode and Pixton also introduce the Goode-Pixton (G-P) splicing system [9]. Furthermore, Yusof-Goode (Y-G) splicing systems [10] were also introduced by Yusof in 2012. The concept for proposing this model is based on the properties of the restriction enzyme itself and the transparent behaviour of the actual process of DNA replication [11]. Formal language theory is used to examine the splicing system in which the resulting molecules from the splicing process develop a language known as splicing language [9].

There are several types of splicing language such as two stages [12], limit [9], transient[13], simple [14], first-order limit [9], second-order limit [15], *n*-th order limit language [16], fuzzy [17] and crips language [17] which different splicing models can generate. This study focuses on the limit language, first proposed by Goode [9] after examining the behaviour of DNA molecules in the last stage. The wet splicing method, which is a laboratory method for splicing DNAs, was then carried out. The DNA molecules were inspected and studied once the process was completed or reached a limit known as limit language. Furthermore, this language is utilised to create the *n*-th order limit language.

Previously, *n*-th order limit language has been introduced by Goode in [9]. According to Goode's definition, the order of the limit language is established by comparing the language generated. The number of rules involved in the splicing system defines the *n*-th order limit language, according to the revised definition [16]. Given that the rules must be of the same length, it guarantees that the string combination matches the pattern shown in the main result. Thus, the *n*-th order limit language is proposed by the number of rules that arise from the string combination created.

The following section provides the basic definitions used in this paper.

2.0 PRELIMINARIES

The concepts related to splicing languages are described in this section. The Head splicing system is given in the formal definition too.

Definition 1: Alphabets, A [18]
An alphabet A is a finite non-empty set of symbols.

Definition 2: Strings [18]
A string is a finite sequence of symbols from the alphabet, A.

Definition 3: Language, L [18]
Language is a set of strings are chosen from A^*, where A is a particular alphabet.

Definition 4: Palindrome [10]
If the sequence from the left to the right side of the upper single strand equals the sequence from the right to the left side of the lower single strand, the string I of dsDNA is said to be a palindrome.

Definition 5: Head Splicing System 4 [3]
A splicing system $S=(A,I,B,C)$ consists comprises four different sets of elements A,I,B and C, elaborated as follows:

A is a set of alphabet,
I is a set of initial string,
B is a set of rules, representing a $5'$-overhang or blunt end,
C is a set of rules, representing a $3'$-overhang.

Next, the definition of limit, second-order limit language and *n*-th order limit language is presented.

Definition 6: Limit Language [9]
Limit language is referred to first-order limit language and described as a splicing language that results from the remaining molecules after the splicing system has reached its equilibrium state or is completed.

Definition 7: Second Order Limit Language [19]
A splicing language is called a second-order limit language if the set of string produced in second order limit words is different from the set of strings of the splicing language. In other words, $L_2(S)\cap L(S)=\varnothing$, $L_2(S)\cap L_1(S)=\varnothing$, and $L_2(S)\not\subset L(S)$.

The derivation of the *n*-th order limit language is obtained from the idea of the second-order limit language and *n*-th order limit language from Goode [9]. The following is the definition of *n*-th order limit language.

Definition 8: *n*-th order Limit Language [9]
Let L_{n-1} be the set of second-order limit words of L, the set L_n of *n*-order limit words of L to be the set of the first-order limit of L_{n-1}. We obtain L_n from L_{n-1} by deleting the words that are transient in L_{n-1}.

Then, the original definition is improvised as follows.

Let $L(S)$ be the splicing language of the splicing system S. We then define $L_n(S)$ such that n represents the order of the limit language. Initial strings of the splicing system S consist of cxd, where c and d are the left and right contexts, respectively and x is the crossing site. The *n*-th order limit language is defined by the number of rules that act on each crossing site, x in which the set of rules is different from each other. Note that the rules must have the same length of crossing sites. A splicing language is called *n*-th order limit language, denoted by $L_n(S)$, if the set of string produce in $L_n(S)$ is different from the set of strings of $L_1(S), L_2(S), \ldots, L_n(S)$ such that $\bigcap_{n=1}^{n} L_n(S)=\varnothing$ and $L_1(S)\not\subset L_2(S)\not\subset \ldots \not\subset L_n(S)$. [16]

3.0 RESULTS AND DISCUSSION

In this section, two lemmas and a theorem are given on the non-existence of *n*-th order limit language in the Head splicing system. The Head splicing system is used throughout the research because it is the pioneer splicing system and, in this research, the focus is on the number of rules. The examples are restricted to a set of initial strings but can be applied to *m* number of initial strings. In addition, the length of set of rules used in the splicing system are set to be different. Meanwhile, Lemma 2 focused on the same rules apply on several crossing sites of the initial strings.

Based on the revised definition of *n*-th order limit language, the following theorem and lemma are used throughout this paper.

Theorem [20]
If a splicing system contains *m* number of initial strings, and *n* number of rules then the splicing system generates *n*-th order limit language.

Lemma [20]
Every *n*-th order limit language produces $(2n-2)$ different combination of strings.

An example is given to enhance the understanding of the above theorem and lemma which focuses on the second order limit language.

Example
Let S be Head splicing system containing an initial string and two rules $S=(\{a,c,g,t\},\{\mu wxzzwxxy\gamma\},\{(w,xz,z),(w,xx,y)\},\varnothing)$, where w and z as well as x and y are complements of each other and $\mu,\gamma,w,x,y,z\in A^*$. The initial string is given as follows:

$$5'-\mu w^{\$}xzzw^{\$}xxy\gamma-3'$$
$$3'-\mu' zyw_{\#}wzyy_{\#}x\gamma'-5'$$

The splicing language produced after the splicing process is given as follows

$$L(S)=\left\{\begin{array}{l}\mu wxxy\gamma,\mu wwyz\mu',\gamma' xxxy\gamma,\\ \mu wyyzwwyz\mu',\mu wxzzwxxy\gamma,\\ \mu wxzzwyyzwwyz\mu',\ldots\end{array}\right\}$$

where $\mu',\gamma'\in A^*$.

Since the number of rules determines the order, the second-order limit language of the splicing system is given as follows:

$$L_2(S)=\left\{\begin{array}{l}\mu w(xzzw\cup yyzw)^* xxy\gamma,\\ \mu w(xzzw\cup yyzw)^* wyz\mu',\\ \gamma' x(xzzw\cup yyzw)xxy\gamma\end{array}\right\}$$

where the symbol $(*)$ and $(\cup)$ in regular expression denotes that the string can occur recursively as well as either string $xzzw$ and $yyzw$ can alternately occur, respectively. The combination of the string of the language for the above splicing language are $xzzw$ and $yyzw$. So from here, if two rules are used in the splicing system, then $n=2$ gives $2n-2=2$. Thus, two string combinations of the language is produced.

In this paper, the Lemmas are given to support the proof of a theorem on the non-existence of the *n*-th order limit language. Then, two lemmas of non-existing of the *n*-th order limit language are presented.

Lemma 1
Let $S=(A,I,B,C)$ be the Head splicing system consisting of *m* number of initial strings and *n* number of rules consisting of different lengths of rules. If no $2n-2$ combination of the strings is produced, then no *n*-th order limit language exists.

Proof

In this lemma, the proving technique is direct proof by showing $P\Rightarrow Q$. Let P be no the combination of the strings and Q *be* no *n*-th order limit language exist. So, in this proving method we show that P then conclude that Q. There are three cases presented below.

Case 1: An initial string and two rules
Let
$S=(\{p,q,r,s\},\{\varpi ppqrqqqrs\xi\},\{p,pq,r\}\{qq,q,rs\},\varnothing)$
be the Head splicing system where p and s, as well as q and r, are the complements of each other, where $\varpi,\xi,p,q,r,s\in A^*$. An initial string and two rules are used in this splicing system. The initial string is given as follows:

$$5'-\varpi p^{\$}pqrqq^{\$}qrs\xi-3'$$
$$3'-\varpi ssr_{\#}qrr_{\#}qp\xi-5'$$

The splicing language produced after the splicing process is as follows:

$$L(S)=\{\varpi p(pqrqq)^* qrs\xi\}.$$

Here, there are two rules used in this splicing system. Based on what we have discussed in the lemma [20], if $n=2$, then $2(2)-2=2$. Thus, there should be two combinations of the string of the language that will be produced. However, in the above splicing language, there is only one combination of strings produced, which is $(pqrqq)$. The length of the rules act on the string is not equal. Then, there is no existence of second-order limit language even two rules are used.

Case 2: An initial string and three rules

Let $S=\left(\begin{array}{l}\{p,q,r,s\},\{\varpi ppqrqqqrs\xi\},\\ \{p,pq,r\}\{qq,q,rs\}\{pp,qq,pp\},\varnothing\end{array}\right)$ be the Head splicing system p and s, as well as q and r, are the complements of each other, and $\varpi,\xi,p,q,r,s\in A^*$. An initial string and three rules are used in this splicing system. The initial string is given as follows.

$$5'-\varpi p^{\$}pqrqq^{\$}qrspp^{\$}qpqpp\xi-3'$$
$$3'-\varpi ssr_{\#}qrr_{\#}qpssrsr_{\#}pp\xi-5'$$

The splicing language produced after the splicing process is as follows:

$$L(S)=\{\varpi p(pqrqqqrspp)^* qpqpp\xi\}.$$

Here, there are three rules used in this splicing system. Based on what we have discussed in the lemma [20], if $n=3$, then $2(3)-2=4$. There should be four string combinations of the language that will be produced. However, in the above splicing language, there is only one combination of strings produced, which is $(pqrqqqrspp)$. The length of the rules act on the string is not equal. Hence, there is no existence of third-order limit language even three rules are used.

Case 3: An initial string and four rules
Let
$S=\left(\begin{array}{l}\{p,q,r,s\},\{\varpi ppqrqqqrsppqqppppppqppp\xi\},\\ \{p,pq,r\}\{qq,q,rs\}\{pp,qq,pp\}\{ppp,q,ppp\},\varnothing\end{array}\right)$
be the Head splicing system p and s, as well as q and r, are the complements of each other, and $\varpi,\xi,p,q,r,s\in A^*$. An initial string and three rules are used in this splicing system. The initial string is given as follows.

$$5'-\varpi p^{\$}pqrqq^{\$}qrspp^{\$}qqppppp^{\$}qppp\xi-3'$$
$$3'-\varpi' ssr_{\#}qrr_{\#}qpssrr_{\#}sssssr_{\#}sss\xi'-5'$$

The splicing language produced after the splicing process is as follows:

$$L(S)=\{\varpi p(pqrqqqrsppqqppppp)^{*}qpqpp\xi\}.$$

Here, there are three rules used in this splicing system. Based on what we have discussed in the lemma [20], if $n=4$, then $2(4)-2=6$. There should be six string combinations of the language that will be produced. However, in the above splicing language, there is only one combination of strings produced, which is $(pqrqqqrsppqqppppp)$. The length of the rules act on the string is not equal. Hence, there is no existence of fourth-order limit language even four rules are used. The summarisation of the cases in Lemma 1 is given in the Table 1.

Table 1 Summarisation of Cases in Lemma 1

Cases	Number of Rules, n	Number of Combination of the strings, $2n-2$	*The* existence of *n*-th Order Limit Language
1	2	1	No
2	3	1	No
3	4	1	No

By using the finding for the above cases, the same findings are predicted until n number of rules when the rules involve in the splicing system have different length. The reason being, it has shown only a combination of string is found in each strings of the language which contradicted with the corallary above. Based on Table 1, the we can see that there are no $2n-2$ combination of the strings produces for all cases, then there is no existence of n-th order limit language. So, $P\Rightarrow Q$. ■

We then proceed to Lemma 2.

Lemma 2
Let $S=(A,I,B,C)$ be the Head splicing system with m number of initial strings in a set of initial strings, I and a rule in a set of rules, B. I contains many crossing sites such that the crossing site is palindrome. If no $2n-2$ combination of the strings is produced, then no n-th order limit language exists.

Proof
In this lemma, the same approach is used which is show that $P\Rightarrow Q$. Let P be no combination of the strings and Q *be* no n-th order limit language exist. So, in this proving method we show that P then conclude that Q. There are three cases presented below.

Case 1: An initial string with two crossing sites
Let $S=(\{p,q,r,s\},\{\varpi ppssppss\xi\},\{p,ps,s\},\varnothing)$ be the Head splicing system where p and s, as well as q and r, are the complements of each other and $\varpi,\xi,p,q,r,s\in A^{*}$. The initial string that contains two crossing sites is given as follows:

$$5'-\varpi p^{\$}pssp^{\$}pss\xi-3'$$
$$3'-\varpi'ssp_{\#}pssp_{\#}p'\xi-5'$$

The splicing language produced after the splicing process is as follows:

$$L(S)=\{\varpi p(pssp)^{*}pss\varpi',\xi'p(pssp)^{*}pss\xi\}$$

where $\varpi',\xi'\in A^{*}$.

Here, the splicing language only has a combination of the string of the language. There is no existence of the n-th order limit language since splicing among the strings of the generated splicing language will regenerate the same splicing language. Here, there is only one combination of the string of the language, which is $(pssp)$.

Case 2: An initial string with three crossing sites
Let $S=(\{p,q,r,s\},\{\varpi ppssppssppss\xi\},\{p,ps,s\},\varnothing)$ be the Head splicing system where p and s, as well as q and r, are the complements of each other and $\varpi,\xi,p,q,r,s\in A^{*}$. The initial string that contains three crossing sites is given as follows.

$$5'-\varpi p^{\$}pssp^{\$}pssp^{\$}pss\xi-3'$$
$$3'-\varpi'ssp_{\#}pssp_{\#}pssp_{\#}p'\xi-5'$$

The splicing language produced after the splicing process is as follows:

$$L(S)=\{\varpi p(pssp)^{*}pss\varpi',\xi'p(pssp)^{*}pss\xi\},$$

where $\varpi',\xi'\in A^{*}$.

Similar to the above case, the splicing language shows only one combination of the string of the language. No presence of the n-th order limit language is generated since splicing among the strings of the generated splicing language will regenerate the same splicing language. Thus, there is only one combination of the string of the language, which is $(pssp)$.

From what we can see in Case 1 and Case 2 above, the splicing language produced for both splicing systems is $L(S)=\{\varpi p(pssp)^{*}pss\varpi',\xi'p(pssp)^{*}pss\xi\}$. The cases in Lemma 2 are summarised in the Table 2.

Table 2 Summarisation of Cases in Lemma 1

Cases	Number of Rules, n	Number of Crossing Site	Number of Combination of the strings, $2n-2$	*The* existence of n-th Order Limit Language
1	1	2	1	No
2	1	3	1	No

Based on the Table 2, we prove that the number of combination of the string produced from the splicing system for both cases 1 and 2, then no n-th order limit language is generated. Since the same rule is applied and the crossing sites is a palindrome in the splicing system, there is no n-th order limit language is generated since splicing among the strings of the generated splicing language will regenerate the same splicing language.

In conclusion, only an initial string is used since in [16] stated that m number of initial string will only affect the pattern of the string in the language not the formation of the n-th order limit language. The same findings are predicted for more than three crossing sites in the set of initial strings. ■

The following theorem is derived by using Lemma 1 and Lemma 2.

Theorem 1
Let $S=(A,I,B,C)$ be the Head splicing system consisting of m number of initial strings and n number of rules. If the m number of initial strings has several crossing sites where the crossing sites are palindrome and the set of the rules has different lengths, there is no existence n-th order limit language.

Proof
Lemma 1 consists of m number of initial strings and n number of rules with different lengths of rules. Based on Case 1 and Case 2, we can see that there is no existence of second and third-order limit languages even though two and three rules are used in the splicing system, respectively. Basically, there is no existence of the n-th order limit language if the rules in the splicing system are of different lengths.

Lemma 2 consists of m number of initial strings in a set of initial strings with many crossing sites such that the crossing site is palindrome. Only a combination of the string is observed in the set of the string produced and there is no existence of the n-th order limit language if the same rules are applied over the splicing system with more crossing sites in the set of initial strings .

So, if the splicing system consist of m number of initial strings has several crossing sites where the crossing sites are palindrome with the rules act on the initial string are of different length, there is only a combination of the string of the language is produced. Therefore, there is no n-th order limit language produce. ■

4.0 CONCLUSION

In this paper, two lemmas and a theorem have been introduced and discussed. Both lemmas are proven by direct proof method. For Lemma 1, we considered m number of initial strings and n number of rules consisting of different lengths of rules. For Case 1, 2 and 3, an initial string with two rules, an initial string with three rules and an initial string with four rules are used in the splicing systems, respectively. Still, there is no existence of second-order third-order and fourth-order limit language since the rule used in the splicing system are not of the same length and there is only a combination of the string of the language. For Lemma 2, m number of initial strings in a set of initial strings contains many crossing sites such that the crossing site is palindrome and a rule in a set of rules. Case 1 and Case 2 used an initial string that contains two and three crossing sites, respectively. The same splicing language is produced after the splicing process occurs. So, no n-th order limit language is formed because splicing among the strings of the produced splicing language regenerates the same splicing language. Based on the lemmas given above, a theorem on the non-existence of the n-th order limit language was given and discussed.

Acknowledgement

The authors would like to thank the Ministry of Higher Education (MOHE) Malaysia for providing financial support under Fundamental Research Grant Scheme (FRGS) No. FRGS/1/2018/STG06/UMP/03/1. (University reference RDU190118).

References

[1] Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K. and Walter, P. 2015. *Molecular Biology of the Cell*. 6th ed. New York: W. W. Norton & Company.
DOI: https://doi.org/10.1016/b978-0-12-279202-1.x5002-7.

[2] Watson, J. and Crick, F. 1953. Molecular Structure of Nucleic Acids: A Structure for Deoxyribose Nucleic Acid. *Nature*.171(4356): 737-738.
DOI: https://doi.org/10.1001/jama.269.15.1966.

[3] Head, T. 1987. Formal Language Theory and DNA: An Analysis of the Generative Capacity of Specific Recombinant Behaviors. *Bulletin of Mathematical Biology*. 49(6): 737-59.
DOI: https://doi.org/10.1016/s0092-8240(87)90018-8.

[4] Head, T. 1998. Splicing Representations of Strictly Locally Testable Languages. *Discrete Applied Mathematics*. 87(1-3): 139-47.
DOI: https://doi.org/10.1016/s0166-218x(98)00053-5.

[5] Head, T. 2000. Splicing Languages Generated with One Sided Context. *Biosystems*. 57(2): 87-93.
DOI: https://doi.org/10.1016/s0303-2647(00)00091-5.

[6] Goode, E. 1999. *DNA Based Computers III*. USA: American

Mathematical Society, Wet Splicing System.
DOI: https://doi.org/10.1090/dimacs/048
[7] Paun, G. 1996. On the Splicing Operation. *Discrete Applied Mathematics*. 70(1): 57-79.
DOI: https://doi.org/10.1016/0166-218x(96)00101-1.
[8] Pixton, D. 1996. Regularity of Splicing Languages. *Discrete Applied Mathematics*. 69(1-2): 101-12.
DOI: https://doi.org/10.1016/0166-218x(95)00079-7.
[9] Goode, E. 2004. *Aspects of Molecular Computing*. New York: Springer-Verlag, Splicing to the Limit.
DOI: https://doi.org/10.1007/978-3-540-24635-0_13.
[10] Yusof, Y. 2012. *DNA Splicing System Inspired by Bio Molecular Operation*. Universiti Teknologi Malaysia.
[11] Yusof, Y., Sarmin, N. H., Goode, T. E., Mahmud, M., Heng, F. W. 2012. Hierarchy of Certain Types of DNA Splicing Systems. *International Journal of Modern Physics: Conference Series 2012*. 271-277.
DOI: https://doi.org/10.1142/s2010194512005314.
[12] Mudaber, M. H., Yusof, Y., and Mohamad, M. S. 2017. Differentiating the Persistency and Permanency of Some Two Stages DNA Splicing Language via Yusof-Goode (YG) Approach. *1st International Conferences on Applied & Industrial Math and Statistics* 2017. 1-10.
DOI: https://doi.org/10.1088/1742-6596/890/1/012081.
[13] Goode, E., & DeLorbe, W. 2007. DNA Splicing Systems. I*nternational Workshop on DNA-Based Computers*. 236-245.
DOI: https://doi.org/10.1007/978-3-540-77962-9_25.
[14] Goode, E. 1999. *Constants and Splicing Systems*. State University of New York.
[15] Ahmad, M. A., Fong, W. H., Yusof, Y. and Sarmin, N. H. 2014. An Extension of First Order Limit Language. *AIP Conference Proceedings* 2014. 627-631.
DOI: https://doi.org/10.1063/1.4882550
[16] Khairuddin, S. H. M., Ahmad, M. A., Adzhar, N. 2020. Effect of m Number of Initial String on the n -th Order Limit Language. *Technology Reports Kansai University*. 62(7): 3681-3689.
[17] Karimi, F. and Turaev, S. 2014. Fuzzy Splicing System. *6th International Conference*, ICCCI 2014. 20-19.
DOI: http://dx.doi.org/10.1007/978-3-319-11289-3_3
[18] Peter, L. An *Introduction to Formal Languages and Automata.* 6th ed. Massachusetts: Jones and Bartlett, LLC; 2016.
DOI: https://doi.org/10.1142/9789812817495_0002.
[19] Ahmad, M. A. 2016. *Second Order Limit Language and its Properties in Yusof-Goode Splicing System*. Universiti Teknologi Malaysia.
[20] Khairuddin, S. H. M., Ahmad, M.A. and Adzhar, N. 2022. An Introduction to n-th Order Limit Language. *Enginering Letters.* 30(1): 29-36.