Folding technique on *n*-cut spliced semigraph in splicing system

Muhammad Nur Syiham Abdul Razak, Wan Heng Fong, and Nor Haniza Sarmin

View Online   Export Citation

ARTICLES YOU MAY BE INTERESTED IN

Folding Technique on *n*-Cut Spliced Semigraph in Splicing System

Muhammad Nur Syiham Abdul Razak,[a)] Wan Heng Fong,[b)] and Nor Haniza Sarmin[c)]

Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia.

[a)]*Corresponding author: mnsyiham@graduate.utm.my*
[b)]*fwh@utm.my*
[c)]*nhs@utm.my*

Abstract. In biology, DNA splicing system models the recombination behaviours of DNA molecules which require the presence of restriction enzymes. Since it is quite incompetent to describe DNA splicing system in one-dimensional string, hence graph theory is used in describing the complexity of DNA splicing system. Thus by graph splicing system, a type of semigraph can be obtained, called as *n*-cut spliced semigraph. Various concepts of graph theory have been applied on semigraph to study its characteristics and properties. In this study, the focus is on the application of the folding technique on *n*-cut spliced semigraph. It is observed that the maximum number of folding for every *n*-cut spliced semigraph is $n + 1$ times. Also, the final form of new semigraphs obtained are determined.

INTRODUCTION

The notion of DNA splicing system was introduced by Head [1] using formal language theory. DNA splicing is a biological process where the DNA strings or called as double-stranded DNA are being decomposed and recombined with the presence of restriction enzymes and ligases. The cleavage pattern of an enzyme determines the cleavage site (site where the cut occur) on a DNA molecule. However, since the complexity of the micro-molecule processes are quite incompetent to be explained in one-dimensional strings, hence Freund [2] introduced graph splicing system in modelling the DNA splicing system.

The DNA splicing system can be illustrated in the form of graphs by using a graph splicing system, where the vertices represents DNA molecules. A splicing scheme is used in a graph splicing system where the splicing rules are stated. In this context, the splicing rules represent the restriction enzymes. Then, Sampathkumar [3] introduced a new notion called semigraph to generalize the concept of graph. This is because semigraph holds various properties through its structure in describing fundamental operations in graph theory. Jeyabharathi et. al [4] introduced a type of semigraph called as *n*-cut spliced semigraph which is obtained after splicing is applied on a graph, which is called as *n*-cut splicing.

In industrial field, the concept of graph folding is applied widely since it has a close connection to most of the industrial problems. For example, the idea of graph folding has been used in robotics and hydraulic tube bending problems by representing them in the form of graphs [5]. Besides that, Lecky [6] used the approach of graph theory on the Programmable Logic Array (PLA) folding problem. Also, Yan [7] applied the concept of graph to study on the protein folding problems and Faisal [8] introduced a hybrid graph to represent the recursive backtracking algorithms which involves foldings and mappings. Generally, a graph folding is a function from a given graph to another graph where all the edges and vertices in the given graph map through the given function to the edges and vertices in the other graph. In this research, the folding technique is applied on the *n*-cut spliced semigraph for every $n \in \mathbb{N}$ to form a new semigraph. The maximum number of foldings for the *n*-cut spliced semigraph is also obtained in which the final form of the new semigraph is also observed.

This paper consists of four sections. The first section is the introduction, followed by the preliminaries. Then, the results are included and discussed in the third section. Finally, the conclusion is summarized in the last section.

PRELIMINARIES

Splicing system models the recombinant behaviors of double stranded DNA molecules with the presence of restriction enzymes and ligases by using formal language [1]. In 1995, Freund [2] introduced graph splicing system to illustrate

DNA splicing system in the form of a graph since it is quite inadequate to describe three-dimensional DNA molecules in one-dimensional strings.

In this section, some definitions that are related to this research are stated.

Definition 1 [9]
A graph is a pair consisting of two sets where the first set is the non-empty set of vertices and the other set is the set of edges. A graph can be denoted as $G = (V,E)$ where V is the set of vertices and E is the set of edges connecting between vertices.

Definition 2 [2]
Let A be a set of finite alphabets. A labeled graph $G = (N,E,L)$ over A consists of a set of vertices, N and lines called edges, E which connect two vertices. A function $L : N \rightarrow A$ assigns a label of A to each node of N.

Definition 3 [2]
A graph splicing scheme is a pair, $\sigma = (A,P)$ where A is a set of finite alphabets and P is a set of finite splicing rules. Meanwhile, a graph splicing system is a set $S = (\sigma, I)$ where σ is a graph splicing scheme and I is a set of the initial strings or graphs.

A finite set P with k number of graph splicing rules, can be written in the form $((h[1], E'[1]), (h[2], E'[2]), ..., (h[k], E'[k]); E)$ such that $k \geq 1$ and for all i with $1 \leq i \leq k$, where

- $h[i] = (N[i], E[i], L[i])$ is a weakly connected graph, where $E[i]$ is the edge of the ith graph splicing rule,

- $E'[i] \subseteq E[i]$, where $E'[i]$ is the cutting pattern for the ith graph splicing rule,

- the nodes $N[i]$ are mutually disjoint,

- E obeys the following rules:

i) Each edge $(n,m) \in E'[i]$ is supposed to be divided into two parts; i.e the start part $(n,m]$ and the end part $[n,m)$,

ii) The elements of E are of the form $((n,m], [n',m'))$, where (n,m) and (n',m') are edges from $\bigcup_{1 \leq i \leq k} E'[i]$,

iii) Every element from $\{(n,m], [n,m) | (n,m) \in \bigcup_{1 \leq i \leq k} E'[i]\}$ must appear exactly once in a pair of E.

A splicing rule can also be represented in the form of a graph.

The following example illustrates DNA molecules in a graph splicing system, which involves one restriction enzyme.

Example 1
Consider the restriction enzyme, AvrII with cleavage pattern (c,ctag,g). In graph splicing, the restriction enzyme AvrII acts as the splicing rule where can be represented as follows:

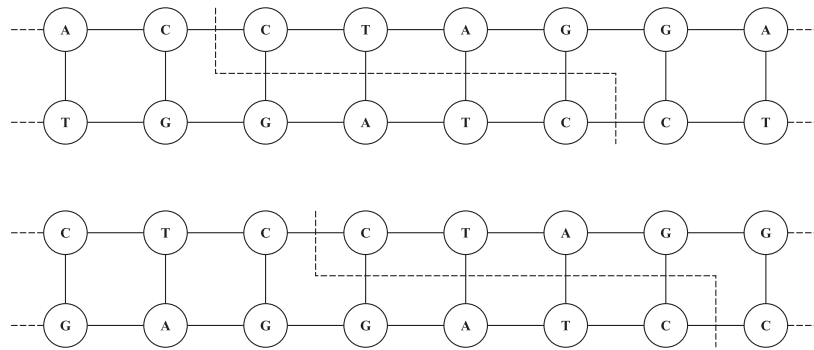$$\{(1,2), (3,4), (5,6), (7,8), (9,10), (11,12)\}).$$
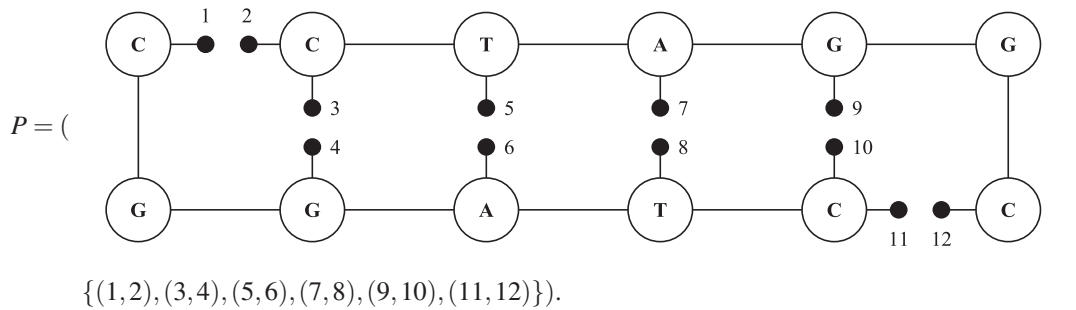
Let acctagga and ctcctagg be two initial strings of double stranded DNA molecules which can be presented in the form of graphs as shown in Figure 1.
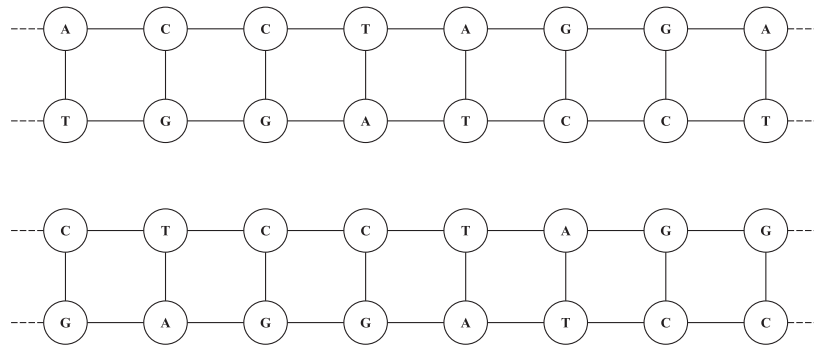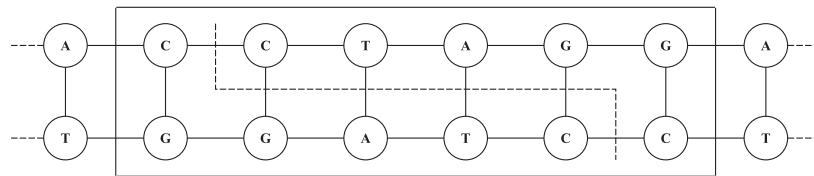
FIGURE 1. The graph representations of acctagga and ctcctagg.

By applying the splicing rule P, the cleavage sites are shown as in Figure 2.

FIGURE 2. The cleavage sites by applying P to the initial DNA molecules.

In splicing system, the part which is of interest is the cleavage pattern or the cleavage site as denoted in Figure 3.

FIGURE 3. The cleavage site.

Then Jeyabharathi [4] introduced n-cut splicing to illustrate the cleavage site of splicing system by using the concept of semigraph. An n-cut splicing will generate two components of n-cut spliced semigraphs by considering $n+2$ bonds to be cut.

Definition 4 [10]
A semigraph G is a 2-tuples (V,X) where V is a non-empty set in which the elements are called vertices of G and X is a set of n-tuples called edges of G of distinct vertices for various $n \geq 2$, satisfying the following conditions:

  1. any two edges have at most one vertex in common,

2. two edges $(u_1, u_2, u_3, ..., u_n)$ and $(v_1, v_2, v_3, ..., v_m)$ are considered to be equal if and only if (a) $m = n$, (b) either $u_i = v_i$ or $u_i = v_{n-i+1}$ for $1 \le i \le n$.

Thus the edge $(u_1, u_2, u_3, ..., u_n)$ are the same as the edge $(u_n, u_{n-1}, ..., u_2, u_1)$.

If $G = (V, X)$ is a semigraph with $E = (u_1, u_2, ..., u_n)$ an element in a set X and u_i for $1 \le i \le n$ are the elements of a set V, hence u_1 and u_n are known as end vertices; while u_i for $2 \le i \le n-1$ are known as middle vertices [10]. If there exists an edge in X in which the middle vertex is an end vertex of another edge in X, then the vertex is called as middle-cum-end vertex [10]. Next, two examples of semigraph are presented in Example 2.

Example 2
In Figure 4, $S_1 = (V, X)$ is a semigraph with $V = \{a, b, c, d, e, f, g\}$ and $X = \{(a, b, c), (d, e, f), (a, d), (b, d), (b, e)\}$.
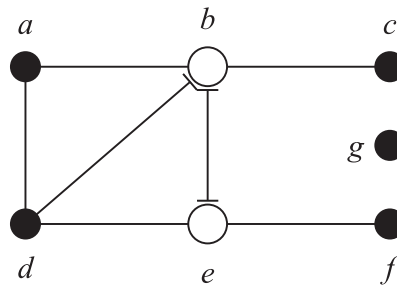
FIGURE 4. The semigraph S_1.

As in Figure 4, the vertices a, c, d, f and g are the end vertices denoted as thick dots while vertex e is a middle vertex and is denoted as an empty circle. Meanwhile, vertex b is said to be a middle-cum-end vertex. The edges approaching at the middle vertex are represented by small tangents.

Next, $S_2 = (V, X)$ is a semigraph where $V = \{a, b, c, d, e, f\}$ with no middle vertex and $X = \{(a, b), (b, c), (c, f), (a, d), (d, e), (e, f), (b, e)\}$ as represented in Figure 5.
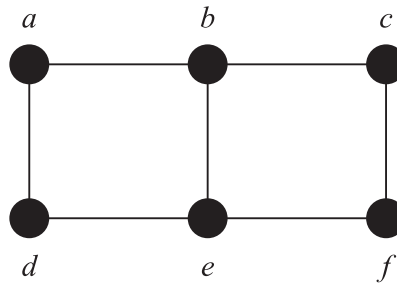
FIGURE 5. The semigraph S_2.

From Figure 5, all vertices are the end vertices and are denoted as thick dots.

Definition 5 [4]
Let G be a graph. By applying splicing to G, new vertices called as semivertices and new edges called as semiedges are obtained by the decomposition of edges. The semivertex and semiedge can be denoted as V' and E' respectively.

Next, the *n*-cut splicing for $n = 1, 2$ and 3 are illustrated in the following table.

TABLE I. The illustration of *n*-cut splicing in producing *n*-cut spliced semigraph.

n	*n*-cut splicing
1	 1-cut splicing
2	 2-cut splicing
3	 3-cut splicing

The concept of mapping and folding on *n*-cut spliced semigraph has been introduced by Jeyabharathi in [4] in which the definition of mapping and semigraph folding are restated in Definition 6.

Definition 6 [4]
Let $SSG_1 = (V_1, E_1)$ and $SSG_2 = (V_2, E_2)$ be two spliced semigraphs and a map $f : SSG_1 \rightarrow SSG_2$ is said to be a spliced semigraph map, if

1. for each vertex and semivertex $v \in V_1$, $f(v)$ is the vertex and semivertex in V_2,

2. for each edge and semiedge $e \in E_1$, $f(e)$ is the edge and semiedge of E_2,

3. for all vertices and semivertices $v \in V_1$, dim $(f(v)) <$ dim (v),

4. for all edges and semiedges $e \in E_1$, dim $(f(e)) <$ dim (e).

where V_1 and V_2 are the set of vertices and semivertices of the spliced semigraphs SSG_1 and SSG_2 respectively; while E_1 and E_2 are the set of edges and semiedges of the spliced semigraphs SSG_1 and SSG_2 respectively. Also, dim(*v*) and dim(*e*) denote the number of vertices and edges, respectively.

Map $f : SSG_1 \rightarrow SSG_2$ is a semigraph folding, if and only if f maps vertices to vertices, semivertices to semivertices, edges to edges and semiedges to semiedges.

The resulted semigraph SSG_n obtained after semigraph foldings is in the form of $\eta(S)$, a 4-tuple such that $\eta(S) = (|V|, |E|, |V'|, |E'|)$, where $|V|$, $|E|$, $|V'|$ and $|E'|$ are the number of vertices, edges, semivertices and semiedges for the new semigraph obtained at the end of the semigraph foldings, respectively.

RESULTS AND DISCUSSION

The process of folding technique on the n-cut spliced semigraph for $n = 1, 2$ and 3 are shown in this section. Besides that, a proposition for the maximum number of folding for n-cut spliced semigraph is also stated and proved by using vertex mapping (condition 1 and 3 from Definition 5). It is shown that the n-cut spliced semigraph can be folded by $n + 1$ times.

Example 3 and 4 illustrate the folding technique on 1-cut spliced semigraph and 2-cut spliced semigraph, respectively. Note that the other possibilities of folding will also result in a new semigraph with the same $\eta(S) = (|V|, |E|, |V'|, |E'|)$.
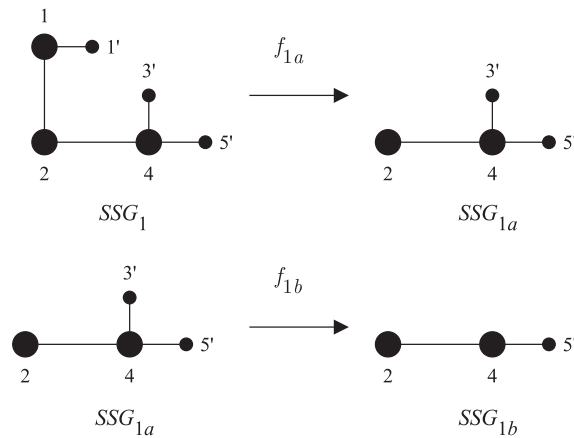
Example 3

Consider a component of a 1-cut spliced semigraph, denoted as SSG_1 as follows.
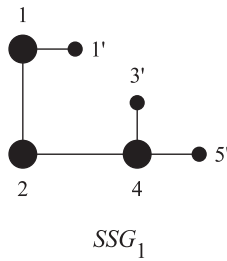
FIGURE 6. The graph representation of a component of a 1-cut spliced semigraph, SSG_1.
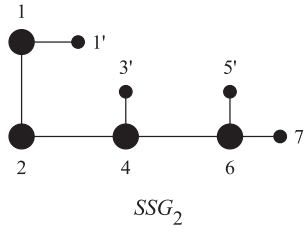
Then, SSG_1 can be folded into a new semigraph by f_{1a} and f_{1b} as shown in the following.

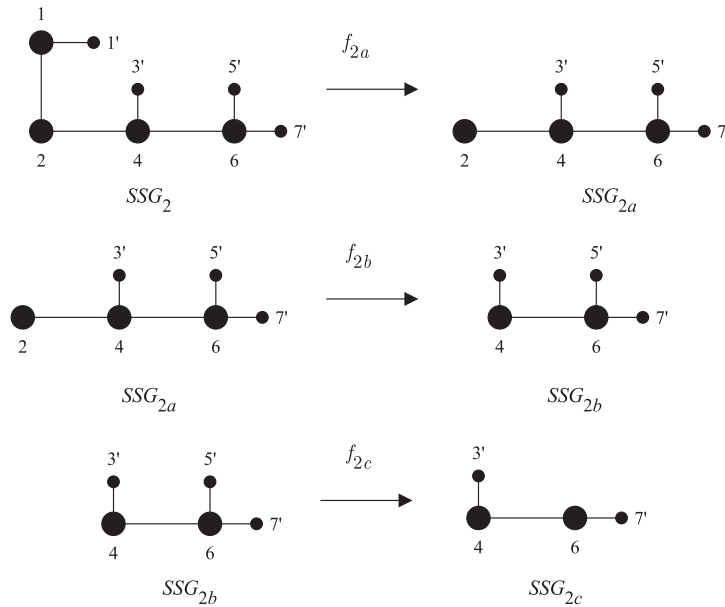**FIGURE 7.** Folding of SSG_1 into a new semigraph, SSG_{1b} through f_{1a} and f_{1b}.

Example 4

Consider a component of a 2-cut spliced semigraph, denoted as SSG_2 as follows.

SSG_2

FIGURE 8. The graph representation of a component of a 2-cut spliced semigraph, SSG_2.

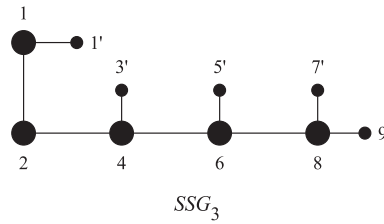Then, SSG_2 can be folded into a new semigraph by f_{2a}, f_{2b} and f_{2c} as shown in the following.

FIGURE 9. Folding of SSG_2 into a new semigraph, SSG_{2c} through f_{2a}, f_{2b} and f_{2c}.

Next, Example 5 illustrates the folding technique on a 3-cut spliced semigraph.
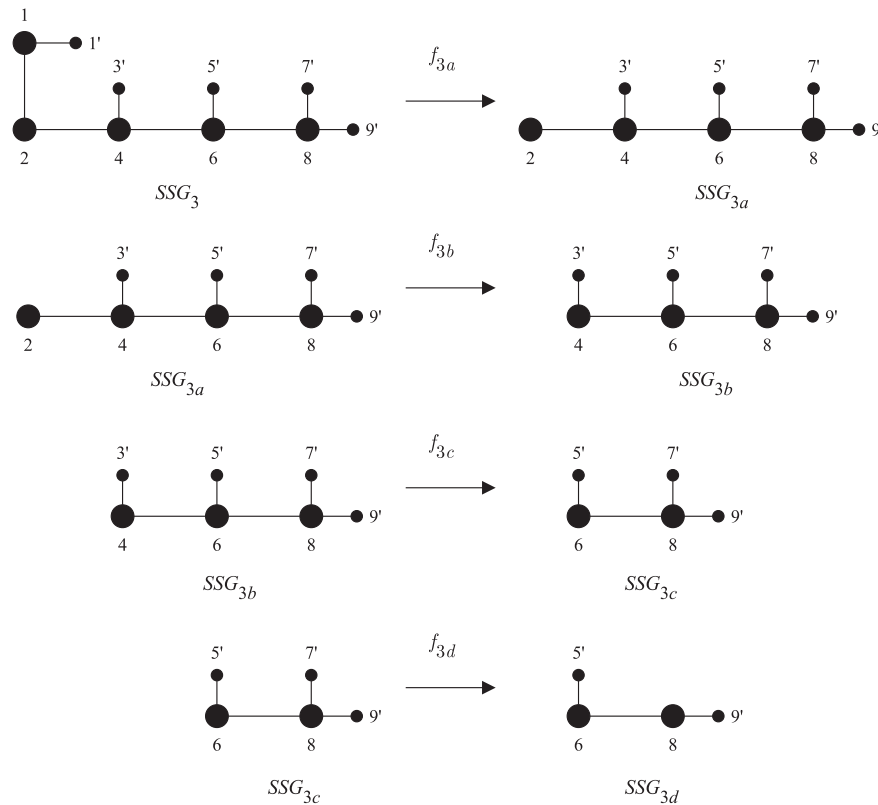
Example 5
A component of a 3-cut spliced semigraph is considered and shown as follows.

FIGURE 10. The graph representation of a component of a 3-cut spliced semigraph, SSG_3.

Then, SSG_3 can be folded into a new semigraph through f_{3a}, f_{3b}, f_{3c} and f_{3d} as shown in the following.


FIGURE 11. Folding of SSG_3 into a new semigraph, SSG_{3d} through f_{3a}, f_{3b}, f_{3c} and f_{3d}.

Recalling from previous examples, the 1-cut spliced semigraph is folded through f_{1a} and f_{1b}. The resulted semigraph can no longer be folded by any folding. Hence there are two times folding. Also, the 2-cut spliced semigraph is folded through f_{2a}, f_{2b} and f_{2c} in which the resulted semigraph can no longer be folded by any folding. Therefore there are three times folding. In Example 5, the 3-cut spliced semigraph is folded through f_{3a}, f_{3b}, f_{3c}, f_{3d} and the resulted semigraph can no longer be folded by any folding. Hence there are four times folding.

By observing the folding technique applied on n-cut spliced semigraph for $n = 1, 2$ and 3, it is shown that every n-cut spliced semigraph can be folded into a new semigraph after $n + 1$ times.

Proposition 1: Every n-cut spliced semigraph can be folded for $n+1$ times.

Proof: Suppose that an n-cut splicing is applied with $n+2$ bonds being cut. Two components of n-cut spliced semigraphs are obtained which are the same. Every bond that is cut produces two semivertices and two semiedges for each component. Therefore, there are $n+2$ semivertices and $n+2$ semiedges for each component. The total number of vertices in each component is also $n+2$.
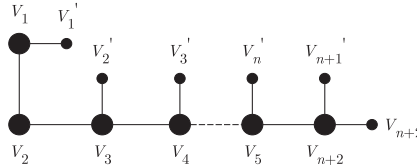
FIGURE 12. A component of n-cut spliced semigraph.

Referring to Figure 12, consider a component of n-cut spliced semigraph, which consists of $n+2$ vertices $(v_1, v_2, ..., v_{n+1}, v_{n+2})$, $n+2$ semivertices $(v'_1, v'_2, ..., v'_{n+1}, v'_{n+2})$ and $n+2$ semiedges. The foldings will follow the following algorithm:

Step 1: Consider the n-cut spliced semigraph where $n \in \mathbb{N}$.

Step 2: Let the vertices be denoted as v_i and the semivertices be denoted as v'_i such that $i = 1, 2, ..., n+2$.

Step 3: Let the start vertex be v_i or v'_i, $i = 1, 2, ..., n+2$.
Also, let $k = 0$ be the initial number of folding.

Step 4: Let $i = 1$, then
Do $f(v_i) = v_{i+2}$ and $f(v'_i) = v'_{i+1}$.
Increase k by 1.
If there exists v_{i+3}, GO TO STEP 5,
else, GO TO STEP 7.

Step 5: Increase i by 1, then
Do $f(v_i) = v_{i+2}$.
Increase k by 1.
If there exists v_{i+3}, GO TO STEP 6,
else, GO TO STEP 7.

Step 6: Increase i by 1, then
Do $f(v_i) = v_{i+2}$ and $f(v'_{i-1}) = v'_{i+2}$.
Increase k by 1.
If there exists v_{i+3}, REPEAT STEP 6,
else, GO TO STEP 7.

Step 7: Increase i by 1, then
Do $f(v'_i) = v'_{i+1}$.
Increase k by 1, and then proceed to the next step.

Step 8: The number of foldings equals to k.

Step 9: The process ends.

By using the above algorithm, it can be shown that every n-cut spliced semigraph can be folded for $n+1$ times. ∎

Besides that, the resulted graph for every folding of n-cut spliced semigraphs is in the form of $\eta(S) = (|V|, |E|, |V'|, |E'|)$ in which $|V|$, $|E|$, $|V'|$ and $|E'|$ denote the number of vertices, edges, semivertices and semiedges for the new semigraph obtained.

From Example 3, the resulted semigraph is in the form of $\eta(S) = (2,1,1,1)$ since there are two vertices, one edge, one semivertex and one semiedge. Meanwhile, from Example 4 and 5, it is shown that the resulted semigraphs are all in the form $\eta(S) = (2,1,2,2)$ since each of them consists of two vertices, one edge, two semivertices and two semiedges.

From the observation, a new semigraph with $\eta(S) = (2,1,1,1)$ is produced by folding 1-cut spliced semigraph for two times; meanwhile by folding n-cut spliced semigraph for $n \geq 2$ for $n+1$ times, new semigraphs are generated with $\eta(S) = (2,1,2,2)$.

CONCLUSION

In this research, the n-cut spliced semigraph is used to study the concept of graph folding. The n-cut spliced semigraph is obtained when a splicing called as n-cut splicing is applied in a graph splicing process which always generates two components of n-cut spliced semigraphs. The examples of folding technique being applied on n-cut spliced semigraph for $n = 1, 2$ and 3 are also shown, where the n-cut spliced semigraphs can be folded into a new semigraph by $n+1$ times. Also, the resulted semigraph obtained is in the form of $\eta(S) = (2,1,1,1)$ for 1-cut spliced semigraph and $\eta(S) = (2,1,2,2)$ for n-cut spliced semigraph for $n \geq 2$.

ACKNOWLEDGMENTS

REFERENCES

1. T. Head, "Formal Language Theory and DNA: An Analysis of The Generative Capacity of Specific Recombinant Behaviors," Bulletin of Mathematical Biology **49(6)**, 737–759 (1987).
2. R. Freund, "Splicing system on graphs," Proceedings of The First International Symposium on Intelligence in Neural and Biological Systems (INBS'95) , 189–194 (1995).
3. E. Sampathkumar, "Semigraph and Their Applications," Report on The DST Project (2011).
4. S. Jeyabharathi, J. Padmashree, S. S. Selvi, and K. Thiagarajan, "Semigraph Structure on DNA Splicing System," Sixth International Conference on Bio-Inspired Computing: Theories and Applications **V**, 182–187 (2011).
5. S. I. Nada and E. H. Hamouda, "On The Folding of Graphs-Theory and Application," Chaos, Solitons and Fractals , 1–8 (2009).
6. J. E. Lecky, O. J. Murphy, and R. Absher, "Graph Theoretic Algorithms for the PLA Folding Problem," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **8(9)**, 1014–1021 (1989).
7. Y. Yan, S. Zhang, and W. Fang-Xiang, "Applications of Graph Theory in Protein Structure Identification," *Proteome science*, **9(Suppl 1)**, S17 (2011).
8. N. A. Faisal, A. L. Michael, E. M. Amer, and P. N. Clinton, "A Hybrid Graph Representation for Recursive Backtracking Algorithms," *International Workshop on Frontiers in Algorithmics*, **6213**, 136–147 (2010).
9. R. Diestel, *Graph Theory*, 5th ed. (Springer-Verlag Berlin Heidelberg, 2017).
10. E. Sampthkumar, C. M. Deshpande, B. Y. Ban, L. Pushpalatha, and V. Swaminathan, *Semigraphs and Their Applications* (Academy of Discrete Mathematics and Application, India, 2019).