Hybridization-Ligation Versus Parallel Overlap Assembly: An Experimental Comparison of Initial Pool Generation for Direct-Proportional Length-Based DNA Computing

Zuwairie Ibrahim*, *Student Member, IEEE*, Yusei Tsuboi, *Student Member, IEEE*, and Osamu Ono

***Abstract*—Previously, direct-proportional length-based DNA computing (DPLB-DNAC) for solving weighted graph problems has been reported. The proposed DPLB-DNAC has been successfully applied to solve the shortest path problem, which is an instance of weighted graph problems. The design and development of DPLB-DNAC is important in order to extend the capability of DNA computing for solving numerical optimization problem. According to DPLB-DNAC, after the initial pool generation, the initial solution is subjected to amplification by polymerase chain reaction and, finally, the output of the computation is visualized by gel electrophoresis. In this paper, however, we give more attention to the initial pool generation of DPLB-DNAC. For this purpose, two kinds of initial pool generation methods, which are generally used for solving weighted graph problems, are evaluated. Those methods are hybridization-ligation and parallel overlap assembly (POA). It is found that for DPLB-DNAC, POA is better than that of the hybridization-ligation method, in terms of population size, generation time, material usage, and efficiency, as supported by the results of actual experiments.**

***Index Terms*—Direct-proportional length-based DNA computing (DPLB-DNAC), hybridization-ligation, parallel overlap assembly (POA), shortest path problem.**

I. Introduction

Currently, there are two kinds of initial pool generation methods of DNA computing for weighted graph problems: hybridization-ligation and parallel overlap assembly (POA). The hybridization-ligation method was first introduced by Adleman [1] to solve a Hamiltonian path problem (HPP) using DNA as a medium of computation. During the operation, the link olligonucleotides or oligos hybridize through the hydrogen bonds by enzymatic reaction. The hybridization-ligation reaction is well shown in Fig. 1 [2] where the arrowhead indicates the $3'$ end of oligos.

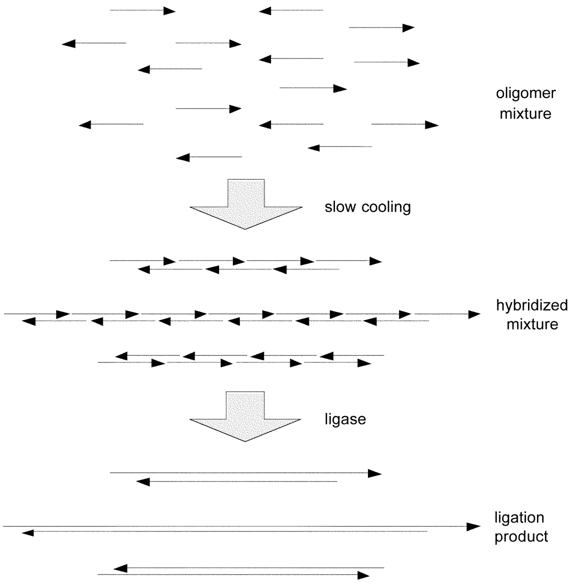

Fig. 1. Hybridization-ligation reaction for initial pool generation of DNA computing. The arrowhead indicates the $3'$ end.

On the other hand, POA has been used [3] and broadly applied in gene construction [4], [5] and DNA shuffling [6]. POA involves appropriate thermal cycles and during the thermal cycles, the position strings in one oligo anneal to the complementary strings of the next oligo. The $3'$ end side of the oligo is extended in the presence of polymerase enzyme to form a longer double-stranded DNA, as depicted in Fig. 2 [2]. In Fig. 2, the thin arrows represent the elongated part during polymerization and the arrowhead indicates the $3'$ end. After a number of thermal cycles, a data pool with all combinations could be generated.

The aim of this paper is to compare the performance of hybridization-ligation and POA, especially for initial pool generation of direct-proportional length-based DNA computing (DPLB-DNAC). From a length-based DNA computing point of view, a shortcoming of this approach has been identified, which is related to errors of the computation, especially if hybridization-ligation is employed during the initial pool generation. This is because the fact that longer weights are encoded as longer sequences is contrary to the biological property of DNA such that the longer the sequences are, the more likely they hybridize with other DNA strands, though we have to find the shortest DNA strands [7]. Hence, an experimental comparison is essential in order to prove the correctness of each initial pool

Manuscript received May 3, 2005; revised December 22, 2005. This research was supported in part by the IEEE Computational Intelligence Society (IEEE-CIS) Walter J. Karplus Student Summer Research Grant 2004. *Asterisk indicates corresponding author.*

*Z. Ibrahim was with the Department of Mechatronics and Robotics, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 UTM Skudai, Malaysia. He is now with the Institute of Applied DNA Computing, Meiji University, Kanagawa 214-8571, Japan (e-mail: zuwairie@ieee.org).

Y. Tsuboi is with Meiji University, Kanagawa 214-8571, Japan (e-mail: tsuboi@isc.meiji.ac.jp).

O. Ono is with Meiji University, Kanagawa 214-8571, Japan (e-mail: ono@isc.meiji.ac.jp).

1536-1241/$20.00 © 2006 IEEE

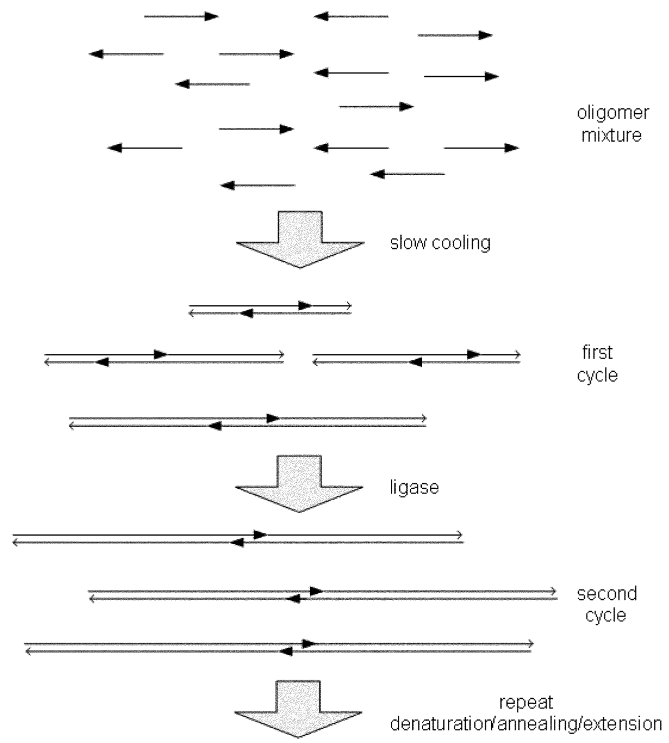

Fig. 2. POA for initial pool generation of DNA computing. The thick arrows represent the synthesized oligos which are the input to the computation. The thin arrows represent the elongated part during polymerization. The arrowhead indicates the 3′ end.

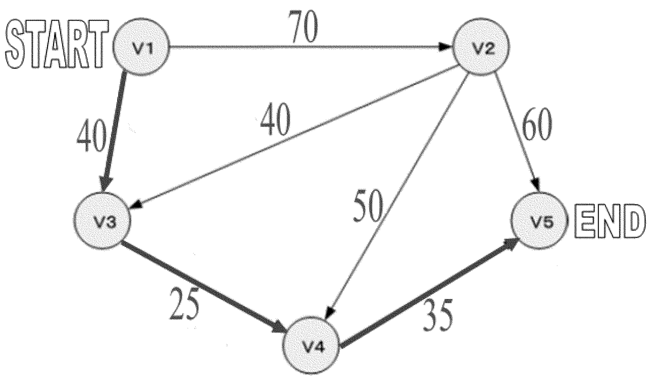

Fig. 3. A weighted undirected graph $G = (V, E)$.

generation methods in order to solve the shortest path problem using DPLB-DNAC.

II. The Shortest Path Problem

The input to the shortest path problem is a weighted directed graph $G = (V, E, \omega)$, a start node u, and an end node v. The output of the shortest path problem is a (u, v) path with the smallest cost. In the case given in Fig. 3, if u is V_1 and v is V_5, the cost for the shortest path will be given as 100 and the optimal path is clearly shown as $V_1 - V_3 - V_4 - V_5$. Even though the shortest path problem belongs to class P, it is worthy of being solved by DNA computing because numerical evaluation is involved during the computation.

TABLE I
Node Sequences for the Case of Hybridization/Ligation

Node, V_i	20-mer sequences (5'-3')	
	V_{ia}	V_{ib}
V_1	AAAGCTCGTC	GTTTAGGAGC
V_2	GCACTAGGGA	TTTGGAGGTT
V_3	GCTATGCCGT	AGTAGAGCGA
V_4	CGATACCGAA	CTGATAAGCG
V_5	CGTGGGTGGC	TCTGTAATAG

TABLE II
Complements of Nodes for the Case of Hybridization/Ligation

Complement node, $\overline{V_i}$	Complement sequences (3'-5')
$\overline{V_1}$	TTTCGAGCAGCAAATCCTCG
$\overline{V_2}$	CGTGATCCCTAAACCTCCAA
$\overline{V_3}$	CGATACGGCATCATCTCGCT
$\overline{V_4}$	GCTATGGCTTGACTATTCGC
$\overline{V_5}$	GCACCCACCGAGACATTATC

III. DNA Sequence Design

Various kinds of methods and strategies for DNA sequence design have been proposed to date. As reviewed by Shin *et al.* [8], those strategies are exhaustive search [9], random search [10], template-map strategy [11], [12], graph method [13], stochastic methods [14], dynamic programming [15], biological-inspired methods [16], [17], and evolutionary algorithms [18]–[22]. However, in the rest of this paper, the DNA sequences used for the computation are designed by using DNASequenceGenerator [13], which can be downloaded for free.[1] Since the hybridization-ligation method and POA are slightly different in principle, two different schemes of DNA sequence design for nodes and edges are developed.

A. DNA Sequence Design for Hybridization-Ligation Method

Let n be the total number of nodes in the graph. The DNA sequences correspond to all nodes and its complements are designed. Let $V_i (i = 1, 2, \ldots, n)$ and $\overline{V_i} (i = 1, 2, \ldots, n)$ be the 20-mer DNA sequences corresponding to the ith node in the graph and its complement, respectively. At first, the DNA sequences V_i and its complement are designed and listed in Tables I and II, respectively. In Table I, V_i is separated into V_{ia} and V_{ib} where V_{ia} is defined as the half-5-end and V_{ib} is defined as the half-3-end of V_i. This separation is important for concatenation during the initial pool generation, by the hybridization-ligation method.

For the case of the hybridization-ligation method, we introduce three rules to synthesize oligos for each edge in the graph as follows.

1) For a connection between V_1 to V_j, synthesize the oligos for the edge as

$$V_1(20) + W_{1j}(\omega - 30) + V_{ja}(10).$$

[1] [Online]. Available: http://ls11-www.cs.uni-dortmund.de/molcomp

TABLE III
DNA SEQUENCES FOR EDGES FOR THE CASE OF HYBRIDIZATION-LIGATION

Edges	DNA Sequences (5'-3')
$V_{4\text{b}}$–W_{45}–V_5	CTGATAAGCGccaagCGTGGGTGGCTCTGTAATAG
$V_{3\text{b}}$–W_{34}–V- 4a	AGTAGAGCGAccgtcCGATACCGAA
V_1–W_{13}–$V_{3\text{a}}$	AAAGCTCGTCGTTTAGGAGC acgtcggttcGCTATGCCGT
$V_{2\text{b}}$–W_{23}–V_3 a	TTTGGAGGTTccgtcttttacccaagtaatGCTATGCCGT
$V_{2\text{b}}$–W_{24}–V_4 a	TTTGGAGGTTacgtgttttaaggaagtacggtaagctgcg CGATACCGAA
$V_{2\text{b}}$–W_{25}–V_5	TTTGGAGGTTgcgtcgcgtaaggcagtaccggactctgcc GGTGGCTCTGTAATAG
V_1–W_{12}–$V_{2\text{a}}$	AAAGCTCGTCGTTTAGGAGC cggtggtttaacgaagtcctgtactatgggttatttgcagGCACTAGGGA

TABLE IV
DNA SEQUENCES FOR WEIGHTS FOR THE CASE OF HYBRIDIZATION-LIGATION

Edges	DNA Sequences (5'-3')
W_{13}	GAACCGACGT
W_{23}	ATTACTTGGGTAAAAGACGG
W_{24}	CGCAGCTTACCGTACTTCCTTAAAACACGT
W_{25}	GGCAGAGTCCGGTACTGCCTTACGCGACGC
W_{12}	CTGCAAATAACCCATAGTACAGGACTTCGTTAAACCACCG

2) For a connection between V_i to V_j, where $i \neq 1, j \neq n$, synthesize the oligos for the edge as

$$V_{ib}(10) + W_{ij}(\omega - 20) + V_{ja}(10).$$

3) For a connection between V_i to V_n, synthesize the oligos for the edge as

$$V_{ib}(10) + W_{\text{in}}(\omega - 30) + V_n(20).$$

The resultant DNA sequences for edges designed based on the rules are listed in Table III. Clearly, the synthesized oligos consist of three segments. The number of DNA bases for each segment is shown in parenthesis. "+" represents the joint and "ω" denotes the weight value for corresponding DNA sequences for weight W_{ij} where W_{ij} denotes the DNA sequences representing a cost between node V_i and V_j. Note that the node segment and edge segment are distinguished by capital and small letters, respectively. For initial pool generation based on hybridization-ligation, DNA sequences for weights are also important. Those sequences are listed in Table IV. Note that in Table IV, only those sequences which are seven or greater in length are shown. This is because the DNA sequences, if less than seven, cannot be synthetically synthesized in the laboratory.

B. DNA Sequence Design for POA

For POA, basically, the DNA sequences designed for nodes are similar to the DNA sequences as listed in Table I without being separated into V_{ia} and V_{ib}. This is because no concatenation occurs during the initial pool generation by POA. Thus, those DNA sequences can be viewed as shown in Table V.

For the case of POA, we introduce three rules to synthesize oligos for each edge in the graph as follows.

TABLE V
NODE SEQUENCES FOR THE CASE OF POA

Node, V_i	20-mer sequences (5'-3')
V_1	AAAGCTCGTCGTTTAGGAGC
V_2	GCACTAGGGATTTGGAGGTT
V_3	GCTATGCCGTAGTAGAGCGA
V_4	CGATACCGAACTGATAAGCG
V_5	CGTGGGTGGCTCTGTAATAG

TABLE VI
DNA SEQUENCES FOR EDGES FOR THE CASE OF POA

Edges	DNA Sequences (5'-3')
V_4–W_{45}–V_5	CGATACCGAACTGATAAGCGccaag CGTGGGTGGCTCTGTAATAG
V_3–W_{34}–V_4	GCTATGCCGTAGTAGAGCGAccgtc CGATACCGAACTGATAAGCG
V_1–W_{13}–V_3	AAAGCTCGTCGTTTAGGAGCacgtcggttc GCTATGCCGTAGTAGAGCGA
V_2–W_{23}–V_3	GCACTAGGGATTTGGAGGTTccgtcttttacccaagtaat GCTATGCCGTAGTAGAGCGA
V_2–W_{24}–V_4	GCACTAGGGATTTGGAGGTT acgtgttttaaggaagtacggtaagctgcg CGATACCGAACTGATAAGCG
V_2–W_{25}–V_5	GCACTAGGGATTTGGAGGTT gcgtcgcgtaaggcagtaccggactctgcc CGTGGGTGGCTCTGTAATAG
V_1–W_{12}–V_2	AAAGCTCGTCGTTTAGGAGC cggtggtttaacgaagtcctgtactatgggttatttgcag GCACTAGGGATTTGGAGGTT

1) For a connection between V_1 to V_j, synthesize the oligo for the edge as

$$V_1(20) + W_{1j}(\omega - 30) + V_j(20).$$

2) For a connection between V_i to V_j, where $i \neq 1, j \neq n$, synthesize the oligo for the edge as

$$V_i(20) + W_{ij}(\omega - 20) + V_j(20).$$

3) For is a connection between V_i to V_n, synthesize the oligo for the edge as

$$V_i(20) + W_{\text{in}}(\omega - 30) + V_n(20).$$

where V, W, and "+" denote the DNA sequences for nodes, DNA sequences for weight, and "joint," respectively. The synthesized oligos consist of three segments; two node segments and an edge segment. "ω" denotes the weight value for corresponding DNA sequences for weight W_{ij} where W_{ij} denotes the DNA sequences representing a cost between node V_i and V_j. The value in parenthesis indicates the number of DNA bases or nucleotides for each segment. The oligo is designed so that the number of DNA bases of that oligo and the cost at the corresponding edge are similar. Table VI lists all the synthesized oligos based on the proposed synthesis rules. Similarly, the node segment and edge segment are distinguished by capital and small letters, respectively.

IV. A BRIEF DESCRIPTION OF DPLB-DNAC

After the DNA sequences are designed, the oligos of the complement of the node sequences and edges sequences are synthesized. Then, all the synthesized oligos are poured into a test tube

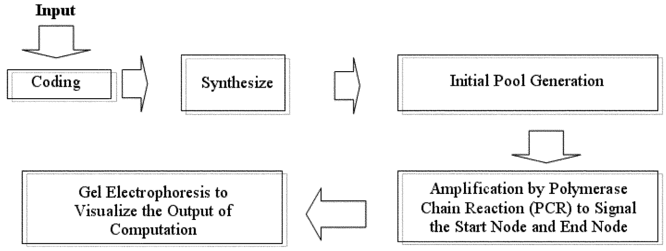
Input
Coding
Synthesize
Initial Pool Generation
Amplification by Polymerase Chain Reaction (PCR) to Signal the Start Node and End Node
Gel Electrophoresis to Visualize the Output of Computation

Fig. 4. The procedure of DPLB-DNAC.

for initial pool generation. In order to generate the initial pool of DPLB-DNAC, initial pool generation methods, such as hybridization-ligation or POA could be employed. Next, the generated initial pool generation is subjected to amplification by polymerase chain reaction (PCR) in order to amplify exponentially, DNA molecules that contain the start node V_1 and end node V_5. After the PCR is accomplished, there should be a big numbers of DNA molecules representing the start node V_1 and end node V_5 traveling through a possible number of nodes.

DNA molecules in a solution can be separated in terms of its length consecutively by means of gel electrophoresis. In fact, the molecules are separated according to their weight, which is almost proportional to their length [23]. Based on DPLB-DNAC, the output solution of PCR operation is brought for gel electrophoresis operation. During this operation, the DNA molecules are separated in term of its length and hence, the shortest DNA molecules in terms of length in base pairs (bp), which representing the shortest path could appear as the shortest band of the output of gel electrophoresis. The procedure of the DPLB-DNAC is shown in Fig. 4.

V. Hybridization-Ligation Versus POA

In this work, we implemented both methods: hybridization-ligation and POA for initial pool generation of DPLB-DNAC. For evaluation purpose, both of the methods are examined whether the correct answer of the shortest path problem can be achieved or not. This can be done by observing and comparing the output of gel electrophoresis with the expected answer of the shortest path problem.

A. In Vitro Computation Based on Hybridization-Ligation Method

Initial pool generation by hybridization-ligation method was performed in a 200-μL solution containing 120-μL oligos (Proligo Primers & Probes, Tokyo, Japan), 2-μL 100-mM ATP (GeneACT, Japan), 20 μL 10× T4 Kinase Buffer (TOYOBO, Japan), 1 μL Kinase (TOYOBO, Japan), and 57 μL H_20 (Maxim Biotech, Japan). The solution was subjected to kination in a thermal cycle at 37 °C for 1 hour. After that, hybridization was done by heating the solution at 94 °C for 10 s and gradually decreased the temperature of the solution 2°C/s to 16 °C.

The resultant solution was then subjected to ligation by pouring 0.5 μL 100 mM ATP (GeneACT, Japan), 5 μL DNA ligation buffer (TOYOBO, Japan), and 0.5 μL T4 DNA ligase (TOYOBO, Japan) into the solution and the solution was kept at 16 °C for 12 h.

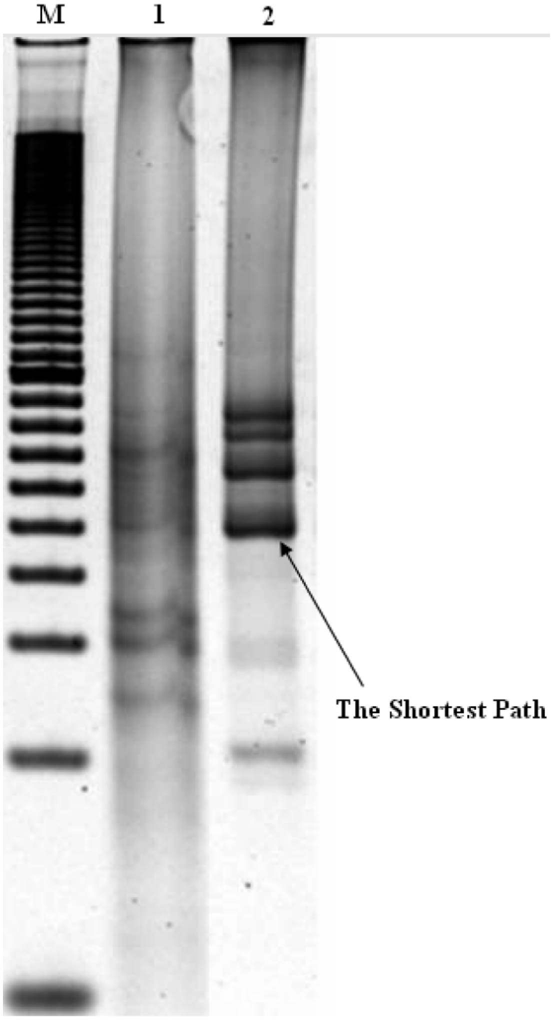

Fig. 5. Experimental results of gel electrophoresis on 10% polyacrylamide gel. Lane M denotes the 20-bp ladder, lane 1 is the product of initial pool generation by hybridization-ligation, and lane 2 is the product of PCR, which represent the output of computation.

For amplification, PCR was performed in a 25 μL solution consists of 2.5 μL solution for each primers, 1 μL template, 2.5 μL dNTP (TOYOBO, Japan), 2.5μL 10× KOD dash buffer (TOYOBO, Japan), 0.125 μL KOD dash (TOYOBO, Japan), and 13.875 μL H_20 (Maxim Biotech, Japan). The reaction consists of 25 cycles and for each cycle, the denaturation, annealing, and polymerization temperatures were 94 °C for 30 s, 55 °C for 30 s, and 74 °C for 10 s, respectively.

The oligo AAAGCTCGTCGTTTAGGAGC was used as forward primer while for the reverse primer, the oligo GCACCCACCGAGACATTATC, was employed. To visualize the result of the computation, the product of PCR was subjected to gel electrophoresis for 40 min at 200 V. After gel electrophoresis was performed, the gel was stained by SYBR Gold (Molecular Probes). The band of DNA was viewed by 300 nm FOTO/Phoresis UV Transluminator (Fotodyne Inc.) and the resultant gel image was captured.

The product of initial pool generation and amplification based on hybridization-ligation method and PCR are shown in Fig. 5. Lane M denotes the 20-bp ladder, lane 1 is the product of hybridization-ligation, and lane 2 is the product of PCR.

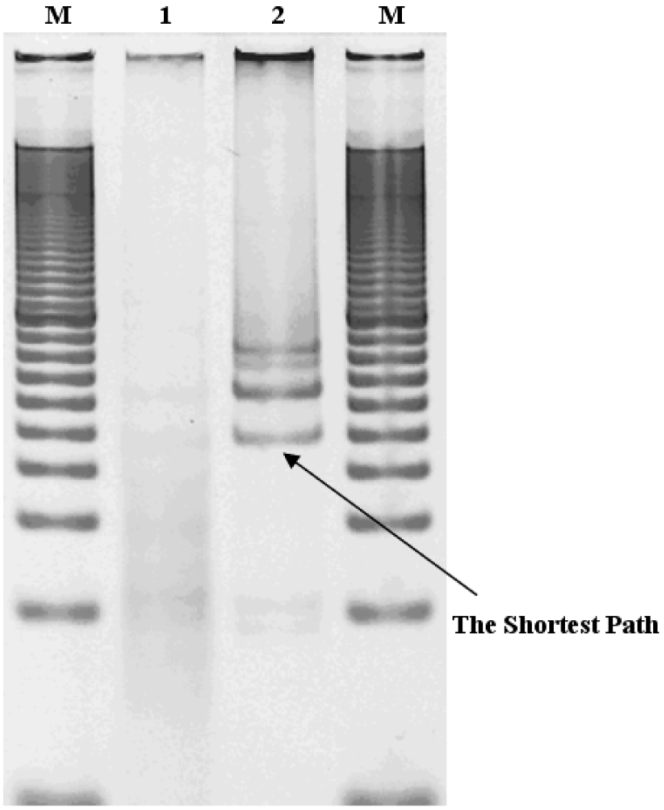

Fig. 6 Experimental results of gel electrophoresis on 10% polyacrylamide gel. Lane M denotes the 20-bp ladder, lane 1 is the product of POA, and lane 2 is the product of PCR, which is the output of the shortest path problem.

B. In Vitro Computation Based on POA

On the other hand, initial pool generation by POA was performed in a 100-μL solution containing 12 μL oligos (Proligo Primers & Probes), 10 μL dNTP (TOYOBO, Japan), 10 μL 10× KOD dash buffer (TOYOBO, Japan), 0.5 μL KOD dash (TOYOBO, Japan), and 67.5 μL H_20 (Maxim Biotech, Japan). The reaction consists of 25 cycles and for each cycle, similar to PCR, the appropriate temperatures were 94 °C for 30 s, 55 °C for 30 s, and 74 °C for 10 s.

As shown in Fig. 6, lane M denotes the 20-bp ladder and lane 1 is the product of initial pool generation based on POA.

For amplification, PCR was performed in a 25-μL solution consisting of 0.5 μL solution for each primers, 1 μL template, 2.5 μL dNTP (TOYOBO, Japan), 2.5 μL 10× KOD dash buffer (TOYOBO, Japan), 0.125 μL KOD dash (TOYOBO, Japan), and 17.875 μL H_20 (Maxim Biotech, Japan). The reaction consists of 25 cycles and, similarly, the denaturation, annealing, and polymerization temperatures were 94 °C for 30 s, 55 °C for 30 s, and 74 °C for 10 s, respectively.

Similarly, the oligo AAAGCTCGTCGTTTAGGAGC was used as forward primer, while for the reverse primer, the oligo GCACCCACCGAGACATTATC was employed. To visualize the result of the computation, the product of PCR was subjected to gel electrophoresis for 40 min at 200 V. After gel electrophoresis was performed, the gel was stained by SYBR Gold (Molecular Probes). The band of DNA was viewed by 300 nm FOTO/Phoresis UV Transluminator (Fotodyne Inc.) and the resultant gel image was captured.

The output of amplification is shown in Fig. 6, where lane M denotes the 20-bp ladder and lane 2 is the product of PCR.

VI. Discussions

The result of the computation, as shown in Figs. 4 and 5, could be obtained if hybridization-ligation method and POA are employed during the initial pool generation of DPLB-DNAC. In both of the figures, lane 2 consists of four bands showing that all the paths which start with V_1 and end at V_5 have been successfully amplified. Those paths are $V_1 - V_3 - V_4 - V_5$ (100 bp), $V_1 - V_2 - V_5$ (130 bp), $V_1 - V_2 - V_4 - V_5$ (155 bp), and $V_1 - V_2 - V_3 - V_4 - V_5$ (170 bp). It is clear that the shortest path computation appears as the shortest band in lane 2.

Even though the same experiment has been done by Lee [2], they did a comparison of hybridization-ligation and POA to solve the weighted graph problem by temperature gradient-based DNA computing [24]. Their findings are important as they found that POA is much better than hybridization-ligation and the advantages of POA over hybridization-ligation can be listed as follows.

- The initial pool size generated from the same amount of initial oligos is about twice larger than that of hybridization-ligation method. Though, if a larger problem is considered, the initial pool size is too small to contain the complete pool. POA, however, with more cycles and large experimental scale, could include the practical pools.
- Initially, two single-stranded DNA molecules partially hybridize during the annealing step and then they are extended by polymerase. The elongated DNA molecules are denatured into two single-stranded DNAs in the next denaturation step, and they are subjected to the annealing reaction at the next cycle. Therefore, POA does maintain the population size and the population size can be decided by varying the initial number of oligos.
- In hybridization-ligation method, the population size decreases as reaction progress by a factor of the number of components composing it in hybridization-ligation method. As the problem size increases, the required initial pool size increases dramatically. Moreover, initial pool generation by POA requires fewer strands than hybridization-ligation method to obtain similar amount of initial pool DNA molecules because complementary strands are automatically extended by polymerase.
- POA does not require phosphorylation of oligos, which is a prerequisite for the ligation of oligos.
- POA demands less time than hybridization-ligation method. Hybridization requires 1.5 h while ligation required more than 12 h. Hence, POA for 34 cycles requires only 2 h. Therefore, POA is much more efficient and economic method for initial pool generation.

In addition, based on our finding, the principle of hybridization-ligation method is different than POA in the sense that hybridization-ligation allows annealing of various length of oligos. As an inherent property of hybridization is that the longer the sequences are, the more likely they hybridize with

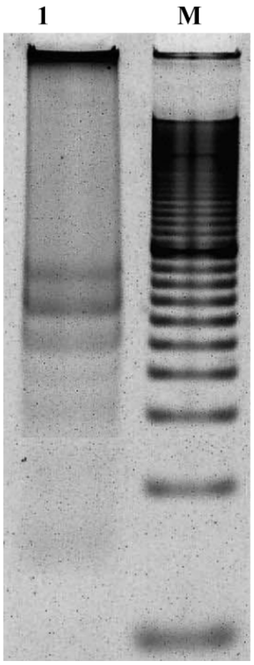

Fig. 7. The example of the output of *in vitro* computation based on hybridization-ligation without weight oligos. Lane M denotes the 20-bp ladder; lane 1 is the product of PCR.

other oligos. However, based on DPLB-DNAC, if hybridization-ligation method is employed, various lengths of oligos should hybridize to generate all the candidate answers in massively parallel fashion; therefore, hybridization-ligation does not seem to be a good method for initial pool generation of DPLB-DNAC. Hence, by employing POA, a shortcoming, which is the biological influence contributed by the length of the oligos, could be eliminated efficiently. In this research, however, we proved that for such a small input graph as shown in Fig. 3, hybridization-ligation is also able to generate the initial pool generation of DPLB-DNAC as POA, without degrading the correctness of *in vitro* computation, but the hybridization-ligation method needs additional input, which are weight oligos in order to generate successfully all the candidate paths. The correct output of the computation is difficult to achieve if the *in vitro* computation is done without weight oligos as shown in Fig. 7.

Note that in this research, conventional PCR is frequently used to amplify exponentially DNA molecules of interest in a solution. Based on conventional PCR, the innovation of real-time PCR technique gaining rapid popularity and played crucial role in molecular medicine and clinical diagnostics. Real-time PCR requires fluorescence reporter such as hydrolysis probes, moelcular beacon, and hybridization probes, and the real-time amplification can be done on a real-time PCR instrument such as LightCycler. Since the *in vitro* amplification can be monitored in real-time, this is a considerable advantage of real-time PCR over the conventional PCR. In future research, the use of real-time PCR will be investigated to advance the DNA computing from an experimental point of view.

VII. Conclusion

An experimental comparison between hybridization-ligation method and POA for an initial pool generation of DPLB-DNAC is done and presented in this paper. Based on the comparison, both of the methods could be used for initial pool generation of DPLB-DNAC without degrading the correctness of the *in vitro* computation. This finding is proven by experimental results, where the correct answer of the shortest path computation has been successfully visualized by gel electrophoresis. However, as opposed to POA, the hybridization-ligation method needs additional input for initial pool generation, which is the oligos for weight. Even though both methods of initial pool generation are usable, since POA offers a lot of advantages over hybridization-ligation method, POA should be selected, especially if bigger size of problem is considered for in *in vitro*.

Acknowledgment

The authors would like to thank M. Yamamoto of Hokkaido University for discussions that led to improvements in this work and his permission to practice various kinds of biochemical experiments in his laboratory, as well, during a research visit in 2004. The authors would also like to thank A. Kameda, S. Kashiwamura, and members of the DNA Computing Laboratory of Hokkaido University for fruitful explanations and kind assistance during the practice of biochemical experiments. The authors would also like to thank Universiti Teknologi Malaysia (UTM) for a study leave in Meiji University under a SLAB-JPA scholarship. Lastly, special thanks go to Prof. M. Khalid from UTM, Malaysia, for reading and improved the manuscript as well as two anonymous reviewers for their valuable advice and comments.

References

[1] L. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, pp. 1021–1024, 1994.

[2] J. Y. Lee, H. W. Lim, S. I. Yoo, B. T. Zhang, and T. H. Park, "Efficient initial pool generation for weighted graph problems using parallel overlap assembly," in *DNA Computing*. Heidelberg, Germany: Springer-Verlag, 2005, vol. 3384, Lecture Notes in Computer Science, pp. 215–223.

[3] P. D. Kaplan, Q. Ouyang, D. S. Thaler, and A. Libchaber, "Parallel overlap assembly for the construction of computational DNA libraries," *J. Theor. Biol.*, vol. 188, no. 3, pp. 333–341, 1997.

[4] S. N. Ho, H. D. Hunt, R. M. Horton, J. K. Pullen, and L. R. Pease, "Site-directed mutagenesis by overlap extension using the polymerase chain reaction," *Gene*, vol. 77, pp. 51–59, 1989.

[5] K. Jayaraman, S. A. Fingar, and J. Fyles, "Polymerase chain reaction-mediated gene synthesis: synthesis of a gene coding for isozymec of horseradish peroxidase," *Proc. Nat. Acad. Sci. USA*, vol. 88, pp. 4084–4088, 1991.

[6] R. DeSalle, M. Barcia, and C. Wray, "PCR jumping in cones of 30-million-year-old DNA fragments from amber preserved termites," *Experientia*, vol. 49, pp. 906–909, 1993.

[7] J. Y. Lee, S. Y. Shin, S. J. Augh, T. H. Park, and B. T. Zhang, "Temperature gradient-based DNA computing for graph problems with weighted edges," in *DNA Computing : 8th International Workshop on DNA-Based Computers*. Heidelberg, Germany: Springer-Verlag, 2003, vol. 2568, Lecture Notes in Computer Science, pp. 73–84.

[8] S. Y. Shin, I. H. Lee, D. Kim, and B. T. Zhang, "Multiobjective evolutionary optimization of DNA sequences for reliable DNA computing," *IEEE Trans. Evol. Comput.*, vol. 9, no. 2, pp. 143–158, Apr. 2005.

[9] A. J. Hartemink, D. K. Gifford, and J. Khodor, "Automated constraint based nucleotide sequence selection for DNA computation," in *Proc. 4th DIMACS Workshop DNA Based Computers*, 1998, pp. 227–235.

[10] R. Penchovsky and J. Ackermann, "DNA library design for molecular computation," *J. Comput. Biol.*, vol. 10, no. 2, pp. 215–229, 2003.

[11] A. G. Frutos, A. J. Thiel, A. E. Condon, L. M. Smith, and R. M. Corn, "DNA computing at surfaces: four base mismatch word design," in *Proc. 3rd DIMACS Workshop DNA Based Computers*, 1997, p. 238.

[12] M. Arita and S. Kobayashi, "DNA sequence design using templates," *New Gener. Comput.*, vol. 20, pp. 263–277, 2002.

[13] U. Feldkamp, S. Saghafi, W. Banzhaf, and H. Rauhe, "DNA sequence generator—A program for the construction of DNA sequences," in *Proc. 7th Int. Workshop DNA Based Computers*, 2001, pp. 179–188.

[14] F. Tanaka, M. Nakatsugawa, M. Yamamoto, T. Shiba, and A. Ohuchi, "Developing support system for sequence design in DNA computing," in *Proc. 7th Int. Workshop DNA Based Computers*, 2001, pp. 340–349.

[15] A. Marathe, A. E. Condon, and R. M. Corn, "On combinatorial DNA word design," in *Proc. 5th DIMACS Workshop DNA Based Computers*, 1999, pp. 75–89.

[16] R. Deaton, J. Chen, H. Bi, M. Garzon, H. Rubin, and D. H. Wood, "A PCR-based protocol for *in vitro* selection of noncrosshybridizing olgionucleotides," in *Proc. 8th Int. Workshop DNA Based Computers*, 2002, pp. 196–204.

[17] C. E. Heitsch, A. E. Condon, and H. H. Hoos, "From RNA secondary structure to coding theory: A combinatorial approach," in *Proc. 8th Int. Workshop DNA Based Computers*, 2002, pp. 215–228.

[18] R. Deaton, M. Garzon, R. C. Murphy, J. A. Rose, D. R. Franceschetti, and S. E. Stevens, Jr., "Reliability and efficiency of a DNA-based computation," *Phys. Rev. Lett.*, vol. 80, no. 2, pp. 417–420, 1998.

[19] B. T. Zhang and S. Y. Shin, "Molecular algorithms for efficient and reliable DNA computing," in *Proc. Genetic Programming*, 1998, pp. 735–742.

[20] M. Arita, A. Nishikawa, M. Hagiya, K. Komiya, H. Gouzu, and K. Sakamoto, "Improving sequence design for DNA computing," in *Proc. Genetic and Evolutionary Computation Conf.*, 2000, pp. 875–882.

[21] A. J. Ruben, S. J. Freeland, and L. Landweber, "PUNCH: An evolutionary algorithm for optimizing bit set selection," in *Proc. 7th Int. Workshop DNA Based Computers*, 2001, pp. 260–270.

[22] S. Y. Shin, D. M. Kim, I. H. Lee, and B. T. Zhang, "Evolutionary sequence generation for reliable DNA computing," in *Proc. Congr. Evolutionary Computation*, 2002, pp. 79–84.

[23] G. Paun, G. Rozenberg, and A. Salomaa*, DNA Computing—New Computing Paradigms*. New York: Springer-Verlag, 1998, pt. 1.

[24] J. Y. Lee, S. Y. Shin, T. H. Park, and B. T. Zhang, "Solving traveling salesman problems with DNA molecules encoding numerical values," *Biosystems*, vol. 78, no. 1–3, pp. 39–47, 2004.

Zuwairie Ibrahim (S'03) received the B.Eng (mechatronics) and M.Eng. (image processing) degrees from Universiti Teknologi Malaysia, in 2000 and 2002, respectively. He is currently working toward the Ph.D. degree at the Institute of Applied DNA Computing, Meiji University, Kanagawa, Japan.

He was with the Department of Mechatronics and Robotics, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, as a Lecturer. His research interests include unconventional computation such as molecular or DNA computing as well as image processing, particularly in automated visual inspection for industrial applications. Recently, he contributed his research to a book entitled *Unconventional Computing 2005: From Cellular Automata to Wetware* (Luniver Press, 2005).

Mr. Ibrahim is a student member of the International Society for Nanoscale Science, Computation and Engineering (ISNSCE). He is a recipient of the IEEE Industrial Electronics Society (IEEE-IES) Student Travel Grant 2004, the IEEE Computational Intelligence Society (IEEE-CIS) Walter J. Karplus Student Summer Research Grant 2004, the IEEE-CIS Student Travel Grant 2005, and a bursary of 8th European Conference on Artificial Life 2005.

Yusei Tsuboi (S'03) received the B.Eng. and M.Eng. degrees from Meiji University, Japan, in 2001 and 2003, respectively. He is currently working toward the Ph.D. degree at the Institute of Applied DNA Computing, Meiji University, Kanagawa, Japan.

Since 2004, he has been a Research Associate in the Department of Electrical and Electronic Engineering, Meiji University. His current research interests are DNA computing, soft computing, and image processing.

Mr. Tsuboi is a student member of the Systems, Man, and Cybernetics Society (SMC) and the Computational Intelligent Society (CIS). He received the Sasakawa Scientific Research Grant from the Japan Science Society and Assistance for Participation in Scientific Meetings Abroad from Inoue Foundation for Science as well, in 2005.

Osamu Ono received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Waseda University, Japan, in 1974, 1976, and 1979, respectively.

Previously, he was a Visiting Professor of the Institute of Automatic Control Technology, Technical University of Munich, Germany, and also a Visiting Professor of National University of Singapore, and Malaysia University of Technology. He is currently a Professor of Electrical and Electronic Engineering at Meiji University, Tokyo, Japan. He is also the Director of Institute of Applied DNA Computing of Meiji University and Chairman of Department of Electrical and Electronic Engineering, Meiji University (2001–2004). He is a Dean and an Editor of *Japan Society for Simulation Technology* (JSST). His research interests include biocomputing and artificial intelligence, evolutional and unconventional computing, biorobotics vision, neural image, signal processing, and humanoid mobile robotics. Currently, his research interest is mainly in the nanobioscience and applications of molecular or DNA computing.

Dr. Ono is a Director of the Tokyo branch of the Japan Institute of Electrical Engineers (JIEE). He is President of Advanced High Speed Image Processing Technology Inc. of the Meiji University Technology Licensing Organization (TLO) Incubation Centre.