Experimental Implementation of Direct-Proportional Length-Based DNA Computing for Numerical Optimization of the Shortest Path Problem

Zuwairie Ibrahim1,2, Yusei Tsuboi1, Osamu Ono1, and Marzuki Khalid3

1 Institute of Applied DNA Computing, Meiji University, 1-1-1 Higashi-mita, Tama-ku, Kawasaki-shi, Kanagawa-ken, 214-8571 Japan
{zuwairie, tsuboi, ono}@isc.meiji.ac.jp
http://www.isc.meiji.ac.jp/~i3erabc/IADC.html
2 Department of Mechatronics and Robotics, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor Darul Takzim, Malaysia
zuwairie@fke.utm.my
3 Center for Artificial Intelligence and Robotics (CAIRO), Universiti Teknologi Malaysia, City Campus, Jalan Semarak, Kuala Lumpur, Malaysia
marzuki@utmkl.utm.my

Abstract. Bio-molecular or DNA computing has emerged as an interdisciplinary field that draws together chemistry, molecular biology, computer science, engineering, and mathematics. From DNA computing point of view, it has been proven that it is possible to solve weighted graph problems such as Traveling Salesman Problem (TSP) and the shortest path problem by exploiting some characteristics of DNA. Those characteristics are length, concentration, and melting temperature of DNA. In this paper, we present an alternative length-based DNA computing approach whereby the cost of each path is encoded by the length of the oligonucleotides in a proportional way. The advantage is such that, after an initial pool generation and amplification, polyacrylamide gel electrophoresis can be performed to separate the respective DNA duplex according to their length which directly decodes the results. For an efficient initial pool generation, parallel overlap assembly method is employed. After amplification is done by polymerase chain reaction (PCR), the result of the computation is visualized by polyacrylamide gel electrophoresis. The experimental results show the effectiveness of the proposed direct-proportional length-based computation and prove that the shortest path problem has been successfully solved on a DNA computer.

1 Introduction

A new computing paradigm based on DNA molecules has appeared in 1994 when Leonard M. Adleman [1] launched a novel in vitro approach to solve the so-called Hamiltonian Path Problem (HPP) with seven vertices by DNA molecules. The goal of the HPP is to determine whether a path exists that commences at the ‘start city’ and finishes at the ‘end city’, and passes through each of the remaining cities exactly once. While in conventional silicon-based computers, information is stored as binary num-
bers in silicon-based memories, in this novel approach, the information of the vertices is encoded by random DNA sequences. The computation is performed in biomolecular reaction fashion which involves hybridization, denaturation, ligation, magnetic bead separation, polymerase chain reaction (PCR), and so on. The output of the computation, also in the form of DNA molecules can be read and visualized by electrophoretical fluorescence operation.

Four years later, in 1998, a length-based DNA computing which is called constant-proportional length-based DNA computing specifically for Traveling Salesman Problem (TSP) is proposed by Narayanan and Zorbalas [2]. A constant increase of DNA strands is encoded according to the actual length of the distance. A drawback of this method is that, there is a possibility of an occurrence of concatenated DNA strands of two distances which could be longer than the DNA strand of the longest distance that has been encoded. This may lead to errors in computing the shortest path [3]. This scheme, however, has not been realized by any laboratory experiment.

On the other hand, Yamamoto et al. [4] presented concentration-controlled DNA computing for accomplishing a local search for the shortest path problem. Although DNA computing with concentration control method enables local search among all the candidate solutions, it cannot guarantee that the most intensive band is the DNA representing the shortest path in the given graph. In addition, it is technically difficult to extract a single optimal solution from the most intensive band [3].

Lee et al. [5] proposed a DNA computing approach based on DNA melting temperature for solving TSP problem. Denaturation Temperature Gradient Polymerase Chain Reaction (DTG-PCR) has been introduced where DNA duplex of correct solutions will be denatured and amplified by the PCR operation. As the denaturation temperature increases, other DNA strands will be also subsequently amplified. However, the amount of correct solutions will also be exponentially increased which does affect the final solution.

Due to the unsolved disadvantages, the constant-proportional length-based DNA computing has not yet been implemented in any laboratory experiments. Hence, with the aim to solve the limitation of the constant-proportional length-based approach, an alternative approach called direct-proportional length-based DNA computing is proposed. The shortest path problem has been selected for consideration of using the proposed technique. In this approach, the cost of an edge is encoded as a direct-proportional length oligos. After an initial pool generation and amplification, since numerous numbers of solution candidates are generated, by using the standard biomolecular laboratory operations, it is possible to extract the optimal combination which represents the solution to the shortest path problem.

2 The Shortest Path Problem

Even though the shortest path problem is belonging to the class P, i.e., it is not hard to solve this problem, it is worth to be solved by DNA computing because numerical evaluations are required during the computation. 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 Figure 1, 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 \( V_1 - V_3 - V_4 - V_5 \).

Fig. 1. Example showing a weighted directed graph \( G = (V, E) \) with the shortest path shown as \( V_1 - V_3 - V_4 - V_5 \)

### 3 DNA Sequence Design and Synthesis

Consider a directed graph and the output of the shortest path computation as shown in Figure 1. 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 correspond to the \( i \)th node in the graph and its complement respectively. By using the available software for DNA sequence design, DNASEquenceGenerator [6], the DNA sequences \( V_i \) is designed and listed in Table 1. Melting temperature, \( T_m \), is calculated based on Sugimoto nearest neighbor thermodynamic parameter [7]. The GC contents (GC%) and melting temperature \( (T_m) \) of each sequence are also shown. Table 2, on the other hand, shows the complement of the node sequences.

<table>
<thead>
<tr>
<th>Node, ( V_i )</th>
<th>20-mer Sequences (5’-3’)</th>
<th>GC%</th>
<th>Melting Temperature, ( T_m ) (°C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V_1 )</td>
<td>AAAGCTCGTCGTATTAGGAGC</td>
<td>50</td>
<td>60.9</td>
</tr>
<tr>
<td>( V_2 )</td>
<td>GCACTAGGGATTTGGAGGTT</td>
<td>50</td>
<td>60.3</td>
</tr>
<tr>
<td>( V_3 )</td>
<td>GCTATGCGTATGAGGGCGA</td>
<td>55</td>
<td>60.5</td>
</tr>
<tr>
<td>( V_4 )</td>
<td>CGATAACCGAATCGTAGAAAGCG</td>
<td>50</td>
<td>60.6</td>
</tr>
<tr>
<td>( V_5 )</td>
<td>CGTGGGTGGCTGCTAGTAGG</td>
<td>55</td>
<td>60.5</td>
</tr>
</tbody>
</table>
Table 2. Complement node

<table>
<thead>
<tr>
<th>Complement Node, $\overline{V}_i$</th>
<th>20-mer Complement Sequences (3’-5’)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\overline{V}_1$</td>
<td>TTTCGAGCAGCAAATCCTCG</td>
</tr>
<tr>
<td>$\overline{V}_2$</td>
<td>CGTGATCCCTAAACCTCCAA</td>
</tr>
<tr>
<td>$\overline{V}_3$</td>
<td>CGATACGGCATCATCTCGCT</td>
</tr>
<tr>
<td>$\overline{V}_4$</td>
<td>GCTATGGCTTGACTATTCGC</td>
</tr>
<tr>
<td>$\overline{V}_5$</td>
<td>GCACCCACCGAGACATTATC</td>
</tr>
</tbody>
</table>

We introduce three rules to synthesize oligos for each edge in the graph as follows:

(i) If there is a connection between $V_i$ to $V_j$, where $j \neq n$, synthesize the oligo for edge as
$$V_i(20) + W_{ij}(\omega - 30) + V_j(20)$$

(ii) If there is a connection between $V_i$ to $V_j$, where $i \neq 1, j \neq n$, synthesize the oligo for edge as
$$V_i(20) + W_{ij}(\omega - 20) + V_j(20)$$

(iii) If there is a connection between $V_i$ to $V_n$, where $i \neq 1$, synthesize the oligo for edge as
$$V_i(20) + W_{in}(\omega - 30) + V_n(20)$$

where $V$, $W$, and ‘+’ denote the DNA sequences for nodes, DNA sequences for weight, and ‘join’ respectively. The synthesized oligos consist of three segments; two node segments and an edge segment. ‘$\omega$’ 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 synthesized so that the number of DNA bases of that oligo and the cost at the corresponding edge are similar. Table 3 lists all the synthesized oligos based on the proposed synthesis rules. Again, DNASequenceGenerator [6] is employed. The node segment and edge segment are distinguished by capital and small letters respectively. The complement sequences of each node are synthesized as well.

4 Direct-Proportional Length-Based DNA Computing for the Shortest Path Problem

Currently, there are two kind of initial pool generation methods for solving weighted graph problem: hybridization/ligation and parallel overlap assembly (POA). The hybridization/ligation method has been firstly introduced by Adleman [1] to solve a HPP. For hybridization/ligation method, during the operation, the link oligos hybrid-
ize through the hydrogen bonds by enzymatic reaction. The hybridization/ligation reaction is well shown in Figure 2 [8].

Table 3. DNA sequences for edges

<table>
<thead>
<tr>
<th>Edge</th>
<th>DNA Sequences</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_4-W_{45}-V_5$</td>
<td>5'-CGATACCGAACTGATAAGCG ccaagCGTGGGTTGGCTCTGTAATAG-3’</td>
</tr>
<tr>
<td>$V_5-W_{34}-V_4$</td>
<td>5'-GCTATGCGTAGTAGAGCGA ccgctCGATACCGAACTGATAAGCG-3’</td>
</tr>
<tr>
<td>$V_{1-13}-V_3$</td>
<td>5'-AAAGCTGCGTTTAGGAGCacgctggttcaacgttacctaatcGCTATGCGTAGTAGAGCGA-3’</td>
</tr>
<tr>
<td>$V_{2-23}-V_3$</td>
<td>5'-GCACTAGGGATTTGAGGTTC cctgttgttgactgttggtgtctactactggagtggagtcctgggttttaacagtcctgtactatgggttatttgcag</td>
</tr>
<tr>
<td>$V_{2-24}-V_4$</td>
<td>5'-GCACTAGGGATTTGAGGTT cctgtgtgcgtgcgtaaggcggtggtggttttaacagtcctgtactatgggttatttgcag</td>
</tr>
<tr>
<td>$V_{2-25}-V_5$</td>
<td>5'-AAAGCTGCGTTTAGGAGCacgctggttcaacgttacctaatcGCTATGCGTAGTAGAGCGA-3’</td>
</tr>
</tbody>
</table>

POA has been used [9] and broadly applied in gene construction [10-12], gene reconstruction [13], and DNA shuffling [14]. POA involves thermal cycle and during the thermal cycle, the position strings in one oligo anneals 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 (dsDNA). One cycle of parallel overlap assembly is depicted in Figure 3 [8]. After a number of thermal cycles, a data pool with all combinations could be built.

Recently, Lee et al. [8] did a comparison between hybridization/ligation method and POA for initial pool generation of DNA computing. They came out with a conclusion that for the initial pool generation of weighted graph problems, POA method is more efficient than that of hybridization/ligation method. According to [8], the advantages of POA over hybridization/ligation method for initial pool generation are as follows:

(i) 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 cycle and large experimental scale could include the practical pools.
Fig. 2. Hybridization/ligation method for initial pool generation. The arrowhead indicates the 3' end.

Fig. 3. Parallel overlap assembly for initial pool generation. 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.
Initially, two single-stranded DNA molecules partially hybridize in the annealing step and then they are extended by polymerase. The elongated DNA molecules are denatured to two single-stranded DNA 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. The population size decreased 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 prerequisite for the ligation of oligos.

POA demands less time than hybridization/ligation method. Hybridization requires one and half hour while ligation required more than 12 hours. Hence, POA for 34 cycles requires only two hours. Therefore, POA is much more efficient and economic method for initial pool generation.

As stated in [3], “In addition, the fact that larger weights are encoded as longer sequences is contrary to the biological fact that; the longer the sequences are, the more likely they hybridize with other DNA strands, though we have to find the shortest DNA strands”. From the biological point of view, this argument is definitely true. In order to overcome the limitation of general length-based DNA computing, the authors discovered that by utilizing POA for initial pool generation, a phase where numerous combinations of random routes of the graph are generated in the solution, a shortcoming, which is the biological influence contributed by the length of the oligos could be eliminated.

POA for the example problem by using POA method, the input to the computation are all the synthesized oligos as listed in Table 3 and the complement sequences for each nodes, which are listed in Table 2. These inputs are poured into a test tube and the cycles begin. In fact, the operation of POA is similar as polymerase chain reaction (PCR) but the difference is that POA operates without the use of primers. As PCR, one cycle consists of three steps: denaturation, hybridization, and extension.

At this stage, an initial pool of solution has been produced and it is time to filter out the optimal combinations among the vast alternative combinations of the problem. Unlike conventional filtering, this process is not merely throwing away the unwanted DNA duplex but rather copying the target DNA duplex exponentially by using the incredibly sensitive PCR process. This can be done by amplifying the DNA duplex that contain the start node $V_1$ and end node $V_5$ using primers. After the PCR operation is accomplished, there should be numerous number of DNA strands representing the start node $V_1$ and end node $V_5$ traveling through a possible number of nodes.

The output solution of the PCR operation then undergoes gel electrophoresis operation. During this operation, the DNA molecules can be separated in terms of its length and hence, the DNA duplex $V_1 - V_3 - V_4 - V_5$ representing the shortest path...
starting from $V_1$ and ending at $V_5$ can be visualized. The overall procedure of direct-proportional length-based DNA computing is given in Figure 4.

![Flowchart](image)

**Fig. 4.** The overall algorithm of direct-proportional length-based DNA computing

5 Experimental Protocols, Results, and Discussions

The initial pool generation by POA is performed in a 100 µl solution containing 12µl oligos (Proligo Primers & Probes, USA), 10 µl dNTP (TOYOBO, Japan), 10 µl 10x KOD dash buffer (TOYOBO, Japan), 0.5 µl KOD dash (TOYOBO, Japan), and 67.5 µl ddH2O (Maxim Biotech). The reaction consists of 25 cycles and for each cycles, the appropriate temperature are as follow:
- 94ºC for 30s
- 55ºC for 30s
- 74ºC for 10s

The product of parallel overlap assembly is shown in Figure 5. According to the gel image, the band in lane 1 is blurs and thus, it is expected that all the candidate answers are successfully generated.

In order to select the paths that begin at $V_1$ and end at $V_5$, DNA amplification is done by employing PCR. The PCR is performed in a 25 µl solution consists of 0.5 µl for each primers, 1 µl template, 2.5 µl dNTP (TOYOBO, Japan), 2.5 µl 10x KOD dash buffer (TOYOBO, Japan), 0.125 µl KOD dash (TOYOBO, Japan), and 15.875 µl ddH2O (Maxim Biotech). The reaction consists of 25 cycles and for each cycles, the appropriate temperature are as follow:
- 94ºC for 30s
- 55ºC for 30s
- 74ºC for 10s

which is the same as POA. The sequences used as primers are AAAGCTCGTCGTTAGGAGC ($V_1$) and GCACCCACCGAGACATTATC ($V_5$).

In order to visualize the result of the computation, the product of PCR is subjected to polyacrylamide gel electrophoresis for 40 minutes at 200V. After that, gel electro-
phoresis, the gel is stained by SYBR Gold (Molecular Probes) and the gel image is captured. Figure 6 shows the product of PCR. Lane 1 consists of four bands showing that all the path that start at $V_1$ and end at $V_5$ have been successfully amplified. Those paths are $V_1 - V_3 - V_4 - V_5$ (100bp), $V_1 - V_2 - V_5$ (130bp), $V_1 - V_2 - V_4 - V_5$ (155bp), and $V_1 - V_4 - V_5 - V_3 - V_4$ (165bp). Clearly, the amplified paths have been sorted in term of length by gel electrophoresis and the output of the shortest path computation appears as the shortest band in lane 1.

Fig. 5. Experimental results of gel electrophoresis on 10% polyacrylamide gel. Lane M denotes 20-bp ladder and lane 1 is the product of POA

6 Lower Bound

The direct-proportional length-based DNA computing is proposed essentially to overcome the shortcoming of constant proportional length-based DNA computing approach. However, by using this approach, the minimum weight of edges that can be encoded is limited and the weight falls in a very narrow range. This is because, the length of the solution is not only proportional to the length of the path it encodes but it also includes the number of vertices in the path. Hence, the lower bound, in term of minimum weight that can be encoded by direct-proportional length-based DNA computing is analyzed. Basically, according to the proposed DNA synthesis rules, the lower bound is achieved when:

$$\omega - \frac{3}{2} \beta = 0$$

(1)

Hence, the minimum weight, $\omega_{\text{min}}$, which can be encoded by oligos is attained as:
\[ \omega_{\text{min}} = \frac{3}{2} \beta \]  

(2)

where \( \beta \) is the number of nucleotides used to encode each node of the input graph.

![The Shortest Path](image)

Fig. 6. Computation output on 10% polyacrylamide gel. Lane M denotes 20-bp ladder and lane 1 is the product of PCR

7 Conclusions

We have presented a new alternative approach called ‘direct-proportional length-based approach’ to solve weighted graph problems using DNA computing. Based on
this approach, it is proposed that the directly proportional length of DNA could be used to encode the cost of each edge. After the initial pool generation and amplification, the DNA duplex is subjected to gel electrophoresis for the separation in term of length. It has been shown that the shortest path is represented by the shortest length of DNA duplex. For the sake of initial pool generation, two kinds of methods are reviewed: hybridization/ligation and POA. For a successful demonstration of direct-proportional length-based DNA computing, we found that POA for initial pool generation is critically important. Finally, it is expected that the proposed approach, would extend the applicability of DNA computing for solving intractable weighted graph problems.

8 Acknowledgements

This research was supported partly by the IEEE Computational Intelligence Society (CIS) Walter J Karplus Student Summer Research Grant 2004 for a research visit in September 2004 at the DNA Computing Laboratory, Graduate School of Information Science and Technology, Hokkaido University, Sapporo, Hokkaido, Japan. The first author would like to thank Masahito Yamamoto for discussions that led to improvements in this work and also the permission to practice various kinds of biochemical experiments in the laboratory. Also, the first author is sincerely grateful to Atsushi Kameda, Satoshi Kashiwamura, and members of DNA Computing Laboratory of Hokkaido University for fruitful explanations and kind assistance during the practice of biochemical experiments, and anonymous reviewers for their important comments. Lastly, the first author is very thankful to Universiti Teknologi Malaysia (UTM) for a study leave.

References


