Comparison of Material Consumption, Experimental Protocols and Computation Time in DNA Computing

N. Rajaee, K. Hong Ping, A. Lit, D. N. S. A. Salleh, and L. Y. Ng

***Abstract*—One of the major constraints in DNA computation is the exponential increase in material consumption and computation time for larger computation size in DNA computing particularly in critical stages such as initial pool generation and extraction during gel electrophoresis. In DNA computation, both the hybridization-ligation method and parallel overlap assembly method can be utilized to generate the initial pool of all possible solutions. In this paper, we discuss and compare the implementation of N × N Boolean matrix multiplication via in vitro implementation between Hybridization-Ligation Method and Parallel Overlap Assembly Method to show that selection of tools and protocols affect the cost effectiveness of a computation in terms of the material consumption, protocol steps and execution time to compute. In general, the the parallel overlap assembly method performs better than hybridization-ligation method in terms of the three parameters mentioned. The calculations are based on approximation of unique sequence strands required for the computation and not actual calculations on the nmol concentration.**

***Index Terms*—DNA computing, material consumption, hybridization-ligation method, parallel overlap assembly method.**

I. Introduction

DNA computing holds the promise for a faster and denser computation with its massively parallel computing capabilities. However, there are several difficulties still remain as stumbling blocks which hinder its development as a practical molecular computing. One of which is the amount of DNA required for a computation that increases exponentially with the size of the problem [1]. Current DNA computing strategies are based on enumerating all candidate solutions and then eliminate incorrect DNA by using selection processes. This requires large numbers of starting molecules at each step and each round of selection, usually via initial pool generation and gel electrophoresis [2].

In solving HPP, the seven-node problem was encoded with 20 oligonucleotide strings. Other problems such as maximal clique problems and encoding DNA words were solved with 28 and 108 encoded strings respectively [3]. Going further, a HPP with 23 nodes would start to require a kilogram quantity of DNA and an increase of nodes from 7 to 70 would require 1025 kg of nucleic acids [4]. Methods proposed for solving TSP, clique problem, vertex-cover problem, clique problem and set packing problems all showed exponentially increasing volumes of DNA and linearly increasing time. LaBean et al (2000) proposed that an n1.89n volume, O ($n2+m2$) time molecular algorithm for the 3-coloring problem and a 1.51n volume, O ($n2m2$) time molecular algorithm for the independent set problem, where n and m are, subsequently, the number of vertices and the number of edges in the problems resolved [5]. Fu (1997) presented a polynomial time algorithm with 1.497n volume for the 3-SAT problem, a polynomial time algorithm with a 1.345n volume for the 3-coloring problem and a polynomial time algorithm with a 1.229n volume for the independent set [6]. Bunow goes on to estimate that an extension combinatorial database would require nearly 10^{70} nucleotides (by comparison, the universe is estimated to contain roughly 10^{80} subatomic particles) [7].

The second problem with DNA computing is its dependency on the reactions produced by the computation via bio-molecular tools. The DNA computing which relies on wet-lab processes is not an exact process. In many situations, the DNA computer may fail to produce exact, algorithmic results due to the concentration of different species, the environment, the temperature and contamination. Errors can be introduced at any protocol steps of the DNA computation which requires utmost care in its preparation and implementation. Thus, an increase in protocol steps will immediately increase the possibilities for errors. The growing numbers of test tubes involved in the computation cause the whole operation to be labor intensive.

From our proposed algorithm and work, the quantity of initial DNA strands to encode the problem is proportionate to the number of vertices and edges existing in the graph problem representing the matrix multiplication. The number of primers to represent the elements in the product matrix is derived from its total number of row and column indicators whereas the total tubes to represent each element in the product matrix is derived from the total number of primer combinations.

Therefore, for an $(m \times k) \bullet (k \times n)$ matrix multiplication problem, the total number of primers is $m + n$ and total number of tubes is $m \times n$. For a 2×2 product matrix, the total number of primers required is 4 and the total number of tubes is also 4. However, as we have calculated, the number of primers and tubes increases drastically for a larger $N \times N$ computation. For a 10×10 product matrix, the total number of primers required is 20 and the total number of tubes to represent all elements in the product matrix is 100. As the size of the problem increases, the volume of DNA increases exponentially and the number of experimental work becomes tedious and impractical to be considered as a viable technology.

Thus it is necessary to study different strategies to encode

Manuscript received March 10, 2014; revised May 15, 2014. This work was supported in part by Universiti Malaysia Sarawak.

N. Rajaee, K. Hong Ping, A. Lit, D. N. S. A. Salleh, and N. Liang Yew are with Faculty of Engineering, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia (e-mail: {rnordiana, hpkismet, lasrani, asdnsdharmiza, ngliangy}@feng.unimas.my).

problems into DNA sequences and varieties of bio-molecular tools to minimize the consumption of materials, experimental protocols and execution time of DNA computation.

In this paper we discuss and compare the material consumption, protocol steps and execution time to compute $N \times N$ Boolean matrix multiplication via in vitro implementation between Hybridization-Ligation Method and Parallel Overlap Assembly Method to support how the choice of methods affect the three parameters mentioned. The calculations are based on approximation of unique sequence strands required for the computation and not actual calculations on the nmol concentration.

II. Hybridization-Ligation Method and Parallel Overlap Assembly Method

Hybridization is the annealing of complementary single stranded molecules to form a double stranded DNA. This is the basis for initial path formation during the reaction step and is subsequently employed during the extraction phase for the isolation of generated path molecules. *Ligation* is a process often invoked after single stranded DNA are annealed and concatenated to each other. Many single stranded fragments are connected in series and ligase is used as "glue" to seal the covalent bonds between the adjacent fragments.

Parallel Overlap Assembly (POA) method was successfully applied by Kaplan *et al.* for initial pool generation consisting of binary numbers to solve a maximal clique problem with DNA computing. The initial pool is a combinatorial library containing numerical or indicative information represented by DNA sequences. Construction of computational DNA libraries is based on a DNA shuffling method consisting of two parts; one is the position string of fixed length and the other is value string (0 or 1) of various lengths. The DNA strands corresponding to the same position string are overlapped during annealing step in the assembly process while the remaining parts of the DNA strands are extended by dNTPs incorporation by polymerase. During each cycle in POA, the DNA strands self assemble and extend/elongate as the denaturation and annealing processes are repeated causing the number of target strands decreasing while the lengths of the newly formed strands increasing.

In DNA computing, both hybridization-ligation method and parallel overlap assembly method can be utilized to generate initial pool of all possible solutions for the DNA computation as reported in works by Lee J. Y. *et al*. in 2004 [8], Ibrahim Z. *et al*. in 2004 [9] and Rajaee *et al*. in 2008 [10].

III. Comparison on Material Consumption

A Boolean matrix multiplication problem is represented by a directed graph problem G. Let V be the total number of vertices and E be the total number of edges in G [11]. Consider the problem is modeled and solved using DNA computation via in vitro implementation utilizing both hybridization-ligation method and parallel overlap assembly method. The calculations for material consumption are defined as:

For Model I: Hybridization-Ligation Method

- For $(m \times k)(k \times n)$:

$$V = 2(m + n) \quad (1)$$

$$E_{max} = mk + kn = k(m + n) \quad (2)$$

- For $(m \times k)(k \times l)(l \times \ldots) \ldots (r \times s)(s \times n)$:

$$V = 2(m + n) \quad (3)$$

$$E_{max} = \Sigma\{(mk) + (kl) + \ldots + (sn)\} + \Sigma\{(kl) + \ldots + (rs)\} \quad (4)$$

- For $(n \times n)^1 (n \times n)^2 \ldots\ldots (n \times n)^p$:

$$V = 4n \quad (5)$$

$$E_{max} = 2n^2(p - 1) \quad (6)$$

For Model II: Parallel Overlap Assembly Method

- For $(m \times k)(k \times n)$:

$$V = m + k \quad (7)$$

$$E_{max} = mk + kn = k(m + n) \quad (8)$$

- For $(m \times k)(k \times l)(l \times \ldots) \ldots (r \times s)(s \times n)$:

$$V = \Sigma(m + k + l + \ldots + n) \quad (9)$$

$$E_{max} = \Sigma\{(mk) + (kl) + \ldots + (sn)\} \quad (10)$$

- For $(n \times n)^1 (n \times n)^2 \ldots\ldots (n \times n)^p$:

$$V = n(p + 1) \quad (11)$$

$$E_{max} = pn^2 \quad (12)$$

The comparisons of material consumption for both methods are shown in Table III and Table IV. Graphical representation of material consumption for both methods is shown in Fig. 1.

IV. Comparison on Protocol Steps

We further compare the performance of protocol steps using both hybridization-ligation method and parallel overlap assembly method. The calculations for protocol steps are based on approximation of test tubes required for each protocol step of the computation. Experimental protocols for Hybridization-Ligation method and Parallel Overlap Assembly method are shown in Table I and Table II respectively. Graphical representation of the protocol steps for both methods is shown in Fig. 2.

For Model I: Hybridization-Ligation Method

TABLE I: Experimental Protocol for Model I

Experiment Protocol		**Tubes**
Step 1:	Preliminary Preparation	$V + E + RE + Primer$
Step 2:	Template	1
Step 3:	Hybridization – Ligation	2
Step 4:	Cutting with Restriction Enzymes	$3(m \times n)$
Step 5:	Copy	$3(m \times n)$
Step 6:	Read	$m \times n$

For Model II: Parallel Overlap Assembly Method

TABLE II: Experimental Protocol for Model II

Experiment Protocol		**Tubes**
Step 1:	Preliminary Preparation	$V + E + Primer$
Step 2:	Template	1
Step 3:	Parallel Overlap Assembly	2
Step 4:	Copy	$2(m \times n)$
Step 5:	Read	$m \times n$

TABLE III: MATERIAL CONSUMPTION FOR HL AND POA (N^{TH} POWER OF O)

	Parallel Overlap Assembly			Hybridization-Ligation		
	$\mathbf{O^2}$	$\mathbf{O^{10}}$	$\mathbf{O^{100}}$	$\mathbf{O^2}$	$\mathbf{O^{10}}$	$\mathbf{O^{100}}$
	$(2\times2)^2$	$(2\times2)^{10}$	$(2\times2)^{100}$	$(2\times2)^2$	$(2\times2)^{10}$	$(2\times2)^{100}$
Vertex	6	22	202	8	8	8
Edge	8	40	400	8	72	792
Primer/RE	4	4	4	8	8	8
Total DNA	18	66	606	24	88	808
	$(10\times10)^2$	$(10\times10)^{10}$	$(10\times10)^{100}$	$(10\times10)^2$	$(10\times10)^{10}$	$(10\times10)^{100}$
Vertex	30	110	1010	40	40	40
Edge	200	1000	10000	200	1800	19800
Primer/RE	20	20	20	40	40	40
Total DNA	250	1130	11030	280	1880	19880
	$(100\times100)^2$	$(100\times100)^{10}$	$(100\times100)^{100}$	$(100\times100)^2$	$(100\times100)^{10}$	$(100\times100)^{100}$
Vertex	300	1100	10100	400	400	400
Edge	20000	100000	1000000	20000	180000	1980000
Primer/RE	200	200	200	400	400	400
Total DNA	20500	101300	1010300	20800	180800	1980800

TABLE IV: MATERIAL CONSUMPTION FOR HL AND POA (O^2 AND O^{10})

	a =	2	3	4	5	6	7	8	9	10
POA	p = 2	18	33	52	75	102	133	168	207	250
	p = 10	66	129	212	315	438	581	744	927	1130
HL	p = 2	24	42	64	90	120	154	192	234	280
	p = 10	88	186	320	490	696	938	1216	1530	1880

TABLE V: PROTOCOL STEPS FOR HL AND POA (N^{TH} POWER OF O)

	Parallel Overlap Assembly			Hybridization-Ligation		
	$\mathbf{O^2}$	$\mathbf{O^{10}}$	$\mathbf{O^{100}}$	$\mathbf{O^2}$	$\mathbf{O^{10}}$	$\mathbf{O^{100}}$
	$(2\times2)^2$	$(2\times2)^{10}$	$(2\times2)^{100}$	$(2\times2)^2$	$(2\times2)^{10}$	$(2\times2)^{100}$
Tubes	36	84	624	55	119	839
	$(10\times10)^2$	$(10\times10)^{10}$	$(10\times10)^{100}$	$(10\times10)^2$	$(10\times10)^{10}$	$(10\times10)^{100}$
Tubes	30	110	1010	40	40	40
	$(100\times100)^2$	$(100\times100)^{10}$	$(100\times100)^{100}$	$(100\times100)^2$	$(100\times100)^{10}$	$(100\times100)^{100}$
Tubes	300	1100	10100	400	400	400

TABLE VI: PROTOCOL STEPS FOR HL AND POA (O^2 AND O^{10})

	a =	2	3	4	5	6	7	8	9	10
POA	p = 2	36	71	118	177	24	331	426	533	652
	p = 10	84	167	278	417	584	779	1002	1253	1532
HL	p = 2	55	108	179	268	375	500	643	804	983
	p = 10	119	252	435	668	951	1284	1667	2100	2583

The comparisons of experimental protocol for both methods are shown in Table V and Table VI. Graphical representation of material consumption for both methods is shown in Fig. 2.

V. Comparison on Computation Time

The calculations for computation time are based on approximation of running time for protocol steps required for the computation. From Adleman's architecture in solving Hamiltonian Path Problem (1994) [12], the hybridization-ligation method is followed by cutting reactions using restriction enzymes. However, Lee J. Y. while solving Weighted Graph Problem (2004) [8], Ibrahim Z while solving Shortest Path Problem (2004) [9] and Rajaee (2008) while solving Boolean matrix multiplications utilize parallel overlap assembly method for DNA computation [10]. We replicate the protocols steps using both methods to calculate their computation times. The computation time for both the Hybridization-Ligation method and Parallel Overlap Assembly method are shown in Table VII and Table VIII respectively.

For Model I: Hybridization-Ligation Method

TABLE VII: Computational Time for Model I

Experiment Protocol		Computation Time
Step 1:	Hybridization – Ligation	8 – 10 hours
Step 2:	Cutting with Restriction Enzymes	1 – 2 hours
Step 3:	Copy	1.5 hours
Step 4:	Read	1 hour

For Model II: Parallel Overlap Assembly Method

TABLE VIII: Computational Time for Model II

Experiment Protocol		Computation Time
Step 1:	Parallel Overlap Assembly	1.5 hours
Step 2:	Copy	1.5 hours
Step 3:	Read	1 hour

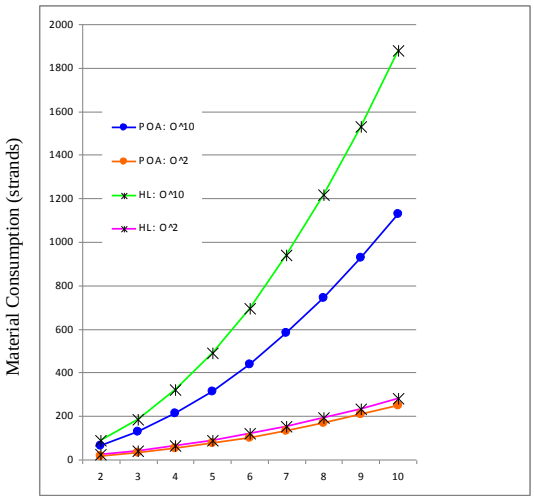

Fig. 1. Material Consumption for HL and POA (O^2 and O^{10})

VI. Results and Discussions

From the results, we have proven that the hybridization-ligation method consume more material, protocol steps and computation time compared to parallel overlap assembly method.

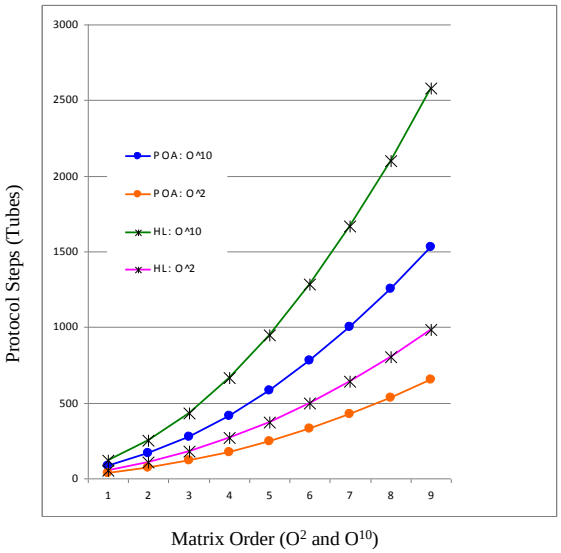

Fig. 2. Protocol steps for HL and POA (O^2 and O^{10}).

A major factor contributing to these differences is the built-up operational structure of parallel overlap assembly method which is extremely similar to massively copying mechanism called polymerase chain reaction (PCR). While hybridization-ligation is more time consuming due to its slow annealing of complementary A-T, C-G pairs for the DNA sequences, parallel overlap assembly generates complementary oligonucleotides by extension of DNTP incorporation by polymerase to the DNA sequences. The hybridization-ligation method is also more prone to mishybridizations if the cooling processes are not slow enough.

Another significant observation between the hybridization-ligation and parallel overlap assembly method is the difference in their execution time. The ligation process which requires phosphorylation of oligonucleotides took 8 hours and the whole process for computation with hybridization-ligation took a total execution time of more than 24 hours whereas the parallel overlap assembly method achieve around a third of hybridization-ligation method execution time.

VII. Conclusion

The performance of parallel overlap assembly method is more cost effective than hybridization-ligation method in generating initial pool of all possible solutions for DNA computation. Based on these results, we can conclude that different experimental protocol yield varied results and the selection of bio-molecular tools and experimental protocols are crucial in determining the material consumption, protocol steps and computation time in DNA computing.

Acknowledgment

The authors would like to thank Universiti Malaysia Sarawak for supporting this research work.

REFERENCES

[1] Z. Ezziane, "DNA computing: applications and challenges," *Nanotechnology*, Institute of Physics Publishing, vol. 17, pp. R27-R39, 2006.

[2] J. C. Coz, D. S. Cohen, and A. D. Ellington, "The complexities of DNA computation," *Trends Biotechnology*, vol. 171, pp. 151-154, 1999.

[3] Q. Ouyang, P. D. Kaplan, S. Liu, and A. Libchaber, "DNA solution of the maximal clique problem," *Science*, vol. 278, pp. 446 – 448, 1997.

[4] M. N. Stojanovic, D. Stefanovic, T. Labean, and H. Yan, "Computing with nucleic acids," *Bioelectronics*, Wiley-Vch Verlag, pp. 427-455, 2004.

[5] LaBean, M. C. Reif, and T. H. Seeman, "Logical computation using algorithmic self-assembly of DNA triple-crossover molecules," *Nature*, vol. 407, pp. 493-496, 2003.

[6] B. Fu and R. Beigel, "A comparison of resource bounded molecular computation models," in *Proc. the 5th Israel Symposium on Theory of Computing and Systems*, 1997, pp. 6-11.

[7] B. Bunow, "On the potential of molecular computing," *Science*, vol. 268, pp. 482-483, 1995.

[8] 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 *Proc. 10th International Meeting on DNA Based Computers*, 2004, pp. 357-364.

[9] Z. Ibrahim, Y. Tsuboi, and O. Ono, "Direct proportional length based DNA computing for shortest path problem," *International Journal of Computer Science & Applications*, vol. 1, no. 1, pp. 46-60, 2004.

[10] N. Rajaee, Y. Kon, K. Yabe, and O. Ono, "Matrix multiplication with DNA based computing: A comparison study between hybridization-ligation and parallel overlap assembly," in *Proc. the 4th IEEE International Conference on Natural Computation*, Jinan, China, 2008, pp. 527-530.

[11] J. S. Oliver, "Matrix multiplication with DNA," *J Mol*, vol. 45, pp. 161-167, 1997.

[12] L. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, no. 5187, pp. 1021-1024, 1994.

Nordiana Rajaee received her BEng (Hons) in electronic and information engineering from Kyushu Institute of Technology, Fukuoka, Japan in 1999, her MSc in microelectronics in University of Newcastle Upon Tyne, United Kingdom in 2003 and PhD in DNA computing from Meiji University, Japan in 2011. Her current interests are in DNA computing, evolutionary computing and microelectronics.

Kismet Hong Ping received his BEng (Hons) in electronics and telecommunications from Universiti Malaysia Sarawak, MSc in digital communication systems from Lougborough University, United Kingdom and PhD in systems science from Nagasaki University Japan. His research interests are biomedical imaging and microwave and antenna technologies.

Asrani Lit received his both B.Eng (Hon.) in computer engineering in 2007 and M.Eng microelectronics & computer system in 2011 from Universiti Teknologi Malaysia. His research interests are in microelectronics, computer system and network-on-chip.

Dyg Nur Salmi Dharmiza Awg Salleh received her BEng in VLSI system design from Ritsumeikan University Japan and MEng in electronics and applied physics in Tokyo Institute of Technology Japan. Her research interests are VLSI design and microelectronics.

Liang Yew Ng received the B.S. degree in electrical engineering degree from University of Alabama, AL, USA in 1990 and M.S. in electrical engineering from Rochester Institute of Technology, Rochester, NY, USA in 1992. He is currently a lecturer in the Department of Electronic Engineering with Universiti Malaysia Sarawak. His research interests include signals processing, communication and integrated circuit design. He is a member of the Institute of Electrical and Electronics Engineers.