DNA Computing Models for Boolean Circuits and Logic Gates

Kuntala Boruah
Department of Electronics and Communication Engineering
Tezpur University
Tezpur, Assam, India
Email: kuntalab@tezu.ernet.in

Jiten Ch. Dutta
Department of Electronics and Communication Engineering
Tezpur University
Tezpur, Assam, India
Email: jitend@tezu.ernet.in

***Abstract*— This paper aims to focus on several practical and theoretical models proposed in one of the emerging subfields of DNA computing which deals with simulating Boolean Circuits and logic gates at molecular level. Deoxyribonucleic acid (DNA) based logic gates are crucial component of molecular computer. Several advantages and shortcomings of some remarkable models employed for implementation of Boolean Circuits and logic gates are discussed and at the end it is concluded that the research on this area towards the realization of molecular computer will be good in near future.**

Keywords—DNA Computing; Boolean Circuit; logic gate; Wet lab experiments

I. INTRODUCTION

The field of DNA Computing has 20 years of dynamic past. Richard Feynman [1] first proposed the possibility of building a computer at "submicroscopic" level but the development in this field geared up after the first experimental demonstration performed by Adleman [2].He solved a seven node Hamiltonian Path Problem (HPP) of a directed graph.

Since then led by Adleman, several authors have proposed how various problems of computation may be simulated and solved using DNA molecules. DNA computing research is not only confined within the context of solving NP class problems, but also spreads to other areas. Some of the proposed models are: Turing machines [3]-[4], splicing [5]-[6], combinatorial optimization [7]-[8] and Boolean circuits and logic gates [9]-[27].With the visionary work of Ogihara and Ray [9] to simulate Boolean circuits and logic gate, a new paradigm of DNA Computing came into existence. Recently significant amount of research interests are drawn by analog molecular logic gates. This paper concentrates on discussing some of the research achievements and challenges in simulating Boolean circuits and logic gates using bio-molecular techniques.

II. DNA MOLECULES AND ITS COMPUTATIONAL PROPERTY

Tremendous parallelism and immense storage capacity of DNA molecule makes it a first choice to an alternate media of computation. Strands of DNA molecule is a polymer composed of finite set of nucleotides; Adenine (A), Guanine (G), Thymine (T) and Cytosine (C). They always prefer to remain in double stranded form and hence migrate towards its complementary strand to attain its double stranded structure, following Watson-Crick complementary rule i.e. A = T and G ≡ C. The two complementary strands must be anti parallel; if one strand is from $5' \rightarrow 3'$ then its complement strand is from $3' \rightarrow 5'$.DNA computing employed techniques from genetic engineering like synthesis, annealing, denaturing, electrophoresis, PCR, digestion by restriction enzyme etc to simulate several computation operations. Like digital computers where all the information are encoded in the form of 1s and 0s, in DNA computing everything is encoded in the form of strings of four alphabet i.e. $\sum = \{A, T, G, C\}$.

III. DNA BOOLEAN CIRCUITS

A Boolean circuit can be represented as a directed graph G (V, E) with n- inputs and m-outputs. All nodes can be categorized into two types: input nodes (in-degree 0) and gate nodes (max in-degree 2). It can be considered as a network of signal processors (gates) carried out the computation of Boolean function. Input variable x_i from the input set ($X_n = x_1, x_2 \ldots x_n$) is associated with input nodes. Boolean function $f_i \in \Omega$ is associated with gate nodes (g_i), where Ω is the circuit basis. Ω = {AND, OR, NOT} are the common bases of any circuit. In some cases NAND gate alone can be used as complete basis. Generally Boolean circuits are leveled structure i.e. all gates in each level are same. The acyclic graph that was first used to implement Boolean circuit is shown in Fig 1.

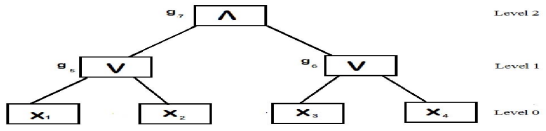

Fig 1: Depth 2 AND-OR Boolean Circuit [9].

The work of Ogihara and Ray[9] is significant as it is the first simulation model of Boolean circuit at molecular level and also it verifies Turing- completeness of DNA computers.

A. Molecular implementation of Ogihara and Ray Model :

The Steps of simulations are as follows :

a) *Step 1: Gates and the edges between them are encoded*: The gates and the input (x_i) were encoded by sequence of DNA oligonucleotides of length ℓ represented as $\partial[i]$ with a specific restriction site at the beginning. The edges

978-1-4799-6023-1/15 $31.00 © 2015 IEEE
DOI 10.1109/CICT.2015.105

$e_{i\to j}$ between two nodes were encoded by concatenation of complements of last half of i^{th} node($3'$ ℓ /2 mer of $\partial[i]$) and the first half of j^{th} node ($5'$ ℓ /2mer of $\partial[j]$).

b) *Step 2: Level 0 simulation*: Only those input variable (x_i) whose value corresponds to 1 ($x_i = 1$) were poured into the test-tube T_1. Thus at the end of level 0 simulation, T_1 contain only those strands which corresponds to value 1.

c) *Step 3: OR gate Simulation*: DNA sequence corresponds to gate g_j i.e. $\partial[j]$ was poured into T_1 which already contains x_i (where $x_i = 1$). Again edge strands $e_{i\to j}$ was added to T_1 and allowed them to hybridize. Presence of any of the two inputs to gate g_j results in forming strands of length 2ℓ. Output of OR gates considered as 1 if DNA strands of length 2ℓ exist in the solution. Gate strands whose value evaluated as 0 were destroyed with the help of restriction enzyme. These DNA strands of length 2ℓ were cut by restriction enzymes for reuse as input to next level.

d) *Step 4: Simulation of AND gate*: After step 3 the test tube T_1 contained the gates sequence whose value evaluated as 1. Again strands of g_j and edge strands $e_{i\to j}$ were poured into the test tube T_1 and allowed them to hybridize. If both of the inputs to the gate g_j were present, strands of length 3ℓ was formed. The output of AND ($\wedge$) gate evaluated as 1 when length of strand 3ℓ exists in the solution. These DNA strands of length 3ℓ were cut by restriction enzymes for reuse. Step 3 and Step 4 are depicted in Fig 2.

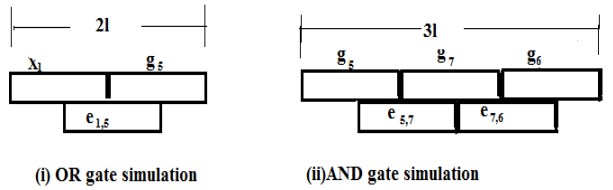

Fig 2: Representing step 3 and step 4 of the Ogihara and Ray experiment [9].

The time and space complexity of this model was proportional to its depth and maximum fan-out of the circuit. Though the theoretical approach was remarkable but the result obtained after lab experiment was ambiguous. The drawback of this model was that it involves several error prone biochemical operations like ligation, hybridization, denaturation and restriction. The pipetting or designing error of edge was also suspected to be the reason behind ambiguous result. This simulation model has a restriction that all the gates in a level of the Boolean circuit must be of same type due to which flexibility in circuit design reduced to a great extant.

Amos and Dunne [11] proposed NAND gate model with same complexity as Ogihara and Ray's model but with easy implementation technique. After this lot of research work implementing Boolean circuits has been performed [12], [23]-[27], [31].

IV. DNA Based Logic Gates

In molecular logic gates input and output are provided in the form of encoded strands of DNA. The gates which act as processing units are specially selected strands of DNA, which are involved in biochemical reactions with the input strand to produce the desired output. Mulawka et al.[13] used Fork-I enzyme of nuclease class II to simulate NAND gate. A new approach of NAND gate simulation and an algorithm for an unbounded fan in Boolean circuit was proposed by Ahrabian and Dalini [15]. Several relevant works are going on in the field of simulating logic gates with DNA [14], [16]-[22]. But all of previous model lacks generalization.

A. Christy M.Gearheart's model:

This is one of the most recent models [18]-[19] proposed to mimic digital data manipulation at molecular level. In this model biochemical reaction is executed in controlled way to attain logical functionality. The presence of double stranded sequence evaluates "true" output otherwise "false".

Implementation of NAND gate:

NAND gate evaluates true result if and only if any or both of the provided input are false. DNA based NAND gate has to simulate this property at molecular level. During this experiment any random sequence was considered as "true" and its complementary strand as "false". Restriction enzyme was added in every step to destroy the entire single stranded DNA (ssDNA) so that double stranded DNA (dsDNA) can be detected easily. The steps of this algorithm are as follows:

a) *Step 1: Base sequence:* The base sequence provided contains predetermined single strands of DNA that represents "true" value.

b) *Step 2: Both inputs as "false"(0,0):* If both of the provided input sequences were "false" i.e. inputs were complementary to the base sequence then annealing occurs between the provided input and the base sequence, hence at the end dsDNA was obtained.

c) *Step 3: Any one input as "false" (0,1 or 1,0)*: If any one of the inputs provided was "false" then due to presence of complementary sequence annealing occurs with the base sequence and dsDNA was obtained.

d) *Step 4: Both input as "true"(1,1):* If both of the inputs provided were "true" then no double stranded structure was obtained because of the absence of complementary strand and the result evaluated as false. The above experiment is depicted in Fig 3.

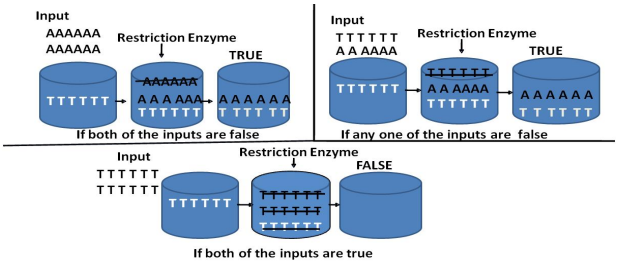

Fig 3: DNA based implementation of NAND gate [18].

Applying similar principle AND, OR, NOT, NOR, and XNOR gates are also implemented. Gearheart et al. [18] later on extended their research work to develop DNA based non-Boolean circuit and shift register [19]. Though this model was easy to implement with less number of operation but it lacked uniform standard in designing the base sequence. The requirement of high rate of human intervention was main drawback of this model.

B. Use of hairpin structure to simulate NAND gate:

Wenbin Liu et al [14] proposed a NAND gate based on induced hairpin formation. Hairpin is the secondary structure motif formed by hybridization of "self-complementary" sequences of DNA or RNA. This structure can be destroyed by two techniques — either by heating or by hybridizing it to complementary strand of the loops part. The formation of hairpin is controlled by adding or removing naphthyridine dimer [28].

The presence of hairpin structure in the gate strand at the end of experiment signifies the output as "1" and its absence represents "0".

Encoded NAND gate consists of two subsequent sequences which correspond to input variables x_1 and x_2 of length ℓ and two complimentary sequences represented by s each of length m with G-G mismatch. Later on during hairpin formation x_1 and x_2 forms the loop portion and the complimentary sequences (s) forms the stem portion. This gate strands are immobilized by attaching to a chemically modified surface with a spacer sequence at the 5′ end. Steps for this algorithm are as follows:

a) *Step 1:* Four different input cases of NAND gate was represented by four different oligonucleotides.

b) Step 2: If either or both input variable x_1 or x_2 was "1" then its complementary strand must be added to the surface otherwise noting to be added and allowed them to anneal.

c) *Step 3:* When napthayridine dimer is added to the surface, hairpin structure was induced. In the case when both of the inputs were "1", the self-loop structure was not possible as hydrogen bond between x_1 and x_2 with its complementary strands are of length 2ℓ which was strong enough to overcome affinity of the self-complementary sequence of stem parts (length m) in both ends Fig 4. Presence of hairpin represents output as "1" otherwise "0".

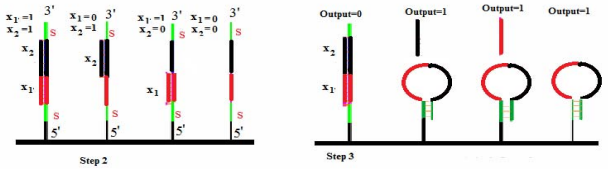

Fig 4: Representing step 2 and step 3 of the experiment [14].

For single NAND gate, output can be detected after Step 3 by technique called Surface Plasmon Resonance (SPR), but to construct network with such NAND gates three more steps needed to be executed.

d) *Step 4:* Hairpin structure was removed by washing away the napthyridine dimer with distilled water.

e) *Step 5:* Now complementary strands 5′ -x_2-s-z- 3′ was added and allowed them to form partial double strand with the gate strands. Again output strand z was added and allowed to anneal. The strand which already had double stranded structure doesn't engage in any hybridization process and output strand z got attached to those gates whose output was "1".

f) *Step 6:* During this step restriction enzyme (Hpa-I) was added so that it cut off the output strand z at restriction site. This restriction site was formed by designing the 3′ end of gate strand with a subsequence 5′ -CAA- 3′ and the output strand was designed with 5′ end with subsequence 5′ -CAA- 3′ Step 5 and Step 6 was shown in Fig 5. Recognition site was formed as follows: 5′ -GTT|AAC- 3′

3′ -CAA|TTG- 5′

The time complexity of this theoretical model increase linearly with the number of level in the circuit but the number of gates doesn't have any effect on the complexity. It was proposed that this technology could be used in next generation of DNA chip. Advantage of this model is that it could be reused in repeated cycle of computation and also the number of inputs may be more than two. Similar works going on in the field of simulating logic gates using the hairpin structure of DNA [21], [29].

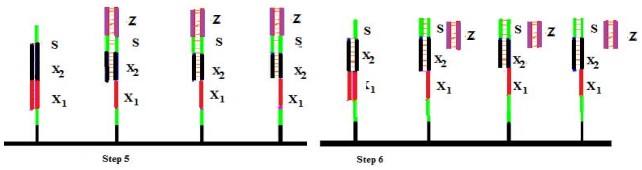

Fig 5: Representing step 5and step 6 of the experiment [14].

TABLE I. TRUTH TABLE FOR MOLECULAR NAND GATE

X_1	X_2	NAND	Gearheart et al model[18]	Wenbin Liu et al model[14]
0	0	1	dsDNA present	Hairpin present
0	1	1	dsDNA present	Hairpin present
1	0	1	dsDNA present	Hairpin present
1	1	0	dsDNA absent	Hairpin absent

C. Use of Strand Displacement behavior to simulate logic gate

The strand displacement is a phenomenon by taking advantage of which several models are proposed. During strand displacement a strand $\overline{A}$ binds to the toe-hold of strand A of AB duplex and displaces the strand B to attain greater stability of this duplex AB.

Shapiro et al [16]-[17], [30] proposed a enzyme driven autonomous DNA computer using strand displacement logic gates as processing units which was coupled with input and output molecule and is capable of diagnosing cancerous or unhealthy cells and accordingly release drug. Several other researchers also employed this strand displacement property in

the design of logic gates [22], [26], [31]-[33]. An enzyme free logic interface by exploiting strand displacement operation has also been developed [34]. Domain-specific language known as DSD is proposed to overcome the problem of designing strand displacement circuits by hand, which was time-consuming and not scalable [35].

Maintaining uniformity in representation of 0s and 1s, reusability of gates and to reduce the chance of error rate are the main challenges faced by researchers while designing molecular gates. Zoraida et al. [21] proposed a generalized algorithm based on hairpin structure of DNA, which seems to address all these problems to great extent. It employs single bio-operation namely hybridization which result in less error rate. This model is reusable and has uniform representation of 0s and 1s.Despite of all the advantages, the algorithm is only theoretic and need to be verified in biochemical lab. Another shortcoming of this model is its limitation in the number of inputs that can be provided to any logic gate.

V. CONCLUSION:

In this paper we have described several models used to simulate Boolean circuits and logic gates at molecular level. Run time complexity of some models are proportional to the size of the circuit while other proportional to the depth of the circuit. In many cases the practical implementation is difficult even though the theoretical approach is remarkable. Work must be carried out to attain reusable and autonomous logic gate. The research work on this new area is therefore expected to witness immense progress in the upcoming years. Developing DNA logic gate might be a new direction of nano scale computing with great possibility of implementation in biomedical field. Research works must be carried out to employ a different input technique in Zoraida et al. model to encode more than two inputs information. It may be concluded that DNA computing towards the realization of analogus logic gate will be good for another few decades.

ACKNOWLEDGMENT

The authors like to extend their gratitude to the Tezpur University for providing facilities in the Laboratory where the literature survey was performed.

REFERENCES

[1] R.P. Feynman, “There’s plenty of room at the bottom,” Engineering and science 23.5 , pp. 22-36,1960.

[2] L.M. Adleman, “Molecular computation of solutions to combinational problems,” Science-AAAS-Weekly Paper Edition 266.5187 , pp.1021-1023, 1994.

[3] P.W. Rothemund, “A DNA and restriction enzyme implementation of Turing machines,” In DNA based Computers,pp.75-120,1996.

[4] W.D. Smith, “DNA computers in vitro and vivo,” In Lipton and Baum, pp. 121–185, 1996.

[5] E.C.Varju, R. Freund, L. Kari and G. Paun, “DNA computation based on splicing:universality results,” First Annual Pacific. Symposium in Biocomputing,pp. 179-190, 1996.

[6] T. Head, “Formal language theory and DNA:an analysis of the generative capacity of specific recombinant behaviors,”Bulletin of Mathematical Biology,pp.737-759,1987.

[7] Q. Ouyang, P.D Kaplan, S. Liu and A. Libchaber, “DNA solution of the maximal clique problem,” Science 278,pp.446-449,1997.

[8] C.G. Owenson, M. Amos and D.A. Hodgsons, “A Gibbsons DNA-based logic,Soft Computing,” pp.102-105. , 2001.

[9] M. Ogihara, and A. Ray, “simulating Boolean circuit on DNA computers, Algorithmica,” vol 2, pp.239-250,1999.

[10] M. Ogihara, and A. Ray, “Executing parallel logical with DNA,” proc. cong. Evolutionary computation, IEEE, pp.972-979, 1999.

[11] M. Amos and P. Dunne, “ DNA simulation of Boolean circuits,” Technical Report CTAG-97009, Department of Computer Science, Universityof Liverpool, 1997.

[12] M. Ogihara and A. Roy, “Simulating Boolean circuits on a DNA computer,” Technical Report 631,University of Rochester, August 1996.

[13] J. Mulawka, P. Wasiewicz, and A. Plucienniczak, "Another logical molecular NAND gate system," IEEE Conference,1999.

[14] W. Liu, X. Shi, S. Zhang, X. Liu, and J. Xu, “A new DNA computing model for the NAND gate based on induced hairpin formation,” Biosystems,pp. 87-92,2004.

[15] H. Ahrabian and A.N. Dalini, "DNA simulation of nand Boolean circuits,"Advanced modeling and optimization ,pp.33-41,2004.

[16] E. Shapiro and B. Gil, "Biotechnology: Logic goes in vitro," Nature nanotechnology ,pp. 84-85,2007.

[17] T. Ran ,S. Kaplan and E. Shapiro,"Molecular implementation of simple logic programs," Nature Nanotechnology ,pp. 642-648, 2009.

[18] C.M. Gearheart, E.C. Rouchka, and B. Arazi, "DNA-based dynamic logic circuitry," Circuits and Systems (MWSCAS),53rd IEEE International Midwest Symposium on. IEEE, 2010.

[19] C.M. Gearheart, E.C. Rouchka, and B. Arazi, "DNA-Based Active Logic Design and Its Implications," Journal of Emerging Trends in Computing and Information Sciences, pp.2079-8407,2012.

[20] H. Jiang, M.D. Riedel and K.K. Parhi, "Digital logic with molecular reactions," Computer-Aided Design (ICCAD), 2013 IEEE/ACM International Conference on. IEEE,pp.721-727, 2013.

[21] B.S.E. Zoraida, M. Arock, B.S.M.Ronald and R..Ponalagusamy, “A novel generalized model for constructing reusable and reliable logic gates using DNA,”In Natural Computation, 2008. ICNC'08. Fourth International Conference on IEEE,pp. 533-537, 2008.

[22] L. Qian and E. Winfree, “A simple DNA gate motif for synthesizing large-scale circuits," Journal of The Royal Society Interface (2011): rsif20100729,2011.

[23] Z.F. QIU and M. Lu, “A new solution for Boolean Circuit with DNA Computer." Department of Electrical Engineering Texas A&M University ,2000.

[24] M. Kadkhoda and A. A. Pouyan, “A DNA-based simulation model for bounded fan-in Boolean circuits," Proceedings of the 10th WSEAS international conference on Computers. World Scientific and Engineering Academy and Society (WSEAS), 2006.

[25] M. Kadkhoda and A.A. Pouyan, "Simulation of Parallel Logical Operations with Biomolecular Computing,”American Journal of Applied Sciences 5.12 (2008): 1758, 2008.

[26] M.R. Lakin, S. Youssef, L. Cardelli and A. Phillips ,“Abstractions for DNA circuit design,” Journal of The Royal Society Interface,pp.470-486, 2012.

[27] M. Amos, “DNA Computation,” PhD thesis, University of Warwick, UK,1997.

[28] E. Smith, M. Kyo, H.Kumasawa, K .Nakatani, I .Saito and R.M. Corn, “Chemically induced hairpin formation in DNA monolayers,” J. Am. Chem. Soc. 124,pp. 6810–6811, 2002.

[29] B.S.E. Zoraida, M. Arock, B.S.M. Ronald and R. Ponalagusamy, “A novel generalized design methodology and realization of Boolean operations using DNA,” Biosystems,pp. 146-153, 2009.

[30] Y. Benenson1, B. Gil, U. B. Dor, R. Adar, and E. Shapiro, "An autonomous molecular computer for logical control of gene expression," Nature journal, vol. 429, pp.423–429, 2004.

[31] G .Seelig, D .Soloveichik, D.Y. Zhang and E. Winfree,“Enzyme-free nucleic acid logic circuits,” Science,pp.1585–1588, 2006.

[32] B.M. Frezza, S.L. Cockroft and M.R. Ghadiri, “Modular multi-level circuits from immobilized DNA-based logic gates,” J Am Chem Soc,pp.14875–14879, 2007.

[33] D. Soloveichik, G. Seelig and E. Winfree , “DNA as a universal substrate for chemical kinetics,” Proc Natl Acad Sci,pp.5393–5398, 2010.

[34] L. Cardelli, “Strand algebras for DNA computing,” In: Deaton R,Suyama A (eds) DNA computing and molecular programming,Chap 2, vol 5877. Springer, Berlin, pp 12–24, 2009 .

[35] R.M. Lakin and A. Phillips, "Compiling DNA strand displacement reactions using a functional programming language," Practical Aspects of Declarative Languages. Springer International Publishing,pp. 81-86, 2014.