IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 3, Issue 12, December 2016

Solving Minimum Independent Dominating Set with Adelman-Lipton model

Ahmed A. Sallam[1], Mohamad Kazem[1], Abdolah B. Askar[1]

Department of Computer Science, Cairo University, Giza, Egypt[1]

Abstract: Adleman showed that deoxyribonucleic acid (DNA) strands could be employed towards calculating solutions to an instance of the Hamiltonian path problem (HPP) [3]. Lipton [5] could solve the Satisfiability problem. In this paper, we use that model for developing a new DNA algorithm to solve minimum independent dominating set problem (MIDSP). In spite of the NP-hardness of minimum independent dominating set problem (MIDSP) our DNA procedures is done in a polynomial time.

Keywords: DNA computing, minimum independent dominating set problem.

I. INTRODUCTION

Recently, DNA computing is one of non-silicon based computing [4]. DNA computing has two very powerful features, Watson-Crick complementarily and massive parallelism [6]. It is clear we cannot solve NP problems with silicon-based computer, But DNA computing provides powerful feature which can solve those problems in polynomial steps. Adleman [1] solved Hamiltonian path problem of size n. That is the first work for DNA computing. Lipton [5] solved the second NP hard problem with this algorithm. Some other NP-hard problems that have been solved [6-21].

In this paper, the DNA operations proposed by Adelman [1] and Lipton [1] are used to solve minimum independent dominating problem.

For a given Graph $G=(V,E)$ we want to find a subset $V^{'} \subset V$ with minimum cardinality such that for all $u \in V-V^{'}$ there is a $u \in V^{'}$ for which $(u,v) \in E$. And, no two vertices in $V^{'}$ are joined by an edge in E .

The rest of this paper is organized as follows. In Section 2, the Adleman–Lipton model is introduced in detail. Section 3 we will present a DNA algorithm for solving the minimum independent dominating set problem and the complexity of the proposed algorithm is described. We give conclusions in Section 4.

II. ADLEMAN-LIPTON MODEL

An easy way to comply with the conference paper formatting requirements is to use this document as a template and simply type your text into it.

Bio-molecular computers work at the molecular level. Since biological and mathematical operations have some similarities, DNA, the genetic material that encodes the living organisms, is stable and predictable in its reactions and can be used to encode information for mathematical problems [7]. DNA algorithms typically solve problems by initially assembling large data sets as input and then eliminating undesirable solutions [14].

A DNA (deoxyribonucleic acid) is a polymer, which is strung together from monomers called deoxyribonucleotides [14]. Distinct nucleotides are detected only with their bases [13].

Those bases are adenine (A), guanine (G), cytosine (C), and thymine (T). Two strands of DNA can form (under appropriate conditions) a double strand, if the respective bases are the Watson–Crick complements of each other, i.e., A matches T and C Matches G; also 3'- end matches 5'- end. For example, strands 5'-ACCGGATGTCA-3' and 3'-TGGCCTACAGT-5' can form a double strand. We also call them as the complementary strand of each other [12].

The length of a single DNA strand is the number of nucleotides comprising the single strand. Thus, if a single DNA strand includes 20 nucleotides, it is called a 20 mer [8]. The length of a double strand (where each nucleotide is base paired) is counted in the number of base pairs [4]. Thus, if we make a double strand from two single strands of length 20 mer, then the length of the double strand is 20 base pairs, also written as 20 bp for more discussion of the relevant biological background, refer to [3]. The DNA operations proposed by Adleman and Lipton [2] are described below.

A (test) tube is a set of molecules of DNA (i.e. a multi-set of finite strings over the alphabet {A, C, G, T}). The following operations perform on tubes [2]:

(1) Merge (T1, T2): for two given test tubes T1, T2 it stores the union $T_1 \cup T_2$ in T1 and leaves T2 empty [4];

(2) Copy (T1, T2): for a, given test tube T1 it produces a test tube T2 with the same contents as T1 [2];

(3) Detect (T): Given a test tube T it outputs ''yes'' if T contains at least one strand, otherwise, outputs ''no'' [2];

(4) Separation (T1, X, T2): for a, given test tube T1 and a given set of strings X it removes all single strands containing a string in X from T1, and produces a test tube T2 with the removed strands [3];

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 3, Issue 12, December 2016

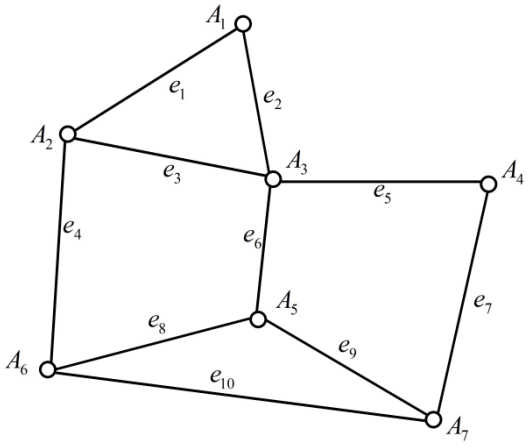

Fig. 1. Graph G.

(5) Selection (T1, L, T2): for a, given test tube T1 and a given integer L it removes all strands with length L from T1, and produces a test tube T2 with the removed strands [8];
(6) Cleavage $(T, \sigma_0\sigma_1)$: for a, given test tube T and a string of two (specified) symbols $\sigma_0\sigma_1$ it cuts each double trend containing $\begin{bmatrix}\sigma_0\sigma_1\\ \sigma_0\sigma_1\end{bmatrix}$ in T into two double strands as follows:

$$\begin{bmatrix}\alpha_0\sigma_0\sigma_1\beta_0\\ \alpha_1\sigma_0\sigma_1\beta_1\end{bmatrix}\Rightarrow\begin{bmatrix}\alpha_0\sigma_0\\ \alpha_1\sigma_0\end{bmatrix},\begin{bmatrix}\sigma_1\beta_0\\ \sigma_1\beta_1\end{bmatrix}$$

(7) Annealing (T): for a, given test tube T it produces all feasible double strands in T. The produced double strands are still stored in T after Annealing [6];
(8) Denaturation (T): for a, given test tube T it dissociates each double strand in T into two single strands [7];
(9) Discard (T): for a, given test tube T it discards the tube T [11];
(10) Append (T, Z): for a, given test tube T and a given short DNA singled strand Z it appends Z onto the end of every strand in the tube T [12];
(11) Read (T): for a, given tube T, the operation is used to describe a single molecule, which is contained in the tube T. Even if T contains many different molecules each encoding a different set of bases, the operation can give an explicit description of exactly one of them [13].

Since these eleven manipulations are implemented with a constant number of biological steps for DNA strands, we assume that the complexity of each manipulation is $O(1)$ steps [14].

III. SOLVING MIDSP BY ADLEMAN-LIPTON MODEL

Let $G=(V,E)$ be an undirected graph with the set of vertices being $V=\{A_k \mid k=1,2,\ldots,m\}$ and the set of edges being $E=\{e_i \mid i=1,2,\ldots,n\}$ [3]. Let |E|=d. In the following, the symbols $0,1,2,\#,X,Y,A_k,B_j,C_j$ $(k=1,2,\ldots,m,\ j=1,2,\ldots,m)$ denote distinct DNA singled strands with same length, say 10-mer. And $||.||$ denotes the length of the DNA singled strand. Obviously, the length of the DNA singled strands greatly depends on the size of the problem involved to distinguish all above symbols and to avoid hairpin formation [3].

Tubes P and Q are defined as follows:
Let
$P=\{j,X,A_1\#,\#B_n,A_kB_{k-1},Y \mid k=1,2,\ldots,n,\ j=01\}$ and
$Q=\{\overline{\#,B_k0A_k},\overline{B_k1A_k} \mid k=1,2,\ldots,n\}$
We design the following algorithm to solve the minimum independent dominating set problem and give the corresponding DNA operations as follows:

IV. PRODUCE EACH POSSIBLE SUBSET FROM E

For a graph with n vertices, each possible subset $V'\subset V$ of vertices is represented by an n-digit number. For example, for graph 1 we can represent $V_1=\{A_1,A_2,A_3\}$ as 0000111 and show $V_2=\{A_5,A_6,A_7\}$ as 1110000, in which 1 in i-th element shows that the vertices A_i is $V'\subset V$, and if j=0 it means that this vertex doesn't exist in that subset.

In this way, we can show all possible subsets with DNA strands. We call this the data pool.
(1-1) Merge (P,Q);
(1-2) Annealing (P);
(1-3) Denaturation (P);
(1-4) Separation $(P,\{A_1\#\},T_{tmp})$*;*
(1-5) Discard (P);
(1-6) Separation $(T_{tmp},\{\#B_n\},P)$*;*

After these six steps, singled strands in tube P will encode all 2^n subset of V. For example, for the graph in Fig. 1 with n=7 we have, e.g. the singled strand $\#B_71A_7B_61A_6B_51A_5B_40A_4B_30A_3B_20A_2B_11A_1\#$
Which denotes the subset $V_1=\{A_1,A_5,A_6,A_7\}$ corresponding to the number 1110001. This operation can be finished in $O(1)$ steps since each manipulation above works in $O(1)$ steps.

V. ELIMINATING INVALID SUBSETS

In definition of problem, no two vertices in V' are joined by an edge in E. It means if for each $(u,v)\in E$, $u\in V'$ and $v\in V'$, those subsets are invalid. For example, $V_1=\{A_1,A_5,A_6,A_7\}$ is an invalid subset because there is an edge between $A_1 -- A_6$.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 3, Issue 12, December 2016

For $i=1$ *to* $d=n$
For $d=1$ *to* $d=n$
if $(A_d, A_j) \in E$
(3-1) *Separation* $(P, \{B_d 1A_d\}, T_1)$
(3-2) *Separation* $(T_1, \{B_i 1A_i\}, T_2)$
(3-3) *Merge* (P, T_1)
(3-4) *Discard* (T_2)
Endif
End for
End for

$B_6 1A_6$, $B_1 1A_1$ mean both of vertices 1 and 6 are in this subset.
If we have an edge between vertices 1 and 6, then this an invalid subset per definition of the problem.
In this part, we remove subsets which have an edge among their vertices.
In the second part, we will remove those subsets that cannot meet this condition, for all $u \in V - V'$ there is a $u \in V'$ for which $(u, v) \in E$. We assume, we have a strand which contains $B_6 0A_6$. It means $6 \in V - V'$ then we are looking for those strands which contain $B_i 0A_i$ and $(6, i) \in E$.
$B_i 0A_i$ means i-th vertices belong to V'.

For $i=1$ *to* $d=n$
(3-1) *Separation* $(P, \{B_d 0A_d\}, T_1)$
For $d=1$ *to* $d=n$
if $(A_d, A_j) \in E$
(3-2) *Separation* $(T_1, \{B_i 1A_i\}, T_2)$
(3-3) *Merge* (T_3, T_2)
(3-4) *Discard* (T_2)
Endif
End for
(3-5) *Merge* (P, T_3)
(3-6) *Discard* (T_2)
End for

$\#B_7 0A_7 B_6 0A_6 B_5 1A_5 B_4 0A_4 B_3 0A_3 B_2 0A_2 B_1 1A_1 \#$
This strand is invalid because in the graph we have an edge between 6 and 7. This strands contain $B_7 0A_7$ And $B_6 0A_6$ .It is obvious those algorithms will terminate in $O(n^2)$.

VI. CALCULATE THE CARDINALITY EACH SUBSETS

The cardinality of each subset is equal to number of vertices in V'. To count the number of vertices in V' we need to count $B_i 1A_i \;\; i=1,2,\ldots,n$. For each strand contain $B_i 1A_i \;\; i=1,2,\ldots,n$ we will add # to the end of those strands.

For $i=1$ *to* $d=n$
(3-1) *Separation* $(P, \{B_d 1A_d\}, T_1)$
(3-2) *Append* $(T_1, \#)$
(3-3) *Merge* (P, T_1)
End for

The number of # demonstrate the cardinality of each subset.
We have n iterations, then this algorithm will terminate at O(n).

VII. FIND THE SUBSET WITH MAXIMUM CARDINALITY

For example, we have V' with n vertices, then the strands of that subsets contain $\frac{\#\#\ldots\#\#}{n}$.

If we found some strands which contain $\frac{\#\#\ldots\#\#}{n}$, those strands will be our solution. Otherwise we will continue with $\frac{\#\#\ldots\#\#}{n-1}$ and $\frac{\#\#\ldots\#\#}{n-2}$,... .

For $i=n$ *to* $i=1$
(2-1) *Separation* $(P, \{\frac{\#\#\ldots\#\#}{i}\}, T_1)$
if T_1 *is not Empty*
(2-2) *Break*
End if
End for

this algorithm will finish in O(n).

VIII. CONCLUSION

In this paper, we propose a procedure for minimum independent dominating set problem in the Adleman–Lipton model. The procedure works in $O(n^2)$ steps for minimum independent dominating set problem of a directed graph with n vertices. All our results in this paper are based on a theoretical model. However, the proposed procedures can be implemented practically since every

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 3, Issue 12, December 2016

DNA manipulation used in this model has been already realized in lab level.

REFERENCES

[1] Leonard M Adleman. Molecular computation of solutions to combinatorial problems. Nature, 369:40, 1994.
[2] Babak Dalvand, Saeed Safaei, and Mojtaba Nazari. Fast parallel molecular solution to the maximum triangle packing problem on massively parallel bio-computing. In FCS, pages 169–173, 2009.
[3] Ardashir Dolati, Mehdi Sohrabi Haghighat, Saeed Safaei, and Hajar Mozaffar. Solving minimum beta-vertex separator problems in the adleman-lipton model. In FCS, pages 97–101, 2008.
[4] Ardeshir Dolati, Mehdi S Haghighat, and Saeed Safaee. Solving bin-packing problem in the adleman–lipton model. In the First Conference and Workshop on Mathematical Chemistry, volume 1, pages 70–74, 2008.
[5] Richard J Lipton. Dna solution of hard computational problems. Science, 268(5210):542, 1995.
[6] Assefi, Mehdi, et al. "An Experimental Evaluation of Apple Siri and Google Speech Recognition." Proccedings of the 2015 ISCA SEDE (2015).
[7] Saeed Safaei, Babak Dalvand, Babak Esmaeili, and Vahid Safaei. Molecular solutions for the minimum edge dominating set problem on dna-based supercomputing. In FCS, pages 32–36, 2009.
[8] Saeed Safaei, Babak Dalvand, Nozar Safaei, and Vahid Safaei. Molecular solutions for the maximum k-facility dispersion problem on dna-based supercomputing. In FCS, pages 513–517, 2011.
[9] Saeed Safaei, Hajar Mozaffar, and Babak Esmaeili. Solving minimum k-center problem in the adleman? lipton model. In FCS, pages 251–255, 2008.
[10] Saeid Safaei, Babak Dalvand, Zahra Derakhshandeh, and Vahid Safaei. Molecular solutions for the bin-packing and minimum makespan scheduling problems on dnabased supercomputing. In PDPTA, pages 512–516, 2009.
[11] Sohrabi-Haghighat, M., Dolati, A., & Safaee, S. (2015). Solving several kinds of constrained shortest path problems via DNA. Applied mathematics in Engineering, Management and Technology, 527-533.
[12] Li, L., Wenhua, L., Wang, Z., Tunhua, W., & Ping, W. (2015). Solving the Maximum Weight Vertex Independent Problem with DNA Molecules in Adleman-Lipton Model. Journal of Computational and Theoretical Nanoscience, 12(8), 1940-1943.
[13] Wang, Z., Ji, Z., Su, Z., Wang, X., & Zhao, K. (2016). Solving the maximal matching problem with DNA molecules in Adleman–Lipton model.International Journal of Biomathematics, 9(02), 1650019.
[14] Zhao, K., Wang, Z., Lu, Y., Qin, J., & Tan, J. (2015). A new biologically DNA computational algorithm to solve the k-vertex cover problem. Journal of Computational and Theoretical Nanoscience, 12(3), 524-526.
[15] Wang, Z., Huang, D., Tan, J., Liu, T., Zhao, K., & Li, L. (2015). A parallel algorithm for solving the n-queens problem based on inspired computational model. BioSystems, 131, 22-29.
[16] Li, L., Zhao, K., & Ji, Z. (2015). A Genetic Algorithm to Solve the Subset Sum Problem based on Parallel Computing. Appl. Math, 9(2), 921-925.
[17] Zheng, X., Wang, B., Zhou, C., Wei, X., & Zhang, Q. (2016). Parallel DNA Arithmetic Operation With One Error Detection Based on 3-Moduli Set. IEEE transactions on nanobioscience, 15(5), 499.
[18] Assefi, M., Liu, G., Wittit, M. P., & Izurieta, C. Measuring the Impact of Network Performance on Cloud-Based Speech Recognition Applications.
[19] Assefi, M., Wittie, M., & Knight, A. (2015, August). Impact of Network Performance on Cloud Speech Recognition. In 2015 24th International Conference on Computer Communication and Networks (ICCCN) (pp. 1-6). IEEE.
[20] Doroodchi, Mahmood, Azadeh Iranmehr, and Seyed Amin Pouriyeh. "An investigation on integrating XML-based security into Web services." GCC Conference & Exhibition, 2009 5th IEEE. IEEE, 2009.
[21] Pouriyeh, Seyed Amin, Mahmood Doroodchi, and M. R. Rezaeinejad. "Secure Mobile Approaches Using Web Services." SWWS. 2010.
[22] Sherafat, A., A. Pouriyeh, and M. Doroodchi. "EV-IMP Model: A comprehensive model for evaluation of an organization's website success." Proceedings of the International Conference on Semantic Web and Web Services (SWWS). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2013.
[23] Moazzam, Akbar, and Babak Dalvand. "Molecular solutions for the Maximum K-colourable Sub graph Problem in Adleman-Lipton model." arXiv preprint arXiv:1610.07294 (2016).