Quest Journals
Journal of Research in Humanities and Social Science
Volume1 ~ Issue 4 (2013) pp: 25-33
ISSN(Online) :2321-9467
www.questjournals.org

Research Paper

A Survey Of Dna Computing

Malasowe Bridget[1], Awodele Oludele[2],
Maitanmi Olusola[3], Emuobonuvie Andy[4]

[1]Computer Science, College of Education, Agbor, Delta, Nigeria
[2]Computer Science, Babcock University, Ilisan Remo, Ogun State, Nigeria
[3]Computer Science, Babcock University, Ilisan Remo, Ogun State, Nigeria
[4]Computer Science, College of Education, Agbor, Delta, Nigeria

Received 18 December, 2013; **A**ccepted 30 December, 2013© The author(s) 2013. Published with open access at **www.questjournal.org**

ABSTRACT:- *In recent times, computer chip manufacturers are frantically racing to make the next microprocessor that will overthrow speed records. So Microprocessors made of silicon will eventually reach their limits of speed and miniaturization and their manufacturers will need a new material to produce faster computing speeds. But fortunately scientists have found the new material they need to build the next generation of microprocessors. Millions of natural supercomputers exist inside living organisms, including our body. DNA (deoxyribonucleic acid) molecules, the material our genes are made of, have the potential to perform calculations many times faster than the most powerful human-built computers. DNA might in the nearest future be integrated into a computer chip to create a so-called BIOCHIP that makes computers even faster. DNA molecules have already been harnessed to perform complex mathematical problems by researchers.While still in its early life,DNAcomputers will be capable of storing billions of times more data than today personal computer. In this article, a survey of recent research on DNA computing is discussed and the possibility that it might take the place of silicon-based computers in the next decade.*

Keywords:- *Complexity, Graphene , Graphs,DNA,Bioinformatic*

I. INTRODUCTION

Biochemical “nanocomputers” already exist in nature. They are manifested in all living things. But they are largely uncontrolled by humans. For instance we cannot program a tree to calculate the digits of pi [1];[2]. To the naked eye, the DNA computer looks like clear water solution in a test tube. There is no mechanical device. A trillion bio-molecular devices could fit into a single drop of water. Instead of showing up on a computer screen, results are analyzed using a technique that allows scientists to see the length of the DNA output molecule. DNA computation is a form of computing which uses DNA and molecular biology, instead of the traditional silicon-based computer technologies. The practical possibility of using molecules of DNA as a medium for computation was first demonstrated by [3]. Adleman's primary intention was to prove the feasibility of biomolecular computation but his work also gave an indication that the emergence of this new computational paradigm could provide an advantage over conventional electronic computing techniques. Specifically, DNA was shown to have massively parallel processing capabilities that might allow a DNA based computer to solve hard computational problems in a reasonable amount of time. However, Adleman's creature force search algorithm is not, and was never meant to be, a practical means of solving such problems; the volume of material required was found to increase exponentially as the complexity of the problem was increased [4]. A brief history, different application areasas well as differnt reseachers findings of DNA Computing will be discussed in this paper.

The computational capability of living systems has intrigued researchers for years. Primarily, the focus has been on implementing aspects of living systems in computational devices. Examples are cellular automata, genetic algorithms, artificial neural networks, and artificial life. The argument has been that universal computational devices are capable of simulating the behavior of physical, living systems through appropriate programming. Therefore, the direction of innovation has been from biology to computer science. This field was

*Corresponding Author: Malasowe Bridget
Computer Science, College of Education, Agbor, Delta, Nigeria

initially developed by [3] of the University of Southern California. In 1994, [3] demonstrated a proof-of-concept use of DNA as form of computation which was used to solve the seven-point Hamiltonian path problem. The electronic computers use two digits that are 0 and 1 known as binary digits, whereas a DNA strand contains four-letter alphabet that is A, T, G and C which can hold large information than earlier type of computers. Since the initial Adleman experiments, advances have been made, and various Turing machines have been proven to be constructable. [5]proposed DNA experiments to solve the satisfiability problem. In 1997, [6] presented a molecular biology based experimental solution to the "maximal clique" problem. In 2000, [7] designed a DNA model system, where a multibased encoding strategy was used in an approach to surfacebased DNA computation. In 2001, [8] analyzed and improved their surface-based method [9]. All these researchers'works used the tools of molecular biology, and all of them demonstrate the feasibility of carrying out computations at the molecular level. One of the formal frameworks for molecular computations is the Head's splicing system, which gives a theoretical foundation for computing based on DNA recombination [10]. Also in 2004, [11] and his co-workers constructed a DNA computer, coupled with an input and output module and is capable of diagnosing cancerous activity within a cell, and then releasing an anti-cancer drug upon diagnosis [12].

II. ALDEMAN'S HYPOTHESIS

In 1994, [3] solved an unremarkable computational problem with a remarkable technique. The type of problem that Adleman solved is formally known as a directed Hamiltonian Path (HP) problem, but is more popularly recognized as a variant of the so-called "traveling salesman problem." A Hamiltonian Path in a connected graph is defined as a closed walk that traverses every vertex of graph 'G' exactly once, except the starting vertex at which the walk also terminates [13];[14]. Fig. 1 shows a connected graph having a Hamiltonian path as ABCDEFGH. The path must pass through each vertex exactly once. The Traveling Salesman Problem can be stated as: A salesman is required to visit a number of cities during a trip. Given the distances between the cities, in what order should he travel so as to visit ever city precisely once and return home, with the minimum mileage traveled? [3] Work is significant for a number of reasons[15];[16]; [17] –

1. It illustrates the possibilities of using DNA to solve a class of problems that is difficult or impossible to solve using traditional computing methods.
2. 2. It is an example of computation at a molecular level, potentially a size limit that may never be reached by the semiconductor industry.
3. 3. It demonstrates unique aspects of DNA as a data structure.
4. 4. It demonstrates that computing with DNA can work in a massively parallel fashion [18]; 19].

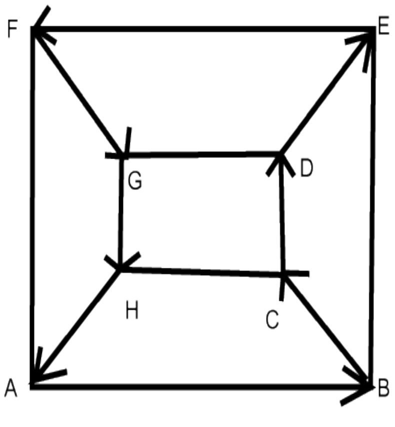

Figure 1.Aconnected graph having a Hamiltonian path

According to [20], DNA has the unique ability to carry out multitasking operations and perform large number of functions simultaneously. Computer based Transistor typically follow the basic von Neumann architecture where instructions are handled sequentially. A von Neumann machine repeats the same "fetch and execute cycle" over and over again; it fetches an instruction and the appropriate data from main memory, and it executes the instruction. DNA computers are **NON**-von Neumann in nature and are stochastic machines that approach computation in a different way from ordinary computers for the purpose of solving a different class of problems[20]. For DNA computation, though, the power comes from the memory capacity and parallel processing [21];[22].

III. DNA LOGIC GATES

Logic gates are a vital part of how your computer carries out functions that you command it to do. These gates convert binary code moving through the computer into a series of signals that the computer uses to perform operations. Currently, logic gates interpret input signals from silicon transistors, and convert those signals into an output signal that allows the computer to perform complex functions. Research have shown that DNA logic gates are the first step toward creating a computer that has a structure similar to that of an electronic personal computer. Instead of using electrical signals to perform logical operations, these DNA logic gates rely on DNA code. These gates are actually tiny DNA processing centers that detect specific fragments of the genetic blueprint as input, and then splice together the fragments to form a single output. For instance, a genetic gate called the “And gate” links two DNA inputs by chemically binding them so they’re locked in an end-to-end structure. The researchers believe that these logic gates might be combined with DNA microchips to create a breakthrough in DNA computation [23].

Implementation techniques: Despite the progress achieved in DNA computation, the main obstacles to creating a practical DNA computer still remain to be overcome. These obstacles are roughly of two types:

1. Practical: Arising primarily from difficulties in dealing with large scale systems and in coping with errors.
2. Theoretical: Concerning the versatility of DNA computers and their capacity to efficiently accommodate a wide variety of computational problems [24].

The synthesis of a DNA strand can sometimes result in the strand annealing to itself and creating a hairpin structure. Even the seemingly straightforward mixing operation can sometimes pose problems. If DNA is not handled gently, the sheer forces from pouring and mixing will fragment it. Also of concern for this operation is the amount of DNA which remains stuck to the walls of the tubes, pumps, pipette tips etc and is thus lost from the computation. Hybridization has also to be carefully monitored because the thermodynamic parameters necessary for annealing to occur are sequence dependent. Besides the accuracy of bio-operations, another challenge of the implementation of DNA computations is the fact that the size of the problem influences the concentration of reactants and this in turn has an effect on the rate of production and quality of final reaction products.

IV. DATA ENCRYPTION STANDARD

Data Encryption Standard (DES) encrypts 64 bit messages and uses a 56 bit key. Breaking DES means that given one (plain-text, cipher-text) pair, we can find a key which maps the plain-text to the cipher-text. A conventional attack on DES would need to perform an exhaustive search through all of the 256 DES keys, which, at a rate of 100,000 operations per second would take 10, 000 years. Thus the molecular programs came into existence that takes about 4 months of laboratory work instead [25]. It is estimated that DNA computation could yield tremendous advantages from the point of view of speed, energy efficiency and economic information storing. DNA computers also have the potential for extraordinary energy efficiency. In principle, one joule is sufficient for approximately 2 X 10^{19} 1igation operations. This is remarkable considering that the second law of thermodynamics dictates a theoretical maximum of 34 X 10^{19} {irreversible} operations per joule (at room temperature). Existing supercomputers are far less efficient, executing at most 109operations per joule. Finally, storing information in molecules of DNA could allow for an information density of approximately 1 bit per cubic nanometer, while existing storage media store information at a density of approximately 1 bit per 10^{12} nm^3[23].

V. OPERABILITY OF DNA-BASED DEVICES

From a practical point of view it maybe not that important to simulate a Turing machine by a DNA computation device. DNAbased devices have been addressed for most models of DNA computation that have so far been proposed [26]; [27]. The existing models of DNA computation are based on various combinations of a few primitive biological operations: -

1 Synthesize: Synthesizing a desired polynomial length strand used in all models.
2 Mixing: Pour the contents of the test tubes into another one to achieve union of bases.
3. Annealing: Bond together two single stranded complementary DNA sequences by cooling the solution.
4. Melting: Break apart a double stranded DNA into its single stranded complementary components by heating the solution .
5 Amplifying: Copying, make copies of DNA strands by using the PolymeraseChain Reaction (PCR).
6 Separation: Separating the strands by length using gel electrophoresis.
7 Extraction: Extracting those strands that contain a given pattern as a substring by using affinity purification [28].
8 Cutting: Cutting DNA double strands at specific sites by using restriction enzymes.
9 Ligation: Paste DNA strands with compatible sticky ends by using DNA ligase.

10 Substitution: Substitute, insert or delete DNA sequences by using PCR site specific oligonucleotide mutagenesis.
11 Marking: Marking single strands by hybridization, complementary sequences are attached to the strands,making them double stranded.
12 Destroying: Destroying the marked strands by using exonucleases.
13 Detecting: Reading the given the contents of a tube and say ‘yes’ if it contains at least one DNA strand and ‘no’ otherwise [29]; [30]; [31].

The bio-operations listed above are used to write programs which receive a tube containing DNA strands as input and return as output either ‘yes’ or ‘no’ or a set of tubes.

VI. MARKOV CHAINS

Various algorithms performing computations over Markov chains have been developed. These determine sequence power of the transition matrix of a Markov chain and their properties of convergence. Some other algorithms help enable to estimate this limit. These also allow the computation of a limit using DNA computation. The states and the transition probabilities have been encoded using strands of DNA for generating paths of the Markov chain [15].

VII. SEQUENCE COMPLEXITY

It has been noticed that randomly generated oligonucleotide populations serve as pools for selecting non-cross-hybridizing sequences, for nanoscale self-assembly and biological and biomedical applications, as well as for DNA computation applications. Various nonlinear kinetic models are present for the complexity estimation of large unknown polynucleotide populations. Models are used to estimate the sequence complexity of the random 20 base-pair population after in vitro denaturation experiments. The kinetic behaviors of the random 20mers can also be evaluated with in vitro thermal melting experiments [32].

7.1 Sticker model

It is essentially easier creating an initial data pool covering answers at first place followed by a series of selection process to destroy the incorrect ones. The surviving DNA sequences are read as the solutions to the problem. But, algorithms are limited to the problem size. As the number of parameters in the studied problem grows, the algorithm becomes impossible owing to the tremendous initial data pool size. The solution sequences are built in parts to satisfy one clause in a step, and eventually solve the whole Boolean formula after a number of steps. The size of data pool grows from one sort of molecule to the number of solution assignments [17].

7.2 Molecular nanotechnology

Researchers have discovered DNA sequences and structures with new functional properties for preventing the expression of harmful genes. Bioinformaticians design rigid DNA structures that serve as scaffolds for the organization of matter at the molecular scale, and can build simple DNA-computing devices, diagnostic machines and DNA motors. The integration of biological & engineering advances offers great potential for therapeutic & diagnostic applications [33].

VIII. CLUSTERING APPROACHES

Clustering is used for revealing a structure in highly dimensional data and arriving at a collection of meaningful relationships in data and information granules. DNA computation has also been used for developing clustering techniques. This is very useful while dealing with huge data sets, unknown number of clusters and encountering a heterogeneous character of available data. It is shown the essential components of the clustering technique through the corresponding mechanisms of DNA computation [34].

IX. DNA-BASED REVERSIBLE GATES

Due to the progress made in areas of nanotechnology, storing information in terms of DNA computing-chips has been possible. The use of reversible logic gates for synthesis of circuits in aspects of quantum computing is being tried [35] Furthermore, it has been seen that in this article, the reversible logic is proposed to be simulated by using DNA molecules and biochemistry operations: the input and the output of a reversible gate or a reversible sequential circuit are both DNA sequences, and the computing progresses correspond to the biochemistry operations. These can also be used for designing the optimal reversible sequential circuits, some new trends in DNA and quantum computing [36].

X. DIGITAL SECTOR

Recent research in the digital sector has revealed that, digital production, transmission and storage have revolutionized to a great extent. But how they are accessed and used have made an increasingly complex task that requires active, continuing maintenance of digital media. This challenge has focused some interest on DNA as an attractive target for information storage because of its capacity for high-density information encoding, longevity under easily achieved conditions[37] According to the researchers previous DNA-based information storage approaches have encoded only trivial amounts of informationor were not amenable to scaling-up, and used no robust error-correction and lacked examination of their cost-efficiency for large-scale information archival. In their paper titled - Towards practical, high-capacity, low-maintenance information storage in synthesized DNA. They described a scalable method that can reliably store more information than has been handled before. They encoded computer files totaling 739 kilobytes of hard-disk storage and with an estimated Shannon information of 5.2×10^6 bits into a DNA code, synthesized this DNA, sequenced it and reconstructed the original files with 100% accuracy. Their theoretical analysis indicates that their DNA-based storage scheme could be scaled far beyond current global information volumes and offers a realistic technology for large-scale, long-term and infrequently accessed digital archiving.

XI. CLASSIC DNA COMPUTATION AND DNA2DNA COMPUTATION

In [38] opined that the primary advantage offered by most proposed models of DNA based computation is the ability to handle millions of operations in parallel. The use of DNA to perform massive searching and related algorithms will be referred to as "**classic**" DNA computation [38]. According to him proposed "classical" models of DNA computers derive their potential advantage over conventional computers from their ability to:

- Perform millions of operations simultaneously;
- Generate a complete set of potential solutions;
- Conduct large parallel searches; and
- Efficiently handle massive amounts of working memory.

They also have some of the following drawbacks:

- Each stage of parallel operations requires time measured in hours or days, with extensive human or mechanical intervention between steps;
- Generating solution sets, even for some relatively simple problems, may require impractically large amounts of memory; and
- Many empirical uncertainties, including those involving: actual error rates, the generation of optimal encoding techniques, and the ability to perform necessary bio-operations conveniently in vitro or in vivo.

His research work also further revealed that another area of DNA computation exists where conventional computers clearly have no current capacity to compete. This is the concept of DNA2DNA computations as suggested by [39] and identified as a potential killer app. DNA2DNA computations involve the use of DNA computers to perform operations on unknown pieces of DNA without having to sequence them first [38]. This is achieved by re-coding and amplifying unknown strands into a redundant form so that they can be operated on according to techniques similar to those used in the sticker model of DNA computation [38].

The potential applications of re-coding natural DNA into a computable form are many and include:

- DNA sequencing;
- DNA fingerprinting;
- DNA mutation detection or population screening; and
- Other fundamental operations on DNA (38).

DNA-based computing could replace silicon-based computing, offering many advantages." [40].A researcher [41] has been working with his students [41] at Nanyang Technical University to propose a way that the manipulation of DNA strands could be used to solve certain types of problems. They are also of the opinion that computations that the human body performs naturally are much faster than even the fastest silicon-based computer. On top of that, Shu points out that silicon is not very environmentally friendly - “There are also heat problems”. DNA-based computing could be faster, friendlier for the environment, and eliminate some of the other problems that come with silicon.” [41]. DNA-based computing could prove especially useful for strategic assignment problems. “Even with developments in silicon-based computing, there are some problems that take even the fastest computers months to solve,” [41]. With DNA-based computing, massive parallel problems, combinatorial problems and AI solving problems could all be addressed with the possibility of greater efficiency [41]. DNA-based computing has the potential to deal with fuzzy data, going beyond digital data [41]. The researcher [41] and his students manipulated strands of DNA at the strand level and at the test tube level. They found that they could fuse strands together, as well as cut them, and perform other operations that would affect the ability of the DNA to compute. In thier model, DNA molecules are used to store information that can be used for computational purposes.

XII. AUTONOMOUS DNA COMPUTATION

A one-pot autonomous DNA computation machine is proposed that is based on photochemical gate transition. Here, photoligation via 5-carboxyvinyldeoxyuridene (cvU) containing oligodeoxynucleotides and photocleavage via carbazole - modified oligodeoxynucleotides, were employed. The binary digit additions are carried out by onetime irradiation at 366 nm in the single test tube. The fluorescence readout by the DNA chip was in good agreement with the correct answer of binary digit additions [42].

XIII. CHALLENGES OF DNA COMPUTING

However, Jian-Jun Shu, points out that DNA-based computing is in the most basic of stages right now. “So far, there are a lot of human manipulations that must be done. In silicon-based computing, the CPU do everything. There is need to get to the point where we just need to provide a command and let the DNA do everything. But mean while Cost is also an issue in DNA computing. At the moment DNA is very expensive and hard to commercialize.

Another challenge with DNA-based computing is the **DISPLAY**. It’s very difficult to display the results from DNA-based computing, since electronics have to be used. “We need to find the missing link between electronic speed, which slow, and DNA speed, which is fast – more like optical speed.” [41]. Despite these challenges, the researcher [41] is optimistic that they have made some progress, and they expect to continue making more progress. DNA is the future of computing.”

Furthermore, DNA computing research has investigated the development of DNA-based nanodevices. Examples are the DNA finite automata [43]; [44] and the realization of logic gates in single deoxyribozymes [44]. For example, instead of utilizing huge amounts of electronic computer power to perform relatively simple analyses on vast quantities of biochemical information, it might be possible to construct a molecular computer, which efficiently processes these data at the molecular level [46]. For the successful implementation of DNA-based computations, the detection of output molecules is of prime importance [46]. According to [46], many of the currently available techniques for the detection of DNA have been used in molecular computing: gel electrophoresis with fluorescent or radiometric visualization, fluorescent labelling and fluorescence resonance energy transfer (FRET), mass spectrometry or surface-based techniques. However, all these methods either detect DNA in bulk quantities or destroy the output molecules. This severely limits the size of the library to be searched: the largest parallel computation reported filtered 2^{20} different molecular species [47], which is less than the number of molecules of one variety necessary for the detection using gel electrophoresis [48]. For molecular automata, this detection limit imposes an equally severe redundancy. Therefore, the application of more sensitive detection technology may significantly enhance the power of DNA computations [46]. In their study, they reported the detection of single molecules of DNA performing a computation. Their procedure for experimental implementation relies on blocking algorithm [49], a parallel search algorithm which involves direct inactivation of those molecules that are not a solution. Fluorescence cross-correlation spectroscopy was employed to detect hybridization between single DNA molecules. They also benchmarked this technology on a small instance of the NP complete SAT problem.

XIV. RESULTS OF THEIR EXPERIMENT - DNA COMPUTATION

SAT problems have frequently been tackled by DNA-based computations by these great researchers – [27];[50];[51] and [47]and their results may be considered a benchmark for new algorithms. The specific SAT problem solved is a 4-variable three SAT problem consisting of four clauses:

$$F = (b \vee c \vee \sim d) \;\&\; (\sim a \vee b \vee \sim c) \;\&\; (a \vee \sim b \vee d) \;\& \\ (\sim a \vee c \vee \sim d), \qquad (1)$$

where a, b, c and d are the four variables with values of true (1) or false (0). OR operations are denoted by ‘∨’, AND operations by ‘&’, while ‘∼’ symbolizes the negation of a variable.

XV. DNA USED TO ASSEMBLE A TRANSISTOR FROM GRAPHENE

As electronics become ever thinner, smaller and faster, scientists always need to think ahead and develop solutions to accommodate the computing needs of the future. DNA is the blueprint for life. Could it also become the template for making a new generation of computer chips based not on silicon, but on an experimental material known as graphene? That’s the theory behind a process that Stanford chemical engineeringused as revealed in Nature Communications [52]. These chemical engineers hope to solve a problem clouding the future of electronics: consumers expect silicon chips to continue getting smaller, faster and cheaper, but engineers fear that this virtuous cycle could grind to a halt. [52]

Silicon has been the most popular semiconductor material used to make chips for a very long time. The basic working unit on a chip is the transistor - they are tiny gates that switch electricity on or off, creating the zeroes (0'^{s}) and ones (1'^{s}) that run the computer software. In other to build more powerful chips, designers have done two major things at the same time: they have the size of the transistors and also swung those gates open and shut faster and faster. Today "We need a material that will let us build smaller transistors that operate faster using less power," [52].

Graphene is a single layer of carbon atoms arranged in a honeycomb pattern. Electrically this lattice of carbon atoms is an extremely efficient conductor.

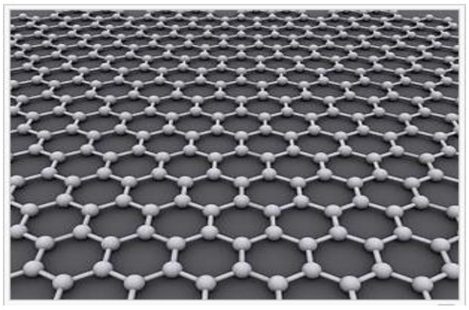

Figure.2 Graphen made of carbon atoms (picture from Wikipedia®, 2013)

The engineers [52] believe that ribbons of graphene (fig 2), laid side-by-side, could create semiconductor circuits. Given the material's tiny dimensions and favorable electrical properties, graphene nano ribbons could create very fast chips that run on very low power. According to the researchers making something that is only one atom thick and 20 to 50 atoms wide was a significant challenge," [52].

To handle this challenge, the Stanford team came up with the idea of using DNA as an assembly mechanism.Physically, DNA strands are long and thin, and exist in roughly the same dimensions as the graphene ribbons that these researchers [52] wanted to assemble. Chemically, DNA molecules contain carbon atoms, the same material that forms graphene.

The real trick was how [52] and her team put DNA's physical and chemical properties to work.

- The researchers started with a tiny platter of silicon to provide a support (substrate) for their experimental transistor. They dipped the silicon platter into a solution of DNA derived from bacteria and used a known technique to comb the DNA strands into relatively straight lines. [52]
- Next, the DNA on the platter was exposed to a copper salt solution. The chemical properties of the solution allowed the copper ions to be absorbed into the DNA.
- Next the platter was heated and bathed in methane gas, which contains carbon atoms.

Once again chemical forces came into play to aid in the assembly process. The heat sparked a chemical reaction that freed some of the carbon atoms in the DNA and methane. These free carbon atoms quickly joined together to form stable honeycombs of graphene."The loose carbon atoms stayed close to where they broke free from the DNA strands, and so they formed ribbons that followed the structure of the DNA," [52].Hence part one of the invention involved using DNA to assemble ribbons of carbon. But the researchers also wanted to show that these carbon ribbons could perform electronic tasks. So they made transistors on the ribbons. These researchers were the first to show that you can use DNA to grow narrow ribbons and then make working transistors.

Their research addresses an important research need for the field. Bao said the assembly process needs a lot of refinement. For instance, not all of the carbon atoms formed honeycombed ribbons a single atom thick. In some places they bunched up in irregular patterns, leading the researchers to label the material Graphitic instead of Graphene.

The researchers concluded that "Our DNA-based fabrication method is highly scalable, offers high resolution and low manufacturing cost," [52]. All these advantages make the method very attractive for industrial adoption." Figure 3 shows the picture of the their experiment.

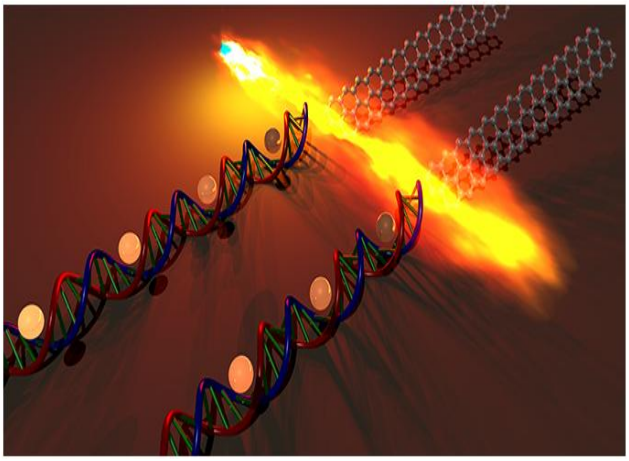

Figure. 3 To the right is a honycomb of graphene atoms. To the left is a double strand of DNA.

The white spheres represent copper ions integral to the chemical assembly process. The fire represents the heat that is an essential ingrdient in the technique. (Anatolly Sokolov). Stanford University

XVI. CONCLUSION AND FUTURE WORK

The apparent ease with which DNA hybridization can be formalized made Adleman's invention very attractive to researchers in the fields of computer science and discrete mathematics. With so many different methods and models emerging from the current research, DNA computing can be more accurately described as a collection of new computing paradigms rather than a single focus. Advancements in DNA computing may also serve to enhance understanding of both the natural and computer sciences. For these reasons, and due to the many areas dependent on each of computer science, mathematics, natural science, and engineering, continued interdisciplinary collaboration is very important to any future progress in all areas of this new field.

With so many possible advantages over conventional techniques, DNA computing has great potential for practical use. Future work in this field should begin to incorporate cost-benefit analysis so that comparisons can be more appropriately made with existing techniques and so that increased funding can be obtained for this research that has the potential to benefit many circles of science and industry.

Furthermore use of DNA computing in an environmental frieindly manner. No doubt it is a good begining for future computing but Green Information Technology should not be left behind because of the issue on GLOBAL WARMING.

REFERENCES

[1]. Melkikh AV (2008)DNA computation, computation complexity and problem of biological evolution rate. Acta Biotheor 56: 285-295.

[2]. Ogasawara S, Ami T, Fujimoto K (2008) Autonomous DNA computation machine based on photochemical gate transition. J Am Chem Soc 130: 10050- 10051.

[3]. Adleman, Leonard M. Molecular Computation of Solutions to Combinatorial Problems. Science, Vol. 266. November 11, 1994, Page 1021.

[4]. Bonsor, Kevin. "How DNA Computers Will Work" 17 November 2000. Retrieved from <http://www.howstuffworks.com/dna-computer.htm> Retrieved date 17 October 2013.

[5]. Lipton RJ (1995) Using DNA to solve NP-complete problems. Priceton University.

[6]. Ouyang Q, Kaplan PD, Liu S, Libchaber A (1997) DNA solution of the maximal clique problem. Science 278: 446-449.

[7]. Liu Q, Wang L, Frutos AG, Condon AE, Corn RM, et al. (2000) DNA computation on surfaces. Nature 403: 175-179.

[8]. Wu H (2001)An improved surface-based method for DNA computation. Biosystems 59: 1-5.

[9]. Macko P, Whelan MP (2008) Fabrication of holographic diffractive optical elements for enhancing light collection from fluorescence-based biochips. Opt Lett 33: 2614- 2616.

[10]. Liu Q, Frutos AG, Thiel AJ, Corn RM, Smith LM (1998) DNA computation on surfaces: encoding information at the single base level. J Comput Biol 5: 269- 278.

[11]. Benenson,Y., Paz,E.T., Adar,R., Keinan,E., Livneh,Z. and Shapiro,E. (2001) Programmable and autonomous computing machine made of biomolecules. Nature, 414, 430–434.

[12]. Los M, Los JM, Wegrzyn G (2008) Rapid identification of shiga toxin-producing Escherichia coli (STEC) using electric biochips. Diagn Mol Pathol 17: 179-184.

[13]. Feldkamp U, Schroeder H, Niemeyer CM (2006) Design and evaluation of single-stranded DNA carrier molecules for DNA-directed assembly. J Biomol Struct Dyn 23: 657-666.

[14]. Liu Y, Xu J, Pan L, Wang S (2002) DNA solution of a graph coloring problem. J Chem Inf Comput Sci 42: 524-528.

[15]. Cardona M,Colomer MA, Conde J, Miret JM, Miró J, et al. (2005) Markov chains: computing limit existence and approximations with DNA. Biosystems 81: 261-266.

[16]. Kim JS, Lee JW, Noh YK, Park JY, Lee DY, et al. (2008) An evolutionary Monte Carlo algorithm for predicting DNA hybridization. Biosystems 91: 69-75.

[17]. Yang CN, Yang CB (2005) A DNA solution of SAT problem by a modified sticker model. Biosystems 81: 1-9.

[18]. Tanaka F, Kameda A, Yamamoto M, Ohuchi A (2005) Design of nucleic acid sequences for DNA computation based on a thermodynamic approach. Nucleic Acids Res 33: 903-911.

[19]. Liu W, Shi X, Zhang S, Liu X, Xu J (2004) A new DNA computation model for the NAND gate based on induced hairpin formation. Biosystems 77: 87-92.

[20]. Somnath Tagore, Saurav Bhattacharya, Md Ataul Islam and Md Lutful Islam. DNA Computation: Applications and Perspectives. OMICS Publishing Group, Journal of Proteomics and Bioinformatics. June 29, 2010

[21]. Schmidt KA, Henkel CV, Rozenberg G, Spaink HP (2004) DNA computation using single-molecule hybridization detection. Nucleic Acids Res 32: 4962- 4968.

[22]. Halpin DR, Harbury PB (2004) DNA display I. Sequence-encoded routing of DNA populations. PLoS Biol 2: E173.

[23]. Ogihara M, Ray A (2000) DNA computation on a chip. Nature 403: 143-144.

[24]. Hug H, Schuler R (2003) Measurement of the number of molecules of a single mRNA species in a complex mRNA preparation. J Theor Biol 221: 615-624.

[25]. Cho A (2000)DNA computation. Hairpins trigger an automatic solution. Science 288: 1152-1153.

[26]. Sakakibara Y, Suyama A (2000) Intelligent DNA chips: logical operation of gene expression profiles on DNA computers. Genome Inform Ser Workshop Genome Inform 11: 33-42.

[27]. Faulhammer D, Lipton RJ, Landweber LF (2000) Fidelity of enzymatic ligation for DNA computation. J Comput Biol 7: 839-848.

[28]. Liu W, Gao L, Liu X, Wang S, Xu J (2003) Solving the 3-SAT problem based on DNA computation. J Chem Inf Comput Sci 43: 1872-1875.

[29]. Aoi Y, Yoshinobu T, Tanizawa K, Kinoshita K, Iwasaki H (1999) Ligation errors in DNA computation. Biosystems 52: 181-187.

[30]. Mills AP, Yurke B, Platzman PM (1999) Article for analog vector algebra computation. Biosystems 52: 175-180.

[31]. Fu B, Beigel R (1999) Length bounded molecular computing. Biosystems 52: 155-163.

[32]. Garzon MH, Jonoska N, Karl SA (1999) The bounded complexity of DNA computation. Biosystems 52: 63-72.

[33]. Condon A (2006)Designed DNA molecules: principles and applications of molecular nanotechnology. Nat Rev Genet 7: 565-575.

[34]. Bakar RB, Watada J, Pedrycz W (2008) DNA approach to solve clustering problem based on a mutual order. Biosystems 91: 1-12.

[35]. Tumpane J,Kumar R, Lundberg EP, Sandin P, Gale N, et al. (2007) Triplex addressability as a basis for functional DNA nanostructures. Nano Lett 7: 3832- 3839.

[36]. Benenson Y, Adar R, Paz-Elizur T, Livneh Z and Shapiro E (2003) DNA molecule provides a computing machine with both data and fuel. Proc Natl Acad Sci USA 100: 2191-2196.

[37]. Nick Goldman, et al Towards practical, high-capacity, low-maintenance information storage in synthesized 2013 DNA 2013

[38]. Joel C. Adams2000. On the Application of DNA Based Computation. Department of Computer Science, University of Western Ontario, Canada

[39]. L. Landweber. R. Lipton. DNA2DNA Computations: A Potential "Killer App"? 3rd DIMACS workshop on DNA based computers. June 1997. 59-68.

[40]. Miranda Marquit May, 2011The next computer: your genes. PhysOrg.com. http://phys.org/news/2011-05-genes.html#jCp

[41]. Jian-Jun Shu, Qi-Wen Want, and Kian-Yan Tong, "DNA-Based Computing of Strategic Assignment Problems," Physical Review Letters (2011). Available online: link.aps.org/doi/10.1103/PhysRevLett.106.188702

[42]. Ogasawara, Shinzi, et al. "Autonomous DNA Computing Machine Based on Photochemical Gate

[43]. Transition." Journal of American Chemistry Society 130.31 (2008): 10050-51.

[44]. Benenson, Y.; Paz-Elizur, T.; Adar, R.; Keinan, E.; Livneh, Z.; Shapiro, E. (2001). "Programmable and autonomous computing machine made of biomolecules". Nature414 (6862): 430–434. doi:10.1038/35106533.

[45]. Benenson,Y., Gil,B., Ben-Dor,U., Adar,R. and Shapiro,E. (2004) An autonomous molecular computer for logical control of gene expression. Nature, 429, 423–429.

[46]. Stojanovic,M.N. and Stefanovic,D. (2003) A deoxyribozyme-based molecular automaton. Nat. Biotechnol., 21, 1069–1074.

[47]. Kristiane A. Schmidt, Christiaan V. Henkel, Grzegorz Rozenberg, Herman P. Spaink. DNA computing using single-molecule hybridization detection. Oxford JournalsNucleic acid research - Life Sciences. Volume 32, Issue 17. Pp. 4962-4968. 2004

[48]. Braich,R.S., Chelyapov,N., Johnson,C., Rothemund,P.W.K. and Adleman,L. (2002) Solution of a 20-variable 3-SAT problem on a DNA computer. Science, 296, 499–502.

[49]. Tuma,R.S., Beaudet,M.P., Jin,X., Jones,L.J., Cheung,C.Y., Yue,S. and Singer,V.L. (1999) Characterization of SYBR gold nucleic acid gel stain: a dye optimized for use with 300-nm ultraviolet transilluminators. Anal. Biochem., 268, 278–288.

[50]. Rozenberg,G. and Spaink,H. (2003) DNA computing by blocking. Theory. Comp. Sci., 292, 653–665.

[51]. Liu Q, Guo Z, Condon AE, Korn RM, Lagally MG, et al. (2000) A surface based approach to DNA computation. In Proceedings of the Second Annual Meeting on DNA Based Computers, 206-216.

[52]. Sakamoto,K., Gouzu,H., Komiya,K., Kiga,D., Yokoyama,S., Yokomori,T. and Hagiya,M. (2000) Molecular computation by DNA hairpin formation. Science, 288, 1223–1226.

[53]. Tom Abate, 2013. Stanford Scientists Use DNA to Assemble a Transistor From Graphene.Stanford Engineering, tabate@stanford.edu, 650-736-2245.