Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2016, 3(2):116-118**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

DNA Computing Made Simple

Matthew NO Sadiku, Adebowale E Shadare, Sarhan M Musa

Roy G. Perry College of Engineering

Abstract DNA computing is essential computation using biological molecules rather than traditional silicon chips. In recent years, DNA computing has been a research tool for solving complex problems. Despite this, it is still not easy to understand. The aim of this paper is present DNA computing in simple terms that a beginner can understand.

Keywords DNA computing, molecular computing, software of life, DNA codes

Introduction

Development in traditional electronic computers is restricted by hardware problems. DNA computing will solve that problem and serve as an alternative technology. DNA computing is also known as molecular computing. It is computing using the processing power of molecular information instead the conventional digital components. It is one of the non-silicon based computing approaches. DNA has been shown to have massive processing capabilities that might allow a DNA-based computer to solve complex problems in a reasonable amount of time. DNA computing was proposed by Leonard Adleman, who demonstrated in 1994 that DNA could be applied in computations [1]. He used DNA to solve a small instance of the traveling salesman problem, in which the objective is to find the most efficient route through seven cities connected by 14 one-way flights. Adleman solved this problem by creating strands of DNA to represent each flight and then combined them to generate every possible route [2, 3]. The graph in Adleman's experiment is shown in Figure1. Adleman's work have set imaginations blazing throughout the world and across disciplines. It introduced a new revolutionary era in the field of computing. DNA computing is now an interdisciplinary research field where chemistry, molecular biology, computer science, mathematics, and technology come together.

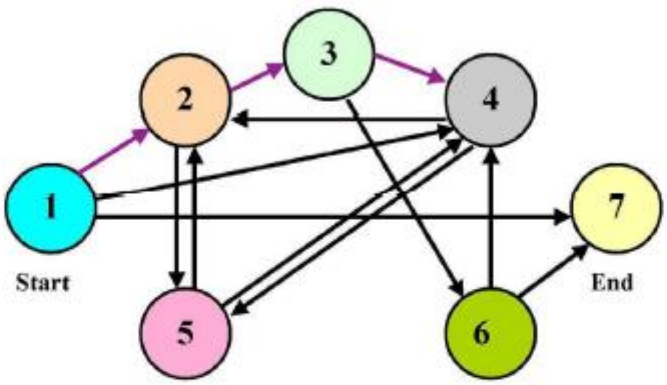

Figure 1: The graph in Adleman's experiment [4].

Computing Model

DNA (Deoxyribo Nucleic Acid) is the molecule that plays an important role in DNA computing. It is found in every living cell. It is used as a medium to store the genetic information of all living beings. It consists of nucleotides which have four different bases: adenine (A), guanine (G), cytosine (C), and thymine (T). Pairs (A,T) and (G,C) are called complimentary. The amount of each nucleotide and the other of their arrangement are unique to every living organism. We define the complement operation as: $\overline{A}=T$, $\overline{T}=A$, $\overline{C}=G$, and $\overline{G}=C$. DNA strands can be regarded as a sequence represented by a combination of four symbols A, G, C, and T. DNA strands are used in encoding the problem, while biological operations are used in simulating the computation [4]. For example, a single-stranded segment consisting of the base sequence TAGCC will stick to a section of another strand made up of the complementary sequence ATCGG. A strand of DNA bears similarity to a Turing machine's tape. DNA is usually double-stranded, consisting of two long strings twisted around each other in a helical form.

DNA computers perform computations by synthesizing DNA strands and allowing them to react in test tubes. DNA can be [4]:

- Synthesized – desired strands can be created
- Separated – strands can be sorted and separated by length
- Merged – pour two test tubes of DNA into one to form union
- Extracted – extract strands that contain a given pattern
- Melted/Annealed – breaking/bonding two DNA molecules with complementary sequences
- Amplified – make copies of DNA strands
- Cut—cut DNA restriction enzymes
- Rejoined – rejoin DNA strands with "sticky ends"
- Detected – confirm presence or absence of DNA

DNA computing model is described in Figure 2. The input consists of DNA fragments and some enzymes. (Enzymes are proteins that accomplish specific functions in the cell.) The output consists of DNA fragments through controllable biochemical reactions [6]. In a DNA computer both the input and output are strands of DNA. DNA computing is a form of parallel computing in that it takes advantage of many different molecules of DNA and tries many different possibilities at once. In order to accomplish DNA computing, it is necessary to have DNA libraries, which are also known as DNA codes.

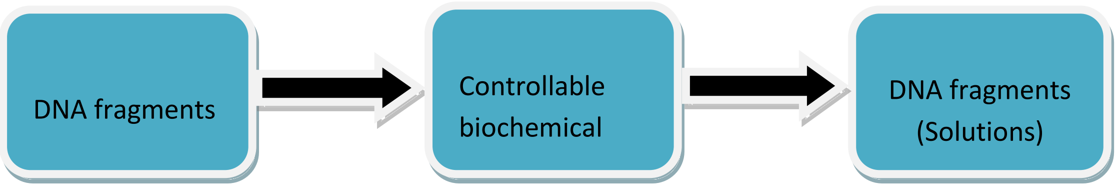

Figure 2: DNA computing model [6].

Limitations

Originally, the objective of DNA computing, as envisioned by Adleman and others, was to solve numerical problems [7]. DNA computing has the potential of performing calculations many times faster than most current digital computers. A current limitation is the use of natural enzymes, which act on certain sequences. Right now, the DNA computer can only perform rudimentary operations. Generating solutions to simple problems may require impractically large amounts of memory. In order to apply DNA computing to a wide range of problems, some procedures for performing primitive operations, such as logic or arithmetic operations, are necessary. Research on DNA computing is still in the proof-of-principle stage. Any practical application is at least some years away.

Because of the limitations of DNA computing, it will not compete with silicon-based technology.

Conclusions

DNA computing was proposed as a way of solving a class of computational problems in which the computation time can grow exponentially with problem size [8]. It should not be viewed as competing with the digital computing. It should be regarded as a platform for new applications. Researchers are still working hard to take advantage of the awesome number-crunching capability of DNA. None of the application is compelling enough to justify the construction of DNA computers. It is not expedient to attempt to predict the future of a new idea like DNA computing.

References

1. L. M. Adleman, “Molecular computation of solutions of combinatorial problems,” *Science*, vol. 266, 1994, pp. 1021-1023.
2. L. M. Adleman, “Computing with DNA,” *Science*, vol. 279, August 1998, pp. 54-61.
3. A. Regalado, “DNA computing,” *Technology Review*, May/June 2000, pp. 80-84.
4. G. Steele and V. Stojkovic, “Agent-oriented approach to DNA computing,” *Proc. Of Computational Systems Bioinformatics Conference*, 2004, pp. 546-551.
5. A. Singh and M. Kaur, “DNA computing approach for automated test pattern generation for digital circuits,” *International Journal of Systems Science*, vol. 39, no. 2, Feb. 2008, pp. 173-180.
6. Y. Huang and L. He, “DNA computing research progress and application,” *The 6th International Conference on Computer Science & Education*, 2011, pp. 232-235
7. D.I. Lewin, “DNA computing,” *Computing in Science and Engineering*, May/June 2002, pp. 5-8.
8. Q. Liu et al., “DNA computing on surfaces,” *Nature*, vol. 403, Jan. 2000, pp. 175-179.

About the authors

Matthew N.O. Sadiku is a professor at Prairie View A&M University, Texas. He is the author of several books and papers. He is a fellow of IEEE.

Adebowale Shadare is a doctoral student at Prairie View A&M University, Texas. He is the author of several papers.

Sarhan M. Musa is an associate professor in the Department of Engineering Technology at Prairie View A&M University, Texas. He has been the director of Prairie View Networking Academy, Texas, since 2004. He is an LTD Spring and Boeing Welliver Fellow.