RESEARCH ARTICLE   OPEN ACCESS

DNA Computer; Present and Future

Amir Abbaszadeh Sori
Department of Computer Engineering and Information Technology, Amir Kabir University of Technology, Tehran, Iran

Abstract
DNA computers use strands of DNA to perform computing operations. The computer consists of two types of strands – the instruction strands and the input data strands. The instruction strands splice together the input data strands to generate the desired output data strand. DNA computing holds out the promise of important and significant connections between computers and living systems, as well as promising massively parallel computations. Before these promises are fulfilled, however, important challenges related to errors and practicality has to be addressed. On the other hand, new directions toward a synthesis of molecular evolution and DNA computing might circumvent the problems that have hindered development, so far. This paper represent present and future DNA computer.

Keywords: DNA Computer, DNA Computing, Parallel Computation, Bio-Molecular.

I. Introduction

DNA (Deoxyribose Nucleic Acid) computing, also known as molecular computing is a new approach to massively parallel computation based on groundbreaking work by Adleman.

DNA computing was proposed as a means of solving a class of intractable computational problems in which the computing time can grow exponentially with problem size (the 'NP-complete' or non-deterministic polynomial time complete problems). A DNA computer is basically a collection of specially selected DNA strands whose combinations will result in the solution to some problem, depending on the problem at hand.

Technology is currently available both to select the initial strands and to filter the final solution. DNA computing is a new computational paradigm that employs (bio) molecular manipulation to solve computational problems, at the same time exploring natural processes as computational models [1, 2].

In 1994, Leonard Adleman at the Laboratory of Molecular Science, Department of Computer Science, University of Southern California surprised the scientific community by using the tools of molecular biology to solve a different computational problem.

The main idea was the encoding of data in DNA strands and the use of tools from molecular biology to execute computational operations. Besides the novelty of this approach, molecular computing has the potential to outperform electronic computers.

For example, DNA computations may use a billion times less energy than an electronic computer while storing data in a trillion times less space. Moreover, computing with DNA is highly parallel: In principle there could be billions upon trillions of DNA molecules undergoing chemical reactions, that is, performing computations, simultaneously.

Paper contents:
• History and Motivation
• DNA Fundamentals
• Principles of DNA Computing
• Present and Future of DNA Computer
• Advantages and Disadvantages
• Conclusions

II. History and Motivation

"Computers in the future may weigh no more than 1.5 tons." So said Popular Mechanics in 1949. Most of us today, in the age of smart cards and wearable PCs would find that statement laughable. We have made huge advances in miniaturization since the days of room-sized computers, yet the underlying computational framework has remained the same.

Today's supercomputers still employ the kind of sequential logic used by the mechanical dinosaurs of the 1930s. Some researchers are now looking beyond these boundaries and are investigating entirely new media and computational models. These include quantum, optical and DNA-based computers. It is the last of these developments that this paper concentrates on.

The current Silicon technology has following limitations:
• Circuit integration dimensions
• Clock frequency
• Power consumption
• Heat dissipation.

The problem's complexity that can be afforded by modern processors grows up, but great challenges

require computational capabilities that neither most powerful and distributed systems could reach [3].

The idea that living cells and molecular complexes can be viewed as potential mechanic components dates back to the late 1950s, when Richard Feynman delivered his famous paper describing "sub-microscopic" computers. More recently, several people have advocated the realization of massively parallel computation using the techniques and chemistry of molecular biology. DNA computing was grounded in reality at the end of 1994, when Leonard Adleman, announced that he had solved a small instance of a computationally intractable problem using a small vial of DNA.

By representing information as sequences of bases in DNA molecules, Adleman showed how to use existing DNA-manipulation techniques to implement a simple, massively parallel random search. He solved the traveling salesman problem also known as the "Hamiltonian path" problem.

There are two reasons for using molecular biology to solve computational problems.

(i)   The information density of DNA is much greater than that of silicon : 1 bit can be stored in approximately one cubic nanometer. Others storage media, such as videotapes, can store 1 bit in 10^{12} cubic nanometer.

(ii) Operations on DNA are massively parallel: a test tube of DNA can contain trillions of strands. Each operation on a test tube of DNA is carried out on all strands in the tube in parallel [4].

III. DNA Fundamentals

DNA (deoxyribonucleic acid) is a double stranded sequence of four nucleotides; the four nucleotides that compose a strand of DNA are as follows: adenine (A), guanine (G), cytosine (C), and thymine (T); they are often called bases. DNA supports two key functions for life:

- coding for the production of proteins,
- self-replication.

Each deoxyribonucleotide consists of three components:

- a sugar — deoxyribose
    - five carbon atoms: 1´ to 5´
    - hydroxyl group (OH) attached to 3´ carbon
- a phosphate group
- nitrogenous base.

The chemical structure of DNA consists of a particular bond of two linear sequences of bases. This bond follows a property of Complementarity: adenine bonds with thymine (A-T) and vice versa (T-A), cytosine bonds with guanine (C-G) and vice versa (G-C). This is known as Watson-Crick complementarity. In Figure 1 can see structure of double-stranded DNA.
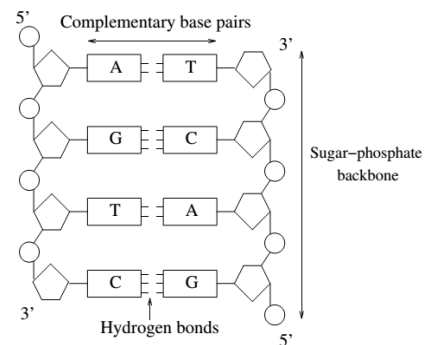

Figure 1: Detailed structure of double-stranded DNA

The four nucleotides adenine (A), guanine (G), cytosine (C), and thymine (T) compose a strand of DNA. Each DNA strand has two different ends that determine its polarity: the 3'end, and the 5'end. The double helix is an anti-parallel (two strands of opposite polarity) bonding of two complementary strands [3, 4].

IV. Principles of DNA Computing

The DNA is the major information storage molecule in living cells, and billions of years of evolution have tested and refined both this wonderful informational molecule and highly specific enzymes that can either duplicate the information in DNA molecules or transmit this information to other DNA molecules. Instead of using electrical impulses to represent bits of information, the DNA computer uses the chemical properties of these molecules by examining the patterns of combination or growth of the molecules or strings.

DNA can do this through the manufacture of enzymes, which are biological catalysts that could be called the 'software', used to execute the desired calculation [5].

A single strand of DNA is similar to a string consisting of a combination of four different symbols A G C T. Mathematically this means we have at our disposal a letter alphabet, Σ = {A GC T} to encode information which is more than enough considering that an electronic computer needs only two digits and for the same purpose. In a DNA computer, computation takes place in test tubes or on a glass slide coated in 24K gold. The input and output are both strands of DNA, whose genetic sequences encode certain information. A program on a DNA computer is executed as a series of biochemical operations, which have the effect of synthesizing, extracting, modifying and cloning the DNA strands.

As concerning the operations that can be performed on DNA strands the proposed models of DNA computation are based on various combinations of the following primitive bio-operations:

- **Synthesizing** a desired polynomial-length strand used in all models.
- **Mixing:** combine the contents of two test tubes into a third one to achieve union.

- **Annealing:** bond together two single-stranded complementary DNA sequences by cooling the solution. Annealing in vitro is known as hybridization (Figure 2).

- **Melting:** break apart a double-stranded DNA into its single-stranded complementary components by heating the solution. Melting in vitro is also known under the name of denaturation (Figure 2).
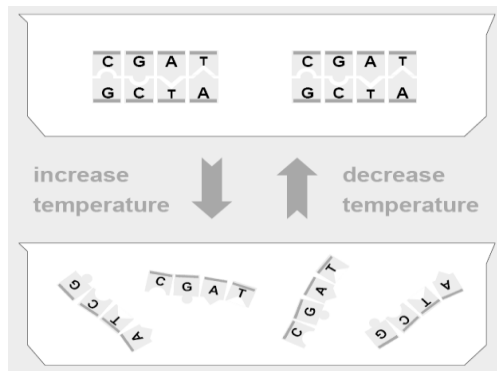

Figure 2: DNA Melting and Annealing

- **Amplifying (copying):** make copies of DNA strands by using the Polymerase Chain Reaction PCR. The DNA polymerase enzymes perform several functions including replication of DNA. The replication reaction requires a guiding DNA single-strand called template, and a shorter oligonucleotide called a primer, that is annealed to it (Figure 3).
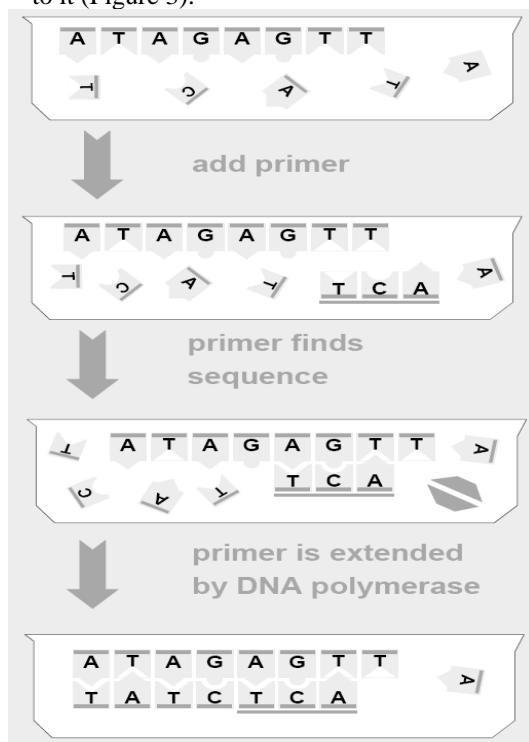

Figure 3: DNA Replication

- **Separating** the strands by length using a technique called gel electrophoresis that makes possible the separation of strands by length.

- **Extracting** those strands that contain a given pattern as a substring by using affinity purification.

- **Cutting** DNA double-strands at specific sites by using commercially available restriction enzymes. One class of enzymes, called restriction endonucleases, will recognize a specific short sequence of DNA, known as a restriction site. Any double-stranded DNA that contains the restriction site within its sequence is cut by the enzyme at that location.

- **Ligating:** paste DNA strands with compatible sticky ends by using DNA ligases. Indeed, another enzyme called DNA ligase, will bond together, or ``ligate'', the end of a DNA strand to another strand.

- **Substituting:** substitute, insert or delete DNA sequences by using PCR site-specific oligonucleotide mutagenesis.

- **Marking** single strands by hybridization: complementary sequences are attached to the strands, making them double-stranded. The reverse operation is unmarking of the double-strands by denaturing, that is, by detaching the complementary strands. The marked sequences will be double-stranded while the unmarked ones will be single-stranded.

- **Destroying** the marked strands by using exonucleases, or by cutting all the marked strands with a restriction enzyme and removing all the intact strands by gel electrophoresis. (By using enzymes called exonucleases, either double-stranded or single-stranded DNA molecules may be selectively destroyed. The exonucleases chew up DNA molecules from the end inward, and exist with specificity to either single-stranded or double-stranded form.)

- **Detecting and Reading:** given the contents of a tube, say ``yes'' if it contains at least one DNA strand, and ``no'' otherwise. PCR may be used to amplify the result and then a process called sequencing is used to actually read the solution.

In Short, DNA computers work by encoding the problem to be solved in the language of DNA: the base-four values A, T, C and G. Using this base four number system, the solution to any conceivable problem can be encoded along a DNA strand like in a Turing machine tape. Every possible sequence can be chemically created in a test tube on trillions of different DNA strands, and the correct sequences can be filtered out using genetic engineering tools. Table 1 is example of DNA binary code [5, 6].

Table 1: An example of a DNA binary code

Strand	Sequence	wxyz
S_0	CAACCCAA	0000
S_1	TCTCAGAG	0001
S_2	GAAGGCAT	0010
S_3	AGGAATGC	0011
S_4	ATCGAGCT	0100
S_5	TTGGACCA	0101
S_6	ACCATTGG	0110
S_7	GTTGGGTT	0111
S_8	CCAAGTTG	1000
S_9	CAGTTGAC	1001
S_{10}	TGGTTTGG	1010
S_{11}	GATCCGAT	1011
S_{12}	ATATCGCG	1100
S_{13}	GGTTCAAC	1101
S_{14}	AACCTGGT	1110
S_{15}	ACTGGTCA	1111

V. Present and Future of DNA Computer

A few years ago, researchers from the Weizmann Institute of Science in Rehovot, Israel, unveiled a programmable molecular computing machine composed of enzymes and DNA molecules instead of silicon microchips. "This re-designed device uses its DNA input as its source of fuel," said Ehud Shapiro, who led the Israeli research team. This computer can perform 330 trillion operations per second, more than 100,000 times the speed of the fastest PC [4, 6].

While a desktop PC is designed to perform one calculation very fast, DNA strands produce billions of potential answers simultaneously. This makes the DNA computer suitable for solving "fuzzy logic" problems that have many possible solutions rather than the either/or logic of binary computers. In the future, some speculate, there may be hybrid machines that use traditional silicon for normal processing tasks but have DNA co-processors that can take over specific tasks they would be more suitable for. Figure 3 is Olympus Optical Company's DNA computer for gene analysis

Figure 3: Japanese DNA computer for gene analysis

The question is: are the problems facing practical, controllable, and reliable DNA computing insurmountable? There are indications that the design of oligonucleotides and molecular biology protocols for error-tolerant computing is as difficult as the original combinatorial problems for which the technique first showed such promise. If this turns out to be the case, then, success may depend on compilation of oligonucleotide collections and protocols which enable successful applications. Current protocols were not developed for computational applications. It may be that with current technology and understanding, a precise enough characterization of DNA chemistry and physics is not possible for building DNA computers. On the other hand, continued exploration of current protocols and techniques may eventually result in a new body of methods that are specifically adapted to computing with DNA. Another problem is the lack of a "killer app". Part of this is certainly associated with the problems that DNA computing has experienced, and the hope is that once the problems are resolved, the applications will come. It has already been suggested that the "killer app" might be the application of DNA computing to problems in molecular biology [7, 8].

Advantages
- Perform millions of operations simultaneously.
- Generate a complete set of potential solutions and conduct large parallel searches.
- Efficiently handle massive amounts of working memory.
- They are inexpensive to build, being made of common biological materials.
- The clear advantage is that we have a distinct memory block that encodes bits.
- Using one template strand as a memory block also allows us to use its compliment. As another memory block, thus effectively doubling our capacity to store information.

Disadvantages
- Generating solution sets, even for some relatively simple problems, may require impractically large amounts of memory (lots and lots of DNA strands are required)
- 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 (for every correct answer there are millions of incorrect paths generated that are worthless).
- DNA computers could not (at this point) replace traditional computers. They are not programmable and the average dunce cannot sit down at a familiar keyboard and get to work [6].

VI. Conclusion

The field of DNA computing and DNA Computer remains alive and promising, even as new challenges emerge. Most important among these are the uncertainty, because of the DNA chemistry, in the Computational results, and the exponential increase in number of DNA molecules necessary to solve problems of interesting size. Despite these issues, de finite progress has been made both in quantifying errors, and in development of new protocols for more efficient and error-tolerant DNA computing. In addition, new paradigms based on molecular evolution have emerged from molecular biology to inspire new directions in DNA computing. As has been the case in the recent development of "hot" fields, only further work will allow the determination of the proper scope and niche of DNA computing.

References
[1] Leonard M. Adleman, *Computing with DNA*, Scientific American, August 1998.
[2] Paun, G., Rozenberg, G., and Salomaa, A., Edited, 2005. *DNA Computing-New Computing Paradigms*, Springer-Verlag Berlin Heide1berg.
[3] Srivastava, A., Pandey, V., "*Inconsistency in DNA computing and it's use in cryptography*", Advance IEEE 3rd International Computing Conference (IACC), 2013.
[4] Sarker, A., Ahmed, T., Rashid, S.M.M., Anwar, S. ; Jaman, L., Tara, N., Alam, M.M., Babu, H.M.H., *"Realization of Reversible Logic in DNA Computing"*, IEEE 11th International Conference on Bioinformatics and Bioengineering (BIBE), 2011.
[5] Xue, B., Xiyu L., "*Design of E-commerce development roadmap for SMEs using DNA computing technique*", IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2013.
[6] Martyn Amos, editor. "*Cellular Computing. Series in Systems Biology*". Oxford University Press, USA, 2004.
[7] Ravinderjit S. Braich, Nickolas Chelyapov, Cliff Johnson, Paul W.K. Rothemund, and Leonard Adleman. "*Solution of a 20-variable 3-sat problem on a DNA computer*". Science, 296:499–502, 2002. Marinesco, S., Dale, N., 2013. *Microelectrode Biosensors*, Springer press.