DNA COMPUTING

H.J. Hoogeboom and G. Rozenberg
Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands

Keywords: DNA computing, miniaturisation, Adleman, Hamiltonian path problem, parallelism

Contents

Summary

Present computer technology may reach its limits. Intrinsically difficult computational problems require new approaches to solve them. In DNA computing one uses DNA strands to code data that is manipulated using bio-chemical techniques. In this way massive parrallelism may be achieved in order to avoid the classical bottleneck of exponentially growing complex problems. This is illustrated by Adleman's solution for the Hamiltonian Path Problem in graphs.

1. Introduction

The development of computer science, that has revolutionised almost all walks of life, has progressed at an unprecedented rate over the past 50 years. However it has become apparent that there are several serious obstacles that may prevent the continuation of this development. On the one hand we have understood the nature of intrinsically difficult problems that require parallel processing on a scale considered impossible by the standards of the current technology, while on the other hand the continuous progress of miniaturisation using silicon technology may be approaching its limits.

As a response to this situation, we have witnessed in recent years tremendous development of *Natural Computing*, i.e., computing that is inspired (gleaned from) nature. *DNA computing* is one of the areas of natural computing. Here, the computing paradigms come from molecular biology, and moreover the hardware for implementing

the algorithms based on these paradigms consists of biomolecules such as DNA and enzymes—such hardware is often referred to as bioware. Since DNA computing develops devices that are nano scale in size, the miniaturisation problem disappears. Since performing a bio-operation on a solution containing trillions of DNA molecules, means that each of the DNA molecules is processed, DNA computing is attractive from the parallel processing point of view.

This paper introduces the area of DNA computing in a tutorial fashion—no previous knowledge of molecular biology or computer science is assumed on the part of the reader.

2. DNA molecules and their processing

We discuss in this section the structure of DNA molecules, notation for them, and the basic biochemical "tool box" for processing DNA molecules. Our presentation of these topics is very simplified—however, it is still adequate for the purpose of this paper.

2.1 Structure of DNA molecules

DNA (deoxyribonucleic acid) molecules are polymers that are built from "simple" monomers called nucleotides. A nucleotide consists of three components: sugar, phosphate, and a base, schematically represented in Figure 1. Here the stick represents the sugar molecule that has five carbon atoms (attachment points) labelled 1′ through 5′. The base is attached to the sugar at 1′, and phosphate is attached at 5′.

Figure 1: Nucleotide with base

Single nucleotides can be strung together to form single stranded DNA molecules. This happens when a strong (covalent) bond is formed through the phosphate between the 3′-attachment point of one nucleotide and the 5′-attachment point of another, as illustrated in Figure 2. This bond is called the *phosphodiester bond*.

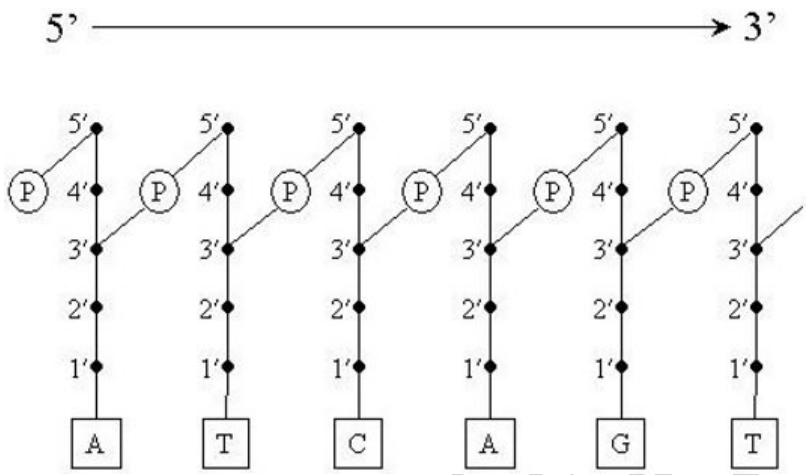

Figure 2: Single stranded DNA molecule

Note that the two nucleotides at the ends of a single stranded DNA molecule are available for further extension: a 5′-attachment point is available, through phosphate, at one end, and a 3′-attachment point at the other end. Since the chemical properties of these two ends are different, one can distinguish between these two ends of a molecule—we refer to them as the 5′-end and the 3′-end, respectively. Hence, a single stranded DNA molecule has polarity: we can read it from the 5′-end to the 3′-end, or we can read it from the 3′-end to the 5′-end. It turns out that biological information is encoded in the 5′–3′ direction.

Nucleotides may differ only by their bases, and there are four possible bases called A, C, G, and T: abbreviations for Adenine, Cytosine, Guanine, and Thymine, respectively. The most important feature of this set {A, C, G, T} of bases is that its elements have pair-wise affinity: A with T, and C with G. This means that both A and T, and C and G like to stick together. This affinity is referred to as the *Watson-Crick complementarity*, and it is central to the formation of double stranded DNA molecules.

Two single stranded molecules can bind to each other to form a double stranded DNA molecule, as illustrated in Figure 3. The single strands bind together, element-wise, through their bases. The bonds here are weak bonds (called *hydrogen bonds*), that can be compared to weak magnets. However, many weak bonds form a bond strong enough to keep the two strands together. The prerequisite for the forming of hydrogen bonds between two bases is that they are complementary. This means that one of them is A and the other T, or one of them is C and the other G. The other requirement for two single stranded molecules to form a double stranded molecule is that the two involved strands are of opposite orientation. This means that the first nucleotide from the 3′-end of one single stranded DNA molecule binds to the first nucleotide from the 5′-end of the

other molecule; then the second nucleotide from the 3′-end binds to the second nucleotide from the 5′-end, etc.

We stress once again that our presentation of DNA molecules in this paper is very simplified, but precise enough for the purpose of this paper. Thus, e.g., the double stranded molecule in Figure 3 is presented in linear form, while in nature it forms the famous double helix.

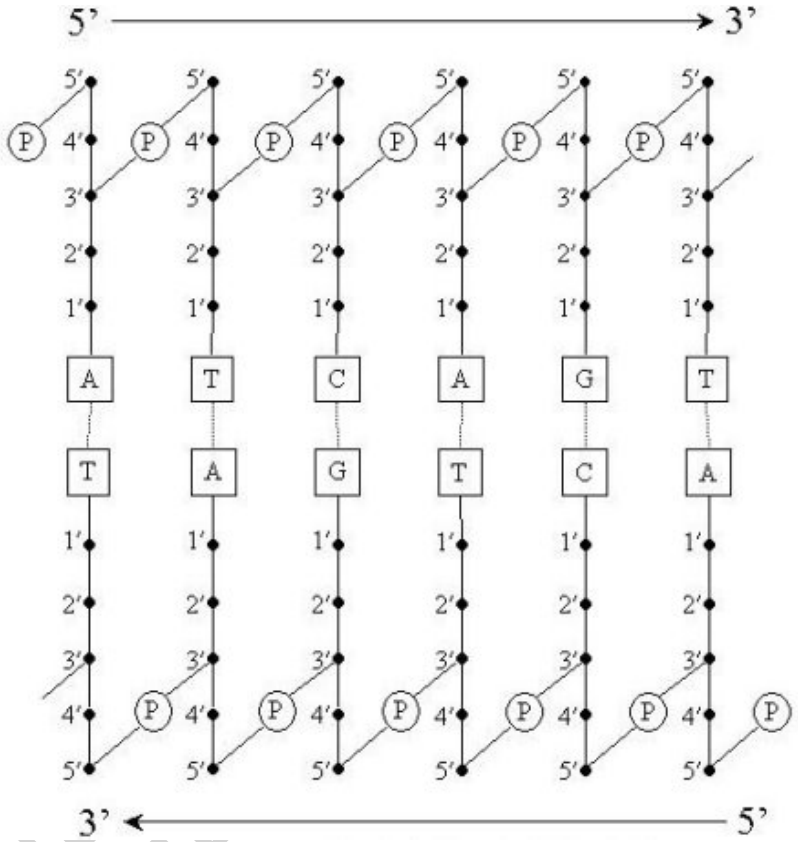

Figure 3: Double stranded DNA molecule

-
-
-

Bibliography

A. Condon, G. Rozenberg. DNA Based Computers VI, Lecture Notes in Computer Science, Springer

Verlag, 2001, to appear. [This conference is a good source for following the progress in DNA computing]

E. Winfree, D. K. Gifford. DNA Based Computers V, proceedings DIMACS Workshop, June 14-15, 1999. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 54, 2000. [This conference is a good source for following the progress in DNA computing]

G. Păun, G. Rozenberg, A. Salomaa. DNA Computing, new computing paradigms. Springer Verlag, Berlin, 1998. [Part 1 is a well accessible introductions to DNA computing]

L. Kari. DNA computing: arrival of biological mathematics. The Mathematical Intelligencer, 19, 1997, 9—22. See http://www.csd.uwo.ca/~lila/intel1.ps.[Well accessible introduction to DNA computing]
L.M. Adleman. Molecular computation of solutions to combinatorial problems. Science, 226 (November 1994), 1021-1024.

Biographical Sketches

G. Rozenberg received his Master and Engineer degree in computer science in 1965 from the Technical University of Warsaw, Poland. In 1968 he obtained his Ph.D. in mathematics at the Polish Academy of Sciences, Warsaw. He is currently a full professor at Leiden Institute of Advanced Computer Science (LIACS), where he is the head of the Theory Group.He is also adjoint professor at the Department of Computer Science of the University of Colorado at Boulder, U.S.A. He is also the scientific director of Leiden Center for Natural Computing (LCNC).

G. Rozenberg has published about 350 papers, 5 books, and is a (co-)editor of about 50 books.

He is the editor of the Bulletin of the EATCS, the editor of the seriesAdvances in Petri Nets (Springer-Verlag), the managing editor of the Series on Natural Computing (Springer-Verlag), and a co-editor of the Monographs and Texts in Theoretical Computer Science (Springer-Verlag). He is an editor-in-chief of the International Journal on Natural Computing (Kluwer Academic Publishers).

G. Rozenberg was the President of the European Association for Theoretical Computer Science (EATCS), in the period 1985-1994, and he was the chairman of the Award Committee for the Goedel Prize 1997.

His current functions include:

- Chairman of the Steering Committee for International Conferences on Theory and Applications of Petri Nets (ICATPN),
- Chairman of the Steering Committee for DNA Based Computers Conferences (DNA),
- Director of the European Molecular Computing Consortium (EMCC).
- Scientific director of the European project "Molecular Computing"
- (MolCoNet) funded by the European Community.

He has been a member of the program committees for practically all major conferences on theoretical computer science in Europe.

G. Rozenberg is involved in a number of externally funded research projects on both national and international levels. His current research interests include:

- DNA computing,
- theory of concurrent systems, in particular theory of Petri nets,
- theory of graph transformations,
- formal language and automata theory,
- Computer Supported Cooperative Work.

He is a Foreign Member of the Finnish Academy of Sciences and Letters, a member of Academia

Europaea, and he is holder of Honorary Doctorates of the University of Turku, Finland, and the Technical University of Berlin,Germany.

G. Rozenberg is a performing magician.

Hendrik Jan Hoogeboom is assistant professor in theoretical computer science at Leiden University. He has published papers on formal languages, formal models for concurrency and DNA computing. His research in DNA Computing includes splicing systems, sticker systems, forbidding-enforcing systems, and formal models for self-assembly.