Molecular Inference Network Experimental Approximation

Piotr Wąsiewicz[1], Andrzej Płucienniczak[2]

[1] Warsaw University of Technology, Institute of Electronic Systems, Warsaw, Poland,
email: pwasiewi@elka.pw.edu.pl
[2] Institute of Biotechnology and Antibiotics, Warsaw, Poland, apl@iba.waw.pl

Abstract. Implementation of the inference system based on DNA chains molecular computing is a new paradigm to perform calculations using nanotechnology means. This work presents a new approach to the implementation of inference engines based on DNA. It introduces the subject of inference methods designed to be used with molecular expert systems. The main part of this work includes the concept of the inference engine based on a rule tree specially customized to allow implementation using deoxyribonucleic acid chains. The approach presented allows the drawing of inferences based on a variable number of predicates using the most reliable techniques employed in standard operations of genetic engineering. In this approach cross cells are bases of multidimensional DNA structures. An experiment was conducted to confirm the capabilities of the implementation suggested. In addition, laboratory evaluation results and perspectives for the further use of the proposed architectural approach are discussed.

1 Introduction

DNA computing in vitro is the main branch of molecular computing. Adleman first showed in [1] that DNA can be used to solve computational problems. In his initial experiment the tools of molecular biology were utilised to solve an instance of the directed Hamiltonian path problem. A new research field had been entered. Research and discussions have been started as to whether computers based on molecular interactions may be a viable alternative to computers based on silicon electronics. Adleman [2] considered practical aspects of constructing a molecular computer and he concluded [3] that the manipulation of molecules to solve mathematical problems had redefined what was meant by "computation". Molecules e.g. DNA oligonucleotides carry information, and chemical reactions are like computing processes. Sequences of such processes are called DNA computing algorithms. DNA Computing research scientists focus on implementing algorithms solving NP-complete problems (nondeterministic-polynomial-time), which are fundamental tasks for any future molecular supercomputing.

Other researchers used DNA to solve different computational tasks e.g. [4, 6–10, 13–28]. Those described in section 2 were chosen from molecular computing bibliography [5] and present experiments, which make use of the DNA capability for performing massively parallel computation in logic and arithmetic systems.

2 Arithmetic, logic, and based on knowledge systems

Arithmetic or arithmetics in common usage is a branch of mathematics which records elementary properties of certain operations on numerals. The traditional arithmetic operations are addition, subtraction, multiplication and division.

In 1854, British mathematician George Boole published a paper detailing a system of logic that would become known as Boolean algebra. His logical system is the basis in expert, inference, logic systems theory and in the development of the binary systems, particularly its implementation in electronic or future molecular circuitry.

Knowledge representation plays an important role in inference systems [12], where the knowledge should be formalized and structured. One of the methods that can support knowledge structuring is known as production rules. Such rules are also referred to as IF-THEN rules. There may be several premise statements within a single rule. All premises have to be included (equal to true or 1) to prove their conclusion. In our approach the premises and conclusions will be represented by DNA molecules, which are described in the next points.

3 Molecular arithmetic and logic system implementations

There have been many attempts to create boolean circuits or logic gates using in vitro molecular computing methods. Ogihara and Ray [14] already in 1996 demonstrated that DNA computers can simulate Boolean circuits with a small overhead. Amos and Dunne [4] described the abstract model and its own laboratory implementation. Hagiya et al [10] designed one molecule DNA computer with data and operations on one DNA strand. Computation of logic function satisfiability was driven by PCR reaction. Wąsiewicz [23, 28] also proposed the evolutionary programming of logic function graphs, the evaluation of which is based on PCR.

Some researchers [6, 8, 15, 22] proved in their experiments that molecular computation can perform arithmetical or logical operations. Braich, Johnson, Rothemund, Hwang, Chelyapov and Adleman [6] used a gel-based DNA computer to solve a satisfability problem, while Oliver [15] used a DNA-Matrix Multiplication method to calculate the product of Boolean matrices. Very interesting results were obtained by Wąsiewicz, Rudnicki, Mulawka and Lesyng [22]. They presented a new algorithm of DNA computing for adding binary integer numbers. It requires the unique representation of bits placed in test tubes treated as registers. The amplification step used for the carry operation in theory allows the addition of numbers for the same quantity of elementary operations, regardless of the number of bits used for representation.

Interesting results were also obtained by Gupta, Parthasarathy, Zaki [9] who reported a method for using DNA molecules to solve the basic arithmetic and logic operations which can be performed in a single test tube, utilizing the output of an operation as an input for the next.

Surface-based methods were presented by Liu, Smith and their research group. In Liu, Frutos, Thiel, Condon, Corn, Lagally and Smith [17] complex combinatorial mixtures of DNA molecules encoding all possible answers to a computational problem were synthesized and attached to the surface of a solid support.

A very promising field in vitro DNA computation is the creation of two dimensional structures, which will fulfil desired requirements. Works in this area include construction,

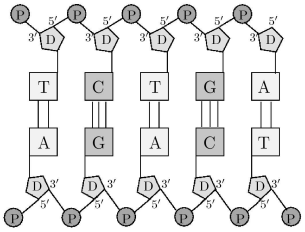

Figure 1. The chemical structure and bonds of DNA double molecule, a joint between single strings

analysis, ligation and self-assembly of DNA triple crossover complexes [7] by (among others) LaBean, Winfree, Reif or the work of Li, Yang, Qi, Seeman [16], where double crossover molecules were presented. Rothemund in his dissertation [18] discussed Tile Model Assembly. This methodology enables for example the construction of binary counters, sierpinski triangles or simulating of Turing Machine on the surface. Sa-Ardyen, Jonoska, Seeman [19] proposed self-assembling DNA graphs by constructing arcs from double strings, which join in the nodes. Tomczuk, Wasiewicz [20] depicted another approach to molecular binary trees.

Wąsiewicz, Mulawka, Dydynski and Tomczuk introduced an original model of molecular neural network based on DNA computing paradigm [24–26]. The essential feature of such an approach emerging from visual inspection of the idea of neural network connectionism is hybridization of pairs of complementary DNA strings and the possibility of representing highly parallel selective operations, which can enable creating alternative, neural architectures based on spacial structures of layer graphs.

The application of molecular computing to problems of expert systems was proposed in [13] and extended in [27]. A biochemical reaction on DNA strands was used to realise the backward chaining algorithm. Horn clause computation by self-assembly of tiles was prepared by Uejima, Hagiya, Kobayashi [21].

4 Inference rule network basis

A single-stranded DNA string has a phospho-sugar backbone with two different, 5' and 3' ends and four bases Adenine, Thymine, Cytosine, Guanine denoted by the symbols A, T, C, and G, respectively. A double-stranded DNA string may be formed of two single strings oriented in opposite directions due to hybridization or annealing reaction, because A is complementary with T, and C is complementary with G as is seen in Fig. 1. Due to this reaction the oligonucleotides may connect with each other during concatenation process called ligation to form longer DNA chains [11]. A sequence of such operations on DNA strings is called an algorithm. Together the genetic operations driven by enzymes, heating and cooling, DNA sequence and a model make computation possible.

A signal is represented by a DNA string called an oligo, an oligonucleotide, a strand, a DNA fragment. Signals can be formed in a line. For example three signals X, Y, Z can be put one by one with the help of black strings (in colour - blue ones) as is seen in Fig. 2 on the left. Even more signals e.g. X, Y, Z, R can join as is depicted in Fig. 3 also on the left. Each added signal requires the next black DNA string. This methodology leads to rule creation. Such a set can be proposed that the R signal depends on X,Y,X premises

and the set can be represented by the schematic symbol as is described in Fig. 2. In the next set the Q conclusion depends on X,Y,Z,R premises and that set can be simplified to the schematic symbol similar to that shown in Fig. 3. This rules can be connected with each other forming inference molecular networks. Schematic representation of such the network is put in Fig 4.

5 Experimental results

During the PCR process the UBI and REP fragments were isolated from plasmids with a help of two kinds of very short primers complementary to their ends. Primers, which were complementary only to the desired single fragments, were added in a much greater quantity than to the second ones. At the conclusion of the PCR process only the single REP and UBI strings were amplified, after all second primers had been built in them. Such a PCR reaction is called linear.

The I and II fragments had to be synthesized. Synthesis machines can generate up to about 50 base long strings. Thus, the I and II fragments were obtained in the form of short DNA strings (Table I) phosphorylated by T4 polinucleotide kinase except for the 5' ends of I fragments. Phosporylation was stopped by heating to 75 ^oC. These short fragments were connected together by ligase concatenation and not phoshorylated I fragments secured the process against unwanted string ligations. After adding ligase enzyme and ATP, one nigth operation execution, stopped by heating and linear PCR, two 186 pMol solutions of I and II single fragments were prepared: $I = A+B+C+D+E$ was hybridized and concatenated with the help of complementary strings: AB, BC, CD, DE; $II = F + B' + D' + G$ was hybridized and concatenated with the help of complementary strings $FB', B'D', D'G$.

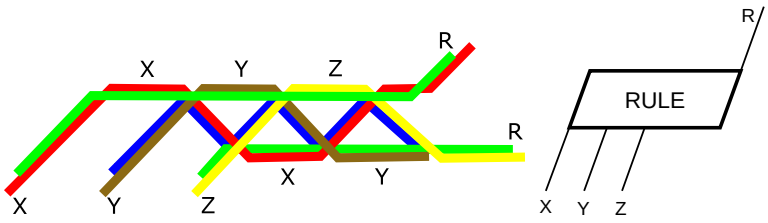

Figure 2. The inference rule with 3 premises

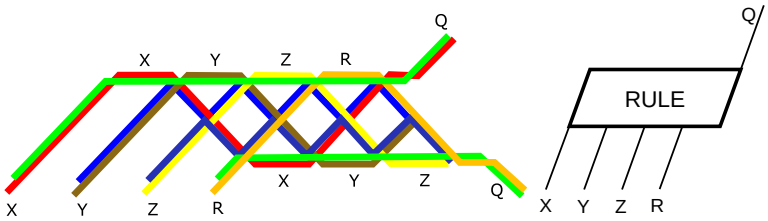

Figure 3. The inference rule with 4 premises

All necessary single strings were kept in special silicon-covered laboratory tubes (single

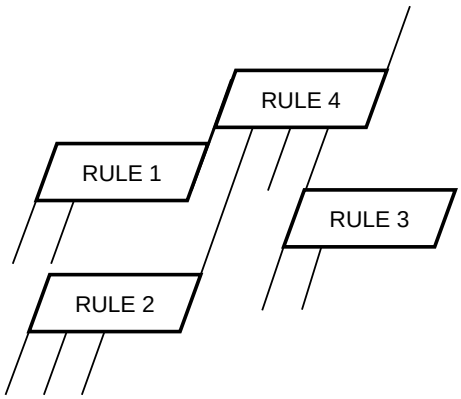

Figure 4. Inference rule network

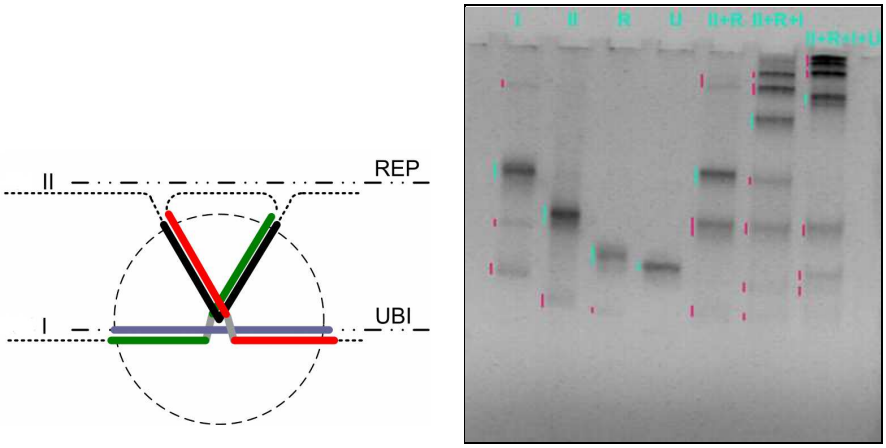

Figure 5. The inference cross cell on the left. DNA electrophoretogram of the hybridization experiment on the right (R=REP, U=UBI).

DNA strings attach themselves to standard tubes) ready to build by self-assembly the cross cell used in our inference systems. Samples of single strand solutions were put for test in the first four lanes of an experiment electrophoretogram as is depicted in Fig. 5 (on the right). All oligos were hybridized in typical B buffer ($100mM$ NaCl, $5mM$ MgCl_2, $1mM$ 2-Mercaptoeturyne, 10mM Tris HCl, $pH = 0,8$) in the following order: after adding 10 μl of REP and 10 μl of II oligonucleotides, the reaction mixture was incubated for one hour and 4 μl sample of this solution was removed and put in the fifth lane as is seen in Fig. 5; after adding 8 μl of the next I oligonucleotide to the mixture and one hour incubation the next 6 μl sample was placed in the sixth lane; and finally after adding 6 μl of the UBI oligo and one hour incubation, the 8 μl sample was put in the seventh lane. All mentioned oligos amounts and lane samples were proportional to the density of all oligos in every corresponding mixture. The final mixture contained 4 μl of each I, II, REP, UBI oligonucleotide.

The results of the hybridization reaction were verified by electrophoresis on the electrophoretogram (single oligos were put in the first four lanes; in the last lane there is a

Table 1. The molecular cross cell oligos with their nucleotide sequences (5'-3')

Oligos	Sequence optimization results
A	tccagatcaacaacgtctaatatttgcaggtaaacagttagaaga
B	acattcaggatccatagaggagattagcac
C	ttgaccagtgttttagtgacaaacgctgcatggtttttcgtagtg
D	agatttcttagtcgacttagagagcacgta
E	gtcgtaccctgtctgattataacattcagaaagaatctaccttac
F	aatttgaaaacatggccatgattcggcgctgaatatcaccgtctt
B'	gtgctaatctcctctatggatcctgaatgt
D'	tacgtgctctctaagtcgactaagaaatct
G	gtatcccttccagacttcatgtggatcgagggcatacgcacaaac
AB	cctgaatgtaaaaatcttctaactgt
BC	aacactggtcaaaaagtgctaatctc
CD	cgactaagaaatctaaacactacgaaaa
DE	ggtacgacaaaaatacgtgctc
FB'	gagattagcacaaaaagacggtga
B'D'	agagagcacgtaaaaacattcaggatcc
D'G	agggatacaaaagatttcttagt
REP	ggtaaggttt gtgcgtatgc cctcgatcca catgaagtct ggaagggata caacactacg aaaaaccatg cagcgtttgt cactaaaaca ctggtcaaca aagacggtga tattcagcgc cgaatcatgg ccatgttttc aaattga
UBI	tctatttctt ccataaggag gtctagttgt tgcagattat aaacgtccat ttgtcaatct tctaccagca tgggacagac taatattgta agtctttctt agatggaatg tagaccagaa tgcagaggcg ccacca

band of the complete inference cross cell, its intermediate states are in the rest of lanes) as is showed in Fig. 5 (on the right). Electrophoresis is the process of distinguishing DNA fragments in gel by length. There is an electric field here. And DNA strings move from the minus to plus side, in this case from top to bottom. Gel net structure makes it more difficult. The lighter the string the further it goes.

In the first lane apart from the main band containing the I fragment there are seen two smaller and one bigger (this top band in the lane) mismatches this means undesirable hybridizations or DNA strings resulting from concatenation and isolation process and marked by dark vertical segments in opposite to correct bands marked with white short lines. In the second lane exists a little lighter mismatch, in the third lane - the smaller one and in the fourth lane there is no additional band. By adding successive oligos to the main mixture new unwanted bands with structure mismatches appear. In the fifth lane the band pattern is similar to that in the first lane, but the I fragment is present only in the next lanes (and in the first one). The absent REP band proves, that all REP strings were hybridized. After adding the I oligo with all its mismatches a new structure mismatches emerges (in the sixth lane) from interference with the previous ones. The first two heaviest macromolecule bands result from the same two mismatches of the previous lane connected with free I and II oligos. The next four incorrect bands emerge from free I and II strings and their mismatches that means they are on the same level of bands from previous lanes. The main cause of these faults may be that the molecular structure is difficult to hybridize and the wrong succession of oligos in the self-assembly process where the longest I string is hybridized almost at the end. In the seventh lane

the mishybridization bands are almost the same, but the correct one consists of at least two alternative spacial forms of the same macromolecule. The double I and UBI string ends move in the electrophoresis gel creating these two structures, which confirms the existence of the predicted structure on the left side of Fig. 5.

6 Summary

The spatial inference system macromolecules described is a new approach to the molecular representation of logic structures. The laboratory implementation of the inference system segment called the cross cell proved that more complicated and larger structures made of spatial inference rules can be self-assembled. Of cource, more careful design is necessary if more solid structures are to be generated without alternative spatial forms. Mishybridizations can be avoided only with the help of more sophisticated nucleotide sequence optimization computer programs and better, more accurate chemical reaction procedures.

Further research should extend ideas and enable propagation of the conclusion sinals between decision rule layers comprised of the proposed inference cross cells, so it can be used in multidimensional molecular constructions. The interference between molecular inference and neural systems in self-assembled macromolecules should be also researched. The neural cells together with cross cells can create macrosegments, which self-assemble the greater structures utilized in molecular computing or in future nanotechnology equipment.

Acknowledgments

I would like to thank Piotr Kiersnowski for his help and work in our group.

Bibliography

[1] L. M. Adleman. Molecular comp. of solutions to combinatorial problems. *Science*, 266, 1994:1021–1024.

[2] L. M. Adleman. On the potential of molecular computing - reply. *Science*, 268(5210), 1995:483–484.

[3] L. M. Adleman. Computing with DNA. *Scientific American*, 279, 1998.

[4] M. Amos and P. E. Dunne. DNA simulation of boolean circuits. Technical Report CTAG-97009, Department of Computer Science, University of Liverpool, UK, 1997.

[5] *The Bibliography of Molecular Computation and Splicing Systems*, Internet.

[6] R. S. Braich, C. Johnson, P. W. K. Rothemund, D. Hwang, N. Chelyapov, and L. M. Adleman. Solution of a satisfability problem on a gel-based DNA computer. In *DNA Computing, 6th international Workshop on DNA-Based Computers, Leiden, The Netherlands*, 2000.

[7] T. H. LaBean and H. Yan and J. Kopatsch and F. Liu and E. Winfree and H.J. Reif and N.C. Seeman. The construction, analysis, ligation and self-assembly of DNA triple crossover complexes. J. Am. Chem. Soc., 122, 2000.

[8] F. Guarnieri, M. Fliss, and C. Bancroft. Making DNA add. *Science*, 273, 1996.

[9] V. Gupta, S. Parthasarathy, and M. J. Zaki. Arithmetic and logic operations with DNA. In *Proceedings of the 3rd DIMACS Workshop on DNA Based Computers, held at the University of Pennsylvania*, 1997.

[10] K. Komiya, K. Sakamoto, H. Gouzo, S. Yokoyama, M. Arita, A. Nishikawa, and M. Hagiya. Successive state transitions with i/o interface by molecules. LNCS 2054, 17-26.

[11] T.Maniatis, E.F.Fritsch, and J.Sambrook. Molecular cloning: a Lab. Manual. Cold Spring Harbor Lab. Press, Cold Spring Harbor, NY, 1982.

[12] J.J. Mulawka. *Expert Systems (in Polish)*. Warsaw: Wydawnictwa Naukowo-Techniczne (WNT), 1999.

[13] J.J. Mulawka, P. Borsuk, and P. Weglenski. Implementation of the inference engine based on molecular computing technique. In *Proc. IEEE Int. Conf. on Evolutionary Computation, Anchorage, USA*, 1998.

[14] M. Ogihara and A. Ray. Simulating boolean circuits on a DNA computer. Technical Report TR 631, University of Rochester, Computer Science Department, August 1996.

[15] J. S. Oliver. Computation with DNA-matrix multiplication. In *Proceedings of the Second Annual Meeting on DNA Based Computers, held at Princeton University*, 1996.

[16] X. Li, X. Yang, J. Qi and N. C. Seeman. Antiparallel DNA Double Crossover Molecules as Components for Nanoconstruction. Journal of the Am. Chemical Society, 118, 1996.

[17] L. M. Smith, R. M. Corn, A. E. Condon, M. G. Lagally, A. G. Frutos, Q. Liu, and A. J. Thiel. A surface-based approach to DNA computation. *Journal of Computational Biology*, 5, 1998.

[18] P. W.K. Rothemund. Theory and experiments in algorithmic self-assembly. *Dissertation for PhD degree*, University of Southern California, 2001.

[19] P.Sa-Ardyen, N.Jonoska, N.S.Seeman. Self-assembling DNA Graphs. LNCS 2568, 1-9.

[20] G. Tomczuk, P. Wąsiewicz. Molecular binary trees. In *Proc. of the VI Polish Conference on Evolutionary Algorithms and Global Optimization, Korbielow*, 249-254.

[21] H. Uejima, M. Hagiya, S. Kobayashi. Horn Clause Computation by Self-Assembly of DNA Molecules. LNCS 2340, 308-320.

[22] P. Wąsiewicz, R. Rudnicki, J.J. Mulawka, and B. Lesyng. Adding numbers with DNA. In *Proceedings 2000 IEEE International Conference on Systems*, 2000.

[23] P. Wąsiewicz, A. Malinowski, R. Nowak, J.J Mulawka, P. Borsuk, P. Weglenski, and A. Plucienniczak. DNA computing: Implementation of data flow logical operations. *Future Generation Computer Systems*, 17/4, 2001, 361-378.

[24] P. Wąsiewicz, G. Tomczuk, A. Plucienniczak. Molecular Neuron Network Experimental Approximation. In *Proc. 7th WSEAS Int. Conf. Automatic Control, Modelling and Simul., Prague, Czech*, 2005, 489-493.

[25] P. Wąsiewicz, A. Dydynski, J. Mulawka. Implementation of molecular neuron model on DNA strings. *Polish-Czech-Hungarian Workshop, Warsaw, Poland*, 2002.

[26] P. Wąsiewicz, A. Dydynski, G. Tomczuk, J.J. Mulawka, A. Plucienniczak. Molecular Neuron Realization. WSEAS Trans. Journal on Biology and Biomedicine, 1(1):73-75.

[27] P. Wąsiewicz, T. Janczak, J.J. Mulawka, A. Plucienniczak. The Inference Based on Molecular Computing. Int. Journal of Cybernetics and Systems 31/3, 2000:283-315.

[28] P. Wąsiewicz, J.J. Mulawka. Molecular Genetic Programming. Soft Computing, Springer-Verlag 5(2), 2001:106-113.