Lab-on-a-Chip Molecular Inference System

Piotr Wąsiewicz, Jan J. Mulawka

Institute of Electronic Systems, Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warsaw, email: pwasiewi@elka.pw.edu.pl

Abstract. Molecular computation on DNA performs calculations using nanotechnology means during chemical reactions. With the help of silicon industry microfluidic processors were invented utilizing nano membrane valves, pumps and microreactors. These so called lab-on-a-chips combined together with molecular computing create molecular-systems-on-a-chips. This work presents an approach to implementation of logic systems on chips. It requires the unique representation of signals by DNA molecules. The main part of this work includes the concept of logic inference based on typical genetic engineering reactions. The presented method uses a lab-on-a-chip approach. Every microreactor of the lab-on-a-chip performs one unique operation on input molecules and can be connected by dataflow output-input connections to other ones.

1 Introduction

The idea of molecular-systems-on-a-chips was first introduced by McCaskill [18, 32]. He used the miniaturization lab-on-a-chip methodology to solve in microreactors computational problems, which were previously computed in genetic engineering laboratory tubes [1, 2, 5, 22].

Molecules e.g. DNA oligonucleotides called oligos, DNA strings or strands, 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 not only NP-complete problems (nondeterministic-polynomial-time), but also executing logic gates and inference rules [3, 12, 17, 20, 30, 35, 40, 41], adding binary numbers [4, 8, 9, 10, 21, 34], constructing nanodevices (nanoscissors, tic-tac-toe automatons, nanorobots, nanotile assembly) in molecule nanoassembly process [16, 19, 24, 26, 27, 28, 29, 39], implementing computation on the molecular surfaces [13, 14, 25].

McCaskill works [18, 32] started research and discussions, whether this joint technology is viable alternative to computers based on silicon electronics (very advanced, but with technology limits) and to molecular computers based on chemical reactions (in embryonic state, but with unknown and very promising future based on miniaturization and massive parallelism) [7, 33].

In this paper we propose another approach to logic inference systems [17] designed for a lab-on-a-chip methodology.

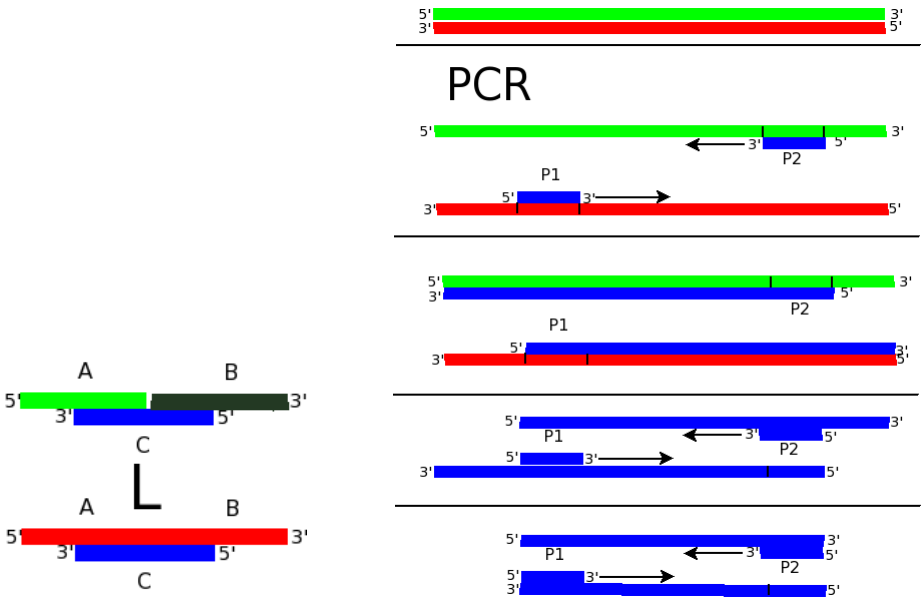

Figure 1. Typical operations of genetic engineering: on the left ligation of A and B oligos in the presence of hybridized third complementary one - C, on the right the extension of primers' 3' ends in the process of PCR

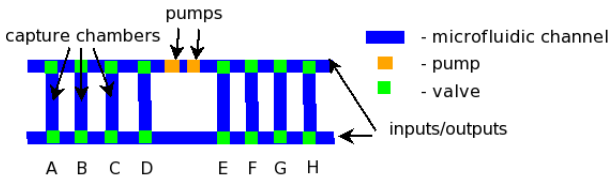

Figure 2. The microfluidic processor with valves, pumps, inputs and eight capture chambers: $A - H$

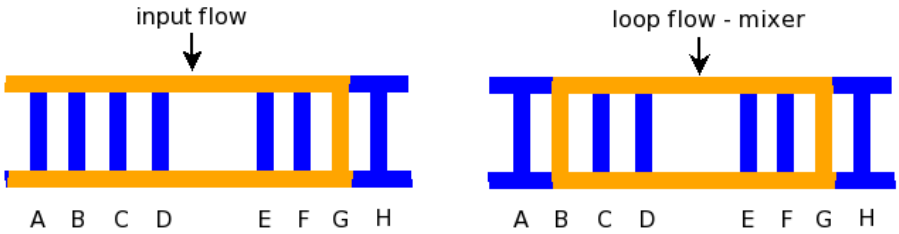

Figure 3. Typical operations of lab-on-a-chip: on the left the input solution flow, on the right the cycle flow in the microfluidic processor

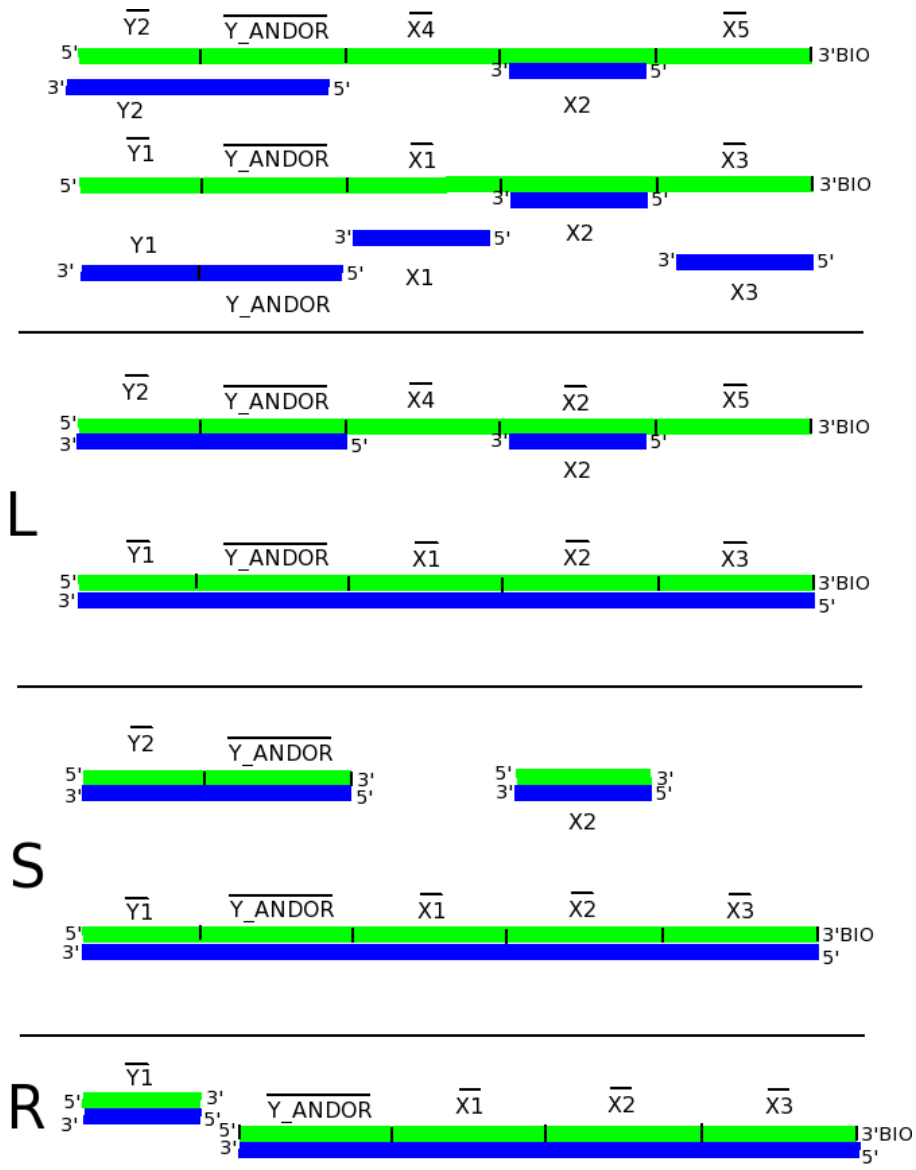

Figure 4. The process of firing a rule - adding $Y1$ conclusion to facts

2 DNA molecules and operations on them

A double helix of DNA is made from two single strands of DNA oriented in opposite directions e.g. $T = ATGC$ and $\overline{T} = TACG$, each of which has two different ends 5', 3' and is a chain of four nucleotides Adenine, Thymine, Cytosine, Guanine denoted by the symbols A, T, C, and G, respectively, due to hybridization (annealing) reaction, because A is complementary with T, and C is complementary with G.

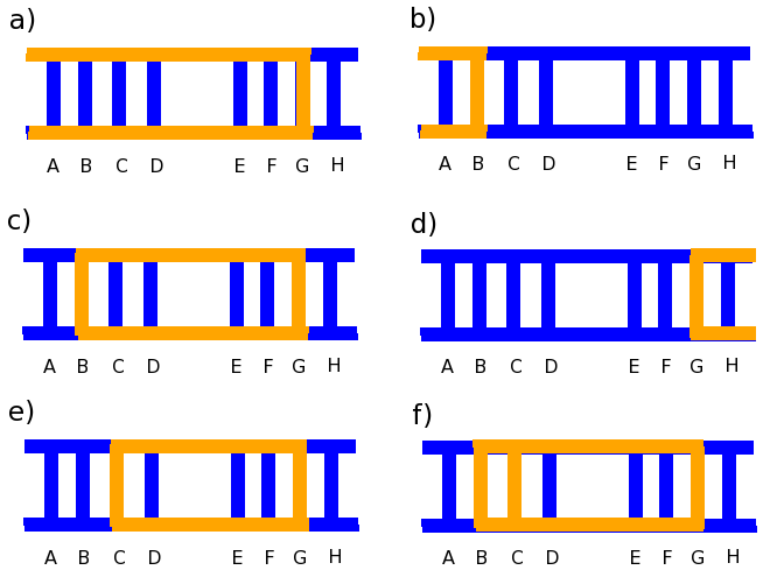

Figure 5. The oligo flows in our exemplary microfluidic processor

Oligonucleotides may connect with each other during concatenation process called ligation L to form longer DNA chains [15] as is shown in Fig. 1 on the left. In order to amplify a target between predefined sequences (primers) a cycle of annealing-melting-extension operations, called Polymerase Chain Reaction (PCR), is utilized with the help of free nucleotides and DNA polymerase which duplicates DNA by adding complementary nucleotides to 3' ends of primers as is depicted in Fig. 1 on the right. One class of enzymes, called restriction endonucleases, recognize a specific short sequence of DNA, known as a restriction site and cut any double-stranded DNA at that location (an operation R). Exonucleases can degrade DNA molecules from the ends in e.g. the S1 nuclease degrades single-stranded nucleic acids (operation S).

Genetic operations driven by enzymes, heating and cooling, DNA sequence and a model make computation possible.

3 Molecular logic system implementations

Ogihara and Ray [20] first demonstrated that DNA computers can simulate Boolean circuits. Klein, Leete and Rubin [11] created universal three input logic gate based on PCR, where a signal equal to 0 is also a DNA string and all permutations of input and output signals equal to 0 or 1 are encoded in DNA molecules. Amos and Dunne [3] described the abstract model and its own laboratory implementation. Hagiya et al [30] designed one molecule DNA computer with data and operations on one DNA strand. Computation of logic function satisfability was driven by PCR reaction. Wąsiewicz [41, 35] also proposed the evolutionary programming of logic function graphs, the evaluation

of which is based on PCR.

Surface-based methods were presented by Liu, Smith and their research group [25, 13, 14]. 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, especially designed for executing logic gates.

4 Molecular-systems-on-a-chip

Significant progress has recently been made in microfluid devices utilizing nano membrane valves, pumps and microreactors. These devices are fabricated on silicon wafers using conventional photolithography. By combining lab-on-a-chips with DNA computing evolves a new area of research, so called molecular systems-on-a-chip [7, 18, 32, 33]. Microfluidic systems on a chip can individually control picoliter-scale quantities of fluids. The following operations are possible: mixing, storage, PCR, heating/cooling, cell lysis, electrophoresis.

As an example McCaskill's on-chip system for serial capture and release of DNA made of H-shaped channel junctions called "selective transfer modules" was a prototype [18] of actual microfluidic DNA computers. In this solution one of two neighboring channels contains input DNA in hybridization buffer and the other contains an alkaline solution with the high pH. Despite the DNA and alkaline streams contacting each other the laminar flow profile limits cross-contamination between the streams. Magnetic beads derivatized with capture oligonucleotides are held by an external magnet in DNA input stream. After capturing DNA strings with correct, perfectly complementary sequence of nucleotides, the beads are transferred magnetically into the alkaline stream, where the high pH destabilizes the DNA duplexes and releases the captured DNA into the alkaline stream. Downstream mixing with fresh hybridization buffer neutralizes the alkaline and makes selected DNA ready for additional capture/release steps. After a series of capture/release steps the remaining DNA represents the correct solution. Initial population encoding all possible answers to the given problem was processed in a single pass through the transfer modules. The most complex DNA computation performed to date was solved using 300 base pair input DNA containing constant 15-base sequences for each of 20-binary bits [5].

Although new solutions of not very good efficiency were proposed e.g. "negative selection", where the alkaline streams are removed, only an invention of monolithic elastomer membrane valves and pumps suitable for large-scale integration into glass microfluidic analysis devices gave a chance to improve capture efficiency, to reduce hybridization time from typical 4 hours to minutes and replace constant multi-base capture sequences with single nucleotides. The air or fluid controlled pumps and valves enable programmable autonomous solution flow and mixing in closed channel sectors [7, 31]. Now it is possible to reuse the same channel several times utilizing different reactions every time. Solution with information can be cycled, processed and stored in separated channels.

An exemplary microfluidic processor [7] is shown in Fig. 2. It has eight reaction chambers containing magnetic beads with capture oligos. Chambers can be open and closed at both sides with the help of microvalves. Pumps in the middle make solution flow. An input DNA flow is shown on the left in Fig. 3. After capturing of input signals in the G chamber, valves near B chamber are closed and solution can flow from the G chamber to the B chamber, where it is captured by other DNA oligos on the beads. After

a series of capture/release operations the correct solution remains in the final chamber. All pumps and valves can be controlled by a computer program [31].

5 The on-a-chip inference system

In our method of inference a fact is represented by a DNA string with a unique sequence of nucleotides optimized by special computer programs [37]. When the fact value is false, the appropriate string is removed and is absent. The rules, formed in a line of DNA strings complementary to premises and a conclusion (shown in Fig. 4), are attached to magnetic beads with one end (denoted by 3'BIO) held in the chamber G after capturing by the external magnet. The appropriate flow is depicted in Fig. 5a.

After filling the B chamber with fact molecules (Fig 5b) and connecting chambers B and G (Fig. 5c), the process of firing rules in the G chamber proceeds as is described in Fig. 4 (molecular operations denoted by L,S,R).

First, all suitable facts are hybridized to rule oligos. Second, the free remained ones are removed in the process from Fig. 5d. Through the same channels the buffer with ligase (concatenating enzyme) is delivered and in the closed completely G chamber all possible joints are made during the L operation. In the same way, the chamber G is filled with the appropriate buffer and the nuclease S1. Operation S in the closed chamber G degrades all single strings. All not bound to beads double strings are removed and the restriction endonuclease with its buffer flows through the same channel as is shown in Fig. 5d. The G chamber is closed and final operation of cutting denoted by L is performed.

The conclusion $Y1$ is introduced as is seen in Fig. 5e to chamber C. Then, destroyed magnetic beads are removed and replaced with new copies, so every cycle of firing rules requires new beads and buffers with enzymes. Facts, in our example being now in chamber B and C, are again used in next hybridization steps with rule molecules, however, after each cycle with a smaller quantity of oligos. To fulfil this procedure every fact should be represented by a large enough group of DNA strings.

The final hypothesis can be decoded during capillary electrophoresis or in special sequencing machines.

6 Summary

It should be mentioned that the lab-on-a-chip methodology evolves as an interesting implementation for inference process to be designed. Comparing DNA computing algorithm performed in genetic engineering laboratory with implementation on a chip, it follows that some operations are preferable for on-a-chip realization. As an example heating problem exists because on a chip reaction chambers are very close to each other and heat propagates in all directions from one to another chamber. Therefore, simultaneous cooling of neighboring chamber can be a problem. The PCR reaction controlled by heating is also difficult to perform in such conditions. In typical computing microfluidic processors there are three basic operations prefered: mixing, deluting and transporting samples of DNA from one place to another on a chip.

To verify practical utility of the method provided here, some practical experiments would be interesting. However, at that time our team is in a state of a lab organization step. We are currently preparing for building our own lab-on-a-chip in cooperation with chemists and biologists.

Bibliography

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

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

[3] Amos M., Dunne P. E.: DNA simulation of boolean circuits. Technical Report CTAG-97009, Department of Computer Science, University of Liverpool, UK, 1997. *http://www.csc.liv.ac.uk/ ctag/archive/t/CTAG-97009.ps.gz*.

[4] Barua R., Misra J.: Binary arithmetic for DNA computers. Poster paper at 8th International Workshop on DNA-Based Computers, DNA 2002, Sapporo, Japan, 2002.

[5] Braich R. S., Johnson C., Rothemund P. W. K., Chelyapov N., and Adleman L. M.: Solution of a 20-variable 3-SAT problem on a DNA computer. Science, 296(5567), 2002, 499-502.

[6] Fink H., Schoenenberg C.: Electrical conduction through DNA molecules. *Nature*, 398, 1999.

[7] Grover W.H.: Microfluidic Molecular Processors for Computation and Analysis. PhD thesis, University of California, Berkeley, 2006

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

[9] Gupta V., Parthasarathy S., and Zaki M. J.: 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] Hug H., Schuler R.: DNA-based Parallel Computation of Simple Arithmetic. LNCS 2340, 321-328.

[11] Klein J.P., Leete T.H., Rubin H.: A biomolecular implementation of logically reversible computation with minimal energy dissipation. Biosystems. 52(1-3), 1999, 15-23.

[12] Kobayashi S., Yokomori T., Sampei G., and Mizobuchi K.: DNA implementation of simple Horn clause computation. *IEEE International Conference on Evolutionary Computation*, 1997.

[13] Liu Q., Frutos A. G., Wang L., Thiel A. J., Gillmor S. D., Strother T., Condon A. E., Corn R. M., Lagally M. G., and Smith L. M.: Progress toward demonstration of a surface based DNA computation: a one word approach to solve a model satisfiability problem. In *Proceedings 4th DIMACS Workshop on DNA Based Computers, held at the University of Pennsylvania*, 1998.

[14] Liu Q., Thiel A. J., Frutos A. G., Corn R. M., and Smith L. M.. Surface-based DNA computation : Hybridization and destruction. In *Proceedings of the 3rd DIMACS Workshop on DNA Based Computers, held at the University of Pennsylvania*, 1997.

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

[16] Michell J.C., Yurke B.: DNA Scissors LNCS 2340, 258-268.

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

[18] van Noort D., Gast F., McCaskill J.S.: DNA Computing in Microreactors. LNCS 2340, 33-45.

[19] Nowak R., Wąsiewicz P., Mulawka J.J., Plucienniczak A.: Processing DNA tokens in parallel computing. In *Proc. International Parallel and Distributed Processing Symposium*, 2001.

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

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

[22] Ouyang Q., Kaplan P. D., Liu S., Libchaber A.: DNA solution of the maximal clique problem. *Science*, 278, 1997.

[23] Lagally E.T., Emrich C.A., Mathies R.A.: Fully integrated PCR-capillary electrophoresis microsystem for DNA analysis. *Lab Chip*, 2001, 1, 102.

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

[25] Smith L. M., Corn R. M., Condon A. E., Lagally M. G., Frutos A. G., Liu Q., and Thiel A. J.: A surface-based approach to DNA computation. *Journal of Computational Biology*, 5, 1988.

[26] Stojanovic M.N., Stefanovic D.: A Deoxyribozyme-Based Molecular Automaton. Nature Biotechnology 21, 2003:1069-1074.

[27] Reif J.H.: The Design of Autonomous DNA Nanomechanical Devices: Walking and Rolling DNA. LNCS 2568, 22-37.

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

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

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

[31] Urbanski J.P., Thies W., Rhodes C., Amarasinghe S., Thorsen T.: Digital microfluidics using soft lithography. *Lab Chip*, 2006, 6, 96-104.

[32] Wagler P. F., Tangen U., Maeke T., Chemnitz S., Juenger M., McCaskill J. S., Varadan V. K.: Molecular systems on-chip (MSoC) steps forward for programmable biosystems, *Proc. of SPIE. Smart Structures and Materials 2004: Smart Electronics, MEMS, BioMEMS, and Nanotechnology* 5389, 2004, 298-305.

[33] Wang Y.: Modeling and Simulation of Lab-on-a-Chip Systems. PhD thesis, Carnege Mellon University, Pittsburgh, 2005.

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

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

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

[37] Wasiewicz P., Tomczuk G., Mulawka J.J. (2002). Hybrid Genetic Approach to Oligo Sets Optimization. In *Proc. of the V Polish Conference on Evolutionary Algorithms and Global Optimization*, Krakow, 2002, 119-130.

[38] Wąsiewicz P., Plucienniczak A.: Molecular Inference Network Experimental Approximation. In *Elektronika* 156, Evolutionary Computation and Global Optimization 2006, 405-412.

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

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

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