Molecular Neuron Network Experimental Approximation

PIOTR WĄSIEWICZ, GRZEGORZ TOMCZUK
The Institute of Electronic Systems
Warsaw University of Technology
Nowowiejska 15/19, 00-665 Warsaw
ANDRZEJ PLUCIENNICZAK
Institute of Biotechnology and Antibiotics
Staroscinska 5, 02-516 Warsaw
POLAND

Abstract: Together with rapidly developing biotechnology, nanotechnology is a real opportunity to test new, maybe revolutionary ideas and algorithms of so called "soft hardware". Self-assembly feature of transforming nano-scale structures, such as DNA macromolecules but not only, from one state to another one in a very well defined way may offer the proper handle for nano-scale computations and play a central role in the development of nano-tech devices in the near future. The Turing machine analogy to information-encoding biochemical reactions on information-carrying molecules inspired our neural network experimental approximation. We describe our original model of molecular neuron network based on genetic laboratory operations.

Key-Words: cellular computing, molecular network, genetic engineering, neural network, molecular computation

1 Introduction

ALGORITHMICS on a nano-scale seems to be about designing data and an algorithm in such a way that self-assembly is able to perform the computation or constructing of molecular devices "by itself" with the help of many processing units so called cells. Their performance cannot be described in the form of the sequential algorithm, more precisely in the form of the operation single list suitable for execution on one processor. Moreover, the NP-difficult problems can be solved in polynomial time only by non-deterministic machine utilizing massive parallelism. Choosing in one moment among millions of solutions ought to be non-deterministic. So chemical reactions as computational processes on molecules carrying encoded information present an obvious analogy to methodologies emerged from nature such as neural networks, evolutionary algorithms and so on.

It is believed that in future, when traditional silicon methodologies meet their technological limits in miniaturization, alternative molecular electronic circuits will be constructed with the use of molecular transistors and logic switches, which have been already invented.

A sequence of operations on DNA executed in parallel on DNA strings called oligonucleotides or oligos is an algorithm. But in the typical DNA computing algorithm this sequence is determined by a model of DNA strings similar to the soft hardware specialized architecture driven here by heating, cooling and connected with them operations on DNA. Together the model and the operation sequence make computation possible. The essential feature of such approach is hybridization of pairs of complementary DNA strings and possibility to represent highly parallel selective operations, which can enable creating alternative, neural architectures on a nanoscale.

Mills has made the first approach to a neural net representation by using DNA matrices on DNA chips. A Hopfield neural network [3] can be realized by implementing memorization and recall. In an experiment data were presented in the form of images of m pixels, which were flashed on a micro-array, whose pixels made of DNA strings attached with one end to the array surface matched the image pixels made of appropriate complementary strings. The sum of all such products becomes the memory matrix. In the experiment roughly two iterations were sufficient to force the mixture into a steady state answer to the query e.g. removing white noise from the query image.

In this paper we consider completely another method in which we focus on modelling a neural network structure made of neural cells presented in [4]. We look for some kind of similarity between real neural structure and something what can be artificially created by self-assembling feature of DNA fragments. The introduced representation requires all the data to be discrete. In our concept we need to design a subset of all single-stranded DNA strings adequate to the particular problem and its data.

2 Neural Network Basis

A neural network is defined in [2] as a parallel, distributed information processing structure consisting of processing elements (which can possess a local memory and can carry out localized information processing operations) interconnected via unidirectional signal channels called connections. Each processing element has a single output connection that branches into as many collateral connections as desired, each carries the same signal - the processing element output sig-

nal. The processing element output signal can be of any mathematical type desired.

In the classical neural architecture each processing element can have multiple input connections (which can originate from other processing elements or from outside the network), but only one output signal. The single output branches into copies (in other words, multiple connections carrying the same signal), which are distributed to other processing elements, or which leave the network altogether. The input to the network can be viewed as a data array x and the output of the network is a data array y. When viewed in this way, the network can be thought of as a function, subroutine, or procedure $y(x)$. The information processing must depend only on the current values of the input signals arriving at the processing element via impinging connections and on values stored in the processing elements' local memory. Neural networks, can be modeled by a general equation:

$$y = f(x, w), \tag{1}$$

where the vectors x, y and w represent inputs, outputs and adjustable weight parameters.

There is no calculation of an energy function or modification of weight values during learning mode in our concept. The main difference from the classical learning mode is caused by its massively parallel computation, which results in a set of all possible self-assembled neural networks architectures being an answer to training data. This massively parallel computation is a major advantage and a key to possible utility of this method. This is a big similarity between our approach and the Monte Carlo method on molecular level.

3 The McCulloch-Pitts Network Model

McCulloch and Pitts described a simple neuron model as a two-value treshold element. The model consists of two components: neuron and synaptic links. The state of the output signal of a neuron is determined by the linear sum of weighted input signals x_{in}. The output signal x_{i+1} of a neuron is 1 if the sum equals or exceeds the threshold value; otherwise it is 0.

The McCulloch-Pitts neuron model with the threshold is shown in Fig. 1, where $x_{i+1} = g(\sum_n x_{in} \times w_{in})$. The traditional scheme of such the neuron has been created during two steps in Fig. 2. The depicted node contains the threshold function and the adder.

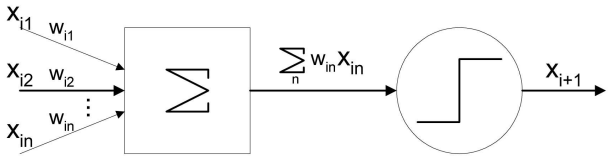

Fig. 1. The McCulloch-Pitts neuron model.

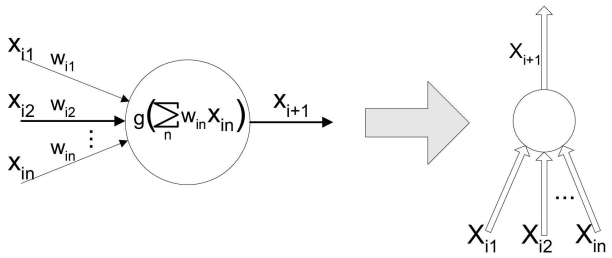

Fig. 2. Creation of the McCulloch-Pitts neuron scheme.

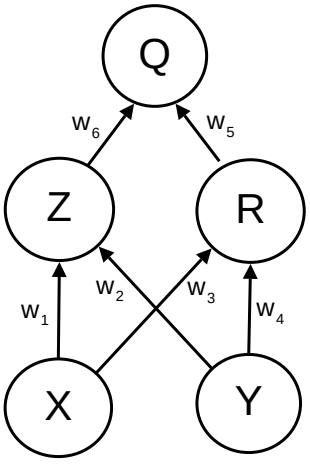

Fig. 3. Neuron network scheme.

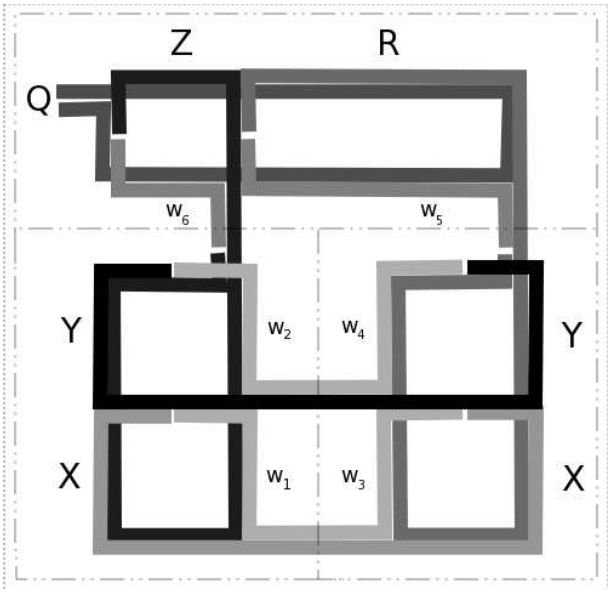

Fig. 4. Molecular neuron network. It consists of three basic cells.

Although its simplicity the McCulloch-Pitts model is a powerful computing tool. McCulloch and Pitts showed that the network formed from such elements as is depicted in Fig. 3 is equivalent to the universal computing machine.

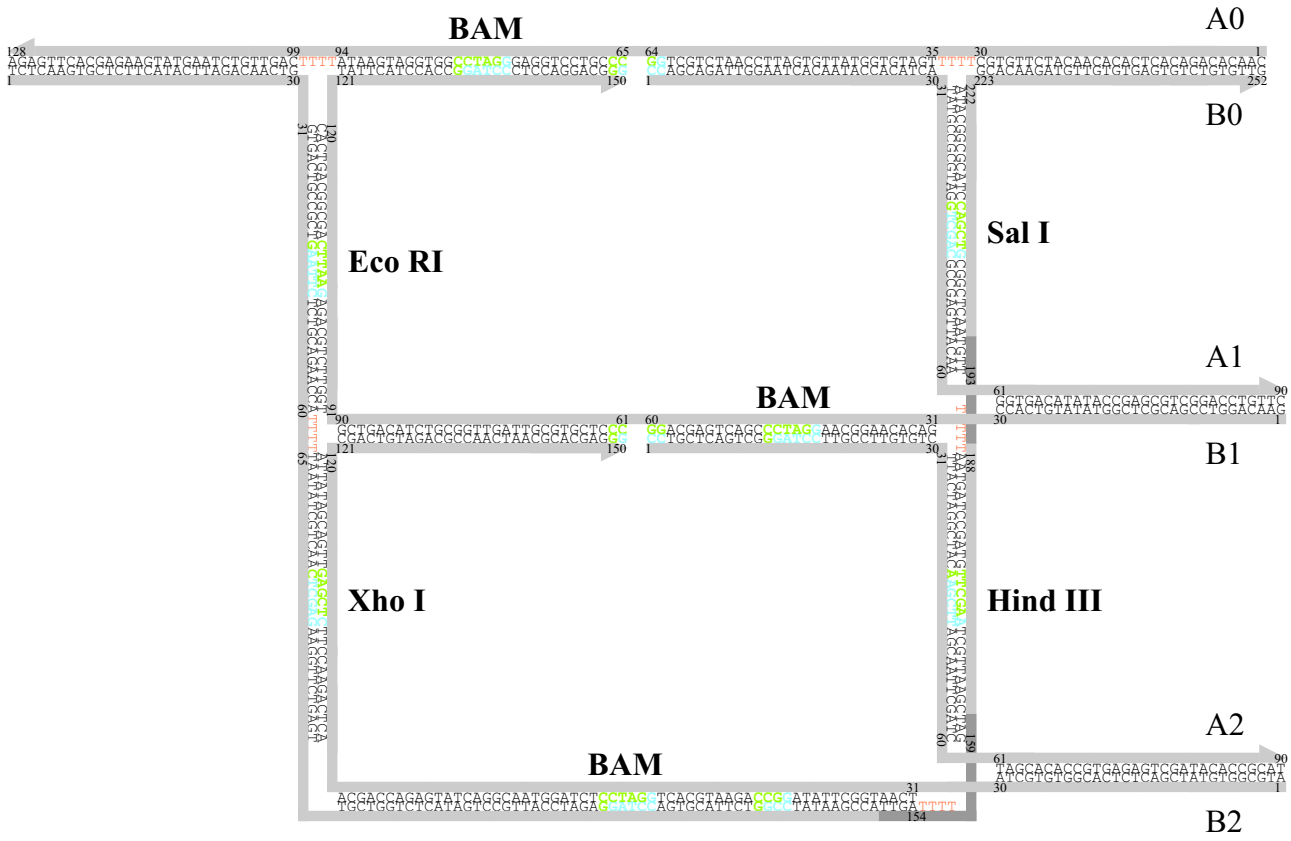

Fig. 5. One basic cell with DNA sequences used in the experiment.

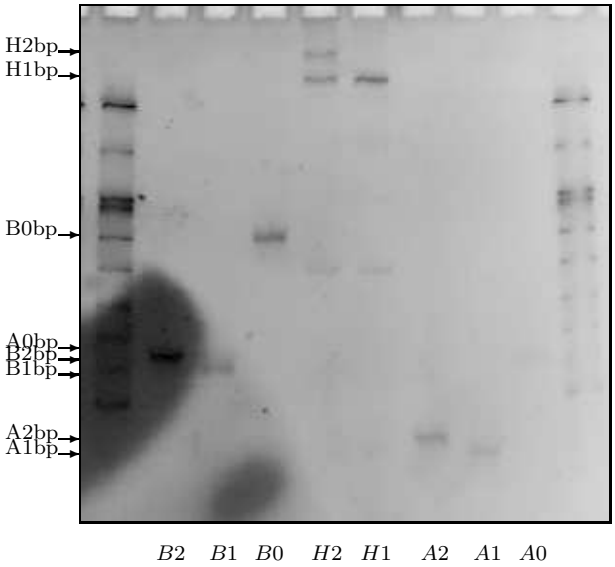

Fig. 6. DNA electrophoretogram of the hybridization experiment.

4 Experimental Approximation of a Molecular Neuron Network by DNA Strings

We consider a class of discrete neural networks, where input and output signals as well as weight coefficients are described by discrete values. In our approach DNA based neuron network model is reported in which standard neuron connections with weigths are formed from oligonucleotide strings resulting in a model that is comparable to that of McCulloch-Pitts one as is shown in Fig. 4. Our molecular model has the same layers, connections and weights. It consists of three basic molecular cells.

The resulting universal basic cell, which can represent neurons in input, hidden, and output layers, but here with two inputs and one output signal was introduced in Fig. 5. It consists of strings $A0, A1, A2, B0, B1, B2$, which were built from shorter synthesized DNA strings shown in Table. 4 e.g. $A0 = A01 + A02 + A03$ was hybridized and ligated with the help of strings: $WA012$ complementary from the 36th to the 57th nucleotide of $A0$, $WA023$ complementary from the 77th to the 98th nucleotide of $A0$; $B1 = B11 + B12 + B13 + B14$ with the help of strings: $WB112$ complementary from the 36th to the 56th nucleotide of $B1$, $WB123$ complementary from the 80th to the 92th nucleotide of $B1$, $WB134$ complementary from the 106th to the 125th nucleotide of $B1$.

5 Experiment Results

In the first experiment every 15 μl of each oligonucleotide (about $30pM$) from a group $B0, B1, B2, A1, A2$ were hybridized together in typical restriction enzyme buffer ($100mM\ NaCl$, $5mM\ MgCl_2$, 10mM $Tris\ HCl$,

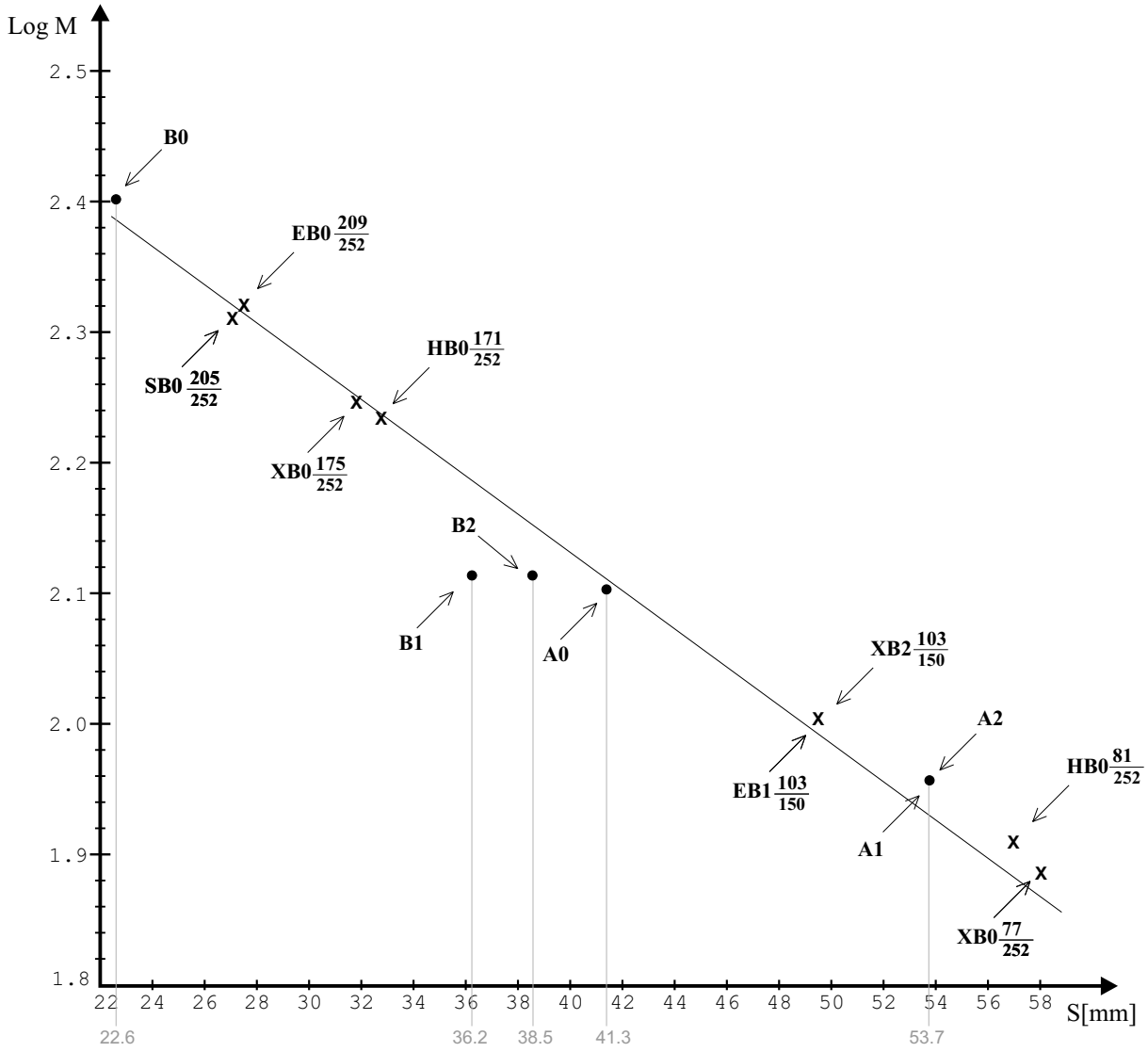

Fig. 8. The diagram of strings before enzyme digestion denoted by big dotes and after cutting denoted by "×".

$pH = 0,8$) for 15 min at 50 ^oC. After adding 15 μl of the next oligonucleotide $B1$, the reaction mixture was incubated for 15 min at 50 ^oC. The sample of this solution was put in a $H1$ lane. And after adding 15 μl of the next oligonucleotide $A0$, the reaction mixture was incubated at 50 ^oC for 15 min, The sample of that solution was put in a $H2$ lane.

The results of the hybridization reaction were verified during electrophoresis (single oligos are put in A and B lanes; in the $H2$ lane there is a band of the complete network cell) and clearly show in Fig. 6 that $A0, A1, A2, B0, B1, B2$ oligonucleotides are hybridized correctly together, because after hybridization reaction of the last five oligos the unique new band $H1$ appears and after hybridization reaction of all six oligonucleotides the next unique new band $H2$ appears larger then the previous one.

The results of the second experiment shown in Figs. 7 and 8 proof that predicted structure (Fig. 5 and Table. 4) really exists in solution. In Fig. 7 the molecular cell typical one hour digestion electrophoretogram is shown. In the lane O cell oligos extracted from the previous experiment $H2$ band are put before digestion e.g.: OB0=B0, OA0=A0. In the rest of lanes after digestion by corresponding enzymes in their typical buffers: B - **BAM**, E - **Eco RI**, H - **Hind III**, S - **Sal I**, X - **Xho I** fragments of cut DNA strings are placed and denoted by enzyme symbol connected with the appropriate string name. In Fig. 8 it is seen the linear dependency between the whole string positions (measured in gel in mm - seen in the electrophoretogram and depended on logarithm of their base pair lengths - $\log M$) denoted by big dots and positions of their fragments obtained after cutting by enzymes. Near the fragment names are placed their lengths in base pairs together with the lengths of their whole predecessors.

TABLE I
The molecular network strings with their nucleotide sequences

Strings	Sequence optimization results
A01 45	AACACAGACACTCACACAACATCTTGTGCTTTTTGATGTGGTAT
A02 43	TTCCAATCTGCTGGCCCGTCCTGGAGGGATCCGGTGGA
A03 40	TGAATATTTTCAGTTGTCTAAGTATGAAGAGCACTTGAGA
B01 44	TCTCAAGTGCTCTTCATACTTAGACAACTGGTGACTGCCGCTGA
B02 44	ATTCTCTGCAGAACCATTTTTAATATCGTCAACTCGAGAAGGTT
B03 43	CTGAGTTGCTGGTCTCATAGTCCGTTACCTAGAGGATCCAGTG
B04 38	CATTCTGGCCTATAAGCCATTGATTTTGATCGAATTGC
B05 42	TAAGCTTGTAGCCTAGTAATTTTTTGTAACTCGGCGTCGACC
B06 41	TACGCGGCATAGCACAAGATGTTGTGTGAGTGTCTGTGTTG
A11 45	CCAGCAGATTGGAATCACAATACCACATCATATGCCGCGTAGGTC
A12 45	GACGCCGAGTTACAAGGTGACATATACCGAGCGTCGGACCTGTTC
B11 45	GAACAGGTCCGACGCTCGGTATATGTCACCGACACAAGGCAAGGA
B12 35	TCCCGACTGAGCAGGCCCTCGTGCGTTAGTTGGCG
B13 35	TCTACAGTCGTGGTTCTGCAGAGAATTCAGCGGCA
B14 35	GTCACTATTCATCCACCGGATCCCTCCAGGACGGG
A21 45	CCTGCTCAGTCGGGATCCTTGCCTTGTGTCTTACTAGGCTACAAG
A22 45	CTTAGCAATTCGATCTAGCACACCGTGAGAGTCGATACACCGCAT
B21 42	ATGCGGTGTATCGACTCTCACGGTGTGCTATCAATGGCTTAT
B22 36	AGGCCAGAATGCACTGGATCCTCTAGGTAACGGACT
B23 37	ATGAGACCAGCAACTCAGAACCTTCTCGAGTTGACGA
B24 35	TATTACGACTGTAGACGCCAACTAACGCACGAGGG

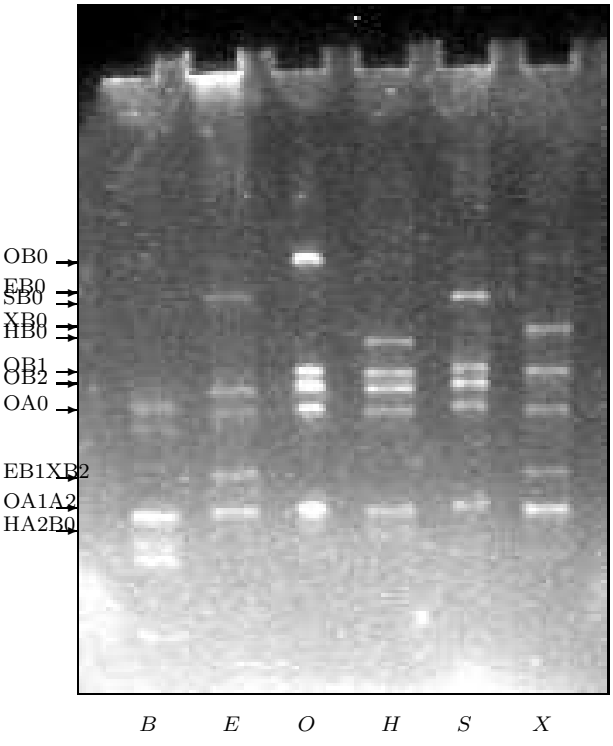

Fig. 7. DNA electrophoretogram of the cell cutting experiment.

6 Summary

After constructing in the laboratory the basic cell, which could be used in building larger and more complex supramolecular structures with many layers like the depicted in Fig. 4 layer graph, it is proved that self-assembly of DNA can be utilized to provide the structure of a adaline-like neural network. Our method is completely original and emerges from visual inspection of the idea of neural network connectionism. It paves a new direction in realization of neural networks by self-organization at molecular scales. It approximates in more natural manner a neural system than simulation by nonlinear operating units, because this approach is based on biochemical reactions, which are basis of every biological process.

Further research should extend ideas and give some approximation of network learning e.g. some kind of backpropagation learning or interference between molecular inference and neural systems in self-assembled macromolecules.

References

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

[2] R. Hoecht-Nielsen, *Neurocomputing*, Addison-Wesley, Reading, USA (1990).

[3] A.P. Mills, Jr., M. Turberfield, A.J. Turberfield, B. Yurke, P.M. Platzman, Experimental aspects of DNA neural network computation, *Soft Computing, Springler-Verlag* **5** (2001) 10-18.

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