

A Chemical System that Recognizes the Shape of a Sphere

K. Giżyński, J. Górecki*

Institute of Physical Chemistry, Polish Academy of Sciences

Kasprzaka 44/52, 01-224 Warsaw, Poland

*E-mail: jgorecki@ichf.edu.pl

Received: 24 November 2016; revised: 29 November 2016; accepted: 29 November 2016; published online: 28 December 2016

Abstract: Unconventional computing devices operating on nonlinear chemical media offer an interesting alternative to standard, semiconductor-based computers. In this work we consider database classifiers formed of interacting droplets in which a photosensitive variant of Belousov-Zhabotinsky (BZ) reaction proceeds. We introduce an evolutionary algorithm that searches for optimal construction of a droplets-based classifier for a given problem. The algorithm is based on maximizing the mutual information between the database and the observed evolution of medium. As an example application of chemical database classifiers we apply the idea to the dataset of points belonging to a unit cube. The dataset contains two output classes: 1 for points belonging to a sphere with radius 0.5 located in the cube center, and 0 for points outside of the sphere. The reliability of optimized chemical classifiers of such database for different numbers of droplets involved in data processing is presented.

Key words: mutual information, Belousov-Zhabotinsky reaction, droplets, evolutionary algorithm

I. INTRODUCTION

Modern information processing is dominated by semiconductor technology. Its success can be measured by Moore's law [1] saying that the circuit complexity, and thus its functionality, doubles in every 18 months. However, it is obvious that the trend predicted by Moore's law cannot be sustained indefinitely. Therefore, despite fantastic progress of microprocessor industry, there is motivation to study alternative methods of computing. The unconventional computing [2-6] is a field of research dedicated to chemistry-, physics- or biology-inspired computational strategies, structures and substrates. In contrast to the conventional computers, based on von Neumann concept of computer architecture [7], unconventional computing interprets the natural time evolution of the medium as a series of information processing operations. In a distributed medium the time evolution of all its parts proceeds simultaneously. As a consequence, typical algorithms executed on an unconventional computer are highly parallel.

Among many implementations of unconventional computers, chemistry-based ones seem especially interesting because chemical reactions are responsible for information processing in living organisms. It is believed that the basic features of biological computing activity are reflected by a reaction-diffusion medium. Chemical reactions showing qualitatively similar evolution as the nerve cells [8] have been studied for many years. The Belousov-Zhabotinsky (BZ) reaction [9] is a chemical process in which an organic substrate is catalytically oxidized in an acidic environment. Among its reagents one can distinguish activator ($HBrO_2$), whose concentration can grow autocatalytically at the appropriate conditions and inhibitors responsible for suppressing the production of activator. Excitation of a medium, seen as a rapid increase in concentration of the activator and next of the inhibitor, can occur when the concentration of reaction activator exceeds a threshold value or concentration of inhibitor fails below a certain level. The reaction is rela-

tively simple for both modeling and experiment because different states of the medium, corresponding to the oxidized and the reduced forms of the catalyst, have different colors. For properly selected initial concentrations of reagents BZ-medium can show simple or complex oscillations or exhibit excitability [10]. These types of behavior are schematically illustrated in Figure 1. In an oscillatory case system states are located on the limit cycle. A trajectory corresponding to the excitable medium also forms a cycle-like structure in the phase space, but the cycle is broken and there is a stable stationary state at one of its ends. Time evolution of homogeneous, excitable medium terminates in this state. An excitable system can be activated by an external perturbation. For a spatially distributed medium propagating pulses or different types of spatio-temporal structures can be observed [11]. If a perturbation leading to a high activator concentration is local, then activator diffuses around. Excitation at a neighboring point can occur if activator concentration at this point becomes sufficiently large. Subsequent excitations of the neighboring points lead to formation of a propagating pulse of excitation (a spike). A spike dissipates the chemical energy of the medium. In a typical excitable medium spike shape converges to a stable form independent of the initial conditions. Many computational applications of a nonlinear medium relate information with the presence of spikes at selected points of the medium. Having in mind that reaction and diffusion determine the time evolution of excitations, the information processing with such system is called reaction-diffusion computing [4].

A version of BZ-reaction catalyzed with a ruthenium complex can be efficiently controlled with illumination [10], because the blue light activates such catalyst and leads to production of Br^- ions that inhibit the reaction [12]. Under illumination the excitability of BZ medium decreases. If a dark medium is self-excitatory (oscillatory) then oscillation period increases with light intensity. For larger illuminations the medium becomes excitable. In the majority of studies on information processing with BZ reaction a compartmentalized excitable medium composed of excitable and non-excitatory (highly illuminated) regions was considered.

Computational properties of the medium are determined not only by the kinetics of chemical processes, but also by the structure of the medium [13]. Geometries of excitable and non-excitatory regions that define the basic logic gates have been described [14-20]. However, the structures of presented devices have to be developed by an intelligent designer and there is no room for their self-generation (although teaching of devices with stiff geometrical structures is possible if the feedback is introduced [21]). Quite recently the interest in computing with BZ-reaction has shifted towards more flexible media in which the structure of medium can be transformed in time [22, 23]. One interesting approach considers droplets containing solution of BZ-reagents (BZ-droplets) surrounded by a solution of lipids in

an organic phase. Droplets are formed when a small amount of BZ medium is immersed into an organic oil phase containing lipids or surfactants. The lipid molecules dissolved in the organic phase cover the surface of a droplet and stabilize it mechanically [23]. Therefore, droplets can be arranged into larger structures that remain stable for a long time. When two droplets come in contact, lipids form a bilayer at the connection surface [24]. Molecules of BZ activator can diffuse through this membrane and excite the medium behind, triggering a chemical wave in the neighboring droplet, thus information coded in excitation pulses can be transmitted between droplets.

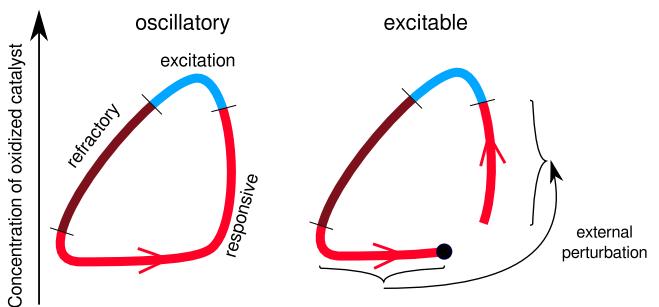


Fig. 1. Time evolution of (a) oscillatory and (b) excitable BZ medium. In the refractory phase (brown) the system is insensitive to external perturbations. Perturbation of the medium in the responsive state (red) can reduce time to excitation. Evolution of unperurbed, homogeneous, excitable medium ends in the stable state

Although it was shown that chemical processing of information with BZ droplets is universal and all logic gates can be constructed from a reaction-diffusion medium [4], neither the speed of such gates nor their size can compete with modern silicon computers. Here we focus on chemical computers that can perform a complex classification function, i.e., that can discriminate a class from a number of input parameters. Each logic gate can be seen as a classification problem for which the database is complete and the class is the gate output. Classification problems are more general than logic gates. Unlike the definition of a logic gate, which lists all possible input states, a dataset considered in a typical classification problem contains a fraction of all possible cases. Each test case contains a number of input variables (predictors) and an assigned category (output class). The aim is to find a generalizing algorithm or a device that correctly classifies the test cases. It is believed that it also gives the correct answer in situations that are not contained in the dataset used for training.

A high accuracy classification algorithm can be obtained as a result of the teaching process. Typically a dataset contains a number of test cases with known output classes that can be used to test the progress of the training. This approach is mainly used in machine learning techniques where the final algorithm is implemented for standard, silicon computers. In the following, we demonstrate that a similar train-

ing strategy can be adopted for designing chemical classifiers implemented in a network of interconnected oscillatory BZ droplets. In such a system, the values of predictors describing a single test case are transformed into input signals applied to droplets working as inputs of the network using an external factor controlling BZ reaction (e.g. light illumination). Interactions between chemical excitations in the medium are responsible for information processing. The time-evolution of chemical activity inside the network can be traced and used as the output signal. Due to the continuous, spatial and temporal character of chemical excitations, a large number of different interactions between the droplets are possible and the output signal can be very complex. We demonstrate that evolutionary algorithm can be applied for finding the conditions at which a given set of BZ-droplets performs a classification task in the optimal way. As an example of a classification problem we considered a dataset of points located inside a unit cube. We demonstrate how several interacting BZ droplets can recognize points belonging to the sphere with radius 0.5 placed in the center of the cube.

II. CHEMICAL CLASSIFICATION COMPUTERS CONSTRUCTED WITH OSCILLATING DROPLETS

We consider a chemical classification computer in the form of a single geometrical structure of $n \times n$ nodes located on a square lattice (Fig. 3). Each node represents a droplet containing oscillating BZ medium. The droplets are interconnected with neighbors according to the von-Neumann neighborhood as presented in Fig. 2. We assume that oscillations in each droplet can be individually controlled by external illumination. After illumination is switched off, chemical oscillations are restored immediately [25]. The time evolution of the network is investigated for 100 s, that is slightly longer than 3 periods of oscillations in a separated droplet.

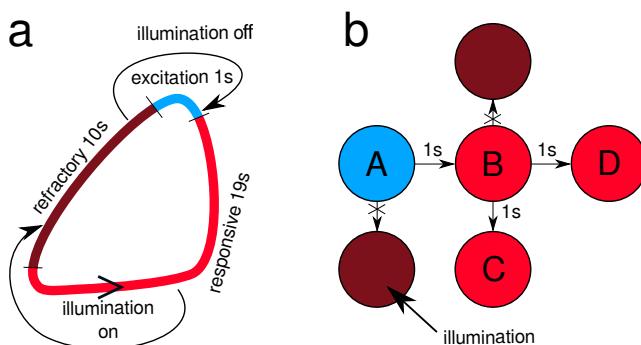


Fig. 2. Typical time evolution observed in oscillatory BZ droplets. (a) The influence of illumination on a BZ droplet. The times of each phase refer to the model used in this paper. (b) If a droplet is excited (A) then, after $\Delta t=1$ s excitation passes to all neighboring droplets in the responsive phase. No droplet in the refractory state is excited

Given a droplet network, one of the droplets is considered as the output and a number of droplets are used as in-

puts (Fig. 3). The remaining droplets, called “normal”, transmit and integrate excitations. Furthermore, they are used to program the network by a sequence of illumination patterns. That is, for different inputs, the input droplets are illuminated differently, while the normal droplets are always illuminated by the same sequence, representing the program.

Because the manual programming of nonlinear media is hard and time consuming, we introduce an evolutionary algorithm (EA) to find an optimal illumination sequence automatically, here for each droplet a time interval during which it is illuminated. Note that classical approaches that combine EA with an artificial neural network (ANN) usually optimize the network’s static parameters [26] or its structure [27, 28]. Here, however, we evolve for the given network a temporal control pattern (the illumination pattern) which is applied together with each input.

The network fitness is evaluated by numerical simulations based on a stochastic model of droplet interactions. The fitness function measures the mutual information [29] between chemical oscillation patterns and the database output classes. The droplet with the highest mutual information generates the output.

II. 1. The simplified event-based model of BZ-droplet network

The evolved networks are simulated using a stochastic, time-continuous model [30]. We assume that all droplets which form a network have the same chemical composition. The process of droplet activation and the excitation transfer can be described by kinetic equations with diffusion terms. However, even the most simplified model like the two variable model (one variable is the concentration of bromous acid that plays the role of reaction activator and another is the concentration of the catalyst in its oxidized form that works as the inhibitor) described in [31] is computationally expensive, especially if it is applied to a three-dimensional, multi-droplet structure. To reduce the computation time required to calculate the time evolution of the network we used an event-based model [25, 30] described below. We assumed that the medium inside droplets oscillates because such behavior was observed in experiments. There are three phases during an oscillation cycle (cf. Fig. 1). A *refractory phase* begins just after excitation and it is characterized by a high level of inhibitor. In this state a droplet becomes insensitive to chemical excitations coming from its neighbors. Next, the inhibitor concentration decreases with time and after crossing some threshold value the droplet enters the *responsive phase* in which a stimulus incoming from any of the neighbors can lead to an excitation. If no external triggering occurs then the amount of inhibitor drops to a level when self-excitation occurs. Rapid production of activator yields oxidation of the catalyst, seen as the change of droplet color from red to blue named below as an *excitation phase*. Having in mind experimental results on interactions between droplets [25], we assumed that the duration of the excitation, refractory and re-

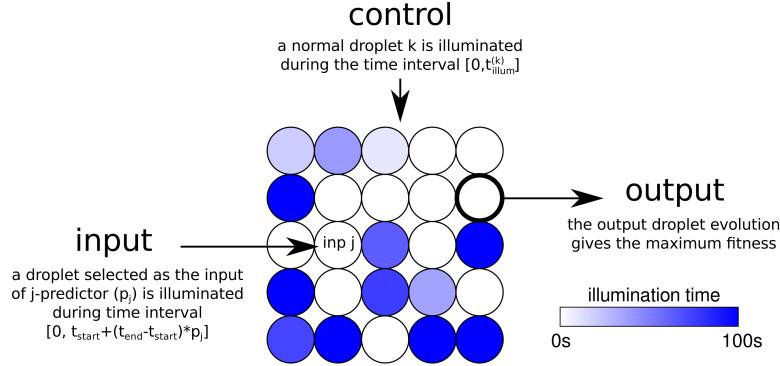


Fig. 3. Illustration of the droplet architecture studied herein. The time evolution of the network is studied for 100 s. An **input** is provided by stimulating selected input droplets by illumination. The summed number of excitations appearing at one, particular droplet is taken as the **output**. The **control** “program”, which we obtain by evolution, is a temporal illumination pattern applied to the remaining “normal” droplets

sponsive phases are 1 s, 10 s, and 19 s respectively. These numbers sum up to the typical oscillation period of 30 s.

Excitation transfer between two interconnected droplets occurs if one droplet is in the excited phase whereas the other is in the responsive state. In our theoretical analysis we assumed that a droplet can be triggered only by the four closest neighbors (von Neumann neighborhood). For the simplicity of simulation we introduced a propagation time parameter $t_{prop}=1$ s, identical for all droplets, regardless of the excitation geometry. For the considered photosensitive reaction, oscillations can be suppressed when a droplet is illuminated with sufficient light intensity (see Fig. 2a). Then, illuminated droplets remain in the refractory phase. When the illumination is switched off the droplet gets excited immediately. The photoinhibition of oscillations is used here to evolve an illumination pattern for the network in order to optimize its functionality. We assume that each droplet i can be separately illuminated, from the beginning of simulation $t_0=0$ to the time $t_{illum}^{(i)}$, therefore the illumination pattern is described by a vector of illumination intervals $[t_0, t_{illum}^{(i)}]$. When the illumination time for a droplet is as long as the total time of the simulation, the droplet remains inactive during the whole simulation and it can be regarded as an empty slot in the network.

Standard computers are built with the assumption that all components are fabricated without any defects and that the fluctuations that are present in the system do not affect the results of the executed programs. However, a highly nonlinear chemical medium, like the BZ reaction, is very sensitive to internal and external fluctuations. For example, it has been recently demonstrated [32] that local charges can change the diffusion of particles (like the activator) through the membranes (see also [33]) and modify times of signal propagation between droplets. In our model the stochastic effects are introduced into simulations in the form of normal distributed noise with the arbitrarily selected standard deviation of 0.05 added to t_{prop} and to the times where a droplet remains in a

particular phase.

In a single simulation we register the time evolution of oscillations in the system, i.e. the times at which excitations occur in each droplet. We assume that a signal represented in this way provides complete information necessary to analyze the network dynamics.

II. 2. Information inflow and outflow

We consider three droplet types: input, normal, and output (see Fig. 3). They are distinguished according to the functionality in the network. Each droplet is assigned either with an input or a normal type. A droplet i of the normal type is inhibited during the time interval $[0, t_{illum}^{(i)}]$.

Input droplets are used to feed the values of predictors of each input case from the dataset into the network. In order to evaluate a single network all test cases have to be simulated separately. Illumination times of normal droplets $t_{illum}^{(i)}$ remain the same for all test cases. The predictor values change according to the test case, thus the illumination time of input droplets for each case is different. If a dataset contains k predictors, then we distinguish k different input types. For the datasets considered here $k = 3$ and the predictors describe $x-$, $y-$ and $z-$ coordinates of a point in the unit cube. Assume that $v_s^j \in [0, 1]$ is the value of the predictor s ($s = 1, \dots, k$) for a test case j from a database. If a droplet is of the input type, corresponding to the predictor s , then during a simulation of this test case it will be illuminated in the time interval $[0, t_{start} + v_s^j(t_{end} - t_{start})]$. The interval $[t_{start}, t_{end}]$ is the same for all input droplets in a particular network, but can differ from genotype to genotype. The values t_{start} and t_{end} are optimized together with the set of initial illuminations.

Note that the types of the droplets are also subjected to evolution and thus the position of inputs might vary according to generation. We also do not introduce any constraints on the number of input droplets in the network. For example, it is possible that there are three droplets with input type

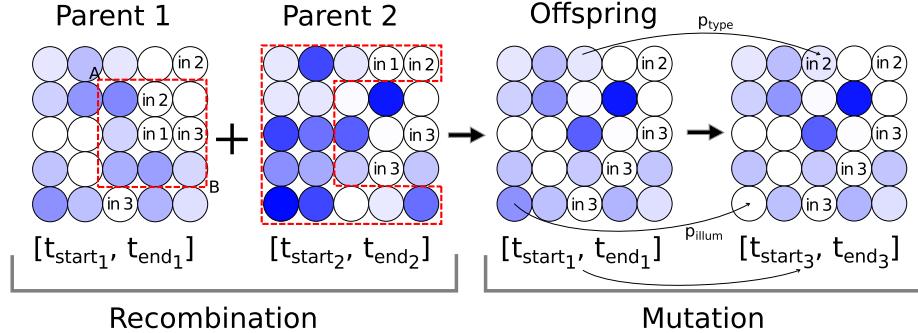


Fig. 4. Randomly selected points A and B in the structure of Parent 1 mark a rectangle, which is copied, along with the illumination interval for inputs, to the Offspring during the recombination process. The other part of the Offspring comes from Parent 2. Then, during the mutation, droplet types and initial illumination times are modified. Intensity of blue color in each droplet is proportional to its illumination time

3 and no droplet with input type 1 in the newly generated network as seen in Fig. 4. In this situation the predictor 3 is provided to all corresponding input droplets and no information about predictor 1 is transferred to the network.

Exactly one droplet of the network is selected as the output droplet. The mutual information with the output class distribution is checked separately for each droplet in the network during the fitness evaluation procedure and the one with the highest value is selected as the output droplet. Since the mutual information in the droplets changes during evolution, the position of the output is not fixed and also can change from generation to generation. We do not exclude the case in which an input droplet is used as the output one.

II. 3. Evolution of BZ networks

In our simulations we consider a classifier formed by a network of $l = n^2$ droplets arranged in a $n \times n$ square lattice where a droplet $i \in \{1, \dots, l\}$ is characterized by its illumination time $t_{\text{illum}}^{(i)}$ and functional type tp_i . The droplet type $tp_i \in \{0, \dots, k\}$ is selected from the k different predictors, as explained in the last section, plus one extra value, e.g. 0. A value of $tp_i = 0$ denotes that the droplet i is not used as classifier input, but should be illuminated according to the value $t_{\text{illum}}^{(i)}$ and thus represents the classification program. More formally, we represent a genotype g as a tuple of the list of functional types \overrightarrow{tp} for each droplet, the list of the illumination times $\overrightarrow{t_{\text{illum}}}$, and the two parameters t_{start} and t_{end} that define the time interval in which input droplets are illuminated. Note that t_{start} and t_{end} are not global values but are part of each genotype and thus they are co-evolved with the network.

$$g = (\overrightarrow{tp}, \overrightarrow{t_{\text{illum}}}, t_{\text{start}}, t_{\text{end}}) \in \{0, \dots, k\}^l \times [0, t_{\text{exp}}]^{l+2} \quad (1)$$

$$0 \leq t_{\text{start}} \leq t_{\text{end}} \leq t_{\text{exp}}$$

$$0 \leq t_{\text{illum}_i} \leq t_{\text{exp}}$$

The evolution scheme is based on the approach presented in [34]. The population size is $\mu = 8$ parents and $\lambda = 30$ offspring. The number of generations is set to 500. Depending on the noise influence, either COMMA or PLUS strategy was applied [35, 36]. The first option assumes that only newly generated offspring are transferred to the next generation. This process is similar to natural procreation in the sense that the parents give birth to the offspring population and die afterwards. In contrast, when the PLUS strategy is applied, a number of $\mu = 5$ best parents are copied to the next generation along with the offspring. In this case, an individual with high fitness can survive for many generations.

Following the typical $(\mu/\rho+, \lambda)$ evolution strategy notation, our evolution hence resembles a $(8/2, 30)$ -strategy for COMMA selection and a $(5/2 + 25)$ -strategy for PLUS selection. Following the standard recombination techniques [37] $\rho = 2$ parents are involved in the procreation of one offspring. A rectangular sub-grid of droplets from Parent 2 constrained by two, randomly selected points A and B is replaced with the corresponding sub-grid in Parent 2 as illustrated in Fig. 4 to yield a new individual. The illumination interval of input droplets co-evolved with the network is copied from the Parent 1 to the offspring. The newly generated individual is subjected to three subsequent mutation operators with fixed mutation rates (see Fig. 4), according to the following order:

1. Input illumination interval mutation

Times determining illumination interval of droplet with input types, i.e. $t_{\text{start}}^{(\text{Offspring})}$, $t_{\text{end}}^{(\text{Offspring})}$ are selected from the normal distributions with the averages $t_{\text{start}}^{(\text{Parent1})}$, $t_{\text{end}}^{(\text{Parent1})}$, respectively, and $\sigma^2 = 10$ as follows:

$$t_{\text{start}}^{(\text{Offspring})} = \mathcal{N}(t_{\text{start}}^{(\text{Parent1})}, \sigma^2) \quad (2)$$

$$t_{\text{end}}^{(\text{Offspring})} = \mathcal{N}(t_{\text{end}}^{(\text{Parent1})}, \sigma^2)$$

$$t_{\text{start}}^{(\text{Offspring})} \leq t_{\text{end}}^{(\text{Offspring})}$$

where $\mathcal{N}(t, \sigma^2)$ is a random number selected from the normal distribution with average t and variance σ^2 . If $t_{\text{start}}^{(\text{Offspring})} > t_{\text{end}}^{(\text{Offspring})}$ both times are swapped, i.e. $t_{\text{end}}^{(\text{Offspring})}$ is replaced with $t_{\text{start}}^{(\text{Offspring})}$ and vice versa.

2. Droplet type mutation

We assume that input droplets can change into normal droplets and vice versa. The probability of droplet transformation is $p_{\text{type}} = 0.04$. Regardless of droplet type, the probability of obtaining an input droplet is $p_{\text{inp}} = 0.12$ and for the normal one $1 - p_{\text{inp}}$.

3. Illumination time mutation

If a droplet i is of the normal type, then with a probability $p_{\text{illum}} = 0.04$ its illumination time is mutated. New illumination time for the Offspring $t_{\text{illum},i}$ is generated from the normal distribution with the average $t_{\text{illum},i}$ and $\sigma^2 = 25$.

The values of all probabilities were selected arbitrarily as being reasonably small, but still large enough to produce noticeable changes in network functionality after mutation. They should have little influence on the final optimized classifier, but definitely decide the rate of convergence towards the optimum solution. Independently of the mutation operators, the fitness function always chooses the output droplet that generates the highest mutual information to the desired output class.

II. 4. Fitness evaluation

The quality of a droplet-based classifier is measured by a fitness function, which takes into account answers from all test cases of the considered dataset as an argument. We evaluate a network's fitness using the mutual information between the total number of excitations in the output droplet and the output class in the dataset. Such mutual information, and hence the fitness value, can be interpreted as the reduction of uncertainty about the output class when seeing a particular number of excitations. After simulating all test cases from the dataset, we sum up all excitations from every pattern (separately for all droplets) to obtain excitation dis-

tribution \mathcal{S}_i for each droplet i . Then the mutual information with the output class distribution \mathcal{P}_o is equal to:

$$I(\mathcal{S}_i : \mathcal{P}_o) = H(\mathcal{S}_i) + H(\mathcal{P}_o) - H(\mathcal{S}_i, \mathcal{P}_o). \quad (3)$$

Here $H(\mathcal{S}_i)$ and $H(\mathcal{P}_o)$ are the Shannon entropies of the summed excitations distribution in droplet (i) and the distribution of output classes in the considered dataset, respectively. $H(\mathcal{S}_i, \mathcal{P}_o)$ is the joint entropy of both distributions. Let us note that if the number of excitations in a droplet is always the same, independently of predictor values, then $H(\mathcal{S}_i)=0$ and $H(\mathcal{S}_i, \mathcal{P}_o) = H(\mathcal{P}_o)$. As a result, $I(\mathcal{S}_i : \mathcal{P}_o)=0$. On the other hand, if the excitation number is perfectly correlated with the database output (for example, two excitations for a point inside the sphere and one excitation in the other cases) then $H(\mathcal{S}_i) = H(\mathcal{P}_o)$ and $H(\mathcal{S}_i, \mathcal{P}_o) = H(\mathcal{P}_o)$. Now $I(\mathcal{S}_i : \mathcal{P}_o) = H(\mathcal{P}_o)$, which means that all output information of the database can be extracted from the excitation number. In practice \mathcal{S}_i are different for various droplets. The droplet with the highest correlation with the dataset becomes the output of the classifier. It means that the signal appearing in this droplet contains the maximum information about the cases from a dataset and thus it can be used for classification. The fact that one finds a droplet giving the best answer does not solve the problem completely. An additional effort needs to be made by the observer to interpret this signal and to find a useful classification rule which can be difficult [38]. In the approach presented here the output of the network is limited, for simplicity, to a single droplet. However, for more complicated classification problems, one can consider signals combined from multiple droplets.

II. 5. Classification datasets

We tested our approach on an artificial dataset with test points (cases) confined in a unit cube. Volume of the cube was subdivided into two parts: (i) internal sphere with $R = 0.5$, centered at $x_0 = y_0 = z_0 = 0.5$ and (ii) remaining part with spherical volume subtracted. The coordinates x,y,z of test cases were generated randomly and assigned with output classes 0 and 1 according to the part of the cube they were

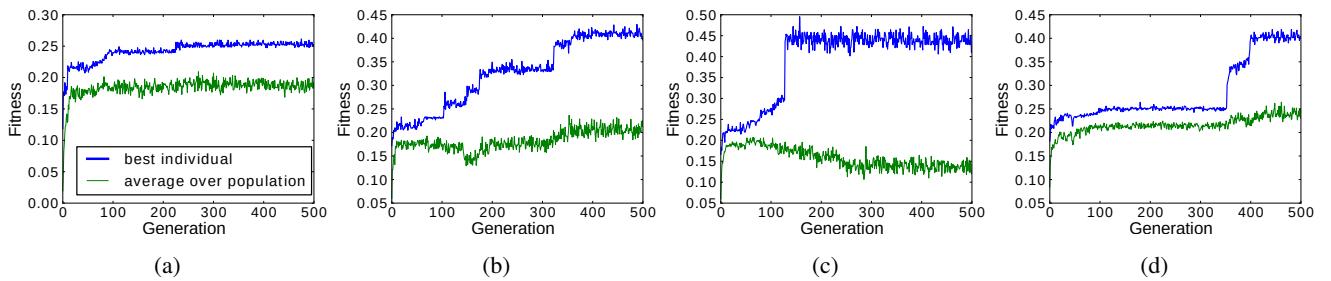


Fig. 5. Fitness evolution for simulations with dataset containing 200 cases. Plots for network with following geometries are shown: (a) 2×2 , (b) 3×3 , (c) 4×4 , (d) 5×5 . The respective amount of mutual information for the best individual found was 0.26 bit, 0.43 bit, 0.46 bit and 0.42 bit

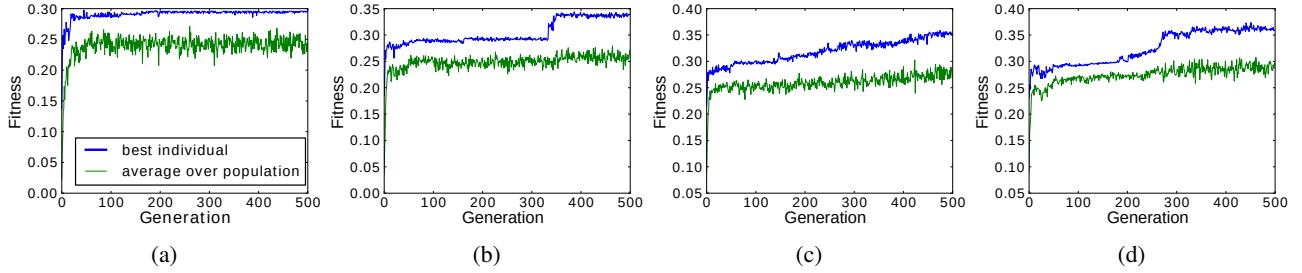


Fig. 6. Fitness evolution for simulations with dataset containing 400 cases. Plots for network with following geometries are shown: (a) 2×2 , (b) 3×3 , (c) 4×4 , (d) 5×5 . The respective amount of mutual information for the best individual found was 0.30 bit, 0.34 bit, 0.36 bit and 0.37 bit

located in (outside and inside of the sphere, respectively). Classification of a test case (x , y , z) can be summarized with the following algorithm:

$$\begin{cases} 1, & \text{if } (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 < R^2, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

The dataset was fixed and the test cases remained the same for all simulations. In our studies we used two datasets containing $N = 200$ or 400 cases. Since the volume ratio of part (i) to part (ii) was 1.09 the number of cases from each output class were adjusted accordingly. That gave 95 and 191 cases from class 0 and 105 and 209 cases from class 1 for

$N = 200$ and $N = 400$, respectively. The entropies of these datasets were $H_{N=200}(\mathcal{P}_o) = H_{N=400}(\mathcal{P}_o) \approx 0.998$ bit.

III. RESULTS AND DISCUSSION

Evolution of droplet classifiers was studied for four sizes of networks with nodes arranged on 2×2 , 3×3 , 4×4 and 5×5 grids. The networks were evolved and tested separately for datasets containing 200 or 400 cases (see Section II. 5.). For each size of the network and the dataset we performed 25 simulations. In the initial part of simulation (first 50 generations) we allowed the system to evolve in a more expansive manner by applying COMMA strategy. Then, to reduce the

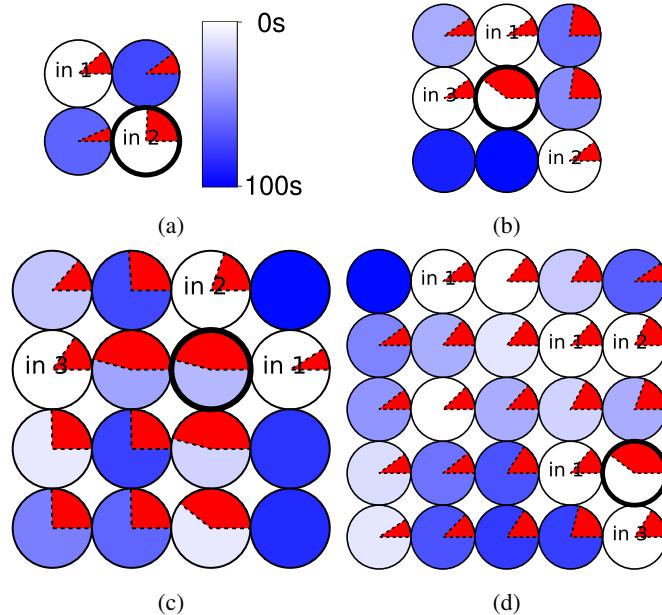


Fig. 7. Illumination patterns for optimized classifiers of the dataset containing 200 test cases. Plots for network with following geometries are shown: (a) 2×2 , (b) 3×3 , (c) 4×4 , (d) 5×5 . Circles represent droplets in a network and brightness of the blue color is proportional to the initial illumination time. The conversion of time to the blue color is shown on a vertical bar in (a). If no blue color is visible then the droplet was active from the beginning of the experiment. High blue color intensity corresponds to illumination time close to the total simulation time. The amount of mutual information contained in each droplet evolution is marked with a red color in the form of a pie chart where the sector size is normalized to the maximal value of mutual information that can be obtained from employed inputs. The output droplet is marked with a wide black border and the numbers of the input droplets correspond to the predictor number in the dataset

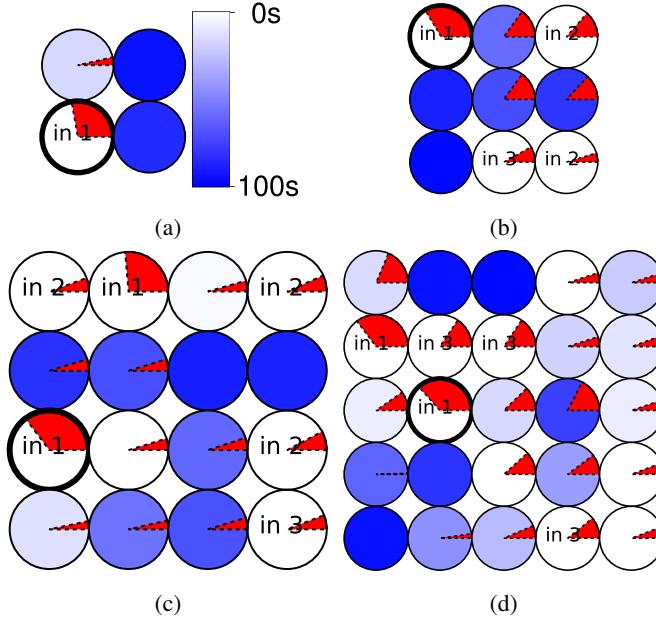


Fig. 8. Illumination patterns for dataset containing 400 test cases. Plots for network with following geometries are shown: (a) 2×2 , (b) 3×3 , (c) 4×4 , (d) 5×5 . Notation as in Fig. 7

noise we switched to PLUS strategy for the remaining 450 generations.

The amount of mutual information in the networks is shown on the fitness evolution plots in Figs. 5 and 6 for dataset containing 200 and 400 cases, respectively. As expected, the mutual information is an increasing function of network size except the case of 5×5 classifiers, for which the optimization seems uncompleted. The smallest classifier (2×2) gave nearly half of the mutual information with the dataset compared to larger networks. It indicates that the number of nodes was insufficient for a reliable classifica-

tion. For $N = 200$ the highest fitness value was observed for 4×4 network (0.49 bit at generation 157). Yet it was a random event and in consecutive generations fitness stabilized around 0.45 bit which was still higher than for the rest of the networks. In the cases (a-c) fitness value was stable for at least 180 generations before optimization was terminated while for 5×5 classifier it was stable within last 100 generations only. We expect that for longer runs of the evolutionary algorithm the larger network could be still improved, finally yielding a fitness value that is higher than for the smaller systems. This observation is supported by the data for $N = 400$

Tab. 1. Amount of mutual information (MI), classification rules and the corresponding classification accuracies for 200 and 400-case datasets, for all considered network sizes. The “stupid” classifier that allocates all points inside the sphere has accuracy ~52%

	200 cases			400 cases		
	MI	Accuracy	Rule	MI	Accuracy	Rule
2×2	0.26 bit	77%	$S \in \langle 2, 3 \rangle \rightarrow 1$ $S < 2 \vee S > 3 \rightarrow 0$	0.30 bit	77.5%	$S \in \langle 3, 4 \rangle \rightarrow 1$ $S < 3 \vee S > 4 \rightarrow 0$
3×3	0.43 bit	84%	$S \leq 5 \rightarrow 1$ $S > 5 \rightarrow 0$	0.34 bit	79%	$S \in \langle 2, 4 \rangle \rightarrow 1$ $S < 2 \vee S > 4 \rightarrow 0$
4×4	0.46 bit	87.5%	$S \leq 3 \rightarrow 1$ $S > 3 \rightarrow 0$	0.36 bit	80%	$S \in \langle 2, 4 \rangle \rightarrow 1$ $S < 2 \vee S > 4 \rightarrow 0$
5×5	0.42 bit	85%	$S \leq 6 \rightarrow 1$ $S > 6 \rightarrow 0$	0.37 bit	81%	$S \in \langle 3, 5 \rangle \rightarrow 1$ $S < 3 \vee S > 5 \rightarrow 0$

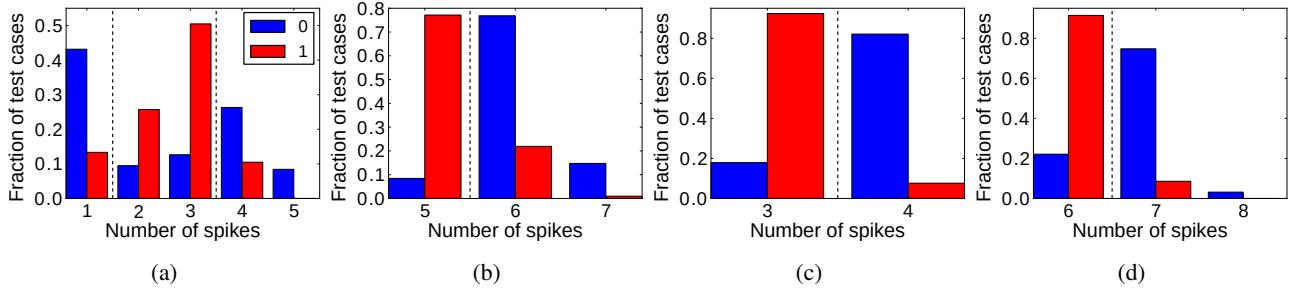


Fig. 9. Distribution of a summed number of excitations, appearing in a single simulation at the output droplet in classifier evolved for 200-case dataset. The height of bars is normalized to the number of instances belonging to a given output class (0 or 1)

test cases shown in Fig. 6. Here, the fitness value after 500 generations was highest for the 5×5 network (0.37 bit at generation 443). However, for a larger dataset the solution did not stabilize within the selected number of generations except for the smallest network. It suggests that longer evolution runs should be applied to optimize the classifiers. In the case of 4×4 network the fitness function has not reached its plateau yet.

The distribution of mutual information and illumination pattern for the networks evolved for dataset containing 200 and 400 cases are shown in Figs. 7 and 8 respectively. For $N = 200$ the classification problem was easier due to a lower number of test cases and the corresponding classifiers were optimized faster than those evolved for $N = 400$ case dataset. For $N = 200$ the optimized classifiers, except the smallest one, contained input droplets for all coordinates, thus they take into account values of all predictors. Usually for both dataset sizes there is just one droplet for which the fitness value is much larger than for the others. The exception is Fig. 7c case as there are three droplets containing a very similar amount of mutual information. It is also interesting to note that the geometry of input droplets and the output ones obtained for 4×4 and 5×5 networks are similar. Remarkably for $N = 200$, only in 2×2 network, an input droplet was selected as the output, whereas for all larger classifiers normal type droplets were the output ones. On the contrary, for $N = 400$, the input droplet for the first predictor (in 1) is always selected as the output (cf. Fig. 8). Such configuration does not seem optimal since the output droplet is dominated

by its own input signal. We expect that the optimized classifier should rather promote integration of signal incoming from many inputs as shown in Figs. 7b-d. In these cases the amount of mutual information (visible as the size of the red sector) was much smaller for individual input droplets than the value of signal integrated at the output.

Allowed optimization of 5×5 network has reached the local plateau (cf. Fig. 6d), but the classifier is missing input droplet for the second predictor (in 2). Therefore the evolutionary optimization has to proceed longer to achieve a physically acceptable result and include all coordinates.

The distribution of a summed number of excitations, for the best found networks are shown in Figs. 9 and 10 for dataset containing 200 and 400 cases, respectively. In such form the signal can be easily interpreted and used to build classification rules. Here we constructed the rules according to the majority principle, i.e. for a given number of excitations we selected the rule that classifies more cases correctly. For example, for the network shown in Fig. 9d the following rule was constructed:

$$\begin{cases} S \leq 6 \rightarrow 1, \\ S > 6 \rightarrow 0, \end{cases} \quad (5)$$

where S is the summed number of excitations at the output droplet. Note that in all cases the distribution of number of excitations can be divided into two or three parts by applying one or two threshold values (marked with dashed lines on the plots). A rule based on two thresholds, constructed

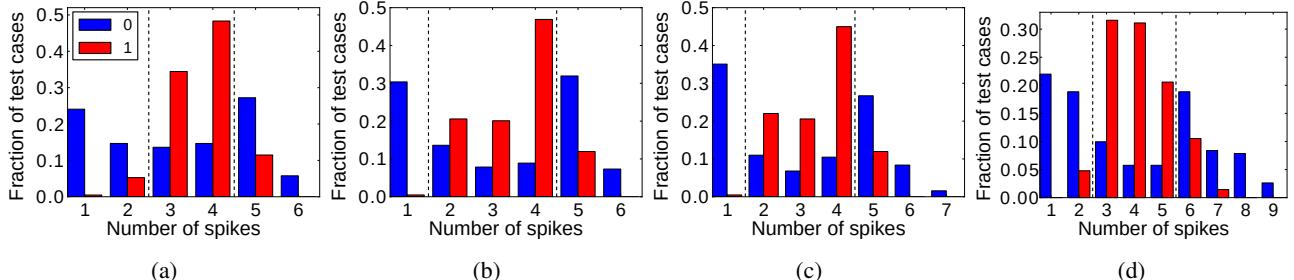


Fig. 10. Distribution of a summed excitations number at the output droplet in classifier evolved for 400-case dataset. Notation as in Fig. 9

for the number of excitations distribution shown in Fig. 9a, can be formulated as:

$$\begin{cases} S \in \langle 2, 3 \rangle \rightarrow 1, \\ S < 2 \vee S > 3 \rightarrow 0. \end{cases} \quad (6)$$

Rules for both datasets and all analyzed network sizes are given in Tab. 1 along with the classification accuracies they provided. For a smaller dataset the best accuracy obtained was 87.5%, using 4×4 droplet network. In case of a large dataset it was 81%, obtained with the largest 5×5 network. Yet, in this case one can expect further improvement if the system is longer evolved. For all studied network sizes we observed correlation between the amount of mutual information and the classification accuracy. Even though we used simple rule constructions the obtained classification accuracies are high. That suggests that classification functionality is learned by the networks during the evolution process rather than implemented later as an intricate set of rules selected by a person interpreting the results.

IV. CONCLUSIONS

In the paper we presented the idea of using evolutionary algorithms for the design of chemical classifiers based on BZ droplet network. The classifier was supposed to predict if a point from a unit cube belongs to a centrally located sphere. Chemical interactions between the nodes of network were adjusted by external illumination to achieve the best match with a database. We considered two databases of a different size and chemical media characterized by various numbers of interacting nodes. Surprisingly, even a relatively small classifier made of 16 droplets can solve the considered geometrical problem with over 85% of classification accuracy. It is hard to expect that such a small system can recognize 3 dimensional geometrical objects yet it was able to determine quite accurately the location of a point with respect to a sphere. It demonstrates the flexibility of classifier design operating on relatively simple nonlinear chemistry.

We believe that classifiers working on chemical processes find applications where chemistry-based information processing seems especially important. One field of applications includes smart drugs with chemical classifiers built into their surrounding capsule. If there are warnings about infection in their neighborhood, then the classifier generates an output signal that opens the capsule and releases the drug. Chemical engineering is another field of potential applications for the classification computers. Many catalytic reactions proceed at high temperatures, where silicon technology does not work. It is known that some catalytic reactions are highly nonlinear and show oscillations and excitability needed for information processing strategies described in this paper.

Acknowledgment

The work was supported by the Polish National Science Centre grant UMO-2014/15/B/ST4/04954.

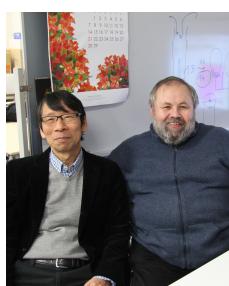
References

- [1] Gordon E. Moore, *Cramming More Components Onto Integrated Circuits*, Proc. IEEE **86**(1), 82–85, (1998), For recent data see for example http://en.wikipedia.org/wiki/Moore's_law.
- [2] T. Gramss, S. Bornholdt, M. Gross, M. Mitchell, and T. Pellegrini, *Non-Standard Computation: Molecular Computation – Cellular Automata - Evolutionary Algorithms - Quantum Computers*, Wiley-VCH Verlag GmbH & Co. KGaA, 2005.
- [3] Cris Calude and G. Paun, *Computing with cells and atoms: an introduction to quantum, DNA and membrane computing*, CRC Press, 2000.
- [4] A. Adamatzky, Ben De L. Costello, and T. Asai, *Reaction-Diffusion Computers*, Amsterdam, Netherlands: Elsevier Science Ltd., 2005.
- [5] A. Adamatzky, L. Bull, and B. De L. Costello, *Unconventional computing*, L. Press, 2007.
- [6] Y. Suzuki, M. Hagiya, H. Umeo, and A. Adamatzky, *Natural Computing: 2nd International Workshop on Natural C. Nagoya, Japan, December 2007, Proceedings*, volume 1, Springer Science & B. Media, 2008.
- [7] J. Von Neumann, *John von Neumann: selected letters*, volume 27, American Mathematical Soc., 2005.
- [8] H. Haken, *Brain dynamics(synchronisation and activity patterns in pulse-coupled neural nets with delays and noise)*, Springer series in synergetics, (2002).
- [9] A.N. Zaikin, A.M. Zhabotinsky, *Concentration Wave Propagation in Two-dimensional Liquid-phase Self-oscillating System*, Nature **225**(5232), 535–537, (1970).
- [10] I.R. Epstein, J.A. Pojman, *An introduction to nonlinear chemical dynamics: oscillations, waves, patterns, and chaos*, Oxford University Press N. York, 1998.
- [11] J. Lindsay, Richard J. Field, and M. Burger, *Oscillations and traveling waves in chemical systems*, Wiley, 1985.
- [12] L. Kuhnert, *A new optical photochemical memory device in a light-sensitive chemical active medium*, Nature, **319**, 393, (1986).
- [13] J. Górecki, J.N. Gorecka, Computing in Geometrical Constrained Excitable C. Systems, In Robert A. Meyers, editor, *Encyclopedia of Complexity and Systems Science*, pages 1352–1376 Springer N. York, 2009.
- [14] Á. Tóth, K. Showalter, *Logic gates in excitable media*, J. Chem. Phys. **103**(6), 2058–2066, (1995).
- [15] O. Steinbock, Á. Tóth, K. Showalter, *Navigating complex labyrinths: optimal paths from chemical waves*, Science **267**(5199), 868, (1995).
- [16] O. Steinbock, P. Kettunen, and K. Showalter, *Chemical Wave Logic Gates*, J. Phys. Chem. **100**(49), 18970–18975, (1996).
- [17] I. Motoike, K. Yoshikawa, *Information operations with an excitable field*, Phys. Rev. E **59**(5), 5354, (1999).
- [18] J. Sielewiesiuk, J. Górecki, *Logical functions of a cross junction of excitable chemical media*, J. Phys. Chem. A **105**(35), 8189–8195, (2001).
- [19] A. Adamatzky, B. De L. Costello, *Experimental logical gates in a reaction-diffusion medium: The XOR gate and beyond*, Phys. Rev. E **66**(4), 046112, (2002).

- [20] K. Yoshikawa, I. Motoike, T. Ichino, T. Yamaguchi, Y. Igarashi, J. Gorecki, and Joanna N. Gorecka, *Basic Information Processing Operations with Pulses of Excitation in a Reaction-Diffusion System*, Int. J. Unconv. Comput., **5**, 3–37, (2009).
- [21] J. Gorecki, J.N. Gorecka, *Information Processing with Chemical Excitations—from Instant Machines to an Artificial Chemical Brain*, Int. J. Unconv. Comput. **2**(4), (2006).
- [22] V.K. Vanag, I.R. Epstein, *Pattern formation in a tunable medium: The Belousov-Zhabotinsky reaction in an aerosol OT microemulsion*, Phys. Rev. Lett. **87**(22), 228301, (2001).
- [23] J. Szymanski, Joanna N. Gorecka, Y. Igarashi, K. Gizynski, J. Gorecki, Klaus-P. Zauner, and Maurits De Planque, *Droplets with information processing ability*, Int. J. Unconv. Comput. **7**(3), 185–200, (2011).
- [24] P.H. King, G. Jones, H. Morgan, M.R.R. de Planque, and K.-P. Zauner, *Interdroplet bilayer arrays in millifluidic droplet traps from 3D-printed moulds*, Lab Chip **14**(4), 722–729, (2014).
- [25] G. Gruenert, K. Gizynski, G. Escuela, B. Ibrahim, J. Gorecki, P. Dittrich, *Understanding networks of computing chemical droplet neurons based on information flow.*, Int. J. Neur. Syst. **25**(7), 1450032, (2015).
- [26] David B Fogel, Lawrence J Fogel, and VW Porto, *Evolving neural networks*, Biol. Cybern. **63**(6), 487–493, (1990).
- [27] D. Cliff, I. Harvey, and P. Husbands, Incremental evolution of neural network architectures for adaptive behaviour, In M. Verleysen, editor, *European Symposium on Artificial Neural Networks (ESANN'93)*, pages 39–44, Brussels, 1993 D Facto.
- [28] X. Yao, Y. Liu, *A new evolutionary system for evolving artificial neural networks*, IEEE Trans. Neural Netw. **8**(3), 694–713, (1997).
- [29] C.E. Shannon, *A mathematical theory of communication*, Bell Syst. Tech. J., **27**, 379–423 and 623–656, (1948).
- [30] G. Gruenert, J. Szymanski, J. Holley, G. Escuela, A. Diem, B. Ibrahim, A. Adamatzky, J. Gorecki, and P. Dittrich, *Multiscale Modelling of Computers Made from Excitable Chemical Droplets*, Int. J. Unconv. Comput. **9**(3-4), 237–266, (2013).
- [31] J. Gorecki, J. Szymanski, and Joanna N. Gorecka, *Realistic Parameters for Simple Models of the Belousov–Zhabotinsky Reaction*, J. Phys. Chem. A **115**(32), 8855–8859, (2011).
- [32] R. Bradbury, M. Nagao, *Effect of charge on the mechanical properties of surfactant bilayers*, Soft Matter **12**(46), 9383–9390, (2016).
- [33] A. Gadomski, P. Beldowski, L. Martínez-Balbuena, I. Santamaría-Holek, and Z. Pawlak, *Unravelling a Self-healing Thermo- and Hydrodynamic Mechanism of Transient Pore's Late-stage Closing in Vesicles, and Related Soft-matter Systems, in Terms of Liaison Between Surface-tension and Bending Effects*, Acta. Phys. Pol. B **47**(5), 1341–1356, (2016).
- [34] H.-G. Beyer, H.-P. Schwefel, *Evolution strategies – A comprehensive introduction*, Nat. Comput. **1**(1), 3–52, (2002).
- [35] H.-P. Schwefel, *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie: mit einer vergleichenden Einführung in die Hill-Climbing- und Zufallsstrategie*, Basel, Switzerland: Birkhäuser, 1977.
- [36] H.-P. Schwefel, *Numerical Optimization of Computer Models*, John Wiley & Sons, Inc., N. York, NY, 1981.
- [37] D.E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, first edition, 1989.
- [38] D. Thurber, *Catfish Are Off the Hook After Tokyo Ends 16-Year Earthquake* P. Study, <http://articles.latimes.com/1992-04-26/news/mn-1403\1\major-earthquake>, 1992, LA Times.



Konrad Giżyński (PhD degree in chemistry, 2015) works as Assistant Professor at the Department of Complex Systems and Chemical Processing of Information at the Institute of Physical Chemistry of the Polish Academy of Sciences in Warsaw. His research interests cover experimental and numerical studies on self-organization phenomena, computer simulations of non-equilibrium systems and unconventional computing.



Jerzy Górecki is Full Professor at the Institute of Physical Chemistry of the Polish Academy of Sciences in Warsaw. He has been leader of the Department of Complex Systems and Chemical Processing of Information since 2008. His research interests cover theory and computer simulations of far-from-equilibrium phenomena in chemical systems, different types of non-equilibrium self-organization and unconventional computation. His recent papers are mainly concerned with strategies of information processing using a spatially distributed reaction-diffusion medium. He used to work as Professor at the Institute of Molecular Science in Okazaki, Japan, and continues research projects with many laboratories in Japan. The photograph shows Jerzy Górecki (right) with his old friend professor Kenichi Yoshikawa at Doshisha University in Kyoto.