# Combining Biochemical Network Motifs within an ARN-Agent Control System

Claire E. Gerrard, John McCall, Christopher Macleod,
IDEAS Research Institute,
Robert Gordon University,
Aberdeen, Scotland.
c.e.gerrard@rgu.ac.uk, j.mccall@rgu.ac.uk,
chris.macleod@rgu.ac.uk

George M. Coghill,
Department of Computing Science
University of Aberdeen
Aberdeen, Scotland.
g.coghill@abdn.ac.uk

*Abstract*—**The Artificial Reaction Network (ARN) is an Artificial Chemistry representation inspired by cell signaling networks. The ARN has previously been applied to the simulation of the chemotaxis pathway of *Escherichia coli* and to the control of limbed robots. In this paper we discuss the design of an ARN control system composed of a combination of network motifs found in actual biochemical networks. Using this control system we create multiple cell-like autonomous agents capable of coordinating all aspects of their behavior, recognizing environmental patterns and communicating with other agent's stigmergically. The agents are applied to simulate two phases of the life cycle of *Dictyostelium discoideum*: vegetative and aggregation phase including the transition. The results of the simulation show that the ARN is well suited for construction of biochemical regulatory networks. Furthermore, it is a powerful tool for modeling multi agent systems such as a population of amoebae or bacterial colony.**

*Keywords— Artificial Reaction Networks; Artificial Chemistry; Swarm Agents*

## I. INTRODUCTION

Artificial Chemistry (A-Chem) is a subfield of A-Life and, in its broadest sense, it describes man-made systems which are similar to real chemical systems [1]. Chemical information processing has many desirable properties, it is: decentralized asynchronous, fault tolerant, evolvable, self-organizing and adaptive [1]. A-Chem focuses on harnessing these properties by either creating Molecular Computing devices where computation is achieved using either real chemicals or by utilizing the principles of the chemical metaphor to construct novel software or hardware architectures *in silico* [1]. The latter is termed Artificial Chemistry Computing (CCM) and is the focus of this paper. In the chemical metaphor, data is stored in the form of molecular species and information processing occurs through interactions (reactions) between these molecules. The result of this computation emerges from the numerous low-level interactions and appears as a global behavior [1]. ACC is used in two main applications: simulating complex systems (biological, social or ecological) and in developing novel solutions to engineering or computational problems. The Artificial Reaction Network (ARN) is an ACC representation inspired by the properties and mechanisms of information processing found in biological Cell Signaling Networks (CSNs). In our previous work, it was applied to simulate the chemotaxis signaling pathway of *E. coli* [2], and later investigated as a means to produce complex temporal waveforms to control limbed robots [3,4].

Within a cell, data is represented by a set of spatially distributed molecular concentrations; CSNs process this information within elaborate hierarchical network control structures which connect species together in productive or inhibitory unions. In this way, cells are able to respond to changes within their environment, communicate with other cells, and perform internal self maintenance operations [5]. The ability of chemical networks to perform computational processing is well documented. For example it has been shown both theoretically and in wet lab experiments that such networks can perform Boolean and Fuzzy logic functions [1,5]. A number of researchers have identified structural motifs in such biochemical networks which can form basic computational processing units [5,6].

The aim of this paper is to show that the ARN is a powerful modeling tool and can produce realistic approximations of the complex network circuitry that exists within and between cells. This ARN is firstly used to construct real biochemical network regulatory motifs. These motifs are then combined to create a control system for an autonomous ARN-agent. The control system implements a set of cell-like behaviors which allow the agent to recognize and respond to environmental patterns by modifying its trajectory. A swarm of ARN-agents are then instantiated within an artificial environment and used to simulate the collective behavior of a population of *D. discoideum* (Dd) cells throughout two phases of the organism's life cycle: vegetative and aggregative.

The paper is structured as follows: Section II provides a summary of the ARN representation; this is followed by a discussion about network motifs and the ARN-agent control system in Section III and IV. The experimental details are discussed in Section V followed by results in Section VI. Finally, Section VII presents the conclusions.

## II. ARTIFICIAL REACTION NETWORKS

A summary of the ARN is provided here. Verification and further discussion of the ARN representation can be found in our previous work [2-4]. The ARN comprises a set of networked reaction nodes (circles), pools (squares), and inputs (triangles) and is depicted as a directed weighted graph as shown in Fig. 1 Each pool stores a current available chemical species concentration (*avail*); thus, the complete set of pool concentrations at time t, corresponds to the current state of the system.

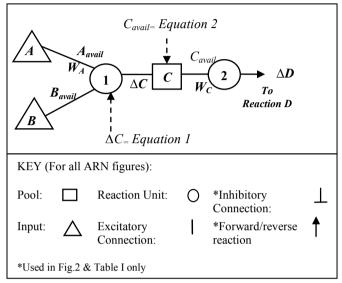

Fig. 1.   The Artificial Reaction Network representation.

$$\Delta C = \left(K_{f(C)}\left(A_{avail}^{W_A} B_{avail}^{W_B}\right) - K_{r(C)}\left(C_{avail}^{W_C}\right)\right)\Delta t \tag{1}$$

$$C_{avail} = C_{avail} + \left(\Delta C - \frac{W_C}{W_C + \alpha}\Delta D\right) \tag{2}$$

$$C_{avail} = C_{avail} + \left(\Delta C - \Delta D W_C\right) \tag{3}$$

Where:
$A, B, C, D$ = Species Concentrations
$W$ = Reaction order (weight)
*avail* = Available species concentration
$K_f$ = Forward rate constant
$\Delta C$ = Change in species concentration C
$K_r$ = Reverse rate constant
$\alpha$ = sum of other incoming weights to D

In this paper and the following example we use conserved mass quantities, however, the ARN may also be used to model non-conserved quantities such as the number of available molecules. The choice affects the way in which each pool is updated and is discussed later. While many ACs assume a well-stirred reactor, the use of pools within the ARN provides a discrete spatial structure. Inside a biological cell concentrations of chemical species are spatially distributed into localized compartments. This localization restricts which molecules may react together, thus affects the overall dynamics of the system. Representing the spatial distribution of chemicals allows fine grained control over the system dynamics and thus is highly beneficial when modeling biochemical circuitry. For example it allows the representation of flow structures such as membrane channels, transport processes; network motifs, and provides a means to explore disease pathways [7]. Thus, in the ARN, each pool is represented as a well-stirred reactor and approximates a spatial compartment. Inputs are a special type of pool; the only difference is that they are of fixed value and thus can be used to represent the continuous flow of environmental inputs or enzymes. Each circle corresponds to a reaction unit, representing a reaction between a number of chemicals. Data is processed by reaction nodes transforming incoming pool values to connected outgoing pool values. Connections symbolize the flow of chemical into and out of reaction units and their weight (*W*) corresponds to reaction order. Connections provide the facility to create complex control structures using combinations of inhibitory and excitatory connections. Fig. 1 shows the reaction between species *A* and *B* to produce species *C*. At time interval Δt, each reaction unit's temporal flux value is calculated by applying Euler's approximation to the differential rate equation as shown in (1). This value is then used to update the current concentration of each reaction's connecting pools as shown in (2). A reaction step may proceed if it meets its preconditions. Preconditions are met if incoming inhibitory pools are inactive, and incoming excitatory pools are active. In a similar way the completion of a reaction step will fulfill a number of post conditions, which depend on the parameters and connections of the reaction step. Pools may asymptotically approach 0, and thus below a particular threshold a pool is considered empty and its value set to zero.

As previously mentioned the ARN is not restricted to representing conserved quantities such as mass, one may choose to model the exchange of molecules between pools. Thus using the example in Fig. 1, to generate one molecule of product *C* requires $W_A$ molecules of reactant *A* and $W_B$ molecules of reactant *B*. In this case pool *C* is updated by applying (3).

## III. BACKGROUND THEORY

The ARN presented in Section IV is composed of regulatory motifs found in real biochemical networks. These motifs are combined in such as way to provide the control system for an ARN-agent. In this Section a summary of two behavioral modes implemented within the ARN-agent control system are described. This is followed by the structure and function of common network motifs, including biological examples.

### A. Foraging and Starvation Modes of ARN-agents

Dd is a cellular slime mould which exists as a collection of amoeba and transitions to a multicellular slug during the aggregation phase of its life cycle. During its vegetative stage, amoebae move up gradients of folic acid (FA) secreted by its bacterial prey. The behavior of these unpolarised cells is characterized as a random biased walk where cells extend random pseudopods in a biased manner toward the source of FA resulting in overall movement up the gradient of FA [8]. Dd amoebae begin to starve when the food resource has been depleted, and begin the aggregation phase. During

aggregation, starving cells secrete cAMP (cyclic adenosine monophosphate) which serves as a signal to attract surrounding amoebae towards a central location [9]. Aggregating Dd cells are polarized, thus one side becomes the leading edge which always faces in the direction of travel [9, 10]. Depending on parameters such as environmental conditions, and Dd population density, migrating cells often form transient emergent patterns such as streams, waves and spirals [9, 10]. Streaming describes a pattern of motion where cells line up in close order files, with the head of one following the rear of another [10].

The agent performs two behavior modes- foraging and starvation based respectively on vegetative and aggregative behaviors of Dd amoebae. During the foraging phase, agent's alternate periods of forward motion termed "runs" and random redirections called "tumbles". The bias is provided by reducing the period of tumbles when moving up the food gradient. At each passing location food is consumed. The agents enter starvation mode if food has not been consumed within a time period. Each starving agent emits a continuous signal of cAMP into its surrounding environment. Instead of turning in a random direction, agents turn in the direction weighted toward the highest concentration of cAMP within its surrounding area.

### B. Functional Motifs In Biochemical Networks

Cell membranes are studded with receptors which are sensitive to external parameters such as chemicals, pH, temperature and light. These receptors are responsible for detecting and transducing environmental signals. These signals trigger information processing events within the CSN which update cell activity such as changes in gene expression.

Components of CSNs are linked through productive unions (union of reactants triggers production/activation of other components) and inhibitory unions (union of two reactants inhibits production/activation). In the same way the network is separated through productive and inhibitory isolations (union does not occur or does not produce any effect). Such links arrange chemicals into elaborate circuitry which functions as the information processing machine of the cell.

A fundamental challenge in Molecular Biology is to understand such signal processing and thus enable the prediction of the effects of disease and intervention of pharmaceuticals. To this end a number of researchers have identified functional structural motifs within these networks [5, 6]. A summary of the structure (in ARN format), function and biological examples of a number of the most common motifs is provided in Table 1. A more detailed discussion is provided by Tyson [6]. Note that these motifs are shown for simplicity as 2 or 3 component but there are larger versions with the same function. For example, an additional component may be added to motif 9 to create a 4 component oscillator.

### IV. ARN-AGENT CONTROL SYSTEM

In this Section we discuss an ARN control system composed of the structural motifs in Table 1. This system is designed to control an autonomous ARN-agent termed a "Cytobot" ("cyto" from Greek for cell, and "bot" from robot).

The cytobot has two behavior modes: foraging and starvation, based on the previously described behavior of Dd amoeba during its respective vegetative and aggregative phases. In the experiments described in Section V a number of these cytobots are instantiated and interact to produce global emergent behavior. The relationship between the ARN control system and the cytobot is similar to that of a CSN to an amoeba. Thus the control system allows a cytobot to recognize environmental patterns, updating its trajectory within an artificial 2D environment and to communicate stigmergically with other cytobots. The environment contains a distribution of artificial chemicals. These chemicals represent attractants of either food or cAMP. When a cytobot moves to a new position, the surrounding level of chemical is used to set the inputs to its ARN. Consequently this changes the internal state of the ARN and updates the agent's trajectory. During this process, the agent modifies the state of the environment by consuming food or releasing cAMP.

The cytobot ARN is composed of 6 subnetworks as shown in Fig. 2. Each subnetwork contributes a functional aspect to either or both starvation and foraging behaviors. The design of the subnetworks is discussed in the following Sections.

### A. The Master Oscillator

The Master Oscillator functions to synchronize the outputs from all the other subnetworks together and is what each cytobot references at each time step to ascertain its current behavior. It is a 4 component oscillator (Table 1 motif 9) with a token unit of chemical cycling around it. It consists of 4 reaction units: M0, M1, M2, and M3 (all with reaction rate of 1) and 4 pools MA, MB, MC and MD and generates a pulsed width modulated waveform. Each pool is associated with 1 of 3 behaviors. Every time step that a particular pool contains the token unit, its corresponding behavior is performed. Pool MA activates turn, MC activates run and MB and MD activate stop. Thus, if pool MC contains a chemical for 10 time steps, the agent will move forward for 10 time steps. Note that this motif could control motor actuators on a simple wheeled robot: MC would switch on all wheel motors, while MA would switch on left wheel motors only, thus turning the robot. The remaining pools would act as off switches. The other subnetworks inhibit (motif 2) or excite (motif 1) the reaction units of the master oscillator to allow or prevent chemical flow.

### B. The Food and Run Length Network

The food network interfaces with the environment at pool FA using an excitatory connection (motif 1) and inhibits the runlength network in accordance with the level of detected food. The forward rate of reaction node F0 is 1, thus the content of FA is transferred to pool FB in a single time step. The presence of chemical FB inhibits (motif 2) R0 for a number of time steps according to the level of food (by setting forward rate of unit F1 to 1 and weight to 0 this can be an exact correlation). The run length network is a 3 component oscillator (motif 9). While reaction R0 is inhibited it prevents pool RC from emptying. RC inhibits reaction M2 (motif 2) of the master oscillator thus preventing pool MC from emptying for the same number of time steps. As discussed previously,

the number of time steps which pool MC contains the token unit represents the number of time steps to move forward.

## C. The Signalling Network

The signaling network functions as a switch between starvation and foraging mode. Low food levels trigger the starvation response and allow the weighted direction network to control each new angle. Sufficient food will switch off the weighted direction network and allow the chaotic network to control each new angle. It is a 3 component oscillator (motif 9) with a token unit of chemical flowing around it. Pool CA acts as a switch between foraging and starvation behavior.

TABLE I EXAMPLES OF FUNCTIONAL MOTIFS FOUND IN BIOCHEMICAL NETWORKS

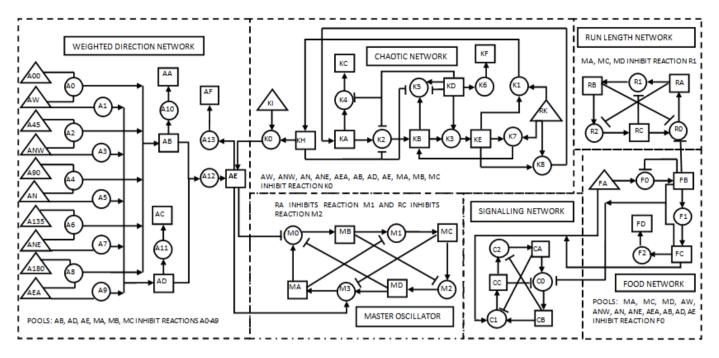| Motif No., Name and Description | Structure (in ARN format) | Biological Example |
|---|---|---|
| **1. Excitatory (E)** <br> The presence of X activates Y |  | Elementary motif common throughout pathways. E.g. Ras is a membrane associated protein that is normally activated in response to the binding of extracellular signals such as growth factors [6]. |
| **2. Inhibitory (Y)** <br> The presense of X inhibits Y. Acts as a NOT gate. |  | Elementary motif common throughout pathways. E.g. E-cadherin (a calcium-dependent cell–cell adhesion molecule) suppresses cellular transformation by inhibiting β-catenin [6]. |
| **3. Positive Feedback Loop (PFL)** <br> The presence of X activates Y and in turn the presence of Y activates X |  | The pathway of caspase activation is essential for apoptosis induction. A PFL exists between caspase-3 and caspase-9 [6]. |
| **4. Negative Feedback Loop (NFL)** <br> The presence of X activates Y and in turn the presence of Y inhibits X |  | The proteins Mdm2 and p53 (p53 is a tumor suppressor protein) are involved in a NFL which functions to keep the level of p53 low in the absence of p53-stabilizing signals [6]. |
| **5. Double-negative Feedback (DNF)** <br> The presence of X inhibits Y and the presence of Y inhibits X |  | BAX is protein which promotes apoptosis by competing with BCL. A DNF is formed between the proteins BAX and BCL [6]. |
| **6. Branch (B)** <br> The presence of X activates Y and Z |  | Transcription factors such as E2F or P53 frequently modulate the expression of more than one gene. Enzymes often modify more than one substrate e.g. CycB-dependant kinase [6]. |
| **7. Logic Gate (LG1)** <br> **AND gate**: 2 excitatory connections from X and Y when both X and Y are present they activate Z <br> **NOR gate**: two inhibitory connections from X and Y. Both X and Y must be absent for Z to be activated <br> **SWITCH:** Excitatory connection from X and inhibitory connection from Y. The presence of X but not Y activates Z |  | AND: The protein gCam 2 kinase becomes active when both calcium ions ($Ca^{2+}$) and Calmodulin (CaM) are present [5]. <br> NOR: The activity of transcription factor E2F is a NOR function of RB and CycB where E2F is active when both RB and CycB are inactive [6]. <br> SWITCH: The enzyme aspartate transcarbamylase is composed of multiple catalytic sites. It is activated by binding of its substrates (aspartate and carbamoyl phosphate) and inactivated by cytidine triphosphate causing its substrates to dissociate [5]. |
| **8. Logic Gate (LG2)** <br> **OR Gate:** : 2 excitatory connections from X and Y when either X or Y are present they activate |  | Ras is a membrane associated protein that is activated by a number of different signals. E.g. in response to the binding of extracellular signals such as a number of growth factors [6]. |
| **9. Oscillator (OSC)** <br> The presence of X activates Y. In turn the presence of Y activates Z but inhibits X. The presence of Z inhibits Y and activates X. |  | There are many examples e.g. in the cyanobacteria clock protein KaiC has a well defined closed cycle of phosphorylation and dephosphorylation states (composed of KaiA, KaiB and KaiC). In the motif shown here, all 3 components oscillate and each inhibits the reaction clockwise left. Oscillators may have less inhibitory connections, the number of which is dependent on the mobility of the reaction species. However, the presence of all inhibitors increases stability in the presence of fluctuating environmental parameters e.g. temperature. Note that this oscillator can also be thought of as a PFL (motif 3) combined with a system of DNFs (motif 5). |
| Key: ──────◆ | | Either inhibitory or excitatory.  X/Y/Z: Chemical species * Note that these motifs may combine arbitrary numbers of components. |

Fig. 2. The cytobot ARN network. Each cytobot is controlled by an instance of this network. The network is composed of 6 subnetworks

Here the presence of chemical in CA inhibits the weighted direction network (motif 2) while its absence switches on the weighted direction network; this in turn inhibits the chaotic network, as shown in Fig. 2. In this oscillatory network, all reaction units have a forward flux of 0.5; which produces a continuous oscillating waveform and ensures a minimum number of time steps for each behavior. A NOR gate (motif 7) activates pool CB in the absence of food chemical in both pools FB and FC of the food network, thus allowing pool CB to empty. While an AND gate (motif 7) will lead to pool CA to eventually refill by activating pool CC only when food is present in input FA and pool FC of the food network.

*D. The Weighted Direction Network*

The weighted direction network senses food within the agents' immediate environment and calculates a tumble angle which is weighted toward higher food levels. This network interfaces with the environment via a number of receptor pools (AW, ANW, AN, ANE, AEA) which sense the level of food around the cytobot. These pools represent receptors and are positioned at points around the front of its perimeter (as shown in Fig. 3), allowing the agent to travel in a similar way to that of a polarized Dd cell. For example, during the aggregation phase of their life cycle, Dd cells are polarized, and one side becomes the leading edge which always faces in the direction of travel [10]. Each receptor input pool forms one input of an AND gate (motif 7), the other input is a static pool containing a fixed level of chemical in correspondence to its direction. Directions start from AW (west) with a corresponding numeric value of 0 (A00) and progress in 45 degree steps through each direction to east. As the receptor positions around the agent are fixed, directions are always relative to that in which the agent is facing. All connections have a weight of 1 with the exception

of the connection between pool AD and reaction A12 which has a weight of -1. This negative connection raises the sum of food detected in AD to -1, which multiplied by AB, allows an average angle to be calculated.
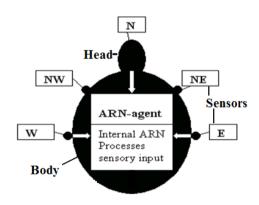


Fig. 3. Location of the ARN-agent (cytobot) sensors around its perimeter.

The calculated angle interfaces with the remaining subnetworks at pool AE. Pool AE is the output of an OR gate (motif 8) where inputs from either the weighted direction network or the chaotic angle network activate AE. AE also forms the inhibitory input of a SWITCH (motif 7) where the presence of chemical in MA and absence in AE activates pool MB of the master oscillator. In an actual organism receptors are set around the cell perimeter and direct movement appropriately. In this simulation, for simplicity, a count of the number of time steps "n" that MA contains the token unit is processed to gain the new heading "h" relative to the agents' current heading "c" using (4). Thus if the number time steps is

120 and the agent is facing north, then the current heading would equal 0 and the new heading would equal 30.

$$h = ((n-90)+c) \bmod 360 \qquad (4)$$

*E. The Chaotic Network*

The chaotic network, as shown in Fig. 2, is responsible for generating pseudo random angles which agents use to perform the foraging mode tumble behavior. It is a networked implementation of a Logistic Map, see (5). Without prior knowledge of the initial conditions the output of the logistic map is unpredictable, while it is deterministic with prior knowledge. Therefore, the series cannot be described as truly random but as pseudo random. Its output has long been proposed as a pseudo-random number generator [11] and it has been successfully used in this capacity by several researchers [12]. The probability density distribution of the Logistic Map is non-uniform and is described in [12]. When λ=4 the distribution is "U" shaped, with higher probability of values closer to the minima and maxima of X and a symmetric distribution at the midpoint. The general shape of the distribution is invariant for the complete range of state variables from 0 to 1.

$$X_{n+1} = \lambda X_n (1 - X_n) \qquad (5)$$

Where:
$X_n$= state variable of value $0 \leq X_n \leq 1$
$\lambda$= system parameter of value $1 \leq \lambda \leq 4$

To implement the logistic map a number of motifs are combined including multiple branch motifs (motif 6- KB activates KD and KE), PFLs (motif 3- a multi component PFL exists where KA leads to activation of KE, which results in the activation of KA) and NFLs (motif 4- KA activates KD which in turn inhibits KA). At the start of the simulation, pools KA and KB are initialized to the same random value (a unique number for each cytobot) between 0 and 1 (to 5 decimal places). This represents the first value of X of (5). All the other pools are initialized to 0 with the exception of the static pools KI and RK whose initial values are 360 and 1 respectively. Reaction K2 is responsible for generating each new value of X and has a forward and reverse rate of 4 (the logistic map exhibits chaotic behavior when λ is 4). The connection between KA and K2 has a weight of 1 and that between K2 and KB has a weight of 2. The remaining series of reactions function to copy the value of X 3 times, where 2 copies serve as the new initial values of KA and KB and one participates in the final output of the network at KH. KI has a fixed value of 360 which allows the network to convert the pseudo random number at KH to an angle value between 0 and 360 at reaction K0. However, reaction K0 cannot proceed until all 11 pools that inhibit it are empty. These inhibitory connections (motif 2) ensure that random angles are not output while the agent is in starvation mode, and that pool AE is empty before adding more chemical. The ARN implementation of the Logistic Map was tested against the recursive relation (5), the details of which are given in our previous work [13]. The frequency distribution gained from the ARN is identical to that of (5).

V. METHODOLOGY

In the following experiments, multiple cytobots are instantiated and used to model aggregating and vegetative Dd cells, where each cytobot represents a cell. Two sets of experiments are performed: aggregation (AG) and foraging to aggregation (AGF). In the AG experiments (AG1-10 of Table 2) only the aggregative phase is modeled, where each experiment is performed at varying population densities of cytobots (p) and different ranges of detection of cAMP (r). In experiments AGF3 and AGF8 the vegetative and aggregative phases are simulated (and the transition between these phases) using the same population density and range as experiments AG3 and AG8 respectively. The emergent patterns, numbers of mounds, and length of time to complete phases is examined and compared in both sets of experiments.

In the AGF experiments, the environment is initialized with a radially outward decreasing gradient of food as described by (6), where x and y are on the horizontal plane. Here the cytobots remain in foraging mode until the food resource is depleted and starvation mode is triggered. In a real environment food is non-uniformly distributed, may be regenerated and can move (in the case of bacterial prey). Thus, this setup is highly simplified, but is comparable to other simulations [14].

$$z = \sqrt{x^2 + y^2} \qquad (6)$$

The results of the AGF experiments are compared with those of the AG experiments, where, the environment never contains food, thus agents immediately enter and stay in the starvation mode. The agents' behavior is initially explored at biologically realistic p and r values and compared with the behavior of the actual organism and other simulations. These parameters are then extended outwith the biological range in order to examine the emergent properties of the system. The Cytobots move within a simulated 2D environment of area 5.06 mm$^2$- approximately half the maximum recorded aggregation territory [9]. Each pixel represents 4.5 μm and the grid is 500 × 500 pixels, giving a total area of 5.06 mm$^2$. In nature, aggregating Dd cell densities are typically 250 per mm$^2$ to $1 \times 10^4$ per mm$^2$ [9]. Due to the computational resources required to manage a population of cytobots within the upper range, two cell densities of 250 agents per mm$^2$ (1250 agents) and 150 per mm$^2$ (750 agents) were chosen. In all experiments, the agents are initialized at random positions within the simulated environment. Foraging cytobots consume food at each passing location, while those in starving mode emit a cAMP signal at equal strength around their circumference into the environment. This cAMP signal is detected by other agents within or equal to r. If an agent in foraging mode detects cAMP it will switch to starvation mode behavior.

In these experiments a range of r values are explored (see Table 2), including that of real cells of 1, 0.5, and 0.1 mm [8].

The actual cAMP signal degrades linearly with increasing distance (d) from the emitting cell. Each agent detects the cAMP signal of all starving cells within or equal to r, and a total value for each direction is calculated. Each cycle represents 1 minute of time. In this time the agent moves 9μm- a distance which corresponds to that reported in the literature [15]. Therefore, after 1 hour motion the agent travels a distance of 540μm. In reality there are always remaining cells that do not aggregate, and thus the simulation runs until 95% of agents are at a distance of less than 0.1mm from their nearest neighbor.

## VI. RESULTS

The results for all 12 experiments are given in Table 2. Each experiment was performed 100 times. In experiments AG8-10 and AGF8 the value of r and d are within the ranges for real Dd cells. These experiments are used to compare the behaviors and aggregation time with the values for real Dd in the literature. In experiments AG8-9 and AFG8 mound formation completes within the range reported for the actual organism of 9-13 hours [14, 16]. These results are comparable with other work. For instance, in [14] the aggregation time reported was 11.6 hours for a cell density of $200mm^2$. In experiment AG10, the population never satisfied the criteria for completion of mound formation. The likely explanation is firstly because the simulation does take into account glycoprotein's which allow aggregating cells to attach together on contact. Furthermore, because r is small, fewer agents are detected by each cytobot. Thus higher numbers of momentarily larger clusters with higher attraction strength go undetected and quickly dissipate. In the AG experiments increasing p by $100mm^2$ the number of mounds formed at each r decrease with the exception of experiment AG6. This is not surprising, as denser populations should have more chance of interacting, and thus form fewer clusters, but with higher numbers of agents. Similarly, decreasing r results in a general increase to the number of mounds formed at both values of p. In the AGF experiments, agents generally focus on consuming food in each remaining highest concentration area (see Fig. 4K-L). Having consumed almost all the food, agents begin switching to starvation mode (Fig. 4M). In these experiments the number and location of resulting mounds differs from that of the AG experiments at the same values of r and d. For example experiment AG8 results in 4.3 mounds while AGF8 results in 6.8 mounds with a general shift in mound formation further away from the centre of the environment (as shown in Fig. 4).
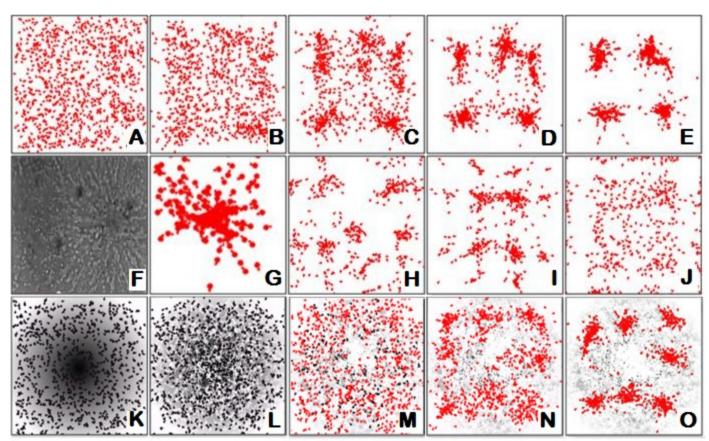


Fig. 4. Screenshots of the Dd simulation. Dots represent the cytobots (black- vegetative and red- aggregative cells), and greyscale color represents the food distribution. A-E: Cytobot aggregation experiment AG8 at A- 1hr, B- 2hr, C-5hr, D- 8hr, E- 10hr; Image F- real Dd cells aggregating; G- Lower right hand corner of image C demonstrating streaming behavior; H-J Shows pattern formation; K-O Cytobot experiment AGF8 at K-0hr vegetation, L-4hr vegetation, M-transition to aggregation 0hr aggregation, N-5hrs aggregation, O-10hr aggregation.

The likely explanation for this is that, at the time of switching to aggregation, the majority of cells had been forced outward, toward the next remaining highest concentration of food. Emergent behaviors and clustering patterns similar to the biological organism were also observed. As previously discussed, the cytobots are polarized. Implementing the agents in this way allowed us to observe whether or not the previously described streaming behavior occurs. A close-up of the right-hand corner of screenshot C is shown in Fig. 4G showing agents beginning to form a cluster. The protruding head of each agent can be seen clearly, where each lines up its head to the rear of another agent and forms a stream. As can be seen in Fig. 4F, this is very similar to the streaming behavior in real cells of Dd. Other emergent patterns occurred during different experiments including spirals (Fig. 4H), symmetric patterns (Fig. 4I), and waves (Fig. 4J).

TABLE II  CYTOBOT SIMULATION RESULTS

| No. | Density (p) per $mm^2$ | Range (r) in mm | Mean No. of mounds; ($\sigma$) | Aggregation Phase Mean time in Hours; ($\sigma$); *Literature 9-13 hours |
|---|---|---|---|---|
| AG1 | 150 | 5 | 1 (0) | 8.98 (0.09) |
| AG2 | 150 | 2.5 | 4 (0.31) | 9.63 (0.17) |
| AGF3 | 150 | 1 | 5.9 (1.16) | 9.47 (0.65) |
| AG3 | 150 | 1 | 5.2 (0.82) | 9.92 (0.34) |
| AG4 | 150 | 0.5 | 8.4 (1.19) | 10.23 (0.59) |
| AG5 | 150 | 0.1 | 14.2 (2.36) | 10.6 (1.82) |
| AG6 | 250 | 5 | 1 (0) | 8.95 (0.11) |
| AG7 | 250 | 2.5 | 1 (0) | 9.6 (0.20) |
| AGF8 | 250 | 1 | 6.8 (1.81) | 9.71 (0.87) |
| AG8 | 250 | 1 | 4.3 (0.37) | 10.05 (0.58) |
| AG9 | 250 | 0.5 | 6.7 (1.62) | 12.65 (1.94) |
| AG10 | 250 | 0.1 | - | - |

## VII.  CONCLUSIONS

The results of the Dd experiments presented above show that ARN-agents are able to simulate individual behaviors, stigmergic interactions and emergent behaviors of unicellular organisms by combining structural motifs found in real biochemical networks. This highlights a potential use as a means to simulate groups of interacting cells - such as a bacterial colony or tissue component within a multicellular organism, including the effects of disease (e.g. faulty gene expression) and pharmaceuticals on global behavior. The results demonstrate the parallels between ARN agents and the biological counterpart from which they were inspired. Like amoebae, their internal network of spatially distributed dynamic chemical species allows them to autonomously coordinate and direct their movement, recognize and respond to patterns in the environment, and produce high-level behavior. This application demands an internal control system which can function without reference to other agents within the environment which are operating in parallel.

In future work, it is intended to further investigate ARN-agents in biological simulations. Importantly a study into ways in which the pathways of ARN-agents can be evolved and how such agents can learn and adapt to the environment autonomously.

REFERENCES

[1] P. Dittrich, J. Zeigler, and W. Banzhaf, "Artificial Chemistries- a reivew". Artifi. Life vol. 7, no. 3, pp. 225-275, 2001.

[2] C. E. Gerrard, J. McCall, G. M, Coghill, and C. Macleod, "Artificial Reaction Networks," Proceedings of the 11th UK Workshop on Computational Intelligence, Manchester, UK, pp. 20-26, September 2011.

[3] C. E. Gerrard, J. McCall, G. M, Coghill, and C. Macleod, "Temporal patterns in Artificial Reaction Networks," Proceedings of The 22nd International Conference on Artificial Neural Networks Lausanne, part 1, vol. 7552, pp. 1-8, September 2012.

[4] C. E. Gerrard, J. McCall, G. M. Coghill, and C. Macleod, "Adaptive Dynamic Control of Quadrupedal Robotic gaits with Artificial Reaction Networks," Proceedings of The 19th International Conference on Neural Information Processing Doha, vol. 7663, part 1, pp. 280-287, November 2012.

[5] D. Bray. "Protein molecules as computational elements in living cells," Nature, vol. 376, no. 6538, pp. 307-12, 1995.

[6] J. J. Tyson, and B. Novák. "Functional motifs in biochemical reaction networks." Annu. Rev. Phys. Chem., vol. 61, pp. 219-240, 2010.

[7] P. Kreyssig, P Dittrich, "Reaction flow artificial chemistries," Advances in Artificial Life, ECAL 2011: Proceedings of the Eleventh European Conference on the Synthesis and Simulation of Living Systems, pp. 431-437. MIT Press, 2011.

[8] C. P McCann, P. W. Kriebel, C. A. Parent, W. Losert, "Cell speed, persistence and information transmission during signal relay and collective migration," J. Cell. Sci., vol. 123, pp. 1724-1731, 2010.

[9] J. C. Dallon, and H. G. Othmer, "A discrete cell model with adaptive signaling for aggregation of Dictyostelium discoideum," Phil. Trans. R. Soc. B, vol. 352, no. 1351, pp. 391–417, 1997.

[10] R. H. Kessin, "Making Streams,"Nature, vol. 422, pp. 481-482, 2003.

[11] S. M. Ulam and J. von Neumann, "On combinations of stochastic and deterministic processes," Bull. Amer. Math. Soc, vol. 53, pp. 1120, 1947.

[12] V. Patidar, K. K. Sud, and N. K Pareek, "A Pseudo Random Bit Generator Based on Chaotic Logistic Map and its Statistical Testing," Informatica, vol. 33, pp. 441-452, 2009.

[13] C. E. Gerrard, J. McCall, G. Coghill and C. Macleod. "Artificial Chemistry Approach to Solving Search Spaces using Artificial Reaction Network Agents", Presented at the IEEE Congress in Evolutionary Computing, Cancún, Mexico, 2013, in press.

[14] M. Becker. "Simulation model for the whole life cycle of slime mould Dictyostelium Discoideum," Proceedings of the European conference on modeling and simulation, pp. 247-253, 2010.

[15] J. L. Rifkin. and R. R. Goldberg. "Effects of chemoattractant pteridines upon speed of D. discoideum vegetative amoeba. Cell Motility and the Cytoskeleton," vol. 63 no.1, pp 1-5, 2006.

[16] D. A. Cotter, T. W. Sands, K. J. Virdy, M. J. North, G. Klein, M. Satre, "Patterning of development in Dictyostelium discoideum: factors regulating growth, differentiation, spore dormancy and germination," Biochem. Cell Biol., vol. 70, no. 10-11, pp. 892-919, 1992.