



A P system model of swarming and aggregation in a Myxobacterial colony

Anthony Nash¹ · Sara Kalvala²

Received: 14 February 2019 / Accepted: 24 April 2019 / Published online: 10 May 2019
© The Author(s) 2019

Abstract

Bacterial communities provide an interesting subject for the study of emergence and complexity as the consequence of many local interactions. In particular, the soil-dwelling social bacterium Myxobacteria demonstrates two distinct types of motility, social motility via the sensing of bacterial slime deposits and adventurous motility. Both modes of motility are governed by local interactions. Using P systems, a membrane computing methodology based on compartmental rewrite rules for modelling computational processes; this work demonstrates how minimal set of rules can model swarming and aggregating behaviour in Myxobacteria bacterial populations. Our model uses a multi-environment P system similar a 2D cellular automaton to represent the substrate environment whilst stochastic rule selection dictates Myxobacterial motion according to behaviour observed in vitro. The rules account for both mechanisms of motility, the deposit and detection of slime, a change in direction due to C-signal induction and the mixing of population numbers. Simulations demonstrate an extensible computational framework for the modelling of bacterial behaviour, with the potential for extension into additional emergent behaviours.

Keywords P systems · Multi-environment P systems · Myxobacteria · Membrane computing · Nature inspired computing · Emergence

1 Introduction

Bacterial populations provide an interesting subject for the study of emergence and complexity phenomena. Bacteria react to the environment in complex ways and use short-distance local communication to start coordinated behaviour that can involve hundreds of thousands of cells. Tissues in multicellular organisms, on the other hand, are often controlled centrally via hormones and other signals. The Gram-negative soil-dwelling bacterium, Myxobacteria, best known for population level rippling and fruiting body stages whilst under starving conditions, provides an interesting and extensively studied vehicle for understanding cellular communication and spatial behaviour in bacteria [1–7]. In particular, phenomena such as swarming, aggregation and rippling in the Myxobacteria *Myxococcus xanthus* continue to elude

researchers in their understanding of the mechanisms by which Myxobacteria communicate and organise themselves to predate prey bacteria [8].

Myxobacteria are able to move over a substrate using slow gliding motions driven by two modes of regulated transport, adventurous (A-motility) and social (S-motility) [7, 9, 10]. Bacteria at the colony edge or under isolation can undergo an adventurous mode of motility through the secretion of slime to propel the organism forward. Through cell-to-cell contact-signal (C-signal) transduction, bacterium undergoes reversal of the intracellular motility mechanism [11]. After a number of hours, the accumulation of C-signals across a population will cause an oscillatory motion within the population, giving a visual impression of a diffusing wave. Although it appears as though colliding waves pass through one another, the head-to-head impact of bacterium causes the wave to reverse resulting in very little net displacement. Under S-motility the cell relies on the extension and retraction of type IV pili: these extend from the leading cell pole and grip the polysaccharide cell surface or solid substrate surface. Both type IV pili-based motility and slime extrusion-based motility

✉ Anthony Nash
anthony.nash@dpag.ox.ac.uk

¹ Department of Physiology, Anatomy and Genetics,
University of Oxford, Oxford, UK

² Department of Computer Science, University of Warwick,
Coventry, UK

require the Frz pathway to induce forward movement in the direction of the long axis of the cell.

P systems are distributed parallel computability models based on cellular membrane compartmentalisation and diffusing chemical signals within and across membranes [12]. A system is defined within a unique outer membrane known as the skin that can hold any number of membrane structures, similar to the organelles inside a living cell. Each membrane has a cellular region which can hold a multiset of objects defined by some alphabet and a set of rewrite rules that enables the multiset of objects to evolve under the correct conditions. Objects can diffuse across membranes analogous to cellular signals, and membranes can dissolve passing their chemical contents onto the parent membrane region. Rules can be assigned a priority rating, giving the set of rules for a single membrane a specific order of execution similar to the downstream transduction of receptor activation. If a high-priority rule cannot execute under current conditions, then any additional rules are tested. The priority of rules is typically denoted as $r_1 > r_2$, to indicate that the first rule has a higher execution priority over the second rule. Additionally, the prioritising of rules can be assigned using a stochastic constant [13]. These stochastic P systems use the Gillespie algorithm to determine which rule to execute and the order.

The evolution of a P system is performed in parallel across all cellular regions similar to the parallel operation of the organelles within the cell and the chemical operations within the main cytoplasmic space itself. Operations are performed either according to the order of each rule, a stochastic selection process, or a mixture of the two. P systems can generally be divided into two categories: the first, non-cooperative, where only single objects can evolve; and the second, cooperative, where multiple objects can evolve in parallel [14]. If every membrane is incapable of evolving over its multiset, the system is said to halt. Upon halting, values can be read from a specific membrane region as defined in the P system formal definition. Under such conditions, the P system is said to be computationally complete, yet unlike the model proposed in this study, these systems would be used to generate a value or a decision rather than spatial behaviour.

Due to inherent compartmentalisation of a P systems model and their encapsulation of chemical diffusion across membranes, they have proved very apt in modelling Quorum sensing [15–19]. The spatial dynamics of diffusing signals can be captured in a grid-like P system, known as a multi-environment, similar to that of a cellular automaton [20]. Further works of modelling biological systems using membrane computing can be seen in Cardona et al. [21, 22], which look at the dynamics of ecosystems. To address the inherent randomness and uncertainty in large

biological systems, multi-environment P systems have been combined with probabilistic terms [23, 24].

We explore the use of the P system formalism to increase our understanding of spatial phenomena in Myxobacteria. Recent extensions to the P system formalism allow simulating the models with a stochastic engine, which is crucial to capture realistic biochemical scenarios [25]. While the spatial behaviour of Myxobacteria is controlled primarily by a non-diffusible contact-based signal, we explore commonalities between this form of communication and chemical quorum sensing which underlies other modelling work using P systems.

2 Methods

The methods proposed in this study begin with a brief description of the formal definition for a P system; the reader is advised to refer to the work of Păun et al. [26]. The P system construct is then described in light of an extension to incorporate a two-dimensional spatial construct. The execution of such a multi-environment P system is covered with a brief description of the Gillespie algorithm. Finally, the two-dimensional model and Myxobacterial rewrite rules are reviewed.

2.1 P system formal definition

P systems can be graphically represented as Venn diagrams and trees. Venn diagrams help illustrate the membrane structure in respects to the physiology of a real biological cell, in addition to showing the multiset of objects and membrane rules.

A P system containing n cellular regions delimited by membranes is defined formally as

$$\Phi_n = (\Sigma, \mu, w_1, \dots, w_n, (R_1, p_1), \dots, (R_n, p_n), i_0),$$

where

- Σ is an alphabet of objects that populate a multiset;
- μ defines the membrane structures including the skin. The labels for the membranes are one-to-one with the number of defined regions. For example, $[\square_2[\square_4]_3]_1$ defines a P system Φ_4 ;
- w_i defines the initial multiset content of each membrane, as indicated by the respective label. The multiset for each region is defined as the set $w_i \in \Sigma^*$;
- (R_i, p_i) refers to the rules and rule priority, where $0 \leq p_i \leq 1$, for the membrane region i . If there is only one rule for a given membrane region the priority is set to 0. Multiple rules for the same region require that each rule be given a priority;

- i_0 defines the output region which should be read only when the system halts.

2.2 Multi-environment P system

A multi-environment P system represents a spatial structure using a grid of membrane regions connected by links. It has the same eight directional properties as a regular two-dimensional cellular automaton using a Moore neighbourhood. The formal definition as presented by Romero-Campero et al. [19]:

$$\text{multi_environment} = (\Sigma, H, G, E_{1,1}, \dots, E_{x,y}, R_{1,1}, \dots, R_{x,y}, \Phi),$$

which we present as a Myxobacterial environment:

- $\Sigma = (\text{slime}, \dots)$ is an alphabet used to construct the contents of a multi-environment region. The alphabet contains chemical signals such as slime and lysing antibiotic chemicals;
- H is a finite set of labels used to distinguish each multi-environment region. We use the Cartesian ordered pair $X \times Y = \{(i, j) : i \in X \text{ and } j \in Y\}$ to layout each multi-environment region similar to the coordinates of a two-dimensional cellular automata;
- $G = (V, S)$ is a graph with nodes $V = ((1, 1), \dots, (x, y))$ represent the multi-environment regions and the edges S define the links between regions. Mycobacteria P systems are able to traverse the multi-environment P system by the link. This is an alternative notation to the P system μ , which is used to outline a membrane structure;
- $E_{i,j} = (w_{i,j}, R_{i,j}, \varphi_n)$, for each environment region $(1, 1) \leq (i, j) \leq (x, y)$, $E_{i,j}$ is the initial configuration of the region. $w_{i,j} \in \Sigma^*$ is a multiset of characters over the alphabet Σ and $R_{i,j}$ defines a set of rules and $\{\varphi_n \in \Phi : 1 \leq n \leq 8\}$ defines a number of bacterial P systems within that multi-environment P system cell, where each bacterial P system can adopt up to one of the eight possible directions according to the underlying Moore neighbourhood arrangement of the multi-environment grid structure;
- $R_{i,j}$ is a finite set of rule for each multi-environment regions. Rule priority is defined on a stochastic basis, using the likelihood that a rule will be executed given certain environmental constraints;
- Φ is the inclusion of n Myxobacterial P systems within this particular multi-environment region. A single Myxobacterial P system is denoted by one of eight directions defined by a Moore neighbourhood. The sum of Myxobacteria present in a multi-environment P system is defined by the sum of k (see below) across the eight possible Myxobacterial P systems present at any one time.

2.3 Simulating P system models

The chemical master equation gives the probability of obtaining a state S at time t from an initial set of conditions [25]. Yet, even in short simulations, this form of stochastic modelling can be very computationally expensive [27]. An alternative means of modelling stochastic processes introduced by Gillespie [28] updates the state of a system using probability to determine the order and time for rule execution. The collections of trajectories are updated after the execution of each rule to reflect the change to any of the system variables. In this work, the Gillespie algorithm is modified to calculate the likelihood of each rule being executed in parallel on a constant time step (a unitless value $\Delta t = 1$). If a region of the multi-environment P system is not populated with a Myxobacterial P system, the stochastic values for that region are set to zero. The algorithm provides a simple yet elegant means of using environmental properties, such as the volume of a biological space and the probability of chemical collision.

2.4 Myxobacterial P system

The structure of a Myxobacterial P system can be defined as

$$\varphi = (\Gamma, \mu, L, E),$$

where

- Γ is an alphabet of multiset elements within the P system region;
- μ represents the structure of a bacterium P system, that is, $[\]_l$ where $l \in L$;
- L is the set $\{\text{myxo}\}$ of labels used on the membrane;
- E represents the initial configuration of the Myxobacterial P system. Each Myxobacterial P system contains a direction, a means of motility and a bacteria population, defined as n, m and k for the purpose of rule presented in this work.

The configuration of a single Myxobacterial P system is defined by a tuple:

$$\Gamma :: \langle N, D, M \rangle,$$

where

- N , where $1 \leq N \leq M$, defines the quantity of Myxobacteria in a P system and M denoted the maximum number of Myxobacterial cells;
- $D = \{N, NE, E, SE, S, SW, W, NW\}$ specifies the direction in which a P system of bacteria faces;
- $M = \{A_m, S_m\}$ specifies the motility engine of the cell.

As with the structure of cellular compartmentalisation of a regular P system, the use of multi-environment model in this study provide cellular space for the inclusion of a Myxobacterial P system and the connection between multi-environment regions allows the diffusion of Myxobacterial P systems between multi-environment regions. For example,

$$[[17, E, -]_{\text{myxo}}]_{i,j} - []_{i+1,j} \xrightarrow{c_i} []'_{i,j} - [[17, E, -]_{\text{myxo}}]'_{i+1,j},$$

signifies the movement (either A-motility or S-motility, as indicated by the wildcard $-$) of a Myxobacteria P system (with a population of seventeen bacterium and orientated toward E) from a multi-environment region at location i, j , into the adjacent multi-environment region $i + 1, j$. The superscript c_i , where $0 \leq c_i \leq 1$, on the rule transformation refers to the stochastic likelihood of this rule executing. A primed region indicates a change in cellular configuration having executed a rewrite rule.

3 Results and discussion

We begin by describing the behaviour of Myxobacteria as a set of multi-component P system rewrite rules. These rules are then explored further via a demonstration of typical motility, complete S-motility, and complete A-motility from the initial conditions of a bacteria colony. Finally, we see in closer detail the behaviour of Myxobacteria in the presence of slime upon the substrate.

3.1 Stochastic rules

The set of P system rewrite rules are projected over the cellular region of the multi-environment P system and not the Myxobacterial P systems themselves (see Table 1 for the complete list). The list of multi-environment P system rules given here acts as examples of possible cellular states that rewrite rules can operate over the included Myxobacterial P systems. Cellular components of Myxobacteria can change, such as direction and motility, leading to additional emergent behaviour. Rule priority is first determined by whether a Myxobacterial P system is in a suitable state or position. Then a random value $[0, 1]$ is calculated and compared to the position of available rules when respective rule priorities are combined and normalised between $[0, 1]$. The availability of some arbitrary n rules implies that $\sum_{i=1}^n c_i = 1$.

The movement of the Myxobacteria is presented in rules r_1 through to r_6 : r_1 uses social motility to move a population of Myxobacteria into an empty neighbouring cell; r_2 applies to Myxobacteria using adventurous motility to propel itself forward into an empty neighbour whilst depositing slime; r_{3a} , r_{3b} and r_{3c} allows the population of Myxobacteria to detect slime perpendicular to its long axis, followed by

movement into the slime whilst itself depositing slime; r_{4a} , r_{4b} and r_{4c} allows slime detection on an angle; and r_5 allows the mixing of a population of Myxobacteria from one neighbouring cell into another, in either motility modes.

The movement of Myxobacteria from one region into another populated region is determined by social motility movement and cell number mixing, perpendicular or on an angle to the moving P system in question, is controlled using r_{6a} , r_{6b} and r_{6c} .

Collision between populations of Myxobacteria can result in a switch in motility mechanism, from adventurous into social as per C-signal induced reversal (r_7).

As the quantity of adventurous Myxobacteria occupy a single region increases there is an ever-greater likelihood of becoming socially mobile, and therefore joining any social Myxobacteria within that region (as per r_8). The probability of switching motility takes on a rate similar in form to a ligand associating with a substrate as per the Michaelis–Menten equation, and in which a surplus of Myxobacterial asymptotes the probability of switching. On the other hand, if the region is sparsely populated with social Myxobacteria, they can switch back into adventurous motility, as seen in r_9 . Myxobacterium can undergo spontaneous reversal via a switch to the internal cellular motility mechanism (r_{10}). This causes the Myxobacteria to spend a lot of its solitary, swarming and aggregating life moving along its long axis.

A simplified multi-component P system at time t , as presented in Fig. 1, illustrates some of the rules used to evolve into the configuration at $t + 1$. Each square represents a multi-environment region whose edges define the membrane of the region. Links between each membrane are used to pass Myxobacterial P systems between regions. This system is evolved by the execution of multi-environment P systems as outlined in Table 1. Myxobacteria in region $i + 1, j + 1$ have moved into region $i, j + 1$ using slime detection. The population of Myxobacteria in region $i, j - 1$ spontaneously reversed direction from facing south to facing north. Population $i - 1, j$ attempts to invade the populated region i, j . By doing so, it causes a head–head collision, thus activating the C-signal pathway, causing a reversal in both P systems. During this time, the population of $i + 1, j$ moved in besides the population of i, j .

3.2 Simulation examples

We present simulations of a Myxobacterial population as a colony. Default stochastic values are presented in Table 2 and draft Java code can be downloaded from <https://github.com/acnash/Psystems>. Video files of the simulations are available as online supplementary material. Each Myxobacterial P system was populated from a normal distribution with at most 400. In each region, the Myxobacteria were randomly assigned an initial direction.

Table 1 The list of multi-component P system rules

Rule ID	Rule	Description
Social motility		
r_1	$[n, E, S_m]_{myxo} \rightarrow [n, E, S_m]_{i+1,j} \xrightarrow{c1} [n, E, S_m]_{i,j}' - [n, E, S_m]_{i+1,j}'$	S-motility
Adventurous motility		
r_2	$[n, E, A_m]_{myxo} \rightarrow [n, E, A_m]_{i+1,j} \xrightarrow{c2} [slime]_{i,j}' - [n, E, A_m]_{i+1,j}'$	A-motility leave slime behind
Adventurous motility into slime		
r_{3a}	$[n, E, A_m]_{myxo} \rightarrow [n, E, A_m]_{i+1,j} \xrightarrow{c3a} [slime]_{i,j}' - [n, E, A_m]_{i+1,j}'$	A-motility into slime
r_{3b}	$[n, E, A_m]_{myxo} \rightarrow [n, E, A_m]_{i+1,j-1} \xrightarrow{c3b} [slime]_{i,j}' - [n, E, A_m]_{i+1,j-1}'$	A-motility into slime
r_{3c}	$[n, E, A_m]_{myxo} \rightarrow [n, E, A_m]_{i+1,j+1} \xrightarrow{c3c} [slime]_{i,j}' - [n, E, A_m]_{i+1,j+1}'$	A-motility into slime
r_{4a}	$[n, NE, A_m]_{myxo} \rightarrow [n, NE, A_m]_{i,j} \xrightarrow{c4a} [slime]_{i,j-1}' - [n, NE, A_m]_{i,j-1}'$	A-motility into and with slime
r_{4b}	$[n, NE, A_m]_{myxo} \rightarrow [n, NE, A_m]_{i+1,j-1} \xrightarrow{c4b} [slime]_{i,j}' - [n, NE, A_m]_{i+1,j-1}'$	A-motility into and with slime
r_{4c}	$[n, NE, A_m]_{myxo} \rightarrow [n, NE, A_m]_{i+1,j} \xrightarrow{c4c} [slime]_{i,j}' - [n, E, A_m]_{i+1,j}'$	A-motility into and with slime
Bacteria mixing		
r_5	$[n, E, A_m]_{myxo} \rightarrow [m, W, -]_{myxo} \xrightarrow{c5} [(n-k), E, AM]_{i,j}' - [(k, E, A_m)_{myxo} [m, W, -]_{myxo}]_{i+1,j}'$	A-motility mixing into neighbour
r_{6a}	$[n, E, S_m]_{myxo} \rightarrow [m, E, S_m]_{i+1,j} \xrightarrow{c6a} [(n-k), E, S_m]_{i,j}' - [(m+k), E, S_m]_{i+1,j}'$	S-motility into neigh- bour
r_{6b}	$[n, E, S_m]_{myxo} \rightarrow [m, NE, S_m]_{i+1,j-1} \xrightarrow{c6b} [(n-k), SE, S_m]_{i,j}' - [(m+k), NE, S_m]_{i+1,j-1}'$	S-motility into neigh- bour
r_{6c}	$[n, E, S_m]_{myxo} \rightarrow [m, NE, S_m]_{i+1,j+1} \xrightarrow{c6c} [(n-k), SE, S_m]_{i,j}' - [(m+k), NE, S_m]_{i+1,j+1}'$	S-motility into neigh- bour
Direction and motility switching		
r_7	$[n, E, -]_{myxo} \rightarrow [m, W, -]_{i+1,j} \xrightarrow{c7} [n, W, -]_{i,j}' - [m, E, -]_{i+1,j}'$	C-signal reversal
r_8	$[n, -, A_m]_{myxo} \xrightarrow{c8} [n, -, S_m]_{i,j}'$	A-motility to S-motility switch
r_9	$[n, -, S_m]_{myxo} \xrightarrow{c9} [n, -, A_m]_{i,j}'$	S-motility to A-motility switch
r_{10}	$[n, E, -]_{myxo} \xrightarrow{c10} [n, W, -]_{i,j}'$	Spontaneous reversal

A complete list would comprise of all directions

3.2.1 Simulation examples of a Myxobacterial colony

A colony of up to 40,000 Myxobacteria was constructed about a 40×40 multi-component P system substrate. The

time step was a single evaluation of rules across the system and the simulation came to an end after 250 time steps had been evaluated. Three simulations were performed: first, a colony of Myxobacteria traversing the substrate under

Fig. 1 An illustrative representation of a multi-environment P system. Each square is a membrane region. Each tuple represents a Myxobacterial population P system

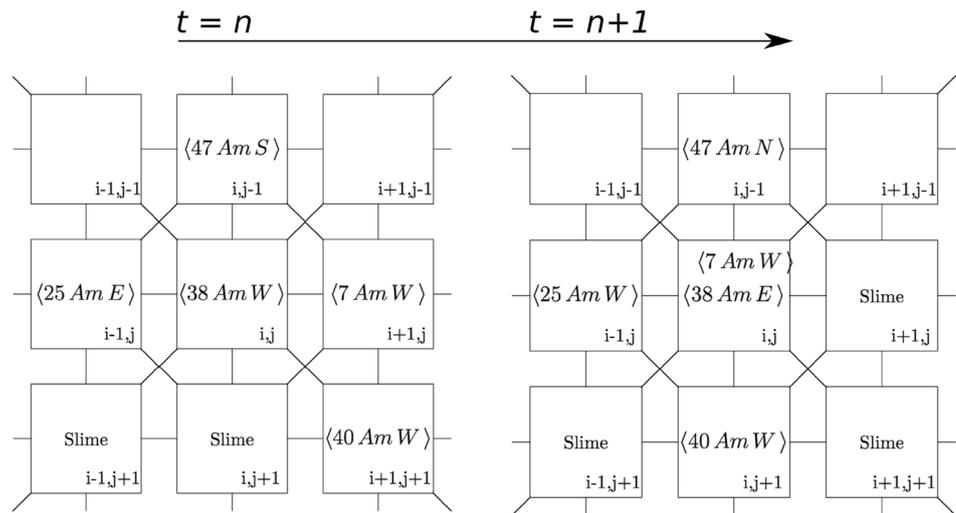


Table 2 Default values for the stochastic selection of rules. Stochastic values are retrieved for valid rules (cellular conditions are correct), arranged descending by size and transformed onto the range (0, 1) for random selection

Rule identifier	Stochastic value	Rule identifier	Stochastic value
r_1	0.05	r_5	0.8
r_2	5	r_{6a}	0.05
r_{3a}	5	r_{6b}	0.05
r_{3b}	5	r_{6c}	0.05
r_{3c}	5	r_7	0.05
r_{4a}	5	r_8	0.05
r_{4b}	5	r_9	0.02
r_{4c}	5	r_{10}	0.05

the influence of social and adventurous motility; second, a colony of Myxobacteria acting under the influence of adventurous behaviour; third, acting under the influence of social coordinated motility. The quantity of Myxobacteria at each multi-component P system region is represented by

the integer values as specified in the Myxobacterial P system definition and is reflected by the peak high in Figs. 2, 3 and 4.

During the early simulation steps of a Myxobacteria colony capable of social and adventurous exploration (Fig. 2), the Myxobacterial P system populations at the colony periphery orientated out towards the empty adjacent multi-component P system regions. Those Myxobacterial P systems within the heart of the colony would have potentially undergone a number of C-signal reversals from head-to-head collisions with immediate neighbours (r_7), spontaneous reversals (r_{10}) or exchange of Myxobacterial sub-populations between neighbouring Myxobacterial P systems (r_{6a} , r_{6b} and r_{6c}). Driven by the adventurous motility (r_2) of the periphery Myxobacterial P systems, immediate neighbours were able to detect local deposits of slime encouraging movement away from the colony. As the initial colony structure reduces, some of those adventurous and free-exploring Myxobacteria switched to S-motility (r_9), enabling neighbouring S-motility cells

Fig. 2 Simulation trajectory snapshots at 50 time intervals. S-motility and A-motility are enabled

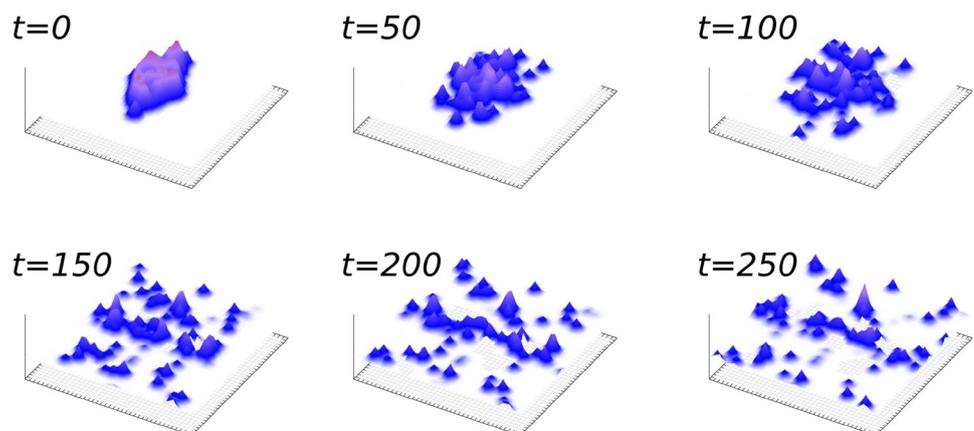


Fig. 3 Simulation trajectory snapshots at 50 time intervals. The stochastic values for S-motility have been set to 0, to ensure bacterial aggregation under complete adventurous motility

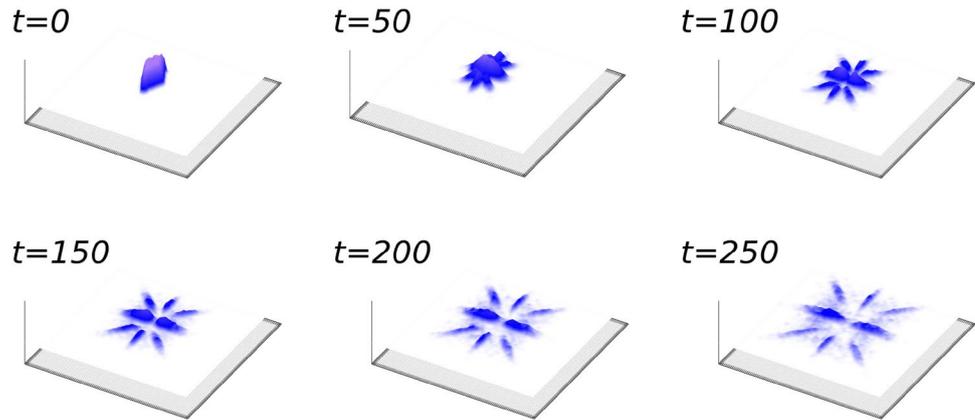
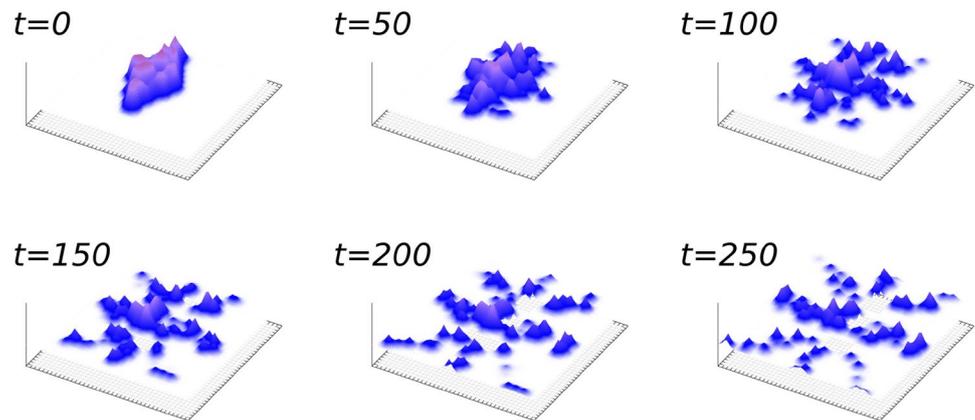


Fig. 4 Simulation trajectory snapshots at 50 time intervals. The stochastic values for A-motility have been set to 0, to ensure bacterial aggregation under complete social motility



to begin mixing and moving amongst themselves (r_{6a} , r_{6b} and r_{6c}). Before too long, the periphery Myxobacterial P systems having broken away from the main colony to setup smaller colonies can switch back into A-motility (r_8) and once again venture out to form additional colonies.

By being able to turn towards the direction of slime deposits (r_{6a} , r_{6b} and r_{6c}), switching direction randomly (r_{10}), and interacting with neighbouring adventurous P systems (r_5), there is a high degree of variability compared to the initial simulation parameters and an increased ability to form further sub-colonies. It is interesting to note, by decreasing the probability of a population switching from A-motility to S-motility, the colony immediately breaks up and begins randomly exploring the environment (data not shown). This is seen when the genes responsible for the Myxobacteria's S-motility are knocked out, preventing the Myxobacteria from gripping onto neighbouring Myxobacteria [29].

A simulation of a Myxobacterial colony using only the A-motility (rules r_2 , r_{3a} , r_{3b} , r_{3c} , r_{4a} , r_{4b} , r_{4c}) rules for movement and removing the ability for the P systems to switch motility was performed for 250 time steps (Fig. 3). Compared to the normal-motility simulation, during the

initial simulation steps the centre of the colony remains intact whilst showing only minor indication of the periphery Myxobacterial P system orientating towards the empty substrate to forward (r_2) into the empty substrate. Without S-motility, the Myxobacterial P systems are unable to maintain a bacterial aggregate, rather they move out into explore the empty substrate (r_2) or follow the slime deposits of Myxobacterial P system directly in front (r_{3a}), or at an angle (r_{3b} , r_{3c}). The eight orientations available can be clearly seen from step 100 to step 250, with a less dense covering of Myxobacteria as populations detected neighbouring slime on the angle off from their long axis.

Similarly, as with A-motility, when the movement of the population has been reduced to only S-motility (r_1 , r_{6a} , r_{6b} , and r_{6c}) (Fig. 4), the eight possible orientations can also be discerned across the Myxobacterial P system population. Without adventurous motility, aggregates of P system Myxobacteria migrate along these orientations with little in the way of break-away Myxobacteria, where each colony having formed due to collisions (r_7) and spontaneous reversal (r_{10}).

To discern spatial differences over time between the three means of motility, the Myxobacterial P System distribution

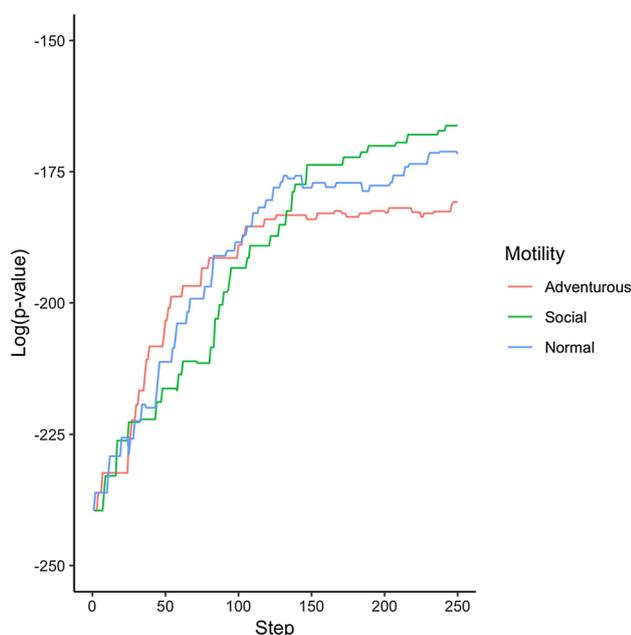


Fig. 5 Log transformation of p values of quadrant counting tests calculations per step

was compared to a uniform Poisson point null model, often called the complete spatial randomness (CRS) model [30]. At each time step, a quadrat count over the multi-environment P system space was calculated. As P system distributions become more random, the quadrat count calculation moves closer to accept the null. Calculations were performed per step over the three motilities and a log transformation of their p value was visualised (Fig. 5). The initial colony structure results in an identical transformed p value over the three models. As each model evolves the transformed p value increases towards the null model ($p > 0.05$). The adventurous motility, as per the visible ordered distribution in Fig. 3, retains the lowest p value. As the simulations evolve, social motility shows the greatest degree of randomness (as per Fig. 4). Combined, the transformed p value of normal motility is neither as random as just social motility nor as ordered as adventurous motility.

4 Conclusion

Using a multi-component P system substrate and a model P system to represent dynamic quantities of Myxobacteria, this work presented simulations of a swarming and aggregating Myxobacterial population. The results demonstrate how both modes of bacterial motility, social and adventurous were necessary to replicate both swarming of bacteria from the colony edge, exploration of the empty substrate and intermittent sub-colony formation. A simulation using only social

behaviour explored the substrate by forming uniformed sub-colonies, yet with no means of acting as solitary agents, the distribution of populations was along the eight orientations. Whereas, adventurous motility only simulations results in Myxobacterial P systems at the colony periphery venturing into the empty substrate and in turn leads to neighbouring P systems breaking from the colony. Yet, with no mixing of P system populations (from social behaviour), had the simulation been left indefinitely, it is quite possible that the complete population would have been distributed equally over the substrate space.

With all languages being Turing equivalent, it is possible to implement a P System model using any language as mathematically, all programming languages are Turing equivalent. However, Java was selected given its native support for encapsulation and inheritance. Object-orientated languages such as Java and C++ lend themselves very well to the paradigm of membrane computing. Although of interest are the efforts to encapsulate P system rules in a formal language [31], and very recently using in vitro techniques to produce a real P system [32].

Despite the swarming and aggregating dynamics observed using only a small set of P system rewrite rules, there are a number of limitations; orientation is limited to the eight adjacent spaces in the multi-environment P system substrate; the time step is a simple count and the execution of rules are determined stochastically with no consideration of temporal dynamics; and, slime remains indefinitely, reducing the A-motility to a random walk as the simulation progresses. Compared with recent particle continuum [33] and continuous [34] based modelling, conventional models of Myxobacteria continue to dominate, chiefly through their ability to model structural properties of bacterium. Finally, there is a degree of abstraction from the traditional cell-line P system paradigm in the way in which the mixing of bacteria is treated. Traditionally, rewrite rules and the diffusion of cell content would be required to combine Myxobacterial populations. However, given the framework presented in this study, further exploration into locally induced global phenomena of bacterial colonies using P systems is entirely possible.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Shimkets, L. J., & Kaiser, D. (1982). Induction of coordinated movement of *Myxococcus xanthus* cells. *Journal of Bacteriology*, *152*, 45–461.
2. Kaiser, D., & Crosby, C. (2005). Cell movement and its coordination in swarms of *Myxococcus xanthus*. *Cytoskeleton*, *3*, 227–245.
3. Igoshin, O., Goldbeter, A., Kaiser, D., & Oster, G. (2004). A biochemical oscillator explains several aspects of *Myxococcus xanthus* behavior during development. *Proceedings of the National Academy of Sciences USA*, *101*, 15760–15765.
4. Sliusarenko, O., Neu, J., Zusman, D. R., & Oster, G. (2006). Accordion waves in *Myxococcus xanthus*. *Proceedings of the National Academy of Sciences USA*, *103*, 1534–1539.
5. Wu, Y., Chen, N., Rissler, M., Jiang, Y., Kaiser, D., & Alber, M. (2006). CA models of Myxobacteria swarming. In S. E. Yacoubi, B. Chopard, & S. Bandini (Eds.), *Cellular automata. ACRI 2006. Lecture notes in computer science* (pp. 192–203). Berlin, Heidelberg: Springer.
6. Welch, R., & Kaiser, D. (2001). Cell behavior in traveling wave patterns of Myxobacteria. *Proceedings of the National Academy of Sciences USA*, *98*, 14907–14912.
7. Wolgemuth, C., Hoiczky, E., Kaiser, D., & Oster, G. (2002). How myxobacteria glide. *Current Biology*, *12*, 369–377.
8. Berleman, J. E., Chumley, T., Cheung, P., & Kirby, J. R. (2006). Rippling is a predatory behavior in *Myxococcus xanthus*. *Journal of Bacteriology*, *188*, 5888–5895.
9. Mauriello, E. M., Yang, T., & Zusman, D. R. (2010). Gliding motility revisited: How do the Myxobacteria move without flagella? *Microbiology and Molecular Biology Reviews*, *74*, 229–249.
10. Muñoz-Dorado, J., Marcos-Torres, F. J., García-Bravo, E., Moraleda-Muñoz, A., & Pérez, J. (2016). Myxobacteria: Moving, killing, feeding, and surviving together. *Frontiers in Microbiology*, *7*, 781.
11. Mignot, T., Merlie, J. P., Jr., & Zusman, D. R. (2005). Regulated pole-to-pole oscillations of a bacterial gliding motility protein. *Science*, *310*, 855–857.
12. Păun, G. (2000). Computing with membranes. *Journal of Computer and System Sciences*, *61*, 108–143.
13. Pu, Y., Yu, Y., Dong, X. (2006). Simulation of biomolecular processes by using stochastic p systems. In: Proc Workshop on High Performance Computing in the Life Sciences. Ouro Preto.
14. Sburlan, D. (2006). Non-cooperative P systems with Priorities Characterize PsETOL. In: Freund R, Lojka G, Oswald M, Păun G, editors. Pre-Proc. of sixth international workshop on membrane computing (WMC6), Vienna, pp. 530–539.
15. Cámara, M. (2006). Quorum sensing: A cell-cell signalling mechanism used to coordinate behavioral changes in bacterial populations. In H. J. Hoogeboom, G. Păun, G. Rozenberg, & A. Salomaa (Eds.), *Membrane Computing. WMC 2006. Lecture notes in computer science* (pp. 42–48). Berlin, Heidelberg: Springer.
16. Bianco, L., Pescini, D., Siepmann, P., Krasnogor, N., Romero-Campero, F. J., & Gheorghe, M. (2006). Towards a P systems *Pseudomonas* quorum sensing model. In H. J. Hoogeboom, G. Păun, G. Rozenberg, & A. Salomaa (Eds.), *Membrane computing. WMC 2006. Lecture notes in computer science* (pp. 197–214). Berlin, Heidelberg: Springer.
17. Bernardini, F., Gheorghe, M., & Krasnogor, N. (2007). Quorum sensing P systems. *Theoretical Computer Science*, *371*, 20–33.
18. Terrazas, G., Krasnogor, N., Gheorghe, M., Bernardini, F., Diggle, S., & Cámara, M. (2005). An environment aware P-system model of quorum sensing. In S. B. Cooper, B. Löwe, & L. Torenvliet (Eds.), *New Computational Paradigms. CiE 2005. Lecture Notes in Computer Science* (pp. 479–485). Berlin, Heidelberg: Springer.
19. Romero-Campero, F. J., & Pérez-Jiménez, M. J. (2008). A model of the quorum sensing system in *Vibrio fischeri* using P systems. *Artificial Life*, *14*, 95–109.
20. Frisco, P., Corne, D.W. (2007). Modeling the dynamics of HIV infection with conformon-p systems and cellular automata. In: Eleftherakis G, Kefalas P, Păun G, Rozenberg G, Salomaa A, editors. *Membrane Computing. WMC 2007. Lecture notes in computer science*, Berlin, Heidelberg, pp. 21–31.
21. Cardona, M., Colomer, M. A., Margalida, A., Pérez-Hurtado, I., Pérez-Jiménez, M. J., & Sanuy, D. (2009). A P system based model of an ecosystem of some scavenger birds. In G. Păun, M. J. Pérez-Jiménez, A. Riscos-Núñez, G. Rozenberg, & A. Salomaa (Eds.), *Membrane computing. WMC 2009. Lecture notes in computer science* (pp. 182–195). Berlin, Heidelberg: Springer.
22. Cardona, M., Colomer, M. A., Pérez-Jiménez, M. J., Sanuy, D., & Margalida, A. (2008). Modeling ecosystems using p systems: The bearded vulture, a case study. In D. W. Corne, P. Frisco, G. Păun, G. Rozenberg, & A. Salomaa (Eds.), *Membrane Computing. WMC 2008. Lecture Notes in Computer Science* (pp. 137–156). Berlin, Heidelberg: Springer.
23. Colomer, M. A., Pérez-Hurtado, I., Pérez-Jiménez, M. J., & Riscos-Núñez, A. (2011). Comparing simulation algorithms for multienvironment probabilistic P systems over a standard virtual ecosystem. *Natural Computing*, *11*, 369–379.
24. García-Quismondo, M., & Martínez-del-Amor, M. A. (2014). Probabilistic guarded P systems, a new formal modelling framework. *International Conference on Membrane Computing., 2014*, 193–214.
25. Gillespie, D. T. (1992). A rigorous derivation of the chemical master equation. *Physica A: Statistical Mechanics and its Applications*, *188*, 404–425.
26. Păun, G., Rozenberg, G., Salomaa, A., editors. (2010). *The oxford handbook of membrane computing*, Oxford University Press, Oxford.
27. Lu, T., Volfson, D., Tsimring, L., & Hasty, J. (2004). Cellular growth and division in the Gillespie algorithm. *Systems Biology*, *1*, 121–128.
28. Gillespie, D. T. (1997). Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry B*, *81*, 2340–2361.
29. Shimkets, L. J. (1990). Social and developmental biology of myxobacteria. *Microbiological Reviews*, *54*, 473–501.
30. Diggle, P. J. (2003). *Statistical analysis of spatial point patterns*. New York: Academic Press.
31. Díaz-Pernil, D., Pérez-Hurtado, I., Pérez-Jiménez, M.J., Riscos-Núñez, A. (2008). A P-lingua programming environment for membrane computing. In: D. W. Corne, P. Frisco, G. Păun, G. Rozenberg, A. Salomaa (Eds.), *Membrane computing. WMC 2008. Lecture notes in computer science*, vol. 5391. Berlin, Heidelberg, pp. 187–203.
32. Mayne, R., Phillips, N., Adamatzky, A. (2018). Towards experimental P-systems using multivesicular liposomes. arXiv, 2018;1812.05476.
33. Degond, P., Manhart, A., & Yu, H. (2018). An age-structured continuum model for myxobacteria. *Mathematical Models and Methods in Applied Sciences*, *29*, 1737–1770.
34. Hendrata, M., Yang, Z., Lux, R., & Shi, W. (2011). Experimentally guided computational model discovers important elements for social behavior in myxobacteria. *PLoS One.*, *2011*, 6.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.