
Morphogenesis through Moving Membranes

Vincenzo Manca · Giovanni Pardini

Abstract We present a methodology for the modelling of spatially-aware biological phenomena, based on the description of the movement of membranes in the Euclidean space. The time evolution of the system is described by an iterative algorithm, which determines the movement of the objects according to the actions they perform, and the constraints they are subjected to.

We exemplify our approach with a model of the morphogenesis of *Dictyostelium discoideum*, and present the results of its simulation.

Keywords Morphogenesis · Interacting membranes · Shape formation · Algorithmic modelling · *Dictyostelium discoideum* · Discrete biological models

1 Introduction

Since a few decades, the constant increase of the power of computing hardware has allowed the simulation of biophysical models which explicitly take into account the spatial arrangement of the various biological entities. However, such models are mostly developed using a bottom-up approach, which consists in the formalisation, using appropriate tools, of the actual known low-level mechanisms, which are used to construct more complex models on top of them. One shortcoming of this approach is that it requires proper low-level knowledge of the biomechanics of membranes and cells.

In this paper, we develop an iterative algorithm for the simulation of spatial systems of moving membranes, using a top-down approach to the modelling. The idea behind our approach is to focus on the definition of a few high-level mechanisms which are able to accurately model the observable behaviour that we want to reproduce. In particular, we explicitly disregard the actual low-level interaction mechanisms among membranes, whose effects are instead to be captured by proper high-level abstract mechanisms. Our aim is to be able to shed light on the key mechanisms driving the particular behaviour in which we are interested, avoiding the difficulty to identify and model the actual biomechanical interactions.

This work started when G. Pardini was employed at Dipartimento di Informatica, Università degli Studi di Verona, Italy.

V. Manca
Dipartimento di Informatica, Università degli Studi di Verona, Strada le Grazie 15, 37134 Verona - Italy
E-mail: vincenzo.manca@univr.it

G. Pardini
Dipartimento di Informatica, Università di Pisa, Largo B. Pontecorvo 3, 56127 Pisa - Italy
E-mail: pardinig@di.unipi.it

This is an author-created version of the paper:

V. Manca, G. Pardini: **Morphogenesis through Moving Membranes**. Natural Computing, Volume 13, Issue 3, pp. 403–419, 2014

The final publication is available at Springer via <http://dx.doi.org/10.1007/s11047-013-9407-4>

Membranes are used to model cells from the real world, which are embedded in a 3D continuous space, and are subjected to proper *displacement actions* which determine how they move in space. Informally, a displacement action is described by a vector, and the resulting movement of a membrane is the result of the sum of all the vectors of the actions affecting the membrane.

We propose a generic framework modelling populations of moving membranes, based on displacement actions and an iterative algorithm for updating the positions of membranes as time passes. The framework includes a few basic common mechanisms which underlie many interesting systems, namely (i) cell adhesion, (ii) substrate repulsion, and (iii) action propagation through attached membranes.

Finally, we instantiate our framework to the development of an iterative algorithm for the simulation of the morphogenesis of *Dictyostelium discoideum* (see, e.g. Maree and Hogeweg (2001)). Normally, cells of the *Dictyostelium discoideum* are spread in the environment, but they are able to aggregate into moving slugs in response to environmental conditions. In particular, when the level of food available for the cells is low, they aggregate to form a slug which moves towards a place suitable for culmination. When a good place is found, the slug stops moving and transforms itself into a fruiting body, composed of a thin stalk with a mound of spores on the top. Finally, the spores are released in the environment. Our iterative algorithm is proved to be able to reproduce the essential logic of this morphogenetic dynamics, for suitable values of parameters which, as shown in the following sections, encode specific aspects of membrane movements and interactions.

2 The framework

Let $I_t = \{1, \dots, n\}$ be the set of membrane indices at a given time t . We associate a constant *radius* r_i with each membrane i . This radius does not have a direct physical meaning, but instead is used to derive a few other parameters, as follows:

- *diameter distance*: the distance, when the system is in a stable state, between two attached cells i, j is the sum of their radii, i.e. $r_i + r_j$;
- *attraction distance*: the maximum distance between two cells for which cells are considered attached.

For simplicity, we assume throughout the paper that the membranes have the same radius of value 1. Therefore the diameter distance between a pair of membranes is 2. Let us consider a coefficient $\theta_{\text{d-attr}}$ which allows us to derive the attraction distance from the diameter distance, namely we assume the attraction distance to be equal to $\theta_{\text{d-attr}}$ times the diameter distance.

Let Σ denote the set of possible displacement actions. Given the current time t , and the corresponding positions $p_1[t], \dots, p_n[t] \in \mathbb{R}^3$ of each membrane, a displacement vector $F_i^\sigma(p_1[t], \dots, p_n[t], t)$ needs to be associated with each action $\sigma \in \Sigma$ and each membrane $i \in I_t$. All the displacements vectors affecting the membrane are summed up to obtain the final displacement vector, which is used to compute the new position $p_i^{(t+1)}$ of the membrane at the following step, as follows:

$$p_i[t+1] = p_i[t] + D_i[t]\Delta t = p_i[t] + \left(\sum_{\sigma \in \Sigma} F_i^\sigma(p_1[t], \dots, p_n[t], t) \right) \Delta t \quad i \in I_t, \quad (1)$$

where Δt denotes the length of the step. In the rest of the paper, we denote by $\|\cdot\|$ the Euclidean norm; thus, given a position $x \in \mathbb{R}^3$, $\|x\|$ corresponds to the *length* of the vector x .

In the following, we introduce the basic displacement actions that we consider, namely: cell adhesion, substrate repulsion, gravity, jitter, and action propagation.

Cell adhesion We model cell adhesion between two cells with a displacement vector which either (i) pushes back the cells if they are colliding, namely if their distance is below their diameter distance, or (ii) attracts the cells if they are not colliding but they are still close enough. No displacement occurs if their distance is equal to their diameter distance.

Given a cell i , the adhesion vector it is subjected to is computed as follows:

$$F_i^{\text{adhesion}}[t] = \sum_{j \in \text{colliding}(i,t)} \theta_{\text{rep}} \cdot v_{j,i}[t] + \sum_{j \in \text{nearby}(i,t)} \theta_{\text{attr}} \cdot v_{i,j}[t]$$

where $v_{i,j}[t] = (p_j[t] - p_i[t]) / \|p_j[t] - p_i[t]\|$ denotes¹ the direction from $p_i[t]$ to $p_j[t]$, $\text{colliding}(i,t)$ denotes the cells whose distance is less than their diameter distance, $\text{nearby}(i,t)$ denotes the non-colliding cells within the attraction distance, as defined by the equations below. Parameters θ_{rep} and θ_{attr} specify the strength of the repulsion and attraction vector, respectively.

$$\begin{aligned} \text{colliding}(i,t) &= \{j \mid \|p_i[t] - p_j[t]\| < r_i + r_j\}; \\ \text{nearby}(i,t) &= \{j \mid r_i + r_j \leq \|p_i[t] - p_j[t]\| < \theta_{\text{d-attr}}(r_i + r_j)\}. \end{aligned}$$

Substrate repulsion We assume a 3D space and a related Cartesian frame. For the sake of simplicity, we also assume a planar “substrate” where a part of our cells lie. Precisely, such a substrate is identified with the xy plane. The presence of this surface, which in general can assume more complex forms, is a postulate for any effective movement. In fact, no object can move without a support with respect to which the movement is realised, where friction causes an opposite action tending to maintain fix the position of the object on the support. Differently from the idealised motion in mechanics, where motion is described with respect to a fixed point (namely, the origin of a Cartesian frame), here we disregard friction, and instead we postulate the motion with respect to the substrate. Thus, in our model, the substrate constitutes the initial entity necessary for providing motion.

In order to prevent membranes from penetrating the substrate as the result of other displacement vectors applied to them, we include a vector modelling the substrate repulsion. This vector is directed along the surface normal, and its strength is determined by the parameter θ_{subrep} , as follows:

$$F_i^{\text{substrate}}[t] = \begin{cases} \theta_{\text{subrep}} \cdot [0, 0, 1]^T & \text{if } (p_i[t])_3 - r < 0 \\ \mathbf{0} & \text{otherwise} \end{cases}$$

where $[0, 0, 1]^T$ is the unit vector along the z axis. Note that vector $F_i^{\text{substrate}}$ is null when the distance of cell from the substrate is greater than the radius.

Gravity We also need to consider gravity, which is simply modelled by the following displacement vector:

$$F^{\text{gravity}}[t] = \theta_{\text{gravity}} \cdot [0, 0, -1]^T$$

where θ_{gravity} denotes the gravity strength.

¹ Term $\|p_j[t] - p_i[t]\|$ corresponds to the Euclidean distance between $p_j[t]$ and $p_i[t]$.

Jitter We also introduce jitter in the movement of membranes, to capture the natural Brownian motion to which all particles in the real world are subjected to. Precisely, we consider a displacement vector for each membrane i having the following form:

$$F_i^{\text{jitter}}[t] = \theta_{\text{jit}} \cdot \hat{u}_i[t]$$

where θ_{jit} denotes the jitter amount, and $\hat{u}_i[t]$ is a random unit vector, corresponding to a uniformly-distributed point in the surface of the unit sphere. Note that, at any given time t , a random unit vector $\hat{u}_i[t]$ is generated for each membrane i in the system.

Action propagation Displacement actions which are exerted from a membrane to the substrate must be handled in a special way, and we cannot simply consider the displacement of a membrane as the sum of the various displacement vectors computed separately. In fact, doing that would neglect the effect of the propagation of forces from a membrane to all the membranes attached. In fact, the membranes which are attached to the surface are the ones which actively exert a force on the substrate, and as a result they tend to move in the opposite direction. The movement of a membrane causes all the connected membranes to move and deform. In the following we formally define the *EFG algorithm*, which ensures that the propagation of movements is handled correctly.

The EFG algorithm

In order to handle the propagation of displacement actions, it is necessary to relax some constraints on the model. In particular, in order to derive the movement of the membranes we need to take into account the position and displacement of all the membranes attached. Therefore, the actual movement of the membranes, described by displacement vectors, depends on the propagation of actions among attached membranes.

The *EFG algorithm* is used to compute the actual displacement of each membrane in a step. Its name stems from the three kinds of displacements that it considers:

- *Externally originated displacements*: these encompass any displacement due to external forces, which are not originated from the membrane itself in response to a signal. This includes most of the displacements presented in the previous sections, namely adhesion, substrate repulsion, gravity, and jitter.
- *Fundamental population displacements*: these displacements are originated from the membrane themselves. For example, it includes the displacements due to forces exerted by a membrane to the substrate, which causes its movement.
- *propagation Generated displacements*: these are the displacements derived from the displacement of all the other membranes attached. Any displacement, irregardless of how it originated, (namely any of E, F, and G), is propagated from one membrane to all attached membranes according to the *action propagation* algorithm.

An important aspect of this model is that it does not take into account the energetic balance involved in cell movements, as it instead happens in physical analysis. This is because membranes can perform autonomous displacements, without considering any notion of force (in the classical sense), as would be necessary.

As regards the propagation of actions through attached membranes, we consider a linear propagation coefficient α such that $0 \leq \alpha \leq 1$. The value of parameter alpha describes how the displacement is propagated from one cell to another. In particular, $\alpha = 1$ means that the displacement is propagated identically to all attached membranes, while $\alpha = 0$ means that only the component of the displacement projected onto the direction connecting the centres of the membranes is propagated. Intermediate values allows precise tuning of the displacement propagation.

Let F denote the displacement vector applied to a cell, corresponding to the sum of all displacements, of any kind, it is subjected to. Moreover, let F_p be the projection

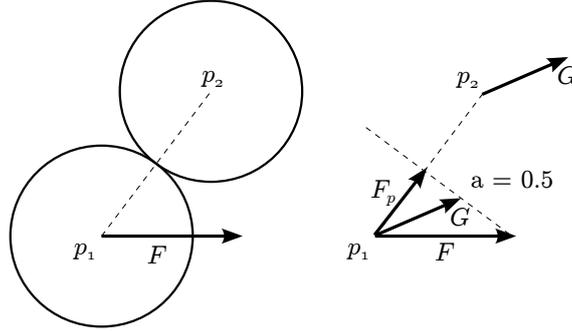


Fig. 1 Propagation of displacement vectors.

of F along the direction connecting the centres of the two membranes, as depicted in Figure 1. The displacement vector G applied to the attached cell, resulting from the propagation of F , is the following:

$$G = \alpha F + (1 - \alpha)F_p$$

Note that, if $\alpha = 1$ then the displacement vector applied to the attached cell is the same, i.e. $G = F$, while if $\alpha = 0$ then the vector applied is just the projection F_p , i.e. $G = F_p$. In the other cases, vector G is obtained as a linear combination of F and F_p . Figure 1 depicts vector G when $\alpha = 0.5$.

Let t be the current time, and let $attached(x)$ denote the set of indices of membranes directly attached to cell x . For the sake of conciseness, let F_1, \dots, F_n be the vectors applied to each cell, namely, for all i , $F_i = F_i^{movement}[t]$. Vectors F_i 's can be non-null only for membranes adjacent to the substrate. Note that we are only interested in the case in which a displacement of a cell occurs with respect to the substrate (which is fixed), and not when a displacement occurs with respect to another attached cell.

Algorithm 1 (*propagateDisplacement*) is used to compute each vector $G_{\alpha,i,j}^{prop}$ propagated from a cell i to any other cell j , using propagation factor α . Algorithm *propagateDisplacement*(i, F_i) computes the propagated displacement $G_{\alpha,i,j}^{prop}$ acting on each cell j , assuming an initial displacement vector F_i applied to cell i . Vector F_i corresponds to the sum of any of external (of kind (E)) and fundamental displacements (of kind (F)) applied to the cell. Given such displacements, the total propagated displacement $G_{\alpha,i}^{prop}$ acting on a cell is:

$$G_{\alpha,i}^{prop} = \sum_j G_{\alpha,j,i}^{prop}.$$

Finally, the total displacement applied to a membrane i , denoted D_i in Equation 1, is given as the sum of all kinds of displacements, thus it includes both externally-originated (E) and fundamental displacements (F), and propagated displacements (G) as computed above.

3 Case study: Dictyostelium Discoideum

In this section we present a model of the morphogenesis of the Dictyostelium Discoideum, using the basic mechanisms presented in Section 2. We represent each cell as a point in space representing its centre of mass, ignoring the actual shape of the cell's membrane. This is motivated by the fact that cell's shape can be always approximated by a sphere throughout the process, with low loss of precision.

The actual process can be described by the following three phases:

Algorithm 1 propagateDisplacement(i, F_i)

```

1:  $\forall j \neq i. G_{\alpha, i, j}^{\text{prop}} = \mathbf{0}$ 
2:  $G_{\alpha, i, i}^{\text{prop}} = F_i$ 
3:  $visited = \{i\}$ 
4:  $Q = \{i\}$ 
5: while  $Q \neq \emptyset$  do
6:   let  $x = head(Q)$ 
7:   Dequeue( $Q$ )
8:   for each  $y \in attached(x) \setminus visited$  do
9:     let  $H$  be the projection of  $G_{\alpha, i, x}^{\text{prop}}$  onto direction  $p_y - p_x$ 
10:     $G_{\alpha, i, y}^{\text{prop}} = \alpha G_{\alpha, i, x}^{\text{prop}} + (1 - \alpha) H$ 
11:     $visited = visited \cup \{y\}$ 
12:    Enqueue( $Q, y$ )
13:   end for
14: end while
15: return  $G_{\alpha, i}^{\text{prop}}$ 

```

1. **aggregation:** a group of cells start secreting a chemical signal in response to an environmental condition, to which the other cells react by moving towards the source of the signal;
2. **slug motion:** once a slug is formed, the slug wanders around in order to find a place suitable for culmination;
3. **formation of the fruiting body:** the slug transforms for spore dispersal

In the real world, each switch from a phase to the subsequent is triggered by specific conditions to which the cells react. In order to keep the model simple, we assume a fixed duration for each of the phases, driving the behaviour of the complete system

In the following sections, we separately describe each phase, and show how they can be modelled using the basic mechanisms presented in the previous section. For each phase, we also present the results of the simulation of different systems, in order to study the role played by the involved parameters in the resulting behaviour of the system.

3.1 Aggregation phase

Let us assume the membranes are initially disposed on the substrate, uniformly distributed on a circular area. The aggregation phase is modelled by assuming a constant displacement vector for each cell, which is directed towards a fixed position representing the centre for the aggregation. Given a proper selection of the parameters, cell adhesion ensures that the cells form a single mound of attached cells. In fact, the constant aggregation displacement vector, together with the substrate repulsion displacement vector (which prevents the interpenetration of the cells), causes that upwards growth of the mound.

Given a cell i , its aggregation vector is:

$$F_i^{\text{aggregation}}[t] = \theta_{\text{aggr}} \cdot dir(p_i[t], \mathbf{0})$$

where $dir(p_i[t], \mathbf{0}) = -p_i[t]/\|p_i[t]\|$ denotes the vector direction from p_i to the centre of aggregation, namely position $\mathbf{0}$. Similarly to cell adhesion, factor θ_{aggr} represents the strength of the aggregation vector.

In order to obtain the aggregate of cells, it is sufficient to apply the aggregation vector to each cell attached to the substrate. In particular, the total displacement acting on each cell is obtained as the sum of the substrate repulsion vector, the adhesion vector, and the aggregation vectors, as follows:

$$\forall i. D_i[t] = F_i^{\text{gravity}}[t] + F_i^{\text{jitter}}[t] + F_i^{\text{substrate}}[t] + F_i^{\text{adhesion}}[t] + F_i^{\text{aggregation}}[t]. \quad (2)$$

3.2 Motion phase

The motion of the slug is coordinated by a small group of pacemaker cells, which are positioned along the leading edge of the slug. They periodically secrete a chemical signal which diffuses along the substrate, and which is received by cells attached to the substrate. When those cells sense the signal, they move towards the source of the signal, bringing with them all the other cells which are on top of them. This causes the entire slug to move ahead in a pulsatile fashion which is driven by the signal waves propagating from the pacemaker cells.

We explicitly model signal propagation from a selected pacemaker cell. When the signal encounters a cell attached to the substrate, a displacement vector is applied to such a cell, which models the movement of the cell in response to the signal. Formally, we consider the following parameters: (i) signal speed $\theta_{\text{sig-speed}}$; (ii) signal duration $\theta_{\text{sig-duration}}$; (iii) period of signal generation τ_{sig} . Each period a signal is generated from the actual position p_i of a pacemaker cell. Such a position is recorded as the source position of that particular signal p^{src} . The movement vector for cells attached to the substrate is the following:

$$F_i^{\text{movement}}[t] = \begin{cases} \theta_{\text{mov}} \cdot v_{i,\text{src}} & \text{if the signal reaches } p_i \text{ in the time interval } [t, t + \Delta t] \\ \mathbf{0} & \text{otherwise} \end{cases}$$

where $v_{i,\text{src}} = \text{dir}(p_i, p^{\text{src}}) = (p^{\text{src}} - p_i) / \|p^{\text{src}} - p_i\|$ denotes the direction from the position of membrane i to the source position of the signal p^{src} .

The displacement vector must also include the displacement vectors propagated through attachments:

$$\forall i. D_i[t] = F_i^{\text{gravity}}[t] + F_i^{\text{jitter}}[t] + F_i^{\text{substrate}}[t] + F_i^{\text{adhesion}}[t] + G_{\alpha_{\text{mov}},i}^{\text{prop}}(t, F_1^{\text{movement}}, \dots, F_n^{\text{movement}}). \quad (3)$$

where $G_{\alpha_{\text{mov}},i}^{\text{prop}}(t, \dots)$ denotes the sum of the movement vectors $F_i^{\text{movement}}[t]$ propagated to the cell.

3.3 Formation of the fruiting body

In this phase, the mound of cells gradually changes shape, forming a stalk of cells which support a globule of cells positioned on top of the stalk. Such a transformation is driven by chemical signals secreted by cells on the top, which cause the upward movement of the other cells. In particular, the globule of cells is formed very early in the process, and it is pushed upwards by the elongation of the stalk beneath it.

In our model we use a different process to describe the formation of the fruiting body. In particular, a small group of cells positioned at the top of the mound is selected. These cells form the proliferating cells, from which the stalk grows. At constant time intervals, these cells duplicate, and each new cell is positioned on top of the old one. Each cell duplicates as long as there are no other cells on top of them, thus causing the stalk to grow mainly from the top. The length of each duplicating period is $\tau_{\text{stalk-dup}}$.

Duplicating cells represent a different kind of cells with respect to the others. Each new cell is initially created of the duplicating kind, and remains such as long as there are no other cells on top of it, i.e. only if it is not *hidden*. This is ensured by a check, performed at the beginning of the step, where a cell i is considered hidden iff there is any other cell j such that $(p_j)_3 > (p_i)_3$ and their distance is less than a given parameter d_{hiding} .

The growth of the stalk begins from the position $p_{\text{center}} = [x_{\text{center}}, y_{\text{center}}, z_{\text{center}}]$, which is selected at the beginning of the step corresponding to the position of the highest

cell (i.e. the position of the cell with the greatest z). Such a position is used to select the initial proliferating cells, which correspond to the cells within distance $d_{\text{init-dup}}$ from p_{center} . Moreover, p_{center} is also used during the growth of the stalk, in order to ensure that the growth is mainly vertical.

In particular, given the current position $p_i[t]$ of the cell, the position of a newly created cell, with new index j , is computed as follows:

$$p_j[t + 1] = p_i[t] + \theta_{\text{center}} \cdot \text{dir}(p_i[t], [x_{\text{center}}, y_{\text{center}}, (p_i[t])_3]) + [x, y, d_z]$$

where $\text{dir}(q_1, q_2) = (q_2 - q_1) / \|q_2 - q_1\|$ denotes the vector direction from q_1 to position q_2 , x, y are random variates of the uniform distribution $\mathcal{U}(-d_{xy}, +d_{xy})$, and d_z is a parameter which determines the position along the z axis.

Two components are summed up to obtain the placement of the new cell: $\theta_{\text{center}} \cdot \text{dir}(p_i[t], [x_{\text{center}}, y_{\text{center}}, (p_i[t])_3])$ means that the cell should be placed towards the centre, while $[x, y, d_z]$, with $d_z > 0$, indicates that the new cell is positioned on top of the other, with some randomisation of the actual (x, y) -position.

When the growth of the stalk terminates, the cells on the top of it start duplicating. Differently from the cells forming the stalk, each cell continues to duplicate, at constant intervals $\tau_{\text{glob-dup}}$, until the phase is complete. Moreover, each new cell is positioned randomly around the originating cell, which causes a rearrangement of the nearby cells to accommodate for the space occupied by the new cell. The duplication causes the formation of the final globule of cells, representing the spores. In this case, the position of each new cell j , originating from cell i , is computed as follows:

$$p_j[t + 1] = p_i[t] + d_{\text{glob-dup}} \hat{u}'_j[t]$$

where $d_{\text{glob-dup}}$ denotes the distance of the new cell from the parent cell, and $\hat{u}'_j[t]$ is a random unit vector, corresponding to a uniformly-distributed point in the surface of the unit sphere.

At each step, after the newly created cells are added to the system, the displacement vectors are computed for all the cells, as follows:

$$\forall i. D_i[t] = F^{\text{gravity}}[t] + F_i^{\text{jitter}}[t] + F_i^{\text{substrate}}[t] + F_i^{\text{adhesion}}[t]. \quad (4)$$

Note that the only mechanisms used are those of gravity, jitter, substrate repulsion, and cell adhesion.

4 Simulation

In this section we present the result of the simulations obtained by varying the parameters of the model. This allows us to discuss the role played by the different parameters of the model, and how they can be tuned to obtain the required behaviour. To this purpose, we graphically show the configurations obtained by simulating different values for the parameters.

The simulator has been implemented in the C++ programming language, using the SDL (Simple DirectMedia Layer) multimedia library [SDL Web page](#), and the OpenSceneGraph OSG [Web page](#) (2013) 3D graphics toolkit. Typically, a complete simulation using the reference parameters takes around 1 hour time, on an Intel Core i5 CPU clocked at 2.40GHz.

The length of each phase is fixed, and set to 500 time units for the aggregation, 600 for the movement, 1200 for the stalk growth, and 250 for the globule formation. Table 1 lists the values of the parameters for which there is a good agreement between the expected behaviour and the simulation results.

The results of the simulation are depicted, at different time instants, in Figure 2. The first figure (a) shows the initial configuration. Figure (b) shows an intermediate step in

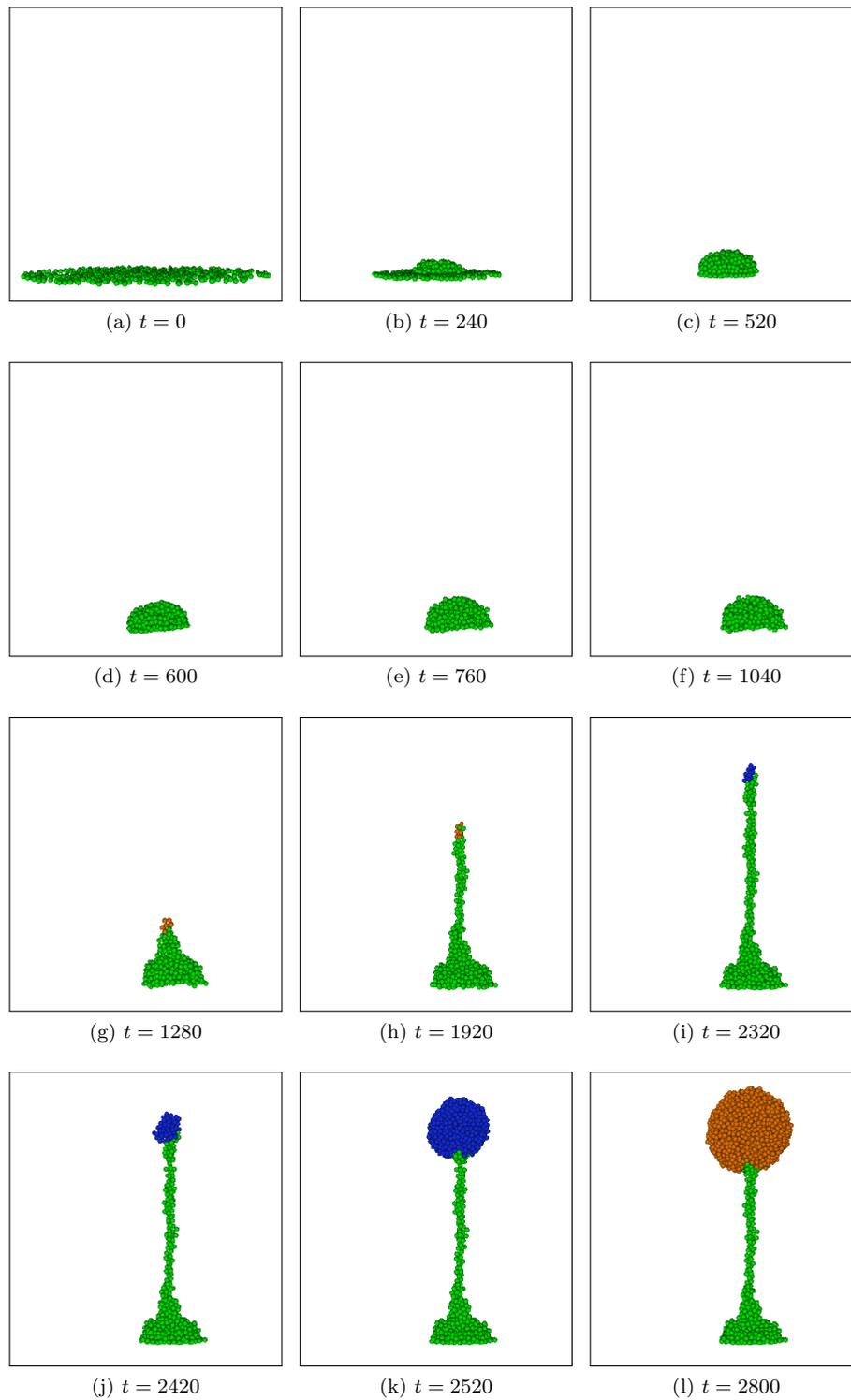


Fig. 2 Snapshots of the simulation encompassing the initial aggregation (a-c), the movement of the slug (d-f), the growth of the stalk (g-h), and the formation of the globule of cells (i-l). (Labels denote simulation times.)

Parameter	Value	Description
Δt	0.2	time step length
τ_1	500	duration of aggregation phase
τ_2	600	duration of movement phase
τ_3	1200	duration of stalk growth
τ_4	250	duration of globule formation
θ_{gravity}	0.003	gravity coefficient
θ_{jitter}	0.01	jitter coefficient
θ_{subrep}	0.7	substrate repulsion coefficient
θ_{rep}	1	repulsion coefficient for cell adhesion
θ_{attr}	0.02	attraction coefficient for cell adhesion
d_{attr}	2.4	attraction distance for cell adhesion
θ_{aggr}	0.1	(aggregation) strength of the aggregation vector
α_{mov}	0.75	(movement) displacement propagation factor for movement phase
θ_{mov}	0.04	(movement) strength of signal-induced movements
$\theta_{\text{sig-speed}}$	0.1	(movement) speed of propagating signals
$\theta_{\text{sig-duration}}$	1.5	(movement) duration of propagating signals
τ_{sig}	10	(movement) period of propagating signals
$\tau_{\text{stalk-dup}}$	20	(stalk formation) duplication period
d_{hiding}	2.003	(stalk formation) maximum distance of upper hiding cells
$d_{\text{init-dup}}$	4	(stalk formation) distance of initial duplicating cells from the top
θ_{center}	0.1	(stalk formation) tendency to align new cells to the center
d_z	0.4	(stalk formation) z -displacement of new cells
d_{xy}	0.2	(stalk formation) maximum x, y -displacement of new cells
$\tau_{\text{glob-dup}}$	30	(globule formation) duplication period
$d_{\text{glob-dup}}$	0.1	(globule formation) distance of newly created cells

Table 1 Values of the parameters used in the simulation.

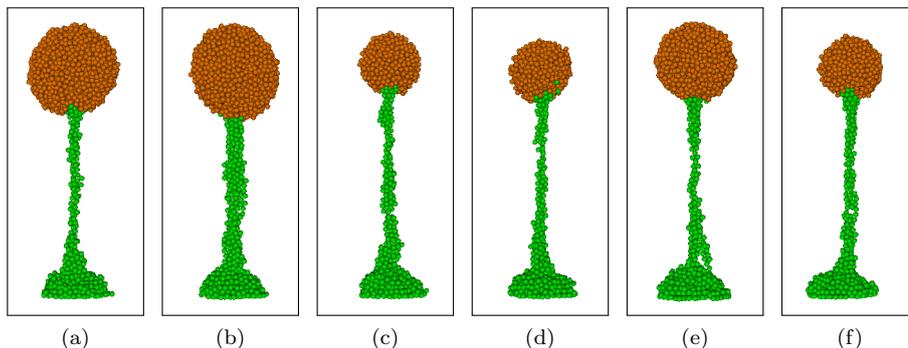


Fig. 3 The resulting shapes, at time $t = 2800$ of different simulations performed with the reference parameters shown in Table 1. (Figure 3-a is the same as Figure 2-l, for reference.)

which the cells are aggregating, forming the slug (c). During the subsequent movement phase (d-f), the shape of the slug changes slightly because of action propagation, which is caused by the different displacement vectors which act upon the cells. The growth of the stalk is depicted in figures (g-h), where the duplicating cells are shown in red. Finally, figures (i-k) show the globule formation, where the duplicating cells forming the globule are shown in blue (in a blank-and-white print they appear darker than those forming the stalk). In figure (l), the cells forming the globule have stopped duplicating (shown in red).

Figure 3 shows the resulting shapes of six different simulations using the parameters shown in Table 1 (Figure 3-a is the same as Figure 2-l for comparison). We recall that there are two sources of randomness involved in the model, namely in jitter and in the positioning of new cells during duplication. Therefore, albeit each simulation is different

from the others, the results look very similar in all the cases considered. In particular, the size of the stalk is more or less the same, except for Figure 3-b where it is visibly thicker than the other simulations. On the other hand, there is a greater variability in the size of the globule of cells. In fact, globules in Figures (a,b,e) are bigger than those in Figures (c,d,f).

The differences in the size of the stalk depend on the number of cells which are selected as duplicating at the beginning of the growth phase. In fact, their number is not fixed, but can randomly vary since it depends on how many cells are within $d_{\text{init-dup}}$ distance from the highest cell. Moreover, the size of the stalk may also change during the growth, usually decreasing to a certain thickness as in Figures (a,c).

The phase of stalk growth and globule formation does not use any other mechanism besides the ones we have already seen, thus cell proliferation is sufficient to form a structures similar to the fruiting body as observed in the real world. Note that the positioning of the new cells created by the duplication, with respect to the originating cells, determines the resulting shape. In fact, by positioning the new cell either on top of the old one, or randomly around the old one, we obtain two very different structures, i.e. the stalk and the globule on top of it.

4.1 Role of common parameters

We discuss in this section how the parameters common to all the phases affect the resulting shape. In the subsequent sections we separately discuss the role of the most important parameters involved in each phase, namely the aggregation and movement phases, and the formation of the stalk. As regards the formation of globule of cells, we have not performed any simulations because the parameters involved affect only the size of the resulting globule.

Gravity coefficient Figure 4 shows the resulting shapes obtained using the following values of gravity coefficients: 0, 0.003, 0.01, 0.1. The height differences between the simulations agree with the intuition, that higher gravity implies shorter structures. Moreover, gravity coefficient 0.1 prevents the growth, while the lower values still permit it. On the other hand, we note that the higher gravity causes a greater displacement of the slug during the movement phase. This is caused by the greater number of cells which are attached to the substrate; in fact, since they are the only ones which push the slug ahead, having more of them causes the slug to move faster.

Jitter coefficient As regards the jitter, Figure 5 shows the results of the simulation using values 0, 0.01, and 0.1. In particular, Figure (c,d) are relative to the simulation using $\theta_{\text{jitter}} = 0.1$, and depict the slug at times $t = 1480$ and $t = 2520$. Clearly, the values 0 and 0.01 allow the formation of the fruiting body, while 0.1 prevents it.

Substrate repulsion coefficient The substrate repulsion coefficient does not play an important role in the formation of the fruiting body. In particular, we tested values 0.01, 0.1, 0.7, 2 (Figure 6) and in all those cases the resulting shape is analogous to those obtained with the reference parameters and shown in Figure 3. Nevertheless, during aggregation, the lower values 0.01 and 0.1 are not sufficient to prevent the interpenetration of the cells into the substrate, as depicted in Figures (a,b), since the substrate repulsion displacements are overcome by the aggregation displacements. Finally, in all those cases, during the subsequent movement phase, since the aggregation displacements disappear, substrate repulsion causes the cells to be pushed upwards, resolving the interpenetrations into the substrate.

Attraction coefficient for cell adhesion Figure 7 shows snapshots of three simulations, performed using values 0.002, 0.02, 0.2 for the attraction coefficient for cell adhesion. Figures (a-c) depict the slug during the movement phase at time $t = 760$, while figures (d-f) show the resulting shapes. As regards the movement phase, using the smallest value 0.002, the slug is quite rough, while in the other cases it is smoother. The final shape is also quite different in the first case than the others, as the stalk is thicker and rougher.

Repulsion coefficient for cell adhesion As regards the repulsion coefficient for cell adhesion, Figure 8 shows snapshots of simulations performed using values 0.1, 1, 3. The moving slug looks the same in all those cases, however the final shape of the fruiting body is quite different from one case to another. In particular, the smallest value 0.1 produces a smaller fruiting body, with a very thin stalk with a diameter of less than two cells (and, consequently, with a small globule of spores on the top). On the other hand, using value 3 causes the formation of a bigger structure (see figure (f)), in which, however, the stalk and the globule on the top are still clearly separated.

4.2 Role of parameters for aggregation phase

Strength of the aggregation vector Increasing the strength of the aggregation vector causes the cells to move faster towards the centre. Figure 9 shows the formation of the slug using value $\theta_{\text{aggr}} = 1$. In this case, the fast moving cells cause a downward displacement which is not completely counteracted by the substrate repulsion. In fact, at least initially (Figures (a-d)), there are cells interpenetrated into the substrate. However, the conflicts are automatically resolved before the end of the aggregation phase, as shown in Figure (e).

4.3 Role of parameters for movement phase

As regards the movement phase, we have performed various simulations for investigating the role of the displacement propagation factor. Figure 10 depicts the slug during the movement phase, using five different values for the propagation factor: 0, 0.1, 0.3, 0.75, 0.9. The simulation using value 0.3 have been performed twice, and the results are shown in figures (k-o) and (p-t). As we can see from the figures, the shape of the slug is not affected by the propagation factor. The only visible difference among the simulations, is that, by increasing the propagation factor, the slug moves faster, which agrees with the intuition behind displacement propagation. The simulations in figures (k-o) and (p-t), both performed using value $\alpha = 0.3$, show that there can be some variability in the actual displacement of the slug among different simulations.

4.4 Role of parameters for stalk formation

Duplication period Varying the duplication period of cell during stalk formation causes the formation of fruiting body having different heights, assuming that the duration of the stalk formation phase remains the same. This is shown in Figure 11, where we have simulated values 1, 5, 50. The smaller values 1, 5 cause the stalk to grow faster, which also causes it having a smaller diameter. On the other hand, by using the bigger period 50, the cells have more time to settle between each duplication, under the effect of gravity. This causes the stalk to become thicker. To enhance the visibility of this effect, we have performed a simulation with $\tau_{\text{stalk-dup}} = 50$ using an extended length $\tau_3 = 3000$ (instead of $\tau_3 = 1200$) for the stalk formation phase. The resulting shape of the extended simulation is depicted in Figure (d).

Maximum distance of upper hiding cells This parameter is also very important in determining the resulting shape. We have performed simulation using values 1.99, 1.995, 2, 2.05, 2.1. As shown in Figure 12, using a value lower than 2 (which corresponds to the diameter distance between two cells, given as the sum of the radii of the cells) does not allow the correct formation of the fruiting body. On the other hand, values 2, 2.05, 2.1 permit the formation of the fruiting body. Moreover, value 2 (Figure (c)) causes the formation of a thicker stalk than the other two cases 2.05, 2.1 (Figures (d,e)).

Distance of initial duplicating cells from the highest cell Figure 13 shows the results of the simulation performed using values 1, 3, 5, 6, 10 for parameter $d_{\text{init-dup}}$, where simulation using value 10 has also been performed with an extended third phase ($\tau_3 = 1800$, Figure (f)). As regards values 1, 3, 5, 6, the resulting shapes are very similar. On the other hand, using the bigger value 10, since many cells are selected as duplicating, this causes the base of the stalk to be very big. However, the diameter of the stalk decreases as it grows until it settles to a diameter similar to the other simulations depicted in figures (a-d). This is particularly evident in Figure (f), where the length of the stalk growth phase has been extended to $\tau_3 = 1800$. In fact, in this case, the upper half of the stalk is similar to the stalk of figures (a-d).

Tendency to align new cells to the centre Figure 14 shows the results of the simulation performed using values 0, 0.1, 0.4, 0.5, 1 for parameter θ_{center} . An interesting result is that, using value 0, allows the fruiting body to form correctly, with just a thicker stalk. Using value 0.1 (Figure (b)) causes the stalk to become thinner. Further increasing the tendency to align to the centre for the duplicating cells causes the stalk to become thinner and shorter, as depicted in Figures (c-e).

Maximum x, y -displacement of new cells Figure 15 shows the simulation performed varying the maximum x, y -displacement of new cells. In particular, we have performed simulations using values 0.05, 0.1, 0.3, 0.4, 0.5. The effect of this parameter on the resulting shape of the fruiting body is limited. In particular, we notice a slight decrease of the height of the stalk as the parameter is increased. This is caused by the gravity, whose effect is more noticeable when the maximum x, y -displacement is bigger.

5 Related works

One of the earliest approaches in the modelling of spatial systems has involved the use of differential equation models. These models have been used to describe many different spatially-aware biological processes, such as reaction diffusion systems, chemotaxis, and the formation of spatial patterns (Murray, 2002, 2003). One shortcoming of models based on differential equations is due to the fact that they are based on continuous variables, thus they may not be adequate to describe discrete biological entities.

In the field of systems biology, many spatial modelling formalisms have been proposed in the last years which include the ability to describe the position of the elements of a system, with respect to some more or less abstract notion of space. The most abstract notion of space is given by compartments, often entailed by membranes, which allow the representation of different locations for the elements of the system. Since membranes play an important role in many biological processes, most computer science formalisms for biology allow some form of compartmental modelling (see, e.g. Regev et al, 2004; Cardelli, 2005).

Cellular Automata (Neumann, 1966) are a computational formalism inspired by biological behaviours, composed of a finite regular grid of cells which evolve in a synchronous way by means of a deterministic *rule*. Such a rule is characteristic of the

particular cellular automaton model, and it is used to determine the state of a cell with respect to the states of nearby cells. Cellular Automata, and their extensions, are particularly suitable for modelling populations of many similar entities, whose behaviour is based on local interactions (e.g. Patel and Nagl, 2006). In order to increase the expressiveness of cellular automata for the modelling of biological systems, a stochastic extension has been proposed in Patel and Nagl (2006).

P systems (Păun, 2000) are a computational and modelling formalism inspired by the functioning of living cells. A P system is composed of a hierarchy of membranes, where each membranes contains both a multiset of *objects* and the evolution rules which act on them. Many variants and extensions of P systems (Păun, 2002; P Systems Web page, 2013) exist that include features which increase their expressiveness and which are based on different evolution strategies. An extension of P system providing a more concrete representation of the space is the Spatial P System (Barbuti et al, 2011b), in which objects and membranes are embedded into a two-dimensional discrete space, similar to the space representation used in cellular automata. In Pardini (2011), a Spatial P system model, based on the Boids model (Reynolds, 1987), capturing the collective movement of herring schools is presented.

Another formalism which has some similarities with the approach proposed in this paper is the Spatial Calculus of Looping Sequences (Spatial CLS) (Barbuti et al, 2011a), an extension of the Calculus of Looping Sequences (CLS) (Barbuti et al, 2006, 2008) which allows both stochastic and spatial modelling. Membranes, which can be nested, may be associated with a precise position in a 2D/3D Euclidean space, and the space occupied by each membrane can be described. In case of conflicts objects and membranes are automatically rearranged. Nevertheless, complex global behaviours such as the propagation of actions among adjacent cells cannot be described. Finally, as regards extensions of computer science formalisms for the modelling of biological systems we cite SpacePI (John et al, 2008) and the 3π process algebra (Cardelli and Gardner, 2010), both extensions of the π -calculus (Milner, 1999). In both formalisms, processes are embedded in a 2D/3D continuous space, with different communication mechanisms. In contrast to the approach proposed in this paper, which supports the use of high-level interaction mechanisms, both SpacePI and 3π provide only low-level interaction capabilities.

Boids model The *Boids model* has been proposed in Reynolds (1987) as a model for the collective movement of groups of autonomous individuals which occurs in nature, such as fish schools and flocks of birds. The key feature of this model is that the collective motion emerges as the result of local interactions between the individuals, since there is no external entity which controls the behaviour of the single individuals. Using this model, a group of individuals positioned randomly in the space is able to organise itself into a single group, in which individuals move in a coordinated manner.

In the Boids model, individuals can “see” only a small space around themselves, in order to determine the relative distance and behaviour of nearby individuals. Then, the distance and direction of movement of nearby individuals is used to compute three different (vector) forces, which are then summed up to compute the resulting movement of the individual. The three forces acting on an individual are the following: a repulsive *separation* force that causes the individual to move away from nearby individuals; an attractive *cohesion* force which drives the individual towards the most crowded direction; and an *alignment* force, which tends to align the direction of the individual to the most common direction among nearby individuals.

The Boids models has some similarities with our model. In particular, the way the direction of an individual is determined in the Boids model is similar to our approach. In fact, in both cases, the resulting position is obtained by summing up some displacement vectors obtained by taking into account the position of the other individuals.

Nevertheless, the Boids model uses only purely local rules, namely each conceptual rule driving the behaviour of each individual takes into account the parameters (position and direction) of the only individuals within its visibility distance. On the other hand, in our model, the global effect of the displacement propagation mechanism cannot be described by local rules, since a displacement action can propagate to an unbounded distance, depending on the relative positions of the cells.

Cellular Potts model A model which has seen a widespread use for the modelling of morphogenesis is the cellular Potts model (Graner and Glazier, 1992). The cellular Potts model is composed of a lattice of discrete sites (similar to cellular automata (Neumann, 1966)), such as a square grid, where a *spin* is associated with each site. The set of sites with the same spin represent a *cell*. Moreover, there can be a finite number of *cell types*. The key concept of the Potts model is the Hamiltonian, namely a function which computes the total energy of the entire system, obtained by summing up the local energy between the spins of each pair of adjacent sites in the lattice. For example, a simple Hamiltonian is $\mathcal{H} = \sum_{(i,j)(i',j')\text{neighbors}} 1 - \delta_{\sigma(i,j)\sigma(i',j')}$; in this case sites with different spins (i.e. belonging to different cells) have energy 1, while energy between internal sites is null.

A Potts model is expected to evolve towards configurations with minimal energy. The simulation can be performed by using the Metropolis algorithm, namely an iterative procedure where in each iteration the spin of a random site is modified, and the consequent variation of system energy $\Delta\mathcal{H}$ is used to decide if the new configuration is accepted or not. In particular, if $\Delta\mathcal{H} \leq 0$, then the configuration is always accepted, since the total energy decreases. On the other hand, an increase of the total energy (i.e. $\Delta\mathcal{H} > 0$) is probabilistically accepted with Monte Carlo probability $P(\sigma(i,j) \rightarrow \sigma'(i,j)) = \exp(-\Delta\mathcal{H}/T)$, where $T > 0$ is the temperature, a parameter used to describe the volatility of the system. Note that, the higher the temperature, the higher the probability of accepting a configuration which increases the total energy.

A model of cell sorting with differential adhesivity is presented in Graner and Glazier (1992). This model, known as the Glazier and Graner model, considers three cell types (there can be many cells with the same type) which are used to describe differential surface energies depending on the types of adjacent cells. That is, according to differential adhesivity, the free energy between each pair of adjacent sites belonging to different cells depends on the actual type of the cells.

In Savill and Hogeweg (1997), a variation of the Glazier and Graner model is also used to model the morphogenesis of *Dictyostelium discoideum* in a 3D setting, from the aggregation phase to the migrating slugs. The aggregation is driven by cAMP signalling, where the cAMP diffusion is described by a partial differential equation (PDE), associated with sites corresponding to the 2D substrate. The movement of cells is based on differential cell adhesivity with volume conservation, together with chemotactic motion depending on local cAMP concentration. Such a model is extended in Maree and Hogeweg (2001) to provide a model of the entire process, from the aggregation phase to the culmination. In this case, the model also includes cell differentiation.

The main difference between our model and the one proposed in Savill and Hogeweg (1997) and Maree and Hogeweg (2001) is methodological: in Maree and Hogeweg (2001) the authors develop a model which accurately captures and reproduces the experimentally elucidated mechanisms known to be involved in the process. Therefore, such an approach is mainly bottom-up, namely their model has been developed by accurately modelling the known biomechanical interactions using an extension of the Glazier and Graner model (Graner and Glazier, 1992). On the other hand, our approach is mainly top-down, since we proposed a few basic high-level mechanisms which are able to reproduce the real behaviour, albeit with less correspondence to the reality. In our opinion, the top-down approach should allow a better understanding of the key mechanisms

driving the process of morphogenesis. Moreover, reusing the mechanisms to describe other systems would allow an empirical validation of the mechanisms chosen.

Biomechanical models A different approach to the description of the movement of cells in *Dictyostelium discoideum* is followed in Palsson and Othmer (2000). The authors propose a model of both the individual and collective movement of cells. The model is based on individual cells embedded in a continuous 3D space, where cells are modelled as deformable ellipsoids, which move under the effect of the forces acting on them. A cell is formally characterised by four parameters: (i) its position, corresponding to its centre of mass; (ii) its orientation; (iii) its stress level; (iv) the active forces it can exert. This modelling allows an accurate description of the observed *viscoelastic* behaviour of a cell when it is subjected to a force (originating either internally or externally), namely it initially responds in an elastic way to stress, up to a certain level, but under sustained force it produces a viscous response. They simulate both the aggregation phase, where cells move individually, and the subsequent slug movement, both phases being driven by the cAMP signalling of pacemaker cells. Since the behaviour is driven by the biomechanical forces acting on the cells, this constitutes a low-level realistic model. Therefore, the main difference with our approach, similarly to Savill and Hogeweg (1997); Maree and Hogeweg (2001), lies in the point of view used for the modelling, which also in this case is based on a low-level bottom-up approach.

6 Conclusions

We have proposed a top-down approach to the modelling of spatial biological systems, in which it is necessary to take into account the exact position of membranes in the space. Our top-down approach explicitly disregards the actual low-level biomechanical interactions which happen in the reality, but instead relies on a few high-level mechanisms. To this aim, we have developed a generic framework for the modelling of populations of moving membranes, based on a few high-level mechanisms: (i) cell adhesion, (ii) substrate repulsion, (iii) propagation of displacement actions through attached cells. These mechanisms are formalised by means of an iterative simulation algorithm, which have been used for the simulation of the morphogenesis of the *Dictyostelium Discoideum* amoeba.

In our opinion, an abstract top-down approach to the modelling of morphogenetic systems could enable a deeper knowledge of the actual high-level capabilities expressed by spatial biological processes. This approach shares some similarities with the development, in the seminal paper by Turing (1952), of an abstract model of morphogenesis of the growing embryo. In fact, in such a paper the mechanical aspects are explicitly ignored, by focussing instead on the study of role played by *morphogens*, i.e. the chemicals driving the morphogenetic process which diffuse from one cell to another.

An advantage of our modelling approach with respect to other models, such as Savill and Hogeweg (1997); Maree and Hogeweg (2001); Palsson and Othmer (2000) for the *Dictyostelium discoideum*, lies in the simplicity of the mechanisms involved in the model. We believe this will allow to shed some light on the mechanisms driving the behaviour of complex systems and, in particular, similarities among different spatial systems to be revealed and recognised more easily.

We have shown how the abstract framework can be exploited for modelling the morphogenesis of the *Dictyostelium Discoideum* amoeba, and that it is able to produce spatial structures of the amoeba similar to those observed in nature. This results is achieved in spite of the fact that we use only simple high-level mechanisms, and that some modelling decisions do not precisely agree with the real-world behaviour (namely the fact that the globule of cells is formed during the final phase instead of growing earlier and being pushed upwards by stalk growth). Nevertheless, by proper tuning the

parameters, we have shown that it is possible to obtain a final spatial structure which resembles the real-world shape of the amoeba with good accuracy.

The key aspect of this result is due to the correct choice of the values for the parameters involved in the simulation. As we have shown, even small variations in the values of the parameters can result in very different shapes being formed. In other words, parameters determine the actual morphogenetic process. This is in agreement with Manca (2013), where the role of parameters in determining the outcome of biological dynamics is discussed.

This work constitutes an initial application of the high-level approach to the modelling of biological systems with spatiality. The framework, since it is based on a few mechanisms, naturally allows the inclusion of many different extensions, which would allow greater expressiveness. A simple extension could allow cells to be distinguished into different kinds, which would react differently when subjected to the available mechanisms. This could be exploited, for example, to extend the model with multiple aggregation points, identified by cells of a different kind (instead of fixed positions), where each cell moves towards their relative nearest aggregation point. Another possible extension would be to allow deformable membranes, for example by adding volume conservation constraints.

An interesting application of the framework, that we leave as a future work, is the modelling of the embryogenesis in animals. This morphogenesis process consists in the initial phases after the zygote, i.e. the fertilised egg, has been formed, which in the end allows the formation of the embryo. An interesting step in this process is the gastrulation phase. We leave as a future work the development, using the framework proposed in this paper, of a model for the gastrulation in the *sea urchin* and similar animals, for which much literature is available (Kominami and Takata, 2004; Davidson et al, 1995; Hardin and Cheng, 1986; Tamulonis et al, 2011; Drasdo and Forgacs, 2000). Finally, another interesting aspect to be investigated involves the relation between our high-level model of the *Dictyostelium discoideum* and other low-level models, in particular as regards the correlation between the values of parameters in the two cases.

References

- Barbuti R, Maggiolo-Schettini A, Milazzo P, Troina A (2006) A calculus of looping sequences for modelling microbiological systems. *Fundam Inf* 72(1-3):21–35
- Barbuti R, Maggiolo-Schettini A, Milazzo P, Tiberi P, Troina A (2008) Stochastic Calculus of Looping Sequences for the Modelling and Simulation of Cellular Pathways. In: Priami C (ed) *Transactions on Computational Systems Biology IX*, Lecture Notes in Computer Science, vol 9, Springer Berlin / Heidelberg, pp 86–113
- Barbuti R, Maggiolo-Schettini A, Milazzo P, Pardini G (2011a) Spatial Calculus of Looping Sequences. *Theoretical Computer Science* 412(43):5976–6001
- Barbuti R, Maggiolo-Schettini A, Milazzo P, Pardini G, Tesi L (2011b) Spatial P systems. *Natural Computing* 10(1):3–16
- Cardelli L (2005) Brane Calculi – Interactions of Biological Membranes. In: Danos V, Schachter V (eds) *Computational Methods in Systems Biology*, Lecture Notes in Computer Science, vol 3082, Springer Berlin / Heidelberg, pp 257–278
- Cardelli L, Gardner P (2010) Processes in Space. In: Ferreira F, Löwe B, Mayordomo E, Mendes Gomes L (eds) *Programs, Proofs, Processes*, Lecture Notes in Computer Science, vol 6158, Springer Berlin / Heidelberg, pp 78–87
- Davidson LA, Koehl MAR, Keller R, Oster GF (1995) How do sea urchins invaginate? Using biomechanics to distinguish between mechanisms of primary invagination. *Development* 121(7):2005–2018
- Drasdo D, Forgacs G (2000) Modeling the interplay of generic and genetic mechanisms in cleavage, blastulation, and gastrulation. *Developmental Dynamics* 219(2):182–191

- Graner F, Glazier JA (1992) Simulation of Biological Cell Sorting Using a Two-Dimensional Extended Potts Model. *Physical Review Letters* 69:2013–2016
- Hardin JD, Cheng LY (1986) The Mechanisms and Mechanics of Archenteron Elongation during Sea Urchin Gastrulation. *Developmental Biology*
- John M, Ewald R, Uhrmacher AM (2008) A Spatial Extension to the π -Calculus. *Electronic Notes in Theoretical Computer Science* 194(3):133–148, proceedings of the First Workshop From Biology To Concurrency and back (FBTC 2007)
- Kominami T, Takata H (2004) Gastrulation in the sea urchin embryo: a model system for analyzing the morphogenesis of a monolayered epithelium. *Develop Growth Differ* 46(4):309–326
- Manca V (2013) *Infobiotics: Information in Biotic Systems*. Springer
- Maree AF, Hogeweg P (2001) How amoeboids self-organize into a fruiting body: Multicellular coordination in *Dictyostelium discoideum*. *Proceedings of the National Academy of Sciences of the United States of America* 98:3879–3883
- Milner R (1999) *Communicating and Mobile Systems: the π -Calculus*. Cambridge University Press
- Murray JD (2002) *Mathematical Biology: I. An Introduction*, 3rd edn. Springer-Verlag
- Murray JD (2003) *Mathematical Biology: II. Spatial Models and Biomedical Applications*, 3rd edn. Springer-Verlag
- Neumann JV (1966) *Theory of Self-Reproducing Automata*. University of Illinois Press
- OSG Web page (2013) OpenSceneGraph library. URL <http://www.openscenegraph.org>
- P Systems Web page (2013) P Systems web page. URL <http://ppage.psyste.ms.eu>
- Palsson E, Othmer HG (2000) A model for individual and collective cell movement in *Dictyostelium discoideum*. *Proceedings of the National Academy of Sciences* 97(19):10,448–10,453
- Pardini G (2011) *Formal Modelling and Simulation of Biological Systems with Spatiality*. PhD thesis, Università di Pisa
- Patel M, Nagl S (2006) *Mathematical Models of Cancer*. In: Nagl S (ed) *Cancer Bioinformatics: From Therapy Design to Treatment*, John Wiley & Sons, chap 4, pp 59–93
- Păun G (2000) Computing with Membranes. *Journal of Computer and System Sciences* 61(1):108–143
- Păun G (2002) *Membrane Computing. An Introduction*. Springer-Verlag, Berlin
- Regev A, Panina EM, Silverman W, Cardelli L, Shapiro EY (2004) BioAmbients: an abstraction for biological compartments. *Theoretical Computer Science* 325(1):141–167
- Reynolds CW (1987) Flocks, herds and schools: A distributed behavioral model. *SIG-GRAPH Comput Graph* 21(4):25–34
- Savill NJ, Hogeweg P (1997) Modelling Morphogenesis: From Single Cells to Crawling Slugs. *Journal of Theoretical Biology* 184(3):229–235
- SDL Web page (2013) Simple DirectMedia Layer (SDL) library. URL <http://www.libsdl.org>
- Tamulonis C, Postma M, Marlow HQ, Magie CR, De Jong J, Kaandorp J (2011) A cell-based model of *Nematostella vectensis* gastrulation including bottle cell formation, invagination and zippering. *Developmental Biology* 351(1):217–228
- Turing AM (1952) The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London Series B, Biological Sciences* 237(641):37–72

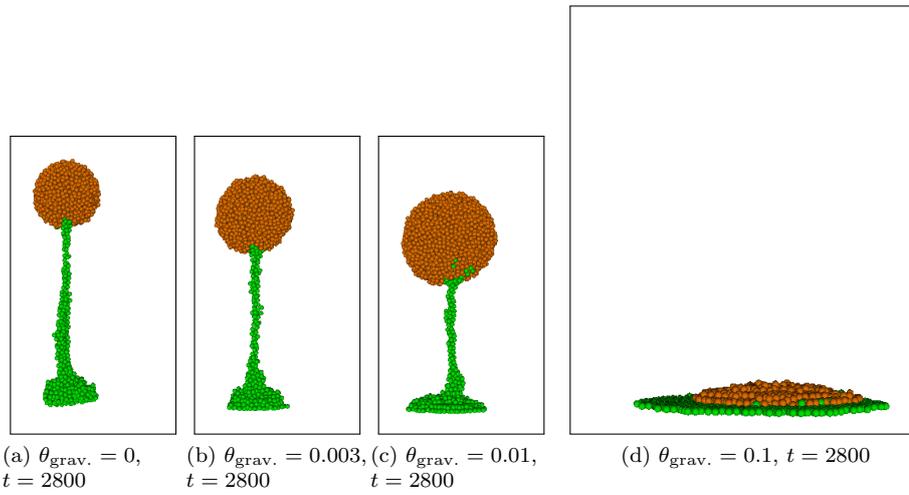


Fig. 4 Resulting shapes with varying gravity coefficient, at time 2800. (Figures are aligned to the left.)

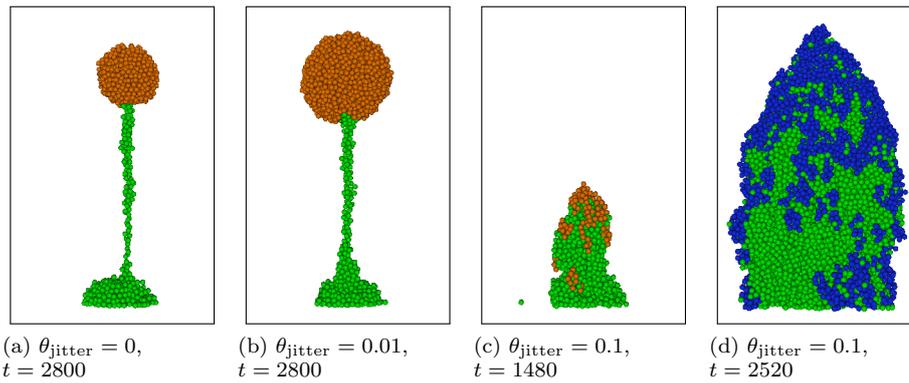


Fig. 5 Resulting shapes with varying jitter coefficient.

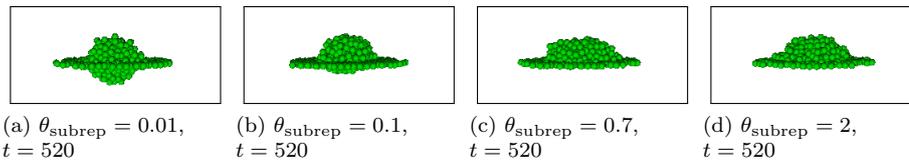


Fig. 6 Shape of the slug during aggregation, with varying substrate repulsion coefficient.

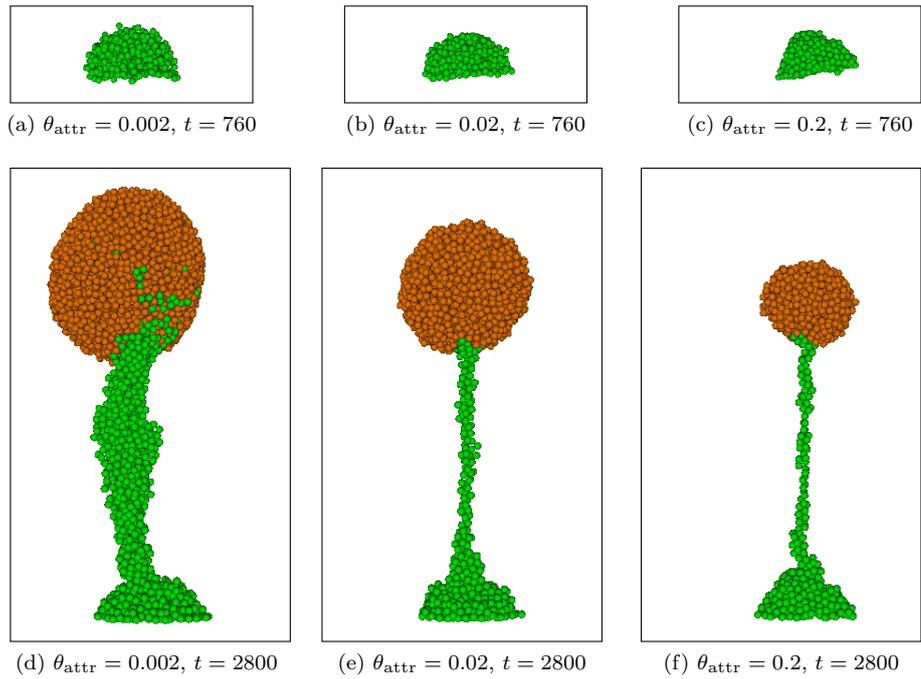


Fig. 7 Shape of the slug during movement (a-c) and resulting shapes (d-f), with varying attraction coefficients for cell adhesion.

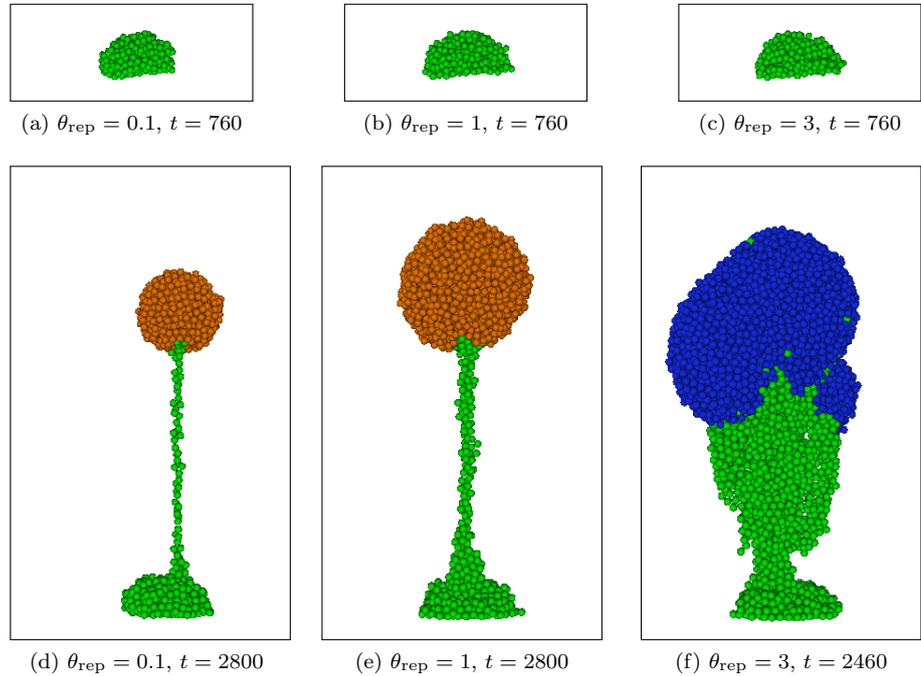


Fig. 8 Shape of the slug during movement (a-c) and resulting shapes (d-f), with varying repulsion coefficients for cell adhesion.

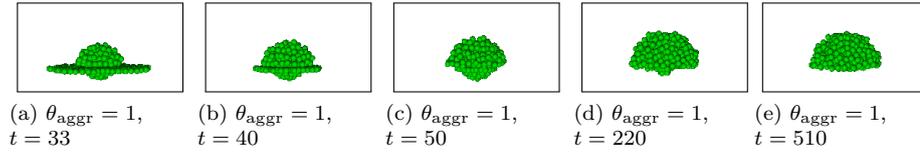


Fig. 9 Shape of the slug during aggregation with aggregation factor $\theta_{\text{aggr}} = 1$.



Fig. 10 Shape of the slug during movement with varying propagation factor.

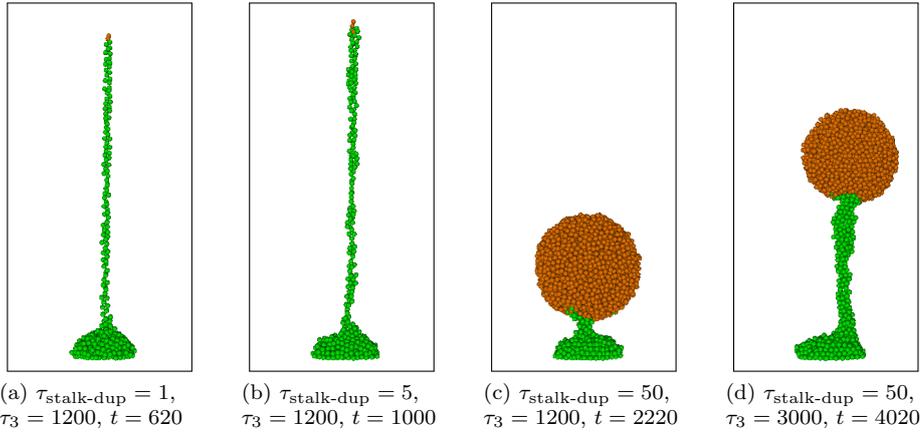


Fig. 11 Snapshots of simulations with varying duplication period for stalk formation. In figure (a,b), the stalk is still growing. (Simulations performed with $\tau_2 = 10$.)

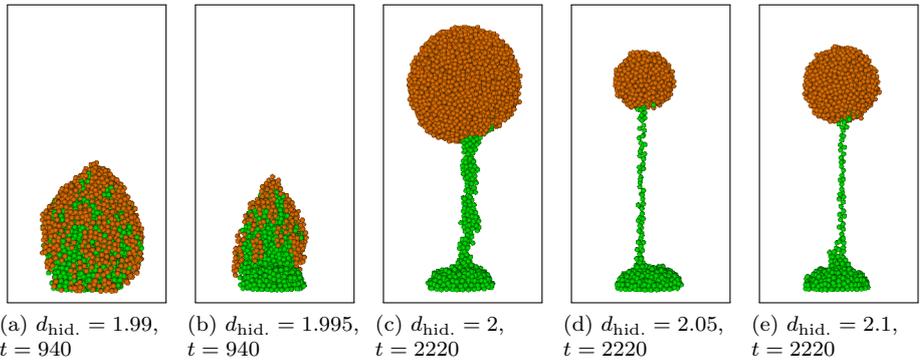


Fig. 12 Snapshots of simulations with varying parameter d_{hiding} , denoting the distance for an upper cell to be considered hiding an underlying cell, during stalk formation. (Simulations performed with $\tau_2 = 10$.)

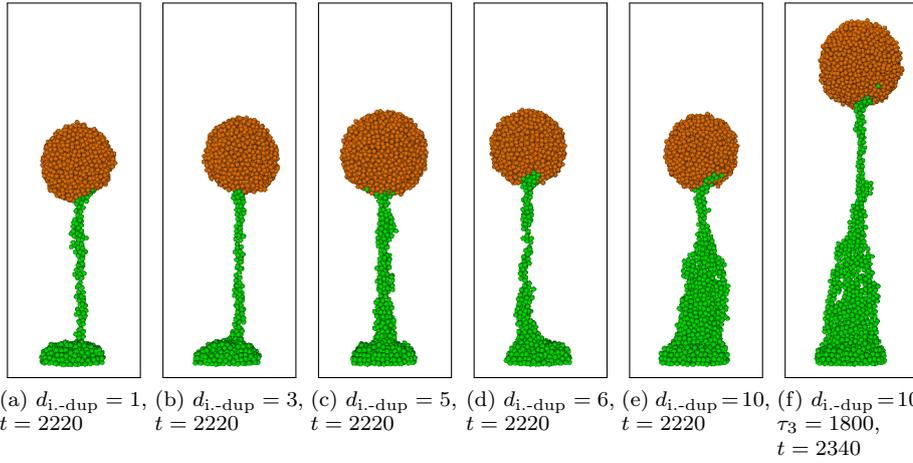


Fig. 13 Resulting shapes with varying parameter $d_{i.-dup}$, denoting the distance of the initial duplicating cell from the highest cell, for stalk formation. (Simulations performed with $\tau_2 = 10$.)

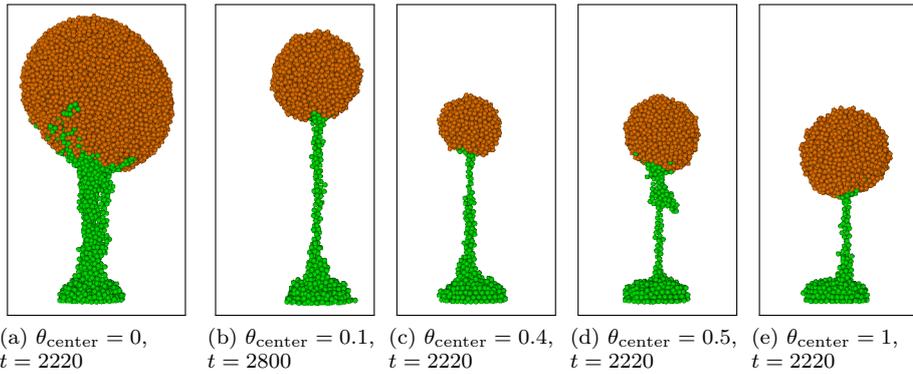


Fig. 14 Resulting shapes with varying tendency of new cells to be aligned towards the center during stalk formation. (Simulations (a,c-e) performed with $\tau_2 = 10$; simulation (b) performed with $\tau_2 = 600$.)

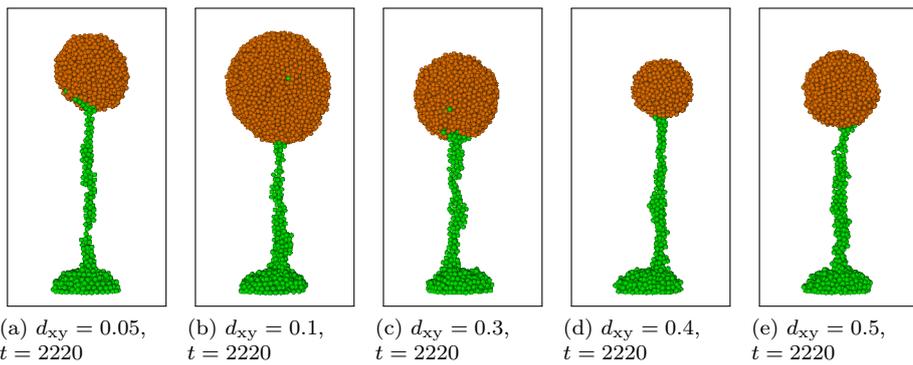


Fig. 15 Resulting shapes with varying maximum x, y -displacement of new cells during stalk formation. (Simulations performed with $\tau_2 = 10$.)