Agent-Based and Continuum Modelling of Populations of Cells

John King Michael Lees Brian Logan

December 2006

1 Introduction

Bacterial biofilms represent systems of considerable complexity, involving phenomena spanning a vast range of spatial scales (from sub-cellular to population) and with scope for generating a huge variety of emergent behaviour. Biofilms comprise communities of diverse individuals which may interact in both mutually beneficial and competitive fashions. They thus provide comparatively simple (and experimentally relatively well characterised) systems in which variety, 'altruism' and antagonism all have scope to flourish as a heterogeneous population develops. Cellular inter-relationships, even in single-species populations, are themselves highly complex, with signalling systems, such as quorum sensing, able to lead to coordinated changes in phenotype (see [10], for example). These quorum-sensing systems are increasingly being understood in terms of the subcellular interactions which govern the production of the relevant signalling molecules. Moreover, mathematical models of these processes are increasingly becoming established and validated, together with those of the corresponding macroscale behaviour (transport of signalling molecules and nutrient, biofilm growth etc.).

Such systems inextricably involve phenomena at both subcellular and macroscopic levels of a type widely studied by mathematicians (appropriate modelling frameworks typically comprising (possibly stochastic) differential equations and differential-delay equations) and interactions between individuals which requires the application of agent-based approaches (whereby signalling between neighbours belonging to distinct bacterial strains, say, leads to phenomena that cannot readily be captured by traditional multi-scale mathematical procedures such as homogenisation). Accounting adequately for the relevant subcellular behaviour in a population of millions of distinct, diverse individuals in order to bridge the scales presents significant modelling and simulation challenges but offers the potential for significant benefits for biology and medicine.

However, while there has been a significant amount of work on continuum and qualitative (process calculi) models of cell level processes on the one hand and both continuum and individual-based models of population scale effects on the other, there has been relatively little work which attempts to span these scales. For example, individual-based models are becoming increasingly widely used in this type of context (see, for example, [6, 3, 5, 2, 8, 7]). Such models use agent-based simulation techniques to investigate the interactions between (often relatively small) groups of cells and their environment. However, existing work in this area has tended to focus on the emergence of complex organisation in biofilms, with the individual cells being treated as 'black boxes' as far as possible, e.g., the use of cellular-automata approaches to describe biofilm growth.

In this working paper we present a hybrid model of the interactions within (multiple-species) populations of bacteria in a developing biofilm which integrates continuum models of population processes (e.g., diffusion of substrates and signalling molecules) with individual-based models of cellular processes (notably growth, division, displacement, and up-regulation). The cell level models in particular are novel in combining aggregated models of continuous processes (growth, division and displacement) for small collections of cells and individual-cell-level models of quorum sensing molecule (QSM) sensing, production and up-regulation which encompass both stochastic and discrete processes.

The model aims extend the state of the art in biofilm modelling (which mostly seems to be limited to qualitative modelling) by providing support for hypothesis testing, e.g., "What would happen if we starved bacteria of this strain or in this region of nutrient?" Such predictions are still qualitative, but could be tested in the lab. While the cell models used in the current prototype are naturally somewhat simplistic, the approach provides a generic framework into which different types of cells and more complex models of signalling pathways and gene networks (which are currently the subject of a vast amount of experimental study) can be plugged.

More generally, the approach embodied in the model described below provides a multi-scale framework for modelling populations of cells, which spans from the cellular level to the population level. In contrast to previous work, e.g., [8], the use of both aggregated and individual models of cellular processes, allows the resolution of the model to be tailored for a particular modelling problem, while at the same time remaining computationally tractable.

The remainder of the paper is organised as follows. In section 2 we delineate the scope of the biofilm model and outline its structure. For each of the main model components (biomass and substrate), we list the values that constitute the state of model at each timepoint. In section 3 we describe the evolution of the state of the model in detail, including diffusion of substrates and QSM, the growth, division, spreading and detachment of biomass, and inter-cell signalling, up-regulation and the production of extracellular polysaccharides. We also briefly outline ways in which the simulation termination condition can be specified. In section 4 we briefly describe the execution of the model and how the state of the model is computed at each timestep. A list of all simulation constants and variables can be found in section 6

2 The model

The overall system is similar to that described in [8]. The model system is a 3D biofilm reactor consisting of two compartments: bulk liquid and biofilm. The bulk liquid compartment contains a (completely mixed) solution of S different soluble substrates with concentrations $C_1^{bulk}, C_2^{bulk}, \ldots, C_S^{bulk}$. The biofilm grows on a planar support surface (substratum) and is assumed to consist of B different types of biomass. In addition to the biomass itself, the biofilm compartment contains a single type of extracellular polysaccharide (EPS) and Q types of quorum sensing molecule. The biofilm and bulk liquid compartments are in contact and exchange solutes only by diffusion. The bulk liquid volume is very large compared with the biofilm volume, and thus the substrate concentrations in the bulk liquid can be considered constant.

Figure 1 illustrates the computational model which is only a small part of the whole system. Within the computational domain the biofilm grows in a rectangular box of dimensions L_X, L_Y, L_Z. The x and y dimensions of the computational domain are periodic. Substrate and biomass which move beyond the x and y boundaries reappear at the opposite boundary. Bacteria, substrates and other material are assumed to be washed away once they reach the z boundary (detachment layer).

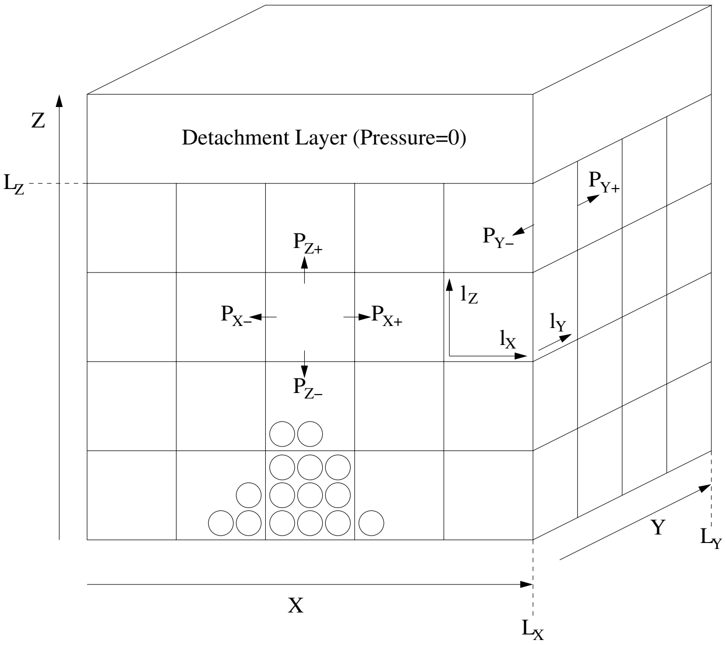

Figure 1: Computational domain

At any given point the state of the model is specified by: the amount and distribution of each type of biomass, the number of up and down-regulated cells of each biomass type, the amount and distribution of extracellular polysaccharide, the concentration and distribution of each type of substrate and quorum sensing molecule, and the pressure distribution. In section 2.1 we consider the biomass state and in section 2.3 we consider the substrate and QSM concentrations and the the pressure distribution.

2.1 Bacterial particles

The model contains zero or more bacterial cells of each biomass type. For efficiency of computation, individual cells are aggregated into bacterial 'particles' as in [8]. Each particle represents a variable number of cells of a single strain.[1] The smallest possible particle is equivalent to a single cell; the largest possible particle is that which will fit in a voxel (see section 2.3). Particles allow aggregated models of of continuous processes (growth, division and displacement) for small collections of cells, and also facilitate visualisation of the relative proportions of different biomass types within a voxel.

[1] In [8] there is also a single type of inert biomass for all particles. The total mass of a particle is the sum of its active biomass and its inert biomass. However, it's not clear what the inert biomass represents in the Picioreanu *et. al.* model, i.e., whether it is dormant or dead cells. We therefore ignore inert biomass, and instead assume that the growth of the biomass in a particle reduces over time (as a result of reduced substrate concentration) until it eventually becomes dormant.

The radius of a particle j at a given timestep is denoted r_j. As j consumes substrate its radius increases until it reaches the maximum particle radius, R, at which point it divides resulting in the creation of a new particle. R is a user-specified parameter which specifies the maximum radius of a particle in microns. Particles of all biomass types (including EPS) are assumed to have the same maximum radius, but their density and the number of bacterial cells they represent depends on the biomass type. Particles of different biomass type with equal radius may therefore have different mass and represent different numbers of cells. For strains of bacteria with larger cells the corresponding particles will contain fewer cells; conversely strains with smaller cells will have particles that contain a larger number of cells.

The mass of a particle j of biomass type b at a particular timestep is denoted m_j and is given by

$$m_j = \frac{4}{3}\pi r_j^3 \rho_b \tag{2.1}$$

where ρ_b is the density of biomass of type b, and the maximum mass of a particle of biomass type b (its mass at division) is

$$M_{b,max} = \frac{4}{3}\pi R^3 \rho_b \tag{2.2}$$

At any given timestep m_j lies between the minimum particle mass after division, $M_{b,min}$, and $M_{b,max}$. $M_{b,min}$ is equal to $0.4M_{b,max}$ (see section 3.3).

The total mass of biomass b in a voxel e at a particular timestep is denoted $m_{b,e}$ and is given by $\sum_{j=1}^{n_{b,e}} m_j$, where $n_{b,e}$ is the total number of particles of type b in e at this timestep, and the total mass of biomass type b in the model at this timestep, m_b is $\sum^{e} \sum_{j=1}^{n_{b,e}} m_j$.

The number of cells represented by a particle j of biomass type b at a particular timestep is denoted by n_j, and is given by

$$n_j = (m_j \mod M_{b,av}) + 1 \tag{2.3}$$

where $M_{b,av}$ is the average mass of a cell of biomass type b.

Cells within a particle can exist in two different states: up-regulated and down-regulated. Particles keep track of the number of up-regulated and down-regulated cells they currently contain and cells can change from one state to another at each timestep (see section 3.6). The number of up-regulated cells in a particle j is denoted u_j and the number of down-regulated cells is denoted by d_j cells.

2.2 EPS particles

Up-regulated cells produce extracellular polysaccharide (EPS) (see section 3.7). The EPS produced by cells in biomass particles is aggregated into EPS particles. EPS particles are assumed to have mass M_{EPS}, where M_{EPS} is the mass of a particle of radius R and density ρ_{EPS}. The number of EPS particles at any given point is therefore simply the quantity of EPS $\mod M_{EPS}$.

2.3 Voxels

The biofilm compartment is discretised into sub-compartments or 'voxels' containing particles, substrate and signalling molecules. We assume that voxels are cubes and that all voxels are the same size. The size of the computational domain (L_X, L_Y, L_Z) is assumed to be an integer multiple of the size of a voxel l_X.

The bulk liquid compartment is represented by a single point for the purposes of discretisation, and that this point is adjacent to all the voxels immediately below the detachment layer. The state of the model in the bulk liquid compartment is given by S constant substrate concentrations, $C_1^{bulk}, C_2^{bulk}, \ldots, C_S^{bulk}$.

Substrate and QSM concentrations are specified relative to a voxel. Each voxel has its own concentrations of the S different substrates and the Q different Quorum Sensing Molecules. The concentration of substrate s in voxel e is denoted by $c_{s,e}$ and the concentration of QSM q in e is denoted $a_{q,e}$. Substrate and QSM concentrations are assumed to be constant throughout the volume of the voxel, and the upper bound on the size of a voxel is chosen such that the substrate and QSM concentration values are 'reasonably close' to the continuous values. The size of voxels l_X is chosen appropriately for the system to be modelled, with smaller values (criteria being deduced from the corresponding continuum models) giving greater resolution at increased computational and communication cost. However the voxels will typically be fairly large in relation to the size of a cell, e.g., each voxel may contain up to 10^4 cells.[2]

The upper bound on voxel size, l_{max}, is given by the equation:

$$l_{max} = \sqrt{\frac{D_{min}}{K \times h}} \tag{2.4}$$

where D_{min} is the smallest diffusion coefficient of all substrates, K is the consumption rate of that substrate and h is the number of particles per unit volume.

In addition the substrate and QSM, each voxel also contains zero or more particles of each biomass type (including EPS). The particles in a voxel exert a 'pressure' on the particles in the neighbouring voxels which is a function of the relative number of particles in the voxels, and these pressures are used to displace particles during the division of biomass. Each voxel has six adjacent voxels, connected at each face, which are considered in determining relative pressures, and into which particles may be displaced. The pressure in voxel e at a given timestep is denoted p_e. Voxels have a pre-determined maximum particle capacity, N, and the pressure in the voxel is considered to be infinite when this maximum is reached. N is calculated using l_X and the maximum radius of a particle, R, assuming simple cubic packing. EPS particles behave in the same way as biomass particles for the purposes of pressure calculation. The arguments used in developing this pressure model are again based on the continuum modelling, in this case building on multiphase formulations for growing populations such as those described in [1].

Each particle has a notional 3D position within its containing voxel which is used for visualisation purposes. These notional positions are chosen such that the particles don't overlap. The pressure model and chosen maximum particle size (see section 3.4) should ensure that there is enough free space in the box for this to be possible.

Note that the resolution of the model with respect to substrate and QSM concentrations is determined by the size of voxels, l_X. The resolution of the model with respect to the distribution of biomass is also determined by the size of a voxel, in that the mass of each biomass type in each voxel is known. Particles and their 3D positions simply make it easier to visualise the distribution of different types of biomass.

[2] Current experiments use a $17 \times 17 \times 17$ micron voxel as in [8], and a maximum cell radius of 0.756 microns, which allows 1371 cells per voxel.

At any given point the state of the model is specified by: the amount and distribution of each type of biomass, the number of up- and down-regulated cells of each biomass type, the amount and distribution of extracellular polysaccharide, the concentration and distribution of each type of substrate and quorum sensing molecule, and the pressure distribution.

3 Model evolution

There are three main processes which determine the evolution of the model: the diffusion of substrate and QSM from voxel to voxel, the displacement of particles between voxels in response to proliferative pressures, and changes in the state of the particles themselves in response to the substrate and QSM concentrations in their containing voxel. The transport of substrate and QSM between voxels corresponds precisely to a simple central-difference discretisation of the relevant continuum reaction-diffusion equations. The pressure model underlying particle displacement builds on multiphase formulations for growing populations as described in [1]. Particles are modelled as agents and implement a simple model of growth and division similar to that in [6], and up-regulation (e.g., the production of extracellular polysaccharide) in the presence of QSM [9]. These processes interact: particles consume substrate and produce QSM, leading to transport associated with the diffusion gradients. Consumption of substrate results in particle growth, which in turn results in increased pressures and particle displacement. Finally, the number of cells in a voxel determines QSM production and hence QSM concentration and up-regulation.

In the remainder of this section we discuss the operation of each of these processes in detail.

3.1 Diffusion

Diffusion is performed globally over all voxels. The diffusion algorithm iterates over all voxels using the concentration of substrate in the neighbouring voxels to determine the change in concentration at the current voxel. More specifically, for each substrate s, the change in concentration at a voxel e is:

$$\frac{dc_{s,e}}{dt} = D_s \left(\sum^{i} \Delta c_{s,i} \right) \frac{dt}{(dx)^2} \tag{3.1}$$

where $c_{s,e}$ is the concentration of substrate s is voxel e, $\sum^i \Delta c_{s,i}$ is the difference in concentration between the voxel e and each of its neighbours ($i = 1, \dots, 6$) and D_s is the diffusivity of substrate s. The bulk liquid compartment is assumed to have constant concentration and the substratum is assumed to have zero concentration. The x and y boundaries are treated as periodic, i.e., the concentration at $(-1, y, z)$ is the same as the concentration at (L_X, y, z). The volume fraction of cells is ignored when doing diffusion calculations—the uptake of substrate by cells (see section 3.2) is much more important.

The concentration at the next timestep can then be calculated as:

$$c'_{s,e} = c_{s,e} + \frac{dc_{s,e}}{dt} \tag{3.2}$$

The diffusion algorithm repeatedly computes the concentration in each voxel using equation (3.2) until the maximum change in concentration for any voxel is less than a pre-defined constant:

$$\frac{dc_{s,e}}{dt} < \delta c_{max}$$

At the voxel level, signalling molecules are treated as a substrate, i.e., the signalling molecule concentration $a_{q,e}$, is constant across a voxel and diffuses between voxels. The signalling molecule concentration(s) in a voxel at a given timestep is therefore given by the amount of signalling molecule produced by all particles in the voxel (see section 3.7) and by diffusion of signalling molecule between surrounding voxels. The diffusion coefficient for the signalling molecule q is denoted D_q.

3.2 Growth

The growth of each particle is a function of the substrate concentration(s) in the voxel containing the particle. The model of particle growth comprises three separate processes: uptake, anabolism (creation of new biomass) and maintenance [6].

At each time-step each particle consumes an amount of substrate proportional to the concentration of substrate in the voxel and the mass of the particle. The uptake of substrate for a particle of mass m_j in voxel e at a given timestep is given by the Best equation:

$$k_{s,j} = \frac{m_j V_{max}(c_{s,e} + K_m + J)(1 - \sqrt{1 - 4c_{s,e}J/(c_{s,e} + K_m + J)^2})}{2J} \tag{3.3}$$

where $V_{max} = \mu_{max}/Y_{max}$, K_m is the half saturation constant, and $J = V_{max}/A\,P$. A is the surface area of a particle[3] and P is the permeability constant, i.e., the diffusion constant during passage through the membrane divided by the thickness of the membrane. μ_{max} is the maximum growth rate. Y_{max} is the (maximum) yield at μ_{max}, i.e., the efficiency with which substrate is converted to biomass at the maximum growth rate, corrected for maintenance. The efficiency with which substrate is converted to biomass is assumed to be constant.

The anabolism of substrate into new biomass is given by:[4]

$$\frac{dm_j}{dt} = k_{s,j}\, Y_{max} \tag{3.4}$$

Maintenance is modelled as consumption of biomass, and is proportional to the mass of the particle:

$$\frac{dm_j}{dt} = m_j\, g\, Y_{max} \tag{3.5}$$

where g is the apparent maintenance rate at zero growth, i.e., $\mu_j = 0$. The overall growth, i.e., change in mass, of a single particle j is therefore:

$$\mu_j = \frac{dm_j}{dt} = (k_{s,j}\, Y_{max}) - (m_j\, g\, Y_{max}) \tag{3.6}$$

Values of V, K and J in eqns. (3.3)–(3.6) are taken from [6], which in turn are based on [4].

[3] The surface area of a particle j is taken to be that of a sphere with radius r_j rather than of a cylinder with hemispheric ends as in [6].

[4] In [6] eqn. 3.4 is incorrectly given as $k_{s,j}/Y_{max}$.

At lower concentrations, the particle grows more slowly, and at sufficiently low concentrations the particle starts to shrink (when it is unable to consume sufficient substrate to satisfy its maintenance requirement). At present we do not consider the death of particles: instead particles continually shrink until the uptake and maintenance balance and the cell becomes dormant.

As noted in section 2.1, a single particle represents a collection of cells some of which may be up-regulated and some down-regulated (the number of cells represented by a particle is given by equation (2.3)). When the number of cells represented by a particle increases as a result of growth, i.e., when m_j increases by $M_{b,av}$, a new down-regulated cell is "created" by incrementing the number of down-regulated cells in a particle, d_j.

The consumption of substrate s by all particles in voxel e is given by $k_{s,e} = \sum_{j=1}^{n_e} k_{s,j}$, where n_e is the total number of particles in voxel e at this time-step, and assuming that $k_{s,j} = 0$ for particles in e which don't consume substrate s. The consumption of substrate is given per unit volume, so the change in concentration of substrate s in voxel e as a result of consumption is given by:

$$c'_{s,e} = c_{s,e} - \frac{k_{s,e}}{{l_X}^3} \tag{3.7}$$

3.3 Division

Particle division occurs when the radius of a particle exceeds the user-specified maximum particle radius,

$$r_j > R$$

i.e., when $m_j > M_{b,max}$. At this point, the original particle is split and a new daughter particle is created in the same voxel as the original particle.[5] The mass of the daughter particle is randomly chosen between $0.4M_{b,max}$ and $0.6M_{b,max}$, with the original particle retaining the remainder of the mass.[6]

The daughter particle is also allocated (up and down regulated) cells from the original particle in proportion to its mass. If the mass of the daughter particle is $m_{j'}$, then $n'_j = \frac{m'_j}{m_j} n_j$ cells are transferred to the daughter particle.[7] The cells transferred from parent to daughter are chosen randomly using a uniform distribution.

3.4 Displacement

As we are not currently considering real positions of particles within voxels (except for visualisation purposes), spreading does not occur within a single voxel.[8] However, spreading and displacement may still occur between voxels.

Following division, particles are transferred between voxels by determining the relative particle pressures in adjacent voxels. The pressure p_e in the voxel e at a given timestep is given by

$$p_e = \frac{n_e}{N - n_e} \tag{3.8}$$

[5] The actual 3D position of the particle is not relevant, since it will be adjusted for the purposes of visualisation.

[6] If a 50–50 split is used this can lead to synchronised division.

[7] If the total number of cells to transfer is not an integer, one of the cells from the parent particle is split at random and a new extra cell is created.

[8] If we do need to model real positions of particles within a voxel, we can model spreading using radii overlap, as in [8].

where n_e is the current total number of particles of all biomass types (including EPS) in the voxel and N is the maximum number of particles in a voxel at close packing.[9] n_e is given as:

$$n_e = \sum_{b=1}^{B} n_{b,e} \tag{3.9}$$

where $n_{b,e}$ is the number of particles of biomass b in the voxel e. The pressure in the substratum is assumed to be infinite[10] and so particles cannot disperse down through the bottom layer. Particle pressure in the bulk liquid is assumed to be zero, so particles can transfer freely into the bulk liquid. Each type of biomass has a *transfer coefficient*, T_b which specifies how easy it is for biomass of that type to be displaced. The number of particles of biomass of type b to be displaced from a voxel e to the neighbouring voxel e' is then

$$\Delta n_{b,e\to e'} = T_b(p_e - p_{e'})(n_{b,e} - n_{b,e'}) \tag{3.10}$$

The total number of particles of biomass type b to be displaced out of the voxel e is given by

$$\Delta n_{b,e} = \sum^{e'} \Delta n_{b,e\to e'} \tag{3.11}$$

and the total number of particles of all biomass types to be displaced out of the voxel e is

$$\Delta n_e = \sum_{b=1}^{B} \sum^{e'} \Delta n_{b,e\to e'} \tag{3.12}$$

The individual particle(s) of each biomass type to displace are chosen randomly.

T_b must be large enough that particles will displace faster than the maximum particle division rate when the voxel is full and small enough that no more than n_e particles are transferred at any timestep. However it can be difficult to determine appropriate values of T_b in advance. In addition, the model outlined above is susceptible to discretisation effects when the number of particles in a voxel is small.

To prevent over sensitivity to values of T_b full voxels are handled specially. If a voxel contains more particles than its maximum capacity, N, i.e., $n_e > N$, the total number of particles to transfer out of the voxel, Δn_e, is increased to be at least $n_e - N$. Conversely, if T_b is too large, too many particles will be transferred at each timestep. For example, consider a situation where a voxel contains 15 particles all of biomass type b and all of its neighbouring voxels are empty. The pressure gradient is equal in all directions and so the same number of particles will be transferred in each direction. However, if $\Delta n_{b,e\to e'} > 3$ there will be insufficient particles to transfer in each direction. To avoid this, we require that no more than a given fraction of the particles in a voxel, $N_\triangle$, can be transferred in any given timestep, such that $\Delta n_{e,max} = \min(\Delta n_e, N_\triangle n_e)$.[11] The number of particles of each biomass type to transfer, $\Delta n_{b,e}$, are then scaled in proportion $\Delta n_{b,e} = \Delta n_{b,e}\frac{\Delta n_{e,max}}{\Delta n_e}$.

The appropriate number of particles of each biomass type are then selected at random for transfer at this timestep. The direction in which to displace each of these particles is chosen probabilistically, where the probability of transferring a particle to a neighbouring voxel e' is proportional to the relative pressure difference between e

[9] For implementation, when $n_e = N$, n_e is reduced slightly so that p_e becomes a large number rather than infinity

[10] In the implementation, a large number is used to approximate infinite pressure.

[11] $N_\triangle$ is currently 25%.

and e' and e and its other neighbours, i.e., the probability of a particle being displaced to voxel e' is $\Delta p_{e,e'} / \sum^{e_i} \Delta p_{e,e_i}$, where e_i are the neighbouring voxels of e such that $\Delta p_{e,e_i} > 0$. This approach also avoids any bias in the direction in which particles are transferred with small values of n_e.

3.5 Detachment of biomass

Detachment is used to keep the biofilm within the specified maximum thickness of the computational domain, L_Z, and is implemented by simply discarding particles which are displaced beyond the boundary layer, i.e., L_Z above the substratum.

3.6 Inter-cell signalling

Quorum sensing molecules (QSM) are generated by particles and provide a form of cell to cell communication known as *quorum sensing*. The molecules, which are typically different for each strain of bacteria, control a number of aspects of bacterial growth and development, including bioluminescence, population expansion by swarming, and virulence. However, only one of these, the production of extracellular polysaccharides, is currently included in the model.

The quorum sensing mechanism involves the QSMs triggering increased expression of of certain genes in the bacterium. One of the genes codes for the QSM itself, creating a positive feedback loop. The QSM therefore functions as an 'autoinducer', and bacteria will create more of the same QSM when they are surrounded by it. A cell that is in a QSM triggered state is referred to as 'up-regulated', and one that is not is referred to as 'down-regulated'. A QSM can combine with a down-regulated cell to produce an up-regulated cell and an up-regulated cell can spontaneously revert to being down-regulated (by the loss of the bound QSM). The probability of a cell changing from down- to up-regulated is given by [9]:

$$P(up) = \alpha a_{q,e} dt$$

where $a_{q,e}$ is the concentration of signalling molecule q in voxel e and α is the conversion rate of down-regulated cells to up-regulated cells due to QSM binding. A cell reverts from up-regulated to down-regulated with constant probability:

$$P(down) = \beta \, dt$$

where β is the spontaneous down-regulation rate.

A down regulated cell produces a QSM q at a low (basal) rate $Z_{q,d}$. Once the cell becomes up-regulated it produces QSM at a much higher rate of $Z_{q,u}$ ($Z_{q,u} > 100Z_{q,d}$). The amount of QSM produced by a particle is determined by the relative number of up-regulated u_j and down-regulated d_j cells within the particle:

$$z_{q,j} = u_j Z_{q,u} + d_j Z_{q,d} dt \tag{3.13}$$

and the change in the concentration of QSM q due to production by particles in voxel e (i.e., ignoring diffusion) at this timestep is then

$$da_{q,e} = \frac{\sum^j z_{q,j}}{{l_X}^3}$$

Other signalling molecules function as inhibitors, which prevent an autoinducer combining with a cell, or prevent up-regulation when a cell combines with the autoinducer. Inhibitors therefore restrict the production of QSM by the bacteria. Inhibitors are particularly interesting from a biological point of view as they offer the ability to control the regulation and hence development of the bacterial colony. The relative ease with which a QSM q and the corresponding inhibitor $\bar{q}$ can combine with a cell is denoted by γ_q and $\gamma_{\bar{q}}$. In what follows, for simplicity we assume that $\gamma_q = \gamma_{\bar{q}} = \gamma$. In the presence of inhibitor, the probability of a cell changing from down to up-regulated is given by:

$$P(up) = \alpha \frac{a_{q,e}}{1 + (\gamma(a_{q,e} + a_{\bar{q},e}))} dt$$

where $a_{q,e}$ is the concentration of signalling molecule q in voxel e, $a_{\bar{q},e}$ is the concentration of inhibitor. A cell reverts from up-regulated to down-regulated with probability:

$$P(down) = \beta \frac{1 + (\gamma a_{\bar{q},e})}{1 + (\gamma(a_{q,e} + a_{\bar{q},e}))} dt$$

We have assumed a positive feedback loop for the production of signalling molecules. In reality, bacteria sense and produce many different types of signalling molecule, and the function used to determine the amount of each signalling molecule produced by a particle at each timestep will take a variety of different signalling molecules as input, and may increase or decrease the particle's production of the molecule at the next timestep.

3.7 Extracellular polysaccharides

Up-regulated cells produce extracellular polysaccharide (EPS) at a constant rate and particles therefore produce EPS at a rate proportional to the number of up-regulated cells they currently contain. All particle types produce the same type of EPS and contribute to an overall amount of EPS within the voxel.

In the initial version of the model, EPS is not produced.

4 Model timestep

In this section we give a high level description of a single iteration of one timestep of the model.

The state of the model at a given timestep t is determined by the state of all the voxels and all the particles at that timestep. The state of each voxel e is given by the number of particles of each biomass type it contains $n_{b,e}$, and the concentrations of each substrate $c_{s,e}$ and signalling molecule $a_{q,e}$. The state of each particle j is given by its biomass type b_j, its mass m_j, the number of up- u_j and down-regulated d_j cells it represents, its containing voxel e and its (notional) 3D position within that voxel.

Determining the state of the model at $t + 1$ involves determining, for each voxel, the change in substrate and signalling molecule concentrations due to diffusion $dc_{s,e}$, consumption $k_{s,e}$ (in the case of substrate) and production $z_{q,j}$ (in the case of signalling molecule), and for each particle, its change in mass over the timestep dm_j (and hence the change in the number of cells the particle represents dn_j), and the change in the number of up- du_j and down-regulated δd_j cells it contains and the particle's containing voxel.

The processing of the voxels at timestep t occurs in two phases. The first involves the execution of the voxels to calculate the consumption of substrate by particles, particle growth and particle division. Processing of phase one within each voxel itself occurs in three steps. Firstly, the growth step increases the mass of each particle given the concentration of substrate at this timestep. (For $t = 0$, the concentrations and number of particles are taken as parameters of the simulation.) This also gives the total consumption of all substrates by all particles in the voxel at this timestep. The second step computes the production of signalling molecule by each particle in the voxel. The third step is particle division: each particle which reached the maximum allowable mass during the growth step is split into two particles, increasing the number of particles in the voxel.

The second phase of the timestep involves computing the changes in substrate and signalling molecule concentrations due to diffusion, as a result of substrate consumption and signalling molecule production at this timestep. In parallel with the diffusion calculation, each voxel also executes a displacement step, which uses the difference in pressure between the voxel and each of its neighbouring voxels to determine movement of particles between voxels at this timestep

The timestep is then incremented and the cycle repeats with the voxels using the newly calculated concentrations and particle counts.

5 Simulation termination

There are several ways in which the simulation termination condition can be specified. Firstly a simulation time can be specified at which the simulation should stop, e.g., after 72 hours of simulated time. Alternatively, termination may occur when the first particle of generation n is created. Yet another approach is to terminate the simulation when the biofilm reaches steady state. As the simulation progresses, biomass closer to the substratum will typically grow more slowly as a result of reduced diffusion and hence reduced substrate concentration until it eventually becomes quiescent (i.e., stops growing and dividing due to lack of substrate).[12] Biomass above the quiescent layer remains active and continues to grow and divide. The quiescent layer grows thicker as the cells at the bottom of the active layer are starved of substrate by new cells growing at the top of the active layer. The active layer moves upwards, leaving mature, quiescent biomass behind. At steady state, the active layer remains the same depth but constantly moves upwards. When this configuration is reached, the simulation run can be considered over. We can detect this state by monitoring the contents of the voxels. Alternatively, if the computational domain is assumed to be large enough for steady state to be reached, the simulation can simply be run until all the voxels are filled with biomass, and the last few cycles discarded.

[12] Assuming the substratum is a sufficient distance from the bulk liquid.

6 Notation

For clarity and ease of reading, the notation used here differs from that in [8]. Table 3 below shows the corresponding notation used in [8].

Parameter	Notation	Value
Size of the computation domain	L_X, L_Y, L_Z	
Number of substrate types	S	
Concentration of substrate s in bulk liquid	C_s^{bulk}	
Effective diffusivity of substrate s	D_s	
Minimum diffusivity of all substrates	D_{min}	
Number of signalling molecule types	Q	
Effective diffusivity of signalling molecule q	D_q	
Basal rate of production of signalling molecule q	$Z_{q,d}$	500 h^{-1}
Up-regulated rate of production of signalling molecule q	$Z_{q,u}$	74000 h^{-1}
Number of biomass types	B	
Density of biomass b	ρ_b	
Density of EPS	ρ_{EPS}	
Average mass of a cell of biomass b	$M_{b,av}$	
Permeability constant	P	
Maximum growth rate	μ_{max}	
Maximum yield (at μ_{max}	Y_{max}	
Maintenance rate (at $\mu = 0$)	g	
Transfer coefficient of biomass b	T_b	
Size of a voxel	l_X	
Upper bound on voxel size	l_{max}	
Number of particles at close packing (capacity of a voxel)	N	
Number of particles per unit volume at close packing	h	
Maximum radius of particle (cell radii)	R	
Minimum mass of particle of biomass b	$M_{b,min}$	
Maximum mass of particle of biomass b	$M_{b,max}$	
Mass of an EPS particle	M_{EPS}	
Maximum (?) consumption of any substrate	K	
Diffusion termination threshold	δc_{max}	
Maximum particle displacement fraction	N_{Δ}	

Table 1: Simulation constants

Parameter	**Notation**	**Units**
Radius of particle j	r_j	
Mass of particle j	m_j	
Number of cells in particle j	n_j	
Number of up-regulated cells in particle j	u_j	
Number of down-regulated cells in particle j	d_j	
Growth of particle j	μ_j	
Concentration of substrate s in voxel e	$c_{s,e}$	
Consumption of substrate s in voxel e	$k_{s,e}$	
Concentration of QSM q in voxel e	$a_{q,e}$	molecules fl^{-1}
Production of QSM q by particle j	$z_{q,j}$	molecules h^{-1}
Number of particles of biomass b in voxel e	$n_{b,e}$	
Number of particles of biomass b in voxels adjacent to voxel e	$n_{b,e\pm\{x,y,z\}}$	
Total number of particles in voxel e	n_e	
Pressure in voxel e	p_e	
Pressure difference between voxels e and e'	$p_{e,e'}$	
Number of particles of biomass b displaced from e to e'	$\Delta n_{b,e\rightarrow e'}$	
Total number of particles of biomass b displaced from voxel e	$\Delta n_{b,e}$	
Total number of particles displaced from voxel e	Δn_e	
Maximum number of particles that can be displaced from voxel e	$\Delta n_{e,max}$	
Total mass of biomass b	m_b	
Biofilm volume	v	

Table 2: Time dependent simulation values

Value	Here	[8]
Mass of a particle p	m_j	$m^(p)_X$
Mass of biomass b in particle p	j_b	$m^(p)_{X,b}$
Radius of particle p	j_r	$R^{(p)}$
Net reaction rate for generation of biomass b	r_b	$r_{X,b}$
Maximum mass of particle	M	M_X
Solute concentrations	$c_{s,e}$	$C_{S,n}$
Concentration of biomass n in bulk liquid	c_b^{bulk}	$C_n^{(b)}$

Table 3: Notational conventions in Picioreanu et. al.

Acknowledgements

This work was supported by EPSRC research grants Nos. EP/C549406/1, EP/C549414/1, GR/S82862/01 and BBSRC grant No. BB/D006619/1.

References

[1] H. M. Byrne, J. R. King, D. L. S. McElwain, and L. Preziosi. A two-phase model of solid tumour growth. *Applied Mathematics Letters*, 16(4):567–573, 2003.

[2] Ivan Chang, Eric S. Gilbert, Natalya Eliashberg, and Jay D. Keasling. A three-dimensional, stochastic simulation of biofilm growth and transport-related factors

that affect structure. *Microbiology*, 149:2859–2871, 2003.

[3] Marta Ginovart, Daniel Lopez, and Joaquim Valls. INDISIM, an individual-based discrete simulation model to study bacterial cultures. *Journal of Theoretical Biology*, 214:305–319, 2002.

[4] A. L. Koch and C. H. Wang. How close to the theoretical diffusion limit do bacterial uptake systems function? *Archives of Microbiology*, 131(1):36–42, February 1982.

[5] Krzysztof Krawczyk, Witold Dzwinel, and David A.Yuen. Nonlinear development of bacterial colony modeled with cellular automata and agent objects. *International Journal of Modern Physics C*, 14(10):1385–1404, 2003.

[6] Jan-Ulrich Kreft, Ginger Booth, and Julian W. T. Wimpenny. BacSim, a simulator for individual-based modelling of bacterial colony growth. *Microbiology*, 144:3275–3287, 1998.

[7] Ray Paton, Richard Gregory, Costas Vlachos, Jon Saunders, and Henry Wu. Evolvable social agents for bacterial systems modeling. *IEEE Transactions on Nanobioscience*, 3(3):208–216, September 2004.

[8] Cristian Picioreanu, Jan-Ulrich Kreft, and Mark C. M. van Loosdrecht. Particle-based multidimensional multispecies biofilm model. *Applied and Environmental Microbiology*, 70(5):3024–3040, May 2004.

[9] J. P. Ward, J. R. King, A. J. Koerber, J. M. Croft, R. E. Sockett, and P. Williams. Early development and quorum sensing in bacterial biofilms. *Journal of Mathematical Biology*, (47):23–55, 2003.

[10] K. Winzer, K. R. Hardie, and P. Williams. Bacterial cell-to-cell communication: sorry, can't talk now—gone to lunch! *Current Opinion in Microbiology*, 5(2):216–222, April 2002.