# Materials and Methods

Individual-based Models (IbM) are characterized for simulating the microbes in an aggregate (granule or biofilm) as a discrete entity with unique traits. Each of the microbe is a small point of reaction that shapes its kinetics function of the local environment and therefore, its growth is affected by the activity of the microbes in the surroundings. Thus, IbM consider the intrinsic interaction between diverse microbes, either a positive (mutualism, syntrophism…) or negative (competition, ammensalism…) interaction. Diffusion of components across the simulation domain is modelled together with the biotic reactions. The substrates and products of this activity are diffusing, and these dynamics are resolved function the boundary conditions imposed over the limits of the domain. This type of models is used to describe microbial growth assuming that cellular division occurs a certain microbe's age, cycle stage or size. Therefore, division or death occurs independently for each of the cells and only function of local conditions, capturing the heterogeneity of the aggregate and supporting non-linear growth models [1].

## Simulation domain

The simulation domain is a two-dimensional ($x$, $y$) and micro-scale space (~$10^{-6}$ m) defined by the user. In it is solved the diffusion of soluble components considered, which are the substrates and products of the microbial activity. The simulation domain is divided in three different zones: the aggregate (biofilm or granule), the boundary layer and the bulk liquid (Figure 1).

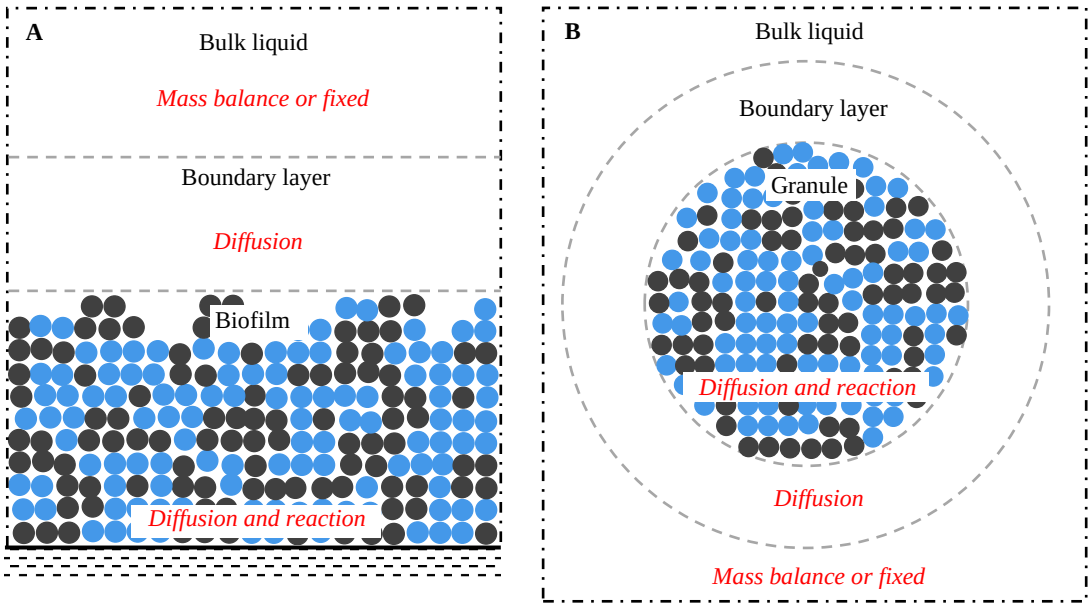


**Figure 1.** Zones of the simulation domain for a biofilm (A) and granule (B)

In the biofilm (Figure 1 A) and granule (Figure 1 B) is where the microbial activity occurs. The microbes are represented like circles with a specific radius. When they grow, they push each

other increasing the height of the biofilm or the radius of granule. Biofilm and granule increase with the microbial growth until a maximum height and radius defined by the user. Diffusion of soluble components occurs through the aggregate at the same time that they are consumed or produced by the microorganisms. The boundary layer is the surrounding space of the biofilm and granule defined to model the gradient of concentrations between the bulk liquid and the surface of the microbial aggregate. Only the diffusion of the soluble components is resolved in this space. At the outside of the boundary layer, identified as bulk liquid, it is considered that the gradient of concentration of all soluble species is negligible. The concentrations are homogenous in space, therefore there is not space discretization and the concentrations in the bulk liquid are only function of time. These can be fixed by the user or also can be calculated fixing a specific hydraulic retention time (HRT) and assuming that the activity of the aggregate modelled is representative of the whole reactor.

In microbial aggregate and boundary layer zones, diffusion equation must be solved. For this, the space is discretized in two coordinates ($x$ and $y$). In an *Excel* sheet read by MATLAB function, the length of the domain over each $x$ and $y$ coordinates are defined. Also, the number of nodes in $x$ and $y$ ($N_x$, $N_y$), and their sizes ($h_x$, $h_y$). Each of the microbe that forms the biofilm or granule is an individual with its own kinetic parameters and specific mass and position in the aggregate. As mentioned above, each microbe acts like a small reactor that grows function of the local environmental condition of the node where is its centre (Figure 2).

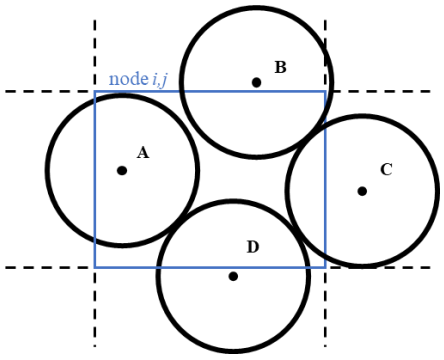


**Figure 2.** Hypothetical position of microbes in a node. Only "microbe A" belongs to node $i, j$ because its centre is inside the node

## Diffusion-reaction of soluble components

The soluble components diffuse through the aggregate, where the biotic reactions occur. To describe it, the Fick's second law equation is integrated over the time ($t$) and in the two dimensions ($x$ and $y$) for each of the soluble components considered. To it, a term of reaction is added function of the position in the space and time ($R(x,y,t)$) (equation 1).

$$\frac{\partial}{\partial t}\phi(x,y,t) = \mathbb{D}(x,y,t)\cdot\nabla^2_{xy}\phi(x,y,t) + R(x,y,t) \quad (1)$$

Where $\phi(x,y,t)$ refers to the concentration of a soluble component in a position of the simulation domain and in a time step and $\mathbb{D}$ refers to the diffusion coefficient. To solve the equation 1, the implicit Crank-Nicholson method, which is unconditionally stable [2], is used to discretize in time the diffusion term. For the reaction term, an explicit forward Euler formula is used (equation 2). This can be done, as the reaction process has a much slower time scale than the diffusion [3].

$$\frac{\phi(x,y)^{n+1} - \phi(x,y)^n}{h_t} = \frac{1}{2}\cdot\mathbb{D}(x,y)^n\cdot[\nabla^2\phi(x,y)^{n+1} + \nabla^2\phi(x,y)^n] + R(\phi(x,y)^n) \quad n \in 1, N_t \quad (2)$$

Where $h_t$ refers to the time step and $N_t$ to the total number of time steps. To approximate the Laplacian of equation 3 in a two-dimensional space ($x$, $y$), this is discretized in $N_x$ and $N_y$ nodes with $h_x$ and $h_y$ length. Therefore, for simplicity, the notation used from now on is presented in equation 3.

$$\phi^n_{i,j} = \phi(x_i, y_j, t_n) \quad (3)$$

Using centred second order finite differences (equations 4 and 5), the Laplacian matrixes [$L_x$] and [$L_y$] are built for both coordinates $x$ and $y$ (equations 6 and 7).

$$\frac{\partial^2}{\partial x^2}\phi^n_{i,j} = \frac{\phi^n_{i-1} - 2\cdot\phi^n_i + \phi^n_{i+1}}{h_x^2} + O(h_x^2) \qquad i \in 1, N_x \quad (4)$$

$$\frac{\partial^2}{\partial y^2}\phi^n_{i,j} = \frac{\phi^n_{j-1} - 2\cdot\phi^n_j + \phi^n_{j+1}}{h_y^2} + O(h_y^2) \qquad j \in 1, N_y \quad (5)$$

$$[L_x] = \begin{pmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & \ddots & \ddots & \ddots \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix} \in M_{N_x \times N_x} \quad (6)$$

$$[L_y] = \begin{pmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & \ddots & \ddots & \ddots \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix} \in M_{N_y \times N_y} \quad (7)$$

This one-dimensional operators (equations 6 and 7) can be extended to two dimensions using the Kronecker product (denoted by ⊗) to generate the two-dimensional Laplacian matrix [$A_\Delta$] (equation 8) [4].

$$[A_\Delta] = \left(\frac{1}{{h_x}^2} \cdot [L_x]\right) \otimes [I_y] + [I_x] \otimes \left(\frac{1}{{h_y}^2} \cdot [L_y]\right) \quad \in \quad M_{N_x \times N_y} \tag{8}$$

Where [$I_x$] and [$I_y$] are the identity matrixes in each *x* and *y* dimension. Owing to the applied Kronecker product (equation 8), the vector of concentrations in the simulation domain for a specific step of time *n* is defined like $\mathbf{\Phi}^n$ (equation 9).

$$\mathbf{\Phi}^n = \begin{pmatrix} \phi_{1,1}^n \\ \phi_{1,2}^n \\ \vdots \\ \phi_{1,N_y}^n \\ \phi_{2,1}^n \\ \vdots \\ \phi_{2,N_y}^n \\ \vdots \\ \phi_{N_x,1}^n \\ \vdots \\ \phi_{N_x,N_y}^n \end{pmatrix} \tag{9}$$

Then, the diffusion-reaction equation (equation 2) can be re-written in form of matrixes system (equation 10).

$$(1 - \Psi \cdot [A_\Delta]) \cdot \mathbf{\Phi}^{n+1} = (1 + \Psi \cdot [A_\Delta]) \cdot \mathbf{\Phi}^n + R(\mathbf{\Phi}^n) \cdot h_t \tag{10}$$

Where Ψ is a vector defined analogously to $\mathbf{\Phi}^n$ (equation 9) where each of the elements is $\psi_{i,j}^n$ (equation 11).

$$\psi_{i,j}^n = \frac{\mathbb{D}_{i,j}^n \cdot h_t}{2} \tag{11}$$

## Boundary conditions of simulation domain

To solve equation 10 in any simulation domain, it is necessary to define the boundary conditions of the problem. In this case, the applied boundary conditions differ if it is desired to simulate a biofilm (Figure 3) or a granule (Figure 4).

**· Biofilm modelling**

In biofilm modelling, three types of boundary conditions are defined (Figure 3). At the sides of the simulation domain (coordinate *x*) we assume a periodic boundary condition. This type of boundary is used because it is assumed that exists a larger domain (the whole reactor or natural

substratum) with similar characteristics than the ones of the simulation domain. To implement this boundary condition the second derivative of the first and last nodes at the *x*-coordinate are approximated by equations 12 and 13.

$$\frac{\partial^2}{\partial x^2}\phi_{1,j}^n \approx \frac{\phi_{N_x,j}^n - 2\cdot\phi_{1,j}^n + \phi_{2,j}^n}{h_x^2} + O(h_x^2) \tag{12}$$

$$\frac{\partial^2}{\partial x^2}\phi_{N_x,j}^n \approx \frac{\phi_{N_x-1,j}^n - 2\cdot\phi_{N_x,j}^n + \phi_{1,j}^n}{h_x^2} + O(h_x^2) \tag{13}$$

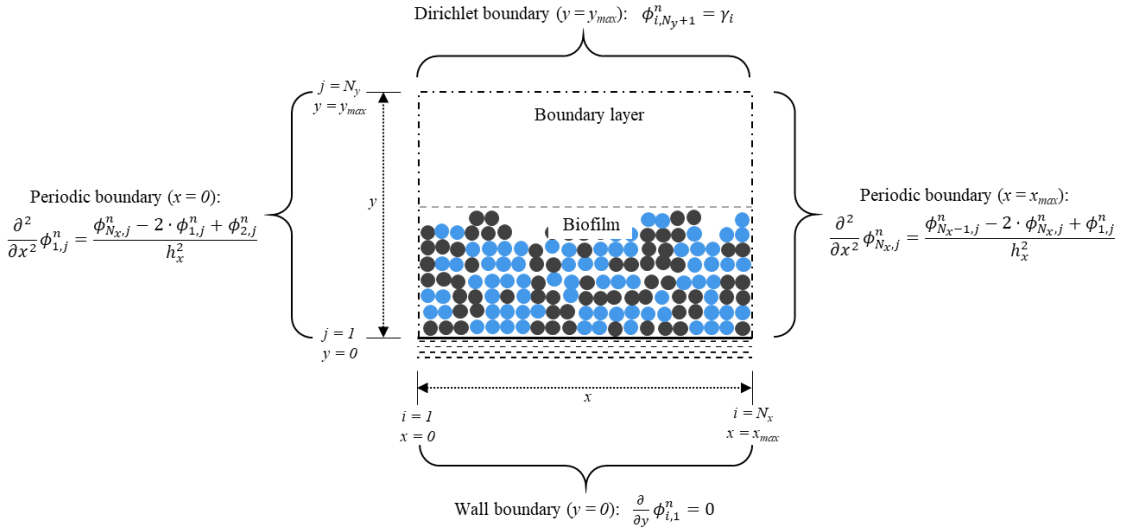


**Figure 3.** Considered boundary conditions in biofilm modelling

At the bottom of the biofilm, it is assumed a solid wall supporting the biofilm, therefore there is not transference of substrates but accumulation of them at the bottom of the simulation domain. Thus, the boundary condition is a null Neumann condition at y = 0 (j = 1, equation 14), which is approximated using a first order finite difference method (equation 15).

$$\frac{\partial}{\partial y}\phi_{i,1}^n = 0 \tag{14}$$

$$\frac{\partial}{\partial y}\phi_{i,1}^n \approx \frac{\phi_{i,2}^n - \phi_{i,1}^n}{h_y} = 0 \tag{15}$$

At the top of the boundary layer, the concentration of substrates and products are defined by the conditions of the bulk liquid. This is modelled implementing a Dirichlet condition, in which is imposed the concentration of the bulk liquid (γ) at the top of the boundary layer ($y = y_{max} + h_y$, $j = N_y + 1$, equation 16). The second derivative at the top of domain simulation is then approximated according to equation 17.

$$\phi_{i,N_y+1}^n = \gamma \tag{16}$$

$$\frac{\partial^2}{\partial y^2}\phi_{i,N_y}^n \approx \frac{\phi_{i,N_y-1}^n - 2\cdot\phi_{i,N_y}^n + \gamma}{h_y^2} + O(h_y^2) \quad (17)$$

To implement the boundary conditions at the sides and at the bottom of the biofilm, the Laplacian matrixes of both coordinates (*x* and *y*) are modified. Equation 18 shows the modification over the Laplacian in the *x*-coordinate for the sides of simulation domain and equation 19 shows the modification over the Laplacian in the *y*-coordinate for the bottom of simulation domain.

$$[L_x^*] = \begin{pmatrix} -2 & 1 & & \mathbf{1} \\ 1 & -2 & 1 & \\ & \ddots & \ddots & \ddots \\ & & 1 & -2 & 1 \\ \mathbf{1} & & & 1 & -2 \end{pmatrix} \in M_{N_x \times N_x} \quad (18)$$

$$[L_y^*] = \begin{pmatrix} \mathbf{-1} & 1 & & \\ 1 & -2 & 1 & \\ & \ddots & \ddots & \ddots \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix} \in M_{N_y \times N_y} \quad (19)$$

The modified two-dimensional Laplacian matrix ($[A_\Delta^*]$) is calculated according equation 8 using $[L_x^*]$ and $[L_y^*]$ instead of $[L_x]$ and $[L_y]$. To include the Dirichlet boundary condition at the top of the biofilm, the matrix $[b_\Delta]$ is defined (equation 20 and 21).

$$[b_y] = \begin{pmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{pmatrix} \in M_{N_y \times N_y} \quad (20)$$

$$[b_\Delta] = 0_{N_x,N_x} \otimes [I_y] + [I_x] \otimes \left(\frac{1}{h_y^2} \cdot [b_y]\right) \in M_{N_x \times N_y} \quad (21)$$

Then, the diffusion-reaction equation including the boundary conditions imposed over the simulation domain is written in equation 22. For each soluble component and each time iteration, equation 22 is solved calculating $\mathbf{\Phi}^{n+1}$.

$$(1 - \Psi \cdot [A_\Delta^*]) \cdot \mathbf{\Phi}^{n+1} = (1 + \Psi \cdot [A_\Delta^*]) \cdot \mathbf{\Phi}^n + 2 \cdot \Psi \cdot \gamma \cdot [b_\Delta] + R(\mathbf{\Phi}^n) \cdot h_t \quad (22)$$

**· Granule modelling**

In granule modelling, only one single boundary condition type must be defined (Figure 4). As in biofilm modelling, the concentrations at outside of the boundary layer are defined by the conditions of the bulk liquid. This is modelled implementing a Dirichlet boundary condition, in which is imposed the concentration of the bulk liquid (γ) at the extreme of the domain simulation (equation 23). The second derivative equations of all domain extremes are then approximated according equations 24-27.

$$\phi^n_{-1,j} = \phi^n_{N_x+1,j} = \phi^n_{i,-1} = \phi^n_{i,N_y+1} = \gamma \tag{23}$$

$$\frac{\partial^2}{\partial x^2}\phi^n_{1,j} \approx \frac{\gamma - 2\cdot\phi^n_{1,j} + \phi^n_{2,j}}{h_x^2} + O(h_x^2) \tag{24}$$

$$\frac{\partial^2}{\partial x^2}\phi^n_{N_x,j} \approx \frac{\phi^n_{N_x-1,j} - 2\cdot\phi^n_{N_x,j} + \gamma}{h_x^2} + O(h_x^2) \tag{25}$$

$$\frac{\partial^2}{\partial y^2}\phi^n_{i,1} \approx \frac{\gamma - 2\cdot\phi^n_{i,1} + \phi^n_{i,2}}{h_y^2} + O(h_y^2) \tag{26}$$

$$\frac{\partial^2}{\partial y^2}\phi^n_{i,N_y} \approx \frac{\phi^n_{i,N_y-1} - 2\cdot\phi^n_{i,N_y} + \gamma}{h_y^2} + O(h_y^2) \tag{27}$$

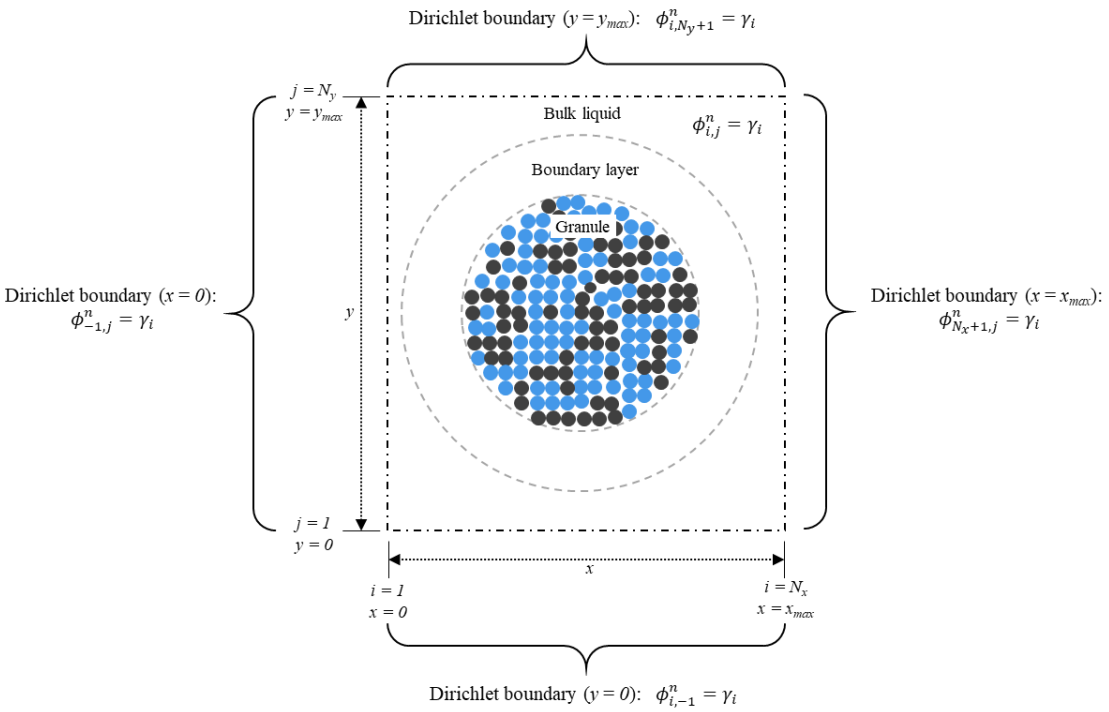


**Figure 4.** Considered boundary conditions in granule modelling

In this case, unlike biofilm modelling, the Laplacian matrixes ($[L_x]$ and $[L_y]$) are not modified because of how we have defined the Dirichlet boundary conditions (equations 25-27). To include the Dirichlet boundary condition at simulation domain extremes, the matrixes $[b_x]$ and $[b_y]$ are defined for both coordinates $x$ and $y$ (equations 28 and 29). Like Laplacian matrixes, Kronecker product is applied to generate the two-dimensional boundary matrix $[b_\Delta]$ (equation 30).

$$[b_x] = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{pmatrix} \in M_{N_x \times N_x} \tag{28}$$

$$[b_y] = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{pmatrix} \in M_{N_y \times N_y} \quad (29)$$

$$[b_\Delta] = \left(\frac{1}{h_x{}^2} \cdot [b_x]\right) \otimes [I_y] + [I_x] \otimes \left(\frac{1}{h_y{}^2} \cdot [b_y]\right) \in M_{N_x \times N_y} \quad (30)$$

Then, the diffusion-reaction equation including the imposed boundary conditions over the simulation domain is written in equation 31. For each soluble component and each time iteration, equation 31 is solved calculating $\mathbf{\Phi}^{n+1}$.

$$(1 - \Psi \cdot [A_\Delta]) \cdot \mathbf{\Phi}^{n+1} = (1 + \Psi \cdot [A_\Delta]) \cdot \mathbf{\Phi}^n + 2 \cdot \Psi \cdot \gamma \cdot [b_\Delta] + R(\mathbf{\Phi}^n) \cdot h_t \quad (31)$$

## The term of reaction

the equations 22 and 31. The reaction of the soluble components in this system is considered function of:

1. The acid-base reactions that occur in the liquid
2. The transference from the liquid to the gas phase and vice-versa
3. The microbial activity of the aggregate function of the environmental conditions

**· The acid-base reactions**

A sub-model is added to IbM to comprehensively describe the pH dynamics of the system. The influent enters in the reactor at a fix pH with a buffer of anions and cations. Then, in the system and due to the microbial activity, the pH varies function of the space and time ($pH_{i,j}^n$). As the acid-base reactions have a fast kinetics, they can be considered instantaneous and it is possible to use and algebraic algorithm to approximate it each time step of the integration [3, 5].

To describe the acid-base reactions taking place, first it is needed to define the chemical species considered and their forms (Table 1). The chemical species are the states of the model: the soluble components that the user is interested to integrate (solutions of equations 22 and 31, $\mathbf{\Phi}^{n+1}$). And some of these chemical species have different chemical forms in water function of the acid-base reactions associated to them. For example, acetic acid in solution tends to deprotonate and therefore is always in equilibrium with its deprotonated form, acetate. The concentration of these two forms is function of the pH of the liquid as the neutrality of charges is maintained in the liquid.

**Table 1.** Structure of the chemical forms of considered components in the model

| State | Chemical Forms(1) | | | | |
|---|---|---|---|---|---|
| Name | $-H_2O$ | $+H_2O$ | $-H^+$ | $-2H^+$ | $-3H^+$ |
| Acetic acid | – | $CH_3COOH$ | $CH_3COO^-$ | – | – |
| Inorganic carbon | $CO_2$ | $H_2CO_3$ | $HCO_3^-$ | $CO_3^{2-}$ | – |

(1) $-H_2O$: *Non hydrated form*
$+H_2O$: *Hydrated form*
$-H^+$: *First deprotonation*
$-2H^+$: *Second deprotonation*
$-3H^+$: *Third deprotonation*

Each of the forms included in the model is associated to its charge and its Gibbs energy value. This allows to directly calculate the acid-base constants for all the chemical species in the solution and to compute the balance of charges in the bulk liquid (equation 32). Because the equilibrium of charges occurs at a fast time scale, this balance is assumed to remain neutral [6]. Therefore, it is used to calculate the pH and the concentration of all the forms of the states, finding the root of equation 32 using an implicit Newton-Raphson scheme [7].

$$0 = [H^+] + \sum_{k=1}^{k=K} z_k \cdot [C_k] \tag{32}$$

Where K is the total number of all the chemical forms present in the bulk liquid, $C_k$ is the concentration of each of the forms, and $z_k$ is its charge. To find the root of the equation 32, it is needed to write the concentrations of the forms function the concentration of protons and the states of the model ($\Phi^{n+1}$), calculated through equations 22 or 31. The equations for each of the chemical forms are present in equations 33-38.

Not hydrated form concentration: $[Not\ Hyd.] = k_d \cdot \frac{\phi_{i,j}^n \cdot [H^+]^3}{\theta}$ (33)

Hydrated and fully protonated form concentration: $[Hyd.] = \frac{\phi_{i,j}^n \cdot [H^+]^3}{\theta}$ (34)

First deprotonated form concentration: $[1^{st} dep.] = K_{a_1} \cdot \frac{\phi_{i,j}^n \cdot [H^+]^2}{\theta}$ (35)

Second deprotonated form concentration: $[2^{nd} dep.] = K_{a_1} \cdot K_{a_2} \cdot \frac{\phi_{i,j}^n \cdot [H^+]}{\theta}$ (36)

Third deprotonated form concentration: $[3^{er} dep.] = K_{a_1} \cdot K_{a_2} \cdot K_{a_3} \cdot \frac{\phi_{i,j}^n}{\theta}$ (37)

$$\theta = (1 + k_d) \cdot [H^+]^3 + k_{a_1} \cdot [H^+]^2 + k_{a_1} \cdot k_{a_2} \cdot [H^+] + k_{a_1} \cdot k_{a_2} \cdot k_{a_3} \quad (38)$$

Where $k_d$, $k_{a1}$, $k_{a2}$ and $k_{a3}$ are the hydration and acid-base equilibrium constants calculated using the Gibbs energy values of formation of the substrate(s) and product(s).

**· Mass transference between liquid and gas phases**

Some of the soluble components might transfer in an important proportion to the gas phase. The equilibrium between the liquid and the gas, disturbed by the acid-base reactions and the microbial activity occurring in the liquid, could move towards production and/or consumption of mass that would affect the microbial growth. The Henry constant ($K_H$) and a volumetric mass transfer coefficient ($k_La$) associated to each soluble component, are used to calculate the rate of liquid transferred to the gas phase or vice-versa due to the environmental conditions in each time step ($r_{L\text{-}G}$ equations 39, 40 and 41).

$$p_{gas}{}_{i,j}^{n\,*} = \frac{\phi_{i,j}^n}{k_H} \; ; \; \phi^{n*} = K_H \cdot p_{gas}^n \quad (39)$$

$$r_{G-L,i,j}^n = k_L a \cdot \left( p_{gas}^n - p_{gas}{}_{i,j}^{n\,*} \right) \quad (40)$$

$$r_{L-G,i,j}^n = -\frac{r_{G-L,i,j}^n \cdot V_{gas}}{R_g \cdot T} \quad (41)$$

Where $p_{gas}{}^n$ is the actual partial pressure of the gas in the head space at a time *n* and $p^n{}_{gasi,j}{}^*$ the partial pressure correspondent to the ideal equilibrium with the concentration in the liquid. $R_g$ is the ideal constant of gases in units of $L \cdot atm \cdot K^{-1} \cdot mol^{-1}$, T is the temperature of the system in K and $V_{gas}$ is the volume of the head space. Assuming constant the pressure at the head space (P), it is possible to calculate the gas flow leaving the liquid phase (equation 42).

$$Q_{gas} = \frac{-(r_{L-G,i,j}{}^n < 0) \cdot r_{L-G,i,j}^n \cdot R_g \cdot T}{P} \quad (42)$$

The transference to the gas or from it ($r^n{}_{L\text{-}G,i,j}$) is calculated in each node of the simulation domain. The gas produced in each of the nodes goes to the head space of the reactor or atmosphere. But, the mass balance of the head space of the reactor or atmosphere is calculated assuming the average of gas transferred from the liquid, considering the head space homogeneous. Therefore, equation 43 shows the mass balance for any specific gas component considered.

$$\frac{dp_{gas}}{dt} = \frac{\sum_{i=1,j=1}^{i=N_x,j=N_y} r_{G-L,i,j}^n}{N_x \cdot N_y} - \frac{Q_{gas}}{V_{gas}} \cdot p_{gas}^n \quad (43)$$

This derivative is integrated over time with an explicit forward Euler scheme to simulate the dynamics of the partial pressures of all the components in the head space of the reactor or atmosphere.

**· Microbial activity (kinetics)**

The kinetics of microbes that grow in the aggregate are calculated function of the local conditions of the node where the cell is located inside the simulation domain. The kinetics parameters associated to each of the microbial species considered are included in the *Excel* file that feeds the model. But also, the *Excel* file needs to indicate the stoichiometry chosen for the anabolic (*Ana*), catabolic (*Cat*) and decay (*Dec*) processes for each of the microbial species. Equations 44-46 show examples of stoichiometries for anabolism, catabolism and decay where $C_s$ and $N_s$ refer to carbon and nitrogen sources, *eD* and *eA* to electron donor and acceptor and *X* to biomass considering an average formula of $C_1H_{1.8}O_{0.5}N_{0.2}$.

$$Ana \ = \ 1 \cdot C_s + 0.2 \cdot N_s \ + \cdots \rightarrow \cdots + \ 1 \cdot \ X \tag{44}$$

$$Cat \ = \ \alpha \cdot \ eD \ + \ \beta \cdot \ eA \ \rightarrow \cdots \tag{45}$$

$$Dec \ = \ 1 \cdot \ X + \cdots \rightarrow \ 1 \cdot C_s + 0.2 \cdot N_s \tag{46}$$

The stoichiometry of the overall metabolism (*Met*) is calculated function of the anabolic and catabolic stoichiometries and the growth yield ($Y_{XS}$, equation 47) [8, 9].

$$Met \ = \ \frac{1}{Y_{XS}} \cdot Cat \ + \ Ana \tag{47}$$

Therefore, if the cell grows, is going to do it function of the metabolic stoichiometry calculated per equation 44. But if the conditions are not favourable, then cell might start to decay function of the stoichiometry fixed per equation 46.

The value of $Y_{XS}$ can be defined initially by the user (through bibliographic review or experimental measure), but also $Y_{XS}$ can be estimated assuming that the Gibbs energy values of the anabolic and catabolic equations plus an energy dissipation term ($\Delta G_{Dis}$) give a fair approximation (equation 48). Because the Gibbs energies for the catabolic and anabolic reactions are updated in the system with the local concentrations and temperature surrounding the microbes, it is possible to approximate changes in the stoichiometry of the metabolism function the environmental conditions [6, 10].

$$Y_{XS} = \frac{\Delta G_{Cat}}{\Delta G_{Ana} + \Delta G_{Dis}} \tag{48}$$

Where $\Delta G_{Cat}$ refers to the Gibbs energy of the catabolic reaction, $\Delta G_{Ana}$ to the Gibbs energy of the anabolic reaction and $\Delta G_{Dis}$ to the energy assumed dissipated in the anabolic process. This energy is assumed only function of the carbon source used [8] (equation 49).

$$\Delta G_{Dis} = 200 + 18 \cdot (6 - NoC)^{1.8} + exp[\{(-0.2 - \gamma)^2\}^{0.16} \cdot (3.6 + 0.4 \cdot NoC)] \quad (49)$$

Where NoC refers to the number of elements of carbon that the C-source has and γ refers to the degree of reduction of the carbon source. When the carbon source is inorganic carbon, a value of 986 kJ/C-mol are assumed for energy dissipation if there is not occurrence of reversed electron transfer and 3500 kJ/C-mol if it occurs [11].

With the metabolic stoichiometry and the kinetic parameters ($\mu^{max}$, $K_S$ and $b^{max}$ (referring to the decay or maintenance)) for each of the microbial species present in the reactor, it is possible to calculate the growth rate ($\mu$) of a microbe *m* and then, through the stoichiometry and the mass of the microbe, the generation/consumption term for the biomass, substrate uptakes and product generation. Equation 50 shows the calculation of the specific growth rate for a cell *m*; equation 51 is the derivative of the mass of a microbe which is integrated in time using a forward Euler scheme; and equation 52 calculates the reaction term for a specific soluble component *s* due to the activity of the microbe *m*.

$$\mu_m^n = \mu_m^{max} \cdot \frac{\phi_{i,j}^n}{K_{S,m} + \phi_{i,j}^n} - b_m^{max} \quad (50)$$

$$\frac{dX_m}{dt} = \mu_m^n \cdot X_m^n \quad (51)$$

$$R_m^n = \frac{\mu_m^n \cdot \delta_{s,m}}{V_{xy}} \quad (52)$$

$X_m$ refers to the mass in moles of the cell *m*, $\delta_{s,m}$ to the stoichiometric coefficient of the substrate S for the cell *m* and $V_{xy}$ to the volume of one node of the simulation domain. Equation 50 would change function of the Monod or inhibition terms that are considered for the species *m*. In non-favourable conditions, the cell might not be able to harvest enough substrate or energy to grow or maintain. In that case, equation 50 returns a negative value which means that the cell is going to lose mass and shrink function of the decay stoichiometry assigned (equation 46) and with the kinetics of equation 53.

$$b_m^n = b_m^{max} - \mu_m^{max} \cdot \frac{\phi_{i,j}^n}{K_{S,m} + \phi_{i,j}^n} \quad (53)$$

The calculation of the reaction term for the soluble components is function of the position in the simulation domain (one reaction term per node for each soluble component). Therefore, equation 54 is used to consider all microorganisms that are contributing to the reaction term of a specific soluble component in the *node i,j*.

$$R_{i,j}^{n} = \sum_{m=1}^{m=K_{i,j}} R_{m}^{n} \quad (54)$$

Where $K_{i,j}$ refers to all microorganism that are in the *node i,j*. The reaction terms $R_{ij}^{n}$ calculated by equation 54 for all the nodes of the simulation domain are included in the equations 22 and 31 to calculate the concentration of each of the soluble components in all the simulation domain.

## The microbial growth

Microorganisms will change their mass function of their kinetics. For each individual, its mass is integrated in time using a forward Euler scheme on equation 51 which allows to describe the dynamics of each of the cell growth (or decay). The model assumes that once cells achieve a maximum mass ($M_{max}$) or size (density is considered constant), they should divide. In this case, a new cell is formed (cell *m+1*, equation 55) with an initial mass that is a random percentage (α is a stochastic parameter with a value between 0 and 1) of the total mass of the mother cell. The mass of the mother cell is updated with the mass remained after the division (equation 56).

$$X_{m+1} = \alpha \cdot X_{m} \quad (55)$$

$$X_{m} = (1-\alpha) \cdot X_{m} \quad (56)$$

At the same time, cells may shrink if they do not have favourable conditions and equation 50 returns negative values. When cells reach a minimum mass ($M_{min}$) or size fixed which is considered negligible, they are removed from the system (Figure 5).

Once the mass of each cell is known, it is possible to estimate how much space a cell is occupying calculating their volume assuming they have the shape of a perfect circle and a specific density [12]. When a cell divides, it is randomly assigned a position for the new individual in the neighbourhood of its mother. Also, when a microorganism disappears, a hole is formed on the granule. Therefore, the dynamics of the cell growth implies certain shoving of the individuals inside the granule. The model assumes only certain percentage of overlapping allowed fixed by the user. Therefore, when a cell divides, a shoving algorithm checks the overlapping of the individuals. When the microbial overlap it is bigger of the maximum overlap accepted, cells move.

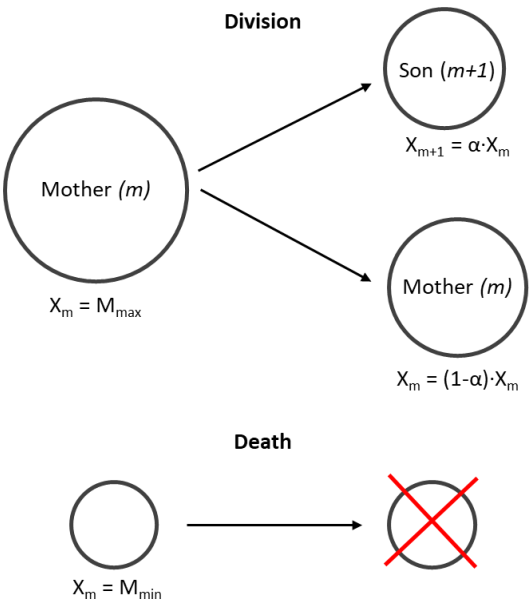


**Figure 5**. Microbial division and death. α is any stochastic value between 0-1. $M_{max}$ and $M_{min}$ refer to the maximum and minimum mass that microbe can reach, respectively.

To compute the shoving of the cells in the aggregate after cell division, first the overlap between microorganisms is checked as per equation 58.

$$|\vec{v}| = \sqrt{(x_m - x_{m+1})^2 + (y_m - y_{m+1})^2} \tag{57}$$

$$overlap = (r_m + r_{m+1}) - |\vec{v}| \tag{58}$$

Where $|\vec{v}|$ is the norm of the vector that links the centres of both cells *m* and *m+1*. If the *overlap* value is bigger than the distance allowed by the user, then microorganisms are pushing each other function their mass and their distance (equations 59-64).

$$\vec{p} = \frac{(r_m + r_{m+1}) - |\vec{v}|}{|\vec{v}|} \tag{59}$$

$$a_m = \frac{X_m}{X_m + X_{m+1}} \; ; \; a_{m+1} = \frac{X_{m+1}}{X_m + X_{m+1}} \tag{60}$$

$$x_{new,m} = x_{old} - (x_{m+1} - x_m) \cdot a_m \cdot \vec{p} \tag{61}$$

$$y_{new,m} = y_{old} - (y_{m+1} - y_m) \cdot a_m \cdot \vec{p} \tag{62}$$

$$x_{new,m+1} = x_{old} + (x_{m+1} - x_m) \cdot a_{m+1} \cdot \vec{p} \tag{63}$$

$$y_{new,m+1} = y_{old} + (y_{m+1} - y_m) \cdot a_{m+1} \cdot \vec{p} \tag{64}$$

## The update of the Dirichlet boundary condition

When the modelled aggregate (biofilm or granule) is considered that growing inside of a reactor, it is assumed that conditions in the bulk liquid are changing due the activity of the whole biomass in it. Considering the activity calculated in the aggregate is representative of the activity of the whole reactor, its average activity is used to integrate the concentration of the soluble components in the bulk liquid of the reactor (S) through a mass balance (equation 65).

$$\frac{dS}{dt} = \frac{1}{HRT} \cdot \left(S_{inf} - S\right) + R \tag{65}$$

Where HRT refers to the hydraulic time fixed in the reactor, $S_{inf,i}$ to the concentration in the influent and $R$ to the reaction term considered in reactor. $R$ is calculated assuming the average of the reaction terms calculated for the granule is representative of the whole reactor activity (equation 66).

$$R = \frac{\sum_{i=1,j=1}^{i=N_x,j=N_y} R_{i,j}^{n}}{N_x \cdot N_y} \tag{66}$$

The concentrations in the influent and the hydraulic retention time are defined by the user. Nevertheless, this last one, can also be dynamically imposed by imposing fixed a concentration of a specific substrate in the reactor (equation 67).

$$HRT = \frac{(S_{inf} - S)}{R_{r,S}} \tag{67}$$

The new concentration S calculated with the integration of equation 65 and using an explicit forward Euler scheme, updates the Dirichlet value (γ) included in equations 22 and 31.

## The integration

Cells are dividing in a time scale much slower than the diffusion-reaction process (~1 hour versus ~$10^{-8}$ hours). To solve the system, the model takes advantage of this time scale differentiation to separate the processes of solving the diffusion-reaction equation and the cell division and its shoving [12]. The overall scheme of the model is presented in Figure 6.

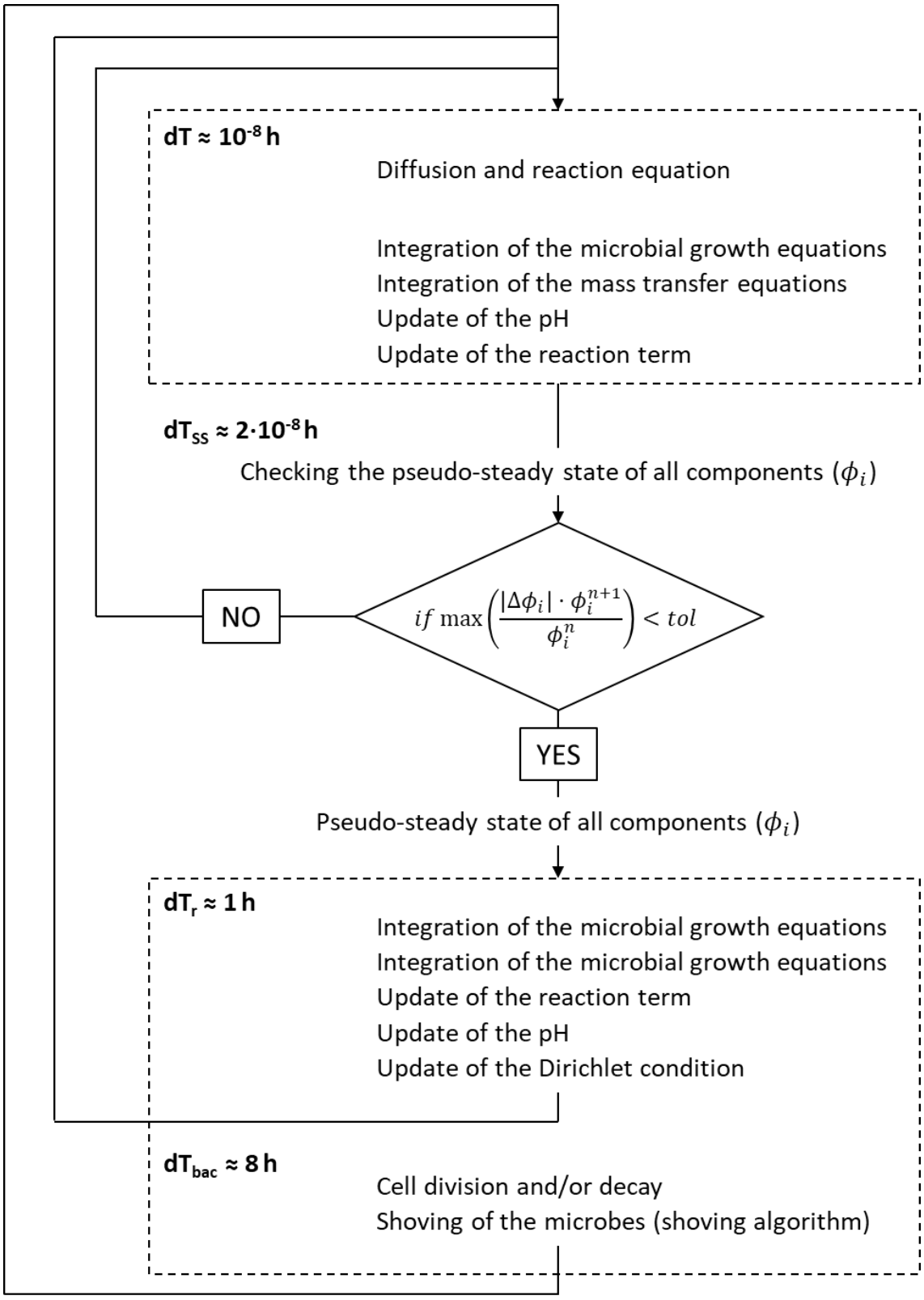


**Figure 6.** Algorithm scheme of the integration process

First, the diffusion-reaction equation is integrated (with a time step *dT*) until it reaches a pseudo-steady state, in which the gradients of the soluble concentrations are varying less than a fixed tolerance defined by the user (*tol*, Figure 6) with each step integrated. The integration

continues updating each *n*-iterations the reaction term togheter with the integration of the microbial growth and the gas-liquid mass transfer and finally, updating the pH in all the nodes of the simulation domain. When this pseudo-steady state is reached then, the mass balances of the overall reactor are integrated in a much bigger time step ($dT_r$), function of the average microbial activity of the aggregate. Also, the biomass growth, and liquid-gas transference are integrated in this bigger time step. At the end of this bigger step, the Dirichlet boundary condition and the reaction term are updated, therefore, the diffusion-reaction equation needs to be integrated again to reach a next pseudo-steady state. Each *n* times that the diffusion-reaction equation reaches a pseudo-steady state ($dT_{bac}$), the cell division is checked and if it happens, the algorithm of the microbial shoving is launched.

## References


1. Hellweger FL, Clegg RJ, Clark JR, Plugge CM, Kreft JU. Advancing microbial sciences by individual-based modelling. *Nat Rev Microbiol* 2016; **14**: 461–471.
2. Thomas JW. Numerical Partial Differential Equations: Finite Difference Methods. 1995. Berlin, New York: Springer-Verlag.
3. Kissel JC, Mccarty PL, Street RL. Numerical simulation of mixed-culture biofilm. *J Environ Eng* 1984; **110**: 393–411.
4. Lynch RE, Rice JR, Thomas DH. Tensor product analysis of partial difference equations. *Bull Am Math Soc* 1964; **70**: 378–384.
5. Batstone DJ, Amerlinck Y, Ekama G, Goel R, Grau P, Johnson B, et al. Towards a generalized physicochemical framework. *Water Sci Technol* 2012; **66**: 1147–1161.
6. González-Cabaleiro R, Lema JM, Rodríguez J. Metabolic energy-based modelling explains product yielding in anaerobic mixed culture fermentations. *PLoS One* 2015; **10**: 1–17.
7. Ypma TJ. Historical development of the Newton-Raphson method. *Soc Ind Appl Math* 1995; **37**: 531–551.
8. Kleerebezem R, Van Loosdrecht MCM. A generalized method for thermodynamic state analysis of environmental systems. *Crit Rev Environ Sci Technol* 2010; **40**: 1–54.
9. Heijnen JJ, van Dijken JP. In search of a thermodynamic description of biomass yields for chemotrophic growth of microorganisms. *Biotechnol Bioeng* 1992; **39**: 833–858.
10. González-Cabaleiro R, Ofiţeru ID, Lema JM, Rodríguez J. Microbial catabolic activities are naturally selected by metabolic energy harvest rate. *ISME J* 2015; **9**: 2630–2641.
11. Tijhuis L, Van Loosdrecht MCM, Heijnen JJ. A thermodynamically based correlation for maintenance gibbs energy requirements in aerobic and anaerobic chemotrophic growth. *Biotechnol Bioeng* 1993; **42**: 509–519.
12. Kreft JU, Picioreanu C, Wimpenny JWT, Van Loosdrecht MCM. Individual-based modelling of biofilms. *Microbiology* 2001; **147**: 2897–2912.