

# **Microbial biofilms structure and dynamics**

## **Simulator User's Guide and Model Documentation**

**N.Mabrouk<sup>(1,2)</sup>, G.Deffuant<sup>(1)</sup>, C.Lobry<sup>(2)</sup>**

### **(1) Laboratoire d'Ingénierie des Systèmes Complexes**

Cemagref, 24 avenue des Landais, Aubière – France

### **(2) Water resource modeling team**

Inria/Inra, 2 pl. Viala 34060 Montpellier - France

September 2007

# Microbial biofilms structure and dynamics

## User's Guide and Model Documentation

### Introduction

We developed an individual-based model to investigate the effect of local bacteria interactions on the global biofilm structure and dynamics. We implemented the model in Java programming language using the Mason framework<sup>1</sup>. To facilitate the communication we named the program **Bacson**. In this document we present the **Bacson** simulator and the underlying individual-based model. The document is organized in three chapters:

- Chapter 1 : Model documentation – describes the model following ODD-protocol
- Chapter 2 : Installation - procedure to install **Bacson** on your pc
- Chapter 3 : User Guide – procedure to use **Bacson** for simulating microbial biofilms

---

<sup>1</sup> <http://cs.gmu.edu/~eclab/projects/mason/>

# Chapter 1: Model documentation

We use an individual-based model to investigate how bacteria interactions affect microbial biofilms structure. The studied system is basically a community of bacterial cells encased in a self-produced viscoelastic polymeric matrix. Such systems (i.e. biofilms) are ubiquitous in nature and have a large and varied role in human activities. The two last decades have shown the development of powerful experimental tools that allowed a detailed experimental examination of biofilms. It became clear that biofilms are highly complex morphological structures with pores and water channels. The mechanisms leading to those complex structures still remain unclear.

In this paper we present a simple individual-based model for the simulation of a microbial biofilm. The model is described according to the ODD framework<sup>2</sup>.

## The Model

### Purpose

Biofilm reactors are systems used in several industrial applications for the cultivation of bacterial cells. The specificity of these systems is that the reactor is generally filled with an inert material typically plastic rings on which surface the bacterial cells adhere naturally and form a complex spatial structure called biofilm. The biofilm structural properties (like, porosity, density...) significantly affect the bioreactor performances. These properties are the result of the local interactions between the bacterial cells and their environment. Our aim is to investigate the relationship between the bacterial cells interaction and the biofilm structural properties using an individual-based model in order to develop controlling strategies of the biofilm structure.

### State variables and scales

The model has two observation levels: the low-level (individual-level) and the system level

Low level: the main state variable at the low level are the attributes of the individuals and the local concentrations of food and polymer (see table 1). The individuals (bacterial cells) are modeled as Brownian particles that grow and divide. They are described by their positions in the spatial domain and their individual masses (or diameters). The individuals are represented as cylinders with a constant height ( $h=1\mu\text{m}$ ) and a variable diameters proportional to their masses (see figure 5).

---

<sup>2</sup> GRIMM V, Berger U, Bastiansen F, Eliassen S, Ginot V, Giske J, Goss-Custard J, Grand T, Heinz S K, Huse G, Huth A, Jepsen J U, Jørgensen C, Mooij W M, Müller B, Pe'er G, Piou C, Railsback S F, Robbins A M, Robbins M M, Rossmannith E, Rüger N, Strand E, Souissi S, Stillman R A, Vabø R, Visser U and DeAngelis D L (2006). **A standard protocol for describing individual-based and agent-based models**. Ecological Modelling. 198 (1-2), 115-126.

Variable	Name	Description	Units
$(x_k, y_k, m_k)$ $k=1..N$ ( $N$ : total number of bacterial cells)	Bacterial cell attributes	A bacterial cell $k$ is totally characterised by its spatial coordinates $(x_k, y_k)$ and its mass $m_k$ : <ul style="list-style-type: none"> <li>● <math>x_k</math>: real-value <math>0 \leq x_k &lt; L</math></li> <li>● <math>y_k</math>: real-value <math>0 \leq y_k &lt; W</math></li> <li>● <math>m_k</math>: real value <math>m_r \leq m &lt; 2m_r</math></li> </ul> $L$ and $W$ denote respectively the length and width of the spatial domain and $m_r$ is a constant (the mass of the cell “at birth”)	(m, m, kg)
$s_{i,j}$	Local food concentration	Food concentration at the grid cell $(i,j)$ $i=0 .. L/h$ $j=0.. W/h$ where $h$ denotes the space step used for the discretisation of the spatial domain into a grid lattice.	kg/m <sup>3</sup>
$p_{i,j}$	Local polymer concentration	Polymer concentration at the grid cell $(i,j)$	kg/m <sup>3</sup>

Table 1: Low-level state variables

Note:

The size of the matrices  $s$  and  $p$  depend on the spatial discretization (denoted  $h$ ) used in solving the diffusion equations for substrate and polymer

System level The simulated world is a continuous thin 3-dimensional domain (see figure 1). This choice is not arbitrary by motivated by the following three reasons: first, several experimental devices reported in bibliography<sup>3</sup> and used for studying biofilm formation consist in a thin microreactor equipped with a microscope and a camera. Second, by choosing a 3 dimensional domain we avoid the representation of abiotic state variables in unrealistic units (for example concentrations would be expressed in kg/m<sup>2</sup> in the case of a 2-dimensional spatial domain). Third, the fact that this domain is thin allow us to simplify most of the model processes by reasonably neglecting the effect of the domain height (for example bacteria motion and diffusion equations can be written in a 2-dimensional form)

<sup>3</sup> Lewandowski, Z, Beyenal, H., Stoodey, D. (2004) Reproducibility of biofilm processes and the meaning of steady state in biofilm reactors. water Science and Technology, vol 49, pp 359-464

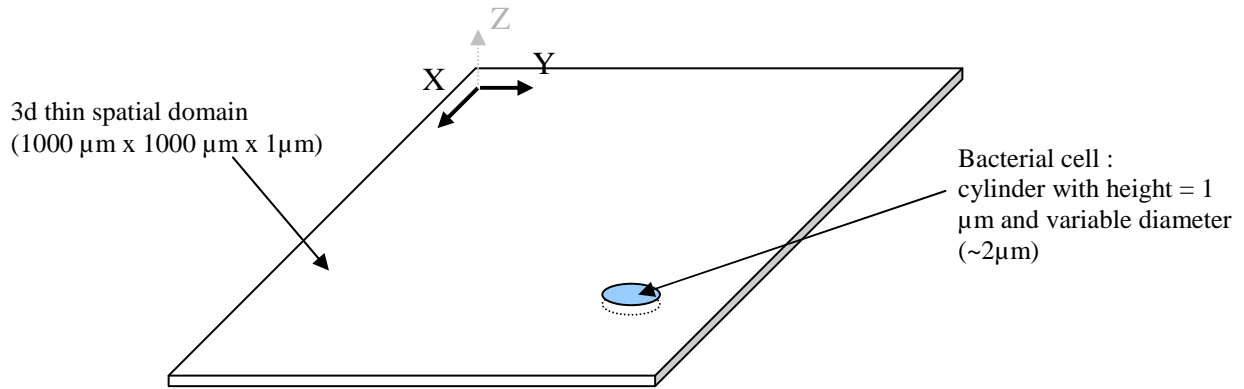


Figure 1. Schematic representation of the simulated world and a bacterial cell

The spatial domain is continuous for the individuals but discrete for the food and polymer concentrations. This means that individuals can occupy any real-valued position within the spatial domain whereas food and polymer concentrations are calculated on a discrete number of grid cells. Though food and polymer are continuous variables the spatial discretization is needed to numerically solve nutrient and polymer diffusion equations, as shown further, using a finite difference scheme.

The state variables calculated at the system level are mainly the mean concentrations of product, food and biomass. Additional system level state variables that may reflect some interesting properties of the biofilm spatial pattern (for example spatial entropy which gives an indication about the complexity of the observed spatial pattern<sup>4</sup>) can be calculated using the low-level state variables (bacteria positions, food and polymer maps). Table 2 lists the basic system level state variables calculated by the simulator.

Variable	Description	Units
$N$	Total number of individuals	-
$s_{\text{mean}}$	Mean food concentration	kg/m <sup>3</sup>
$p_{\text{mean}}$	Mean polymer concentration	kg/m <sup>3</sup>
$b_{\text{mean}}$	Mean biomass concentration	kg/m <sup>3</sup>

Table 2: System-level state variables

### Process overview and scheduling

Figure 2 summarize the different processes scheduled in our model. These processes can be classified in two groups: processes related to the individuals (showing relaxation, biological growth and mobility) and processes related to the spatial domain (substrate/polymer diffusion, feeding process and detachment of biofilm).

<sup>4</sup> Xinmin Yang, Haluk Beyenal, Gary Harkin, and Zbigniew Lewandowski. **Quantifying biofilm structure using image analysis**. Journal of Microbiological Methods, 39:109–119, 2000.

Group 1: Individual level processes (table 3)

**Mobility:** The bacterial cells are modeled as Brownian particles which mobility is described by diffusion constant  $D_f$ .  $D_f$  depends on the local concentration of polymer such that a high polymer concentration reduces the bacteria mobility

**Biological growth and division:** the bacteria consume locally available substrate and use part of it ( $Y_s$ ) for growth and the other part ( $Y_p = 1 - Y_s$ ) for the production of a polymer released in the environment. When the cell size reaches a critical mass (twice it's initial mass), it divides into two identical daughter cells

**Shoving relaxation process:** when two bacterial cells overlap, they are artificially pushed away from each other

Group 2 : system-level processes (table 4)

**Substrate diffusion:** the substrate transfer is described by a diffusion reaction equation with user specified boundary conditions

**Polymer diffusion:** the substrate transfer is described by a diffusion reaction equation with user specified boundary conditions

**Detachment:** a detachment event consists of removing a part of the biofilm. In real system detachment process is mainly due to shear effect.

**Feeding:** The system can be continuously fed with substrate. During the feeding process the bacteria which are not enclosed in a polymeric matrix can be washed out of the domain

The time scale is discretized with a constant time step  $\Delta t$  also used for the discretization of the diffusion equations of food and polymer. At every time step the new state of the system is calculated.

Process	Description
<b>Shoving relaxation process</b>	Check if the individuals overlap. If it the case, the individuals are moved in the opposite direction to relax the overlapping
<b>Biological growth process</b>	Let every individual consume locally available food, produce polymer and update its mass. If the new mass is high enough the individual give birth to a new bacteria (division)
<b>Brownian motion of individuals</b>	Calculate the new position for each individual

*Table 3: Individuals processes description*

<b>Process</b>	<b>Description</b>
<b>Substrate diffusion process</b>	The process solves the 2D substrate diffusion equation
<b>Polymer diffusion process</b>	The process solves the 2D polymer diffusion equation
<b>Feeding process</b>	This is the feeding process. The simulated world can be continuously (and uniformly) fed with food
<b>Biofilm detachment process</b>	A detachment event consist in emptying a grid cell from its polymer and individuals content

*Table 4: Spatial processes description*

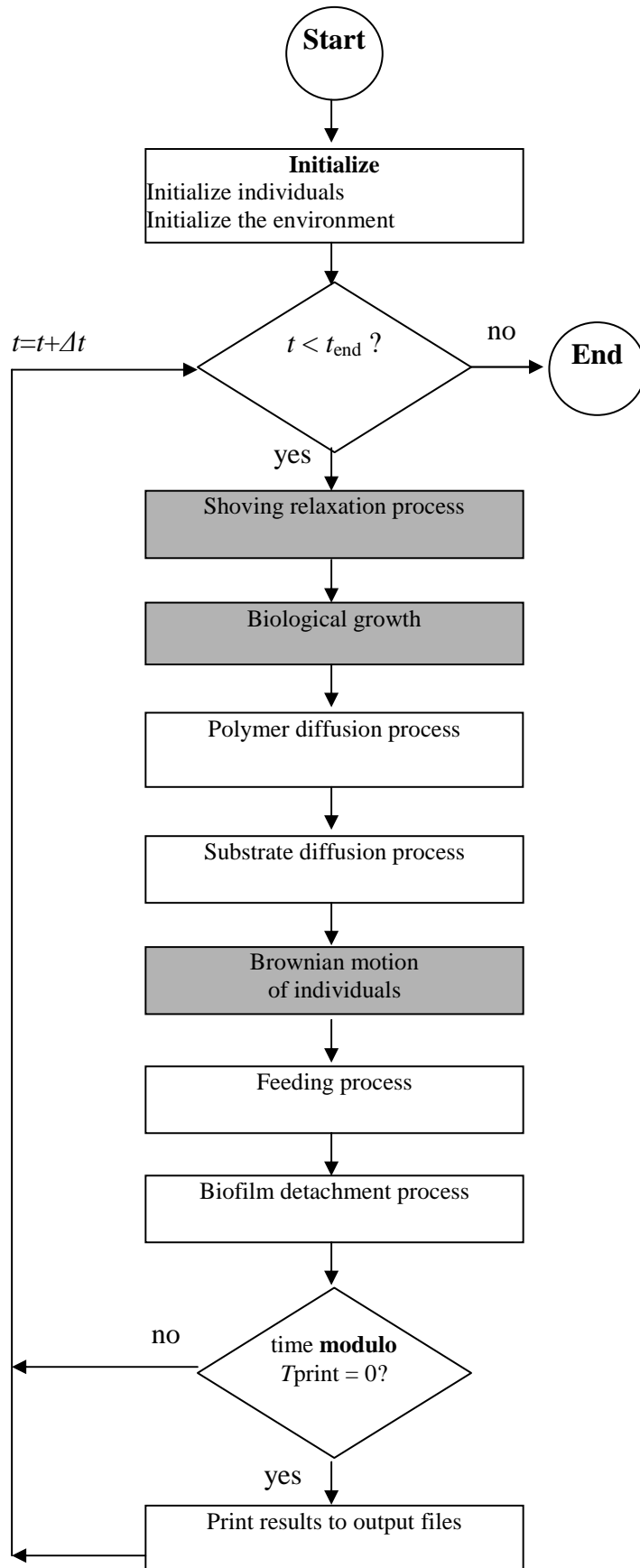


Figure 2: The sequence of events during each time step. A dark box indicates that the event is performed for each individual in the individuals list.

## Design concepts

Bacterial cells are purely reactive. They react to local substrate and polymeric substance concentration. The local substrate concentration affects the growth rate of the bacterial cell whereas the polymeric substance concentration affects the cell mobility. The bacterial cells have no explicit aim or any adaptive process.

### *Emergence*

The biofilm structure emerges from local interactions between the bacterial cell and the environment (figure 3). The biofilm is defined as the regions where the polymeric substance concentration is higher than a threshold. Experimental evidence showed that a biofilm is mainly formed by the polymeric substance and that most of the structural properties of the biofilm are defined by the properties of the polymeric matrix.

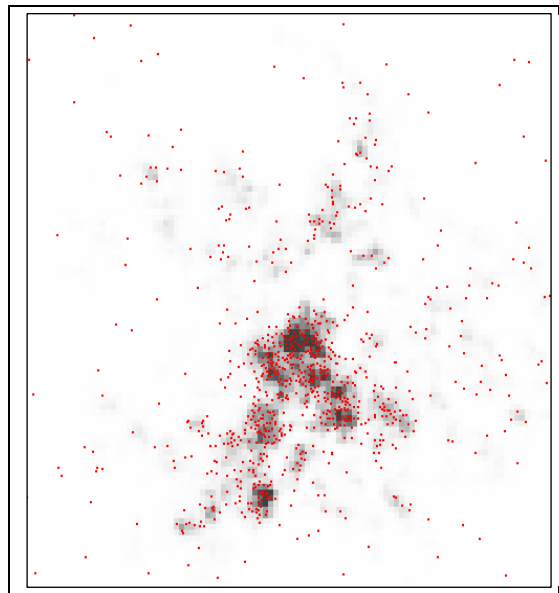


Figure 3: Emergence of the biofilm spatial pattern from individual interactions (Bacteria: red dots, Polymeric substance: grey level)

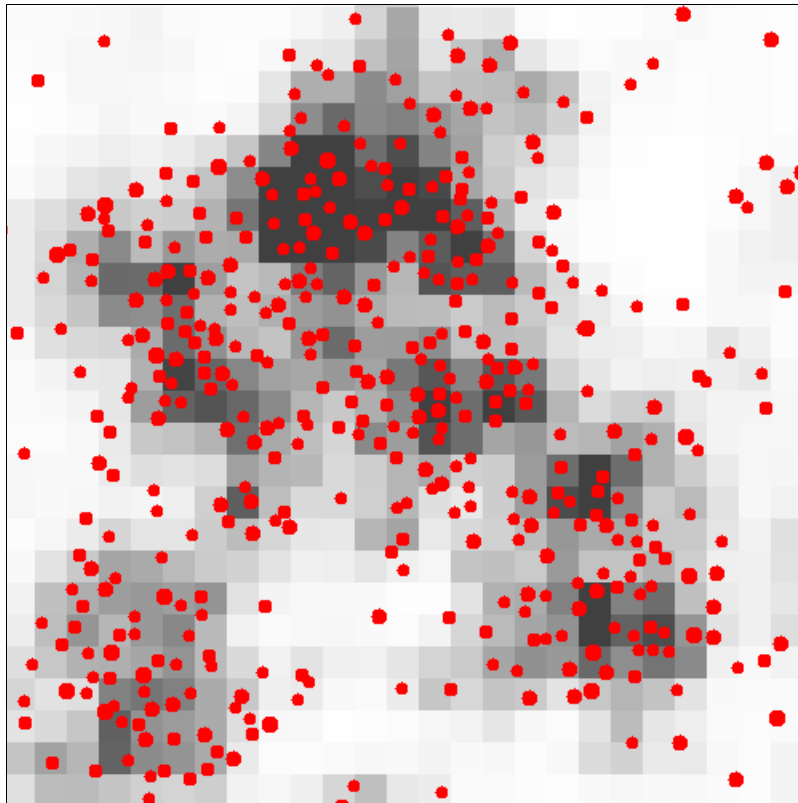


Figure 4: (Zoom in of figure 3) Emergence of the biofilm spatial pattern from individual interactions (Bacteria: red dots, Polymeric substance: grey level). The environment is discretized into a grid of cells to solve the diffusion equations. The substrate and polymer concentrations are calculated on each grid cell using the Alternating Direction Implicit numerical method (Reference). The size of a grid cell is a model parameter. The bacteria positions are real-valued. A bacterial cell may consume food from several grid cells at the same time if it is not entirely contained in one grid cell.

### **Interaction**

The individuals interact indirectly through the consumption of the nutrient and the release of the polymeric substance. Individuals also interact through the shoving relaxation process.

### **Stochasticity**

Bacteria motion is assumed to be a Brownian process. This is the simplest way to model bacteria mobility. According to this assumption the bacterial cell has no preferential direction and behaves like an inert particle which diffuses in an aqueous phase. At the low-level, diffusion is modeled as brownian motion.

### **Sensing**

The individual detect the local concentrations of nutrients and polymeric substance.

### **Observation**

In our model we observe the following quantities:

- Total bacterial concentration: which is the sum the individuals masses divided by the domain volume.
- Mean and local substrate concentration

- Mean and local polymeric substance concentration
- Bacterial cells positions

### **Initialization**

Bacterial cells: The bacterial cells are randomly distributed over the spatial domain, or situated in user defined location. The initial mass of each cell corresponds to the minimum mass of a bacterial cell (denoted  $m_r = 3.1415 E-15$  kg)

Substrate and product: The initial substrate and product concentration are uniform and specified by the user.

### **Submodels**

#### Bacteria mobility

Bacteria are represented as Brownian particle which mobility is described using the following equations:

$$y(t + dt) = y(t) + \sqrt{2D_b dt} N_2(0,1) \quad (\text{eq.1})$$

$$x(t + dt) = x(t) + \sqrt{2D_b dt} N_1(0,1) \quad (\text{eq.2})$$

Where  $D_b$  is the apparent diffusion coefficient,  $N_1(0,1)$  and  $N_2(0,1)$  are two independent standard Gaussian variables and  $dt$  is the time step

The apparent diffusion factor of the bacterial is assumed to be a function of the local polymeric substance concentration. We use the following equation:

$$D_b = \frac{D_{b0}}{1 + \beta \cdot p(x, y)} \quad (\text{eq.3})$$

Where :

$D_{b0}$  is the apparent diffusion constant of the bacterial cell in the absence of polymeric substance in its microenvironment.

$P(x,y)$  is the polymeric substance concentration at the bacterial cell location

$\beta$  is a factor (expressed in  $\text{m}^3/\text{kg}$  so that  $\beta \cdot p(x,y)$  is adimensional)

#### Bacteria growth and division

A bacterial cell consumes locally available substrate and use of part of the consumed nutrient (denoted  $Y_s$ ) for growth. The growth rate is given by the following model:

$$\frac{dm_i}{dt} = Y_s \mu(s) m_i \quad (\text{eq.3.1})$$

Where  $m_i$  is the mass of a given bacterial cell,  $s$  is the local concentration at the bacterial cell position and  $\mu(s)$  is the growth function given by the Monod equation:

$$\mu(s) = \mu_{max} \frac{s}{s + k_s} \quad (\text{eq.3.2})$$

Where  $\mu_{max}$  is the maximum growth rate [ $T^{-1}$ ] and  $k_s$  is the half-saturation constant [ $ML^{-3}$ ]. Equation 3.1 is discretized with respect to time using an explicit Euler numerical scheme:

$$\frac{m_i^{t+dt} - m_i^t}{dt} = Y_s \mu(s^t) m_i^t \quad (\text{eq.4})$$

Where  $dt$  is the discretization time step. Given the mass of the cell  $i$  at the instant  $t$ , the new mass at  $t+dt$  is then given by:

$$m_i^{t+dt} = m_i^t + dt Y_s \mu(s^t) m_i^t \quad (\text{eq.4.1})$$

When the mass of the cell reaches a critical mass, denoted  $2m_r$ , it divides into two identical daughter cells each with a mass  $m_r$ . Both daughter cells occupy the same position as the mother cells, unless the shoving relaxation process described further is switched on.

#### Shoving relaxation process

The formulation of the shoving relaxation process is inspired by the work of Kreft *et al.*, (1998)<sup>5</sup>. A given cell  $i$  with overlapping  $n_i$  neighbors is shifted with a displacement vector  $\mathbf{r}_{0i}$  such that:

$$\vec{r}_{0i} = \sum_{k=1}^{n_i} \vec{r}_k \quad (\text{eq.4.2})$$

$\mathbf{r}_k$  are the overlapping vectors defined as :

$$\vec{r}_k = \frac{0.5(d_i + d_k) - d_{ik}}{d_{ik}} \vec{c}_{ik} \quad (\text{eq.4.3})$$

$d_i$  : diameter of the cell  $i$

$d_k$  : diameter of the cell  $k$

$d_{ik}$  : distance between the centers of the cells  $i$  and  $k$

$\mathbf{c}_{ik}$  : the vector going from the center of the cell  $i$  to the center of the cell  $k$

Figure 5 illustrates an example of shoving relaxation for a given cell  $i$  having two overlapping neighbors ( $k=1,2$ )

---

<sup>5</sup> Jan Ulrich Kreft, Ginger Booth and Julian W.T. Wimpenny. **Bacsim, a simulator for individual-based modelling of bacterial colony growth.** Microbiology, 144, pp3275-3287

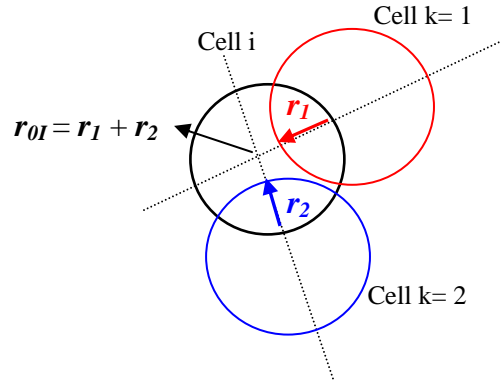


Figure 5 : Illustration of the shoving relaxation process for a bacterial cell  $i$  with two neighbors :  $n_i = 2$  (two neighbors),  $\mathbf{r}_{0I}$  : displacement vector calculated by the shoving relaxation process,  $\mathbf{r}_1$  : shoving vector of cell ( $k=1$ ) with the cell  $i$ ,  $\mathbf{r}_2$  : shoving vector of the cell ( $k=2$ ) with the cell  $i$

### Substrate consumption

The substrate concentrations are evaluated at discrete positions corresponding to the discretization of the diffusion equation. For a given spatial element  $i$  containing  $k$  bacterial cells, the amount of consumed substrate is given by:

$$\frac{ds_i}{dt} = \frac{1}{V_i Y_s} \sum_{j=1}^{j=k} \frac{dm_j}{dt} \quad (\text{eq.5})$$

Where  $V_i$  is the volume of the spatial element  $i$ . This means that the variation of local substrate concentration due to biological consumption is the sum of the action of the  $k$  bacterial cells located in the spatial element  $i$ .

### Polymeric substance production

$$\frac{ds_i}{dt} = \frac{1 - Y_s}{V_i Y_s} \sum_{j=1}^{j=k} \frac{dm_j}{dt} \quad (\text{eq.6})$$

The part of the substrate that is not used for the growth is assumed to be transformed into polymeric substance released by the bacteria in the environment

### Substrate diffusion process

The substrate diffuses in the environment. The diffusion process is modeled using the following diffusion equation:

$$\frac{\partial s}{\partial t} = D_s \left( \frac{\partial^2 s}{\partial x^2} + \frac{\partial^2 s}{\partial y^2} \right) - \left( \frac{ds}{dt} \right)_{bio} \quad (\text{eq.7})$$

Where  $D_s$  is the substrate diffusion constant. The term  $(ds/dt)_{bio}$  denotes the variation of the substrate concentration due to biological consumption. This term is calculated separately during the growth process.

Diffusion equations (eq.7 and eq.8 – see below) are discretized and solved used the alternating direction implicit method (ADI)<sup>6</sup>

#### Polymer diffusion process

$$\frac{\partial p}{\partial t} = D_p \left( \frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \right) + \left( \frac{dp}{dt} \right)_{bio} \quad (\text{eq.8})$$

Where  $D_p$  is the substrate diffusion constant. The term  $(dp/dt)_{bio}$  denotes the polymer production rate by the bacterial cells.

#### Detachment process

Detachment is the process by which a part of the biofilm is removed from the system. In real systems this is mainly due to the shear effect produced by the mixing in the reactor. In our model the biofilm is defined as the regions where the polymeric substance accumulates. A detachment event consists in emptying a given region from its polymeric substance content and bacterial cells content. A spatial region is at least equal to a spatial element (used for the spatial discretisation of the polymeric substance diffusion equation).

At each time step and for each spatial element we calculate the detachment probability for each spatial element. This probability is given by:

$$Pr(\text{Detachment position } i, j) = \Delta t \frac{p(i, j)}{p_{max}} \quad (\text{eq.9})$$

Where  $i, j$  are the coordinate of the spatial element,  $p(i, j)$  is the polymer concentration in the element  $(i, j)$  and  $p_{max}$  a parameter (maximal polymer concentration) and  $\Delta t$  is the time step used in the simulation.

#### Feeding process

The spatial domain can be fed with a constant flux, denoted  $Q$  (in m<sup>3</sup>/s). We consider that the spatial domain volume,  $v$ , is constant. This means the input flow is equal to the output flow (figure 6). The input flow may contain substrate at an input concentration  $s_{input}$ . We suppose that the fed amount of substrate is distributed uniformly over the spatial domain. For a given spatial element  $(i, j)$  we can write :

$$\frac{v}{N} \frac{ds(i, j)}{dt} = \frac{Q}{N} (s_{input} - s(i, j)) \quad (\text{eq.10})$$

---

<sup>6</sup> William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery. **Numerical Recipes in Fortran 77**. Cambridge University Press, 2<sup>nd</sup> edition (1997) – (see chapter 19 for an introduction to the ADI method)

Where  $V$  is the domain volume,  $N$  the number of spatial elements (used in the spatial discretization of the diffusion equation),  $s(i,j)$  the substrate concentration for the spatial element  $(i,j)$  and  $s_{input}$  is the input concentration of substrate.

We propose a similar equation for the polymer:

$$\frac{v}{N} \frac{dp(i,j)}{dt} = \frac{Q}{N} (p_{input} - p(i,j)) \quad (\text{eq.11})$$

The feeding flow is assumed to be bacteria-free. We model the wash-out of bacterial cells by assigning a washout probability to each cell given by:

$$Pr(\text{washout of a given cell}) = \Delta t \frac{Q}{V} \quad (\text{eq.12})$$

Additionally we assume that when a bacterial cell is enclosed into a polymeric matrix, it has a lower probability to be washed out than a free bacterial cell. The washout probability of a bacterial cell is:

$$Pr(\text{washout of a given cell}) = \Delta t \frac{Q}{V} \left( 1 - \frac{p(i,j)}{p_{max}} \right) \quad (\text{eq.13})$$

Where  $p(i,j)$  is the polymer concentration at the bacteria position  $(i,j)$  and  $p_{max}$  is a parameter (maximum polymer concentration)

## Simulation Experiments

Here we present a simple simulation experiment using this model. We used the Mason framework to implement the model. Mason is a discrete event multi-agent simulation library code in pure Java developed at the George Mason University for implementing multi-agent models (Luke et al., 2004)<sup>7</sup>.

The simulator gets a parameter file as input (see Appendix A) in which the initial and boundary conditions and the parameters describing of the processes (kinetic parameters of the bacteria, diffusion factors, feeding flow and input concentrations) are specified. At each time step the model calculates the new position and size of the bacteria, the new substrate and polymer concentration maps. The results are stored in five output files. The output files can then be analyzed using standard tools like Matlab or R.

### Case study

#### Description

In this example we propose to simulate a simple bacterial culture. The bacterial cells are supposed to be cultivated in a batch bioreactor (closed domain without any mass exchange through the domain boundary). An initial number ( $N = 100$ ) of immobile bacterial cells are randomly situated in the studied domain. The bacterial cells consume an initial stock of food (uniformly distributed)

---

<sup>7</sup> LUKE S., CIOFFI-REVILLA C., PANAIT L., SULLIVAN K., (2004) MASON : A Java Multi-Agent Simulation Toolkit. *Proceedings of the 2004 SwarmFest Workshop*.

and transform it entirely to new biomass (new bacterial cells). The parameter file is listed in Appendix A.

### Results analysis

Bacson allows the user to visualize the time series of three main state parameters: mean biomass concentration, mean substrate concentration and total number of cells (figure 6,7,8)

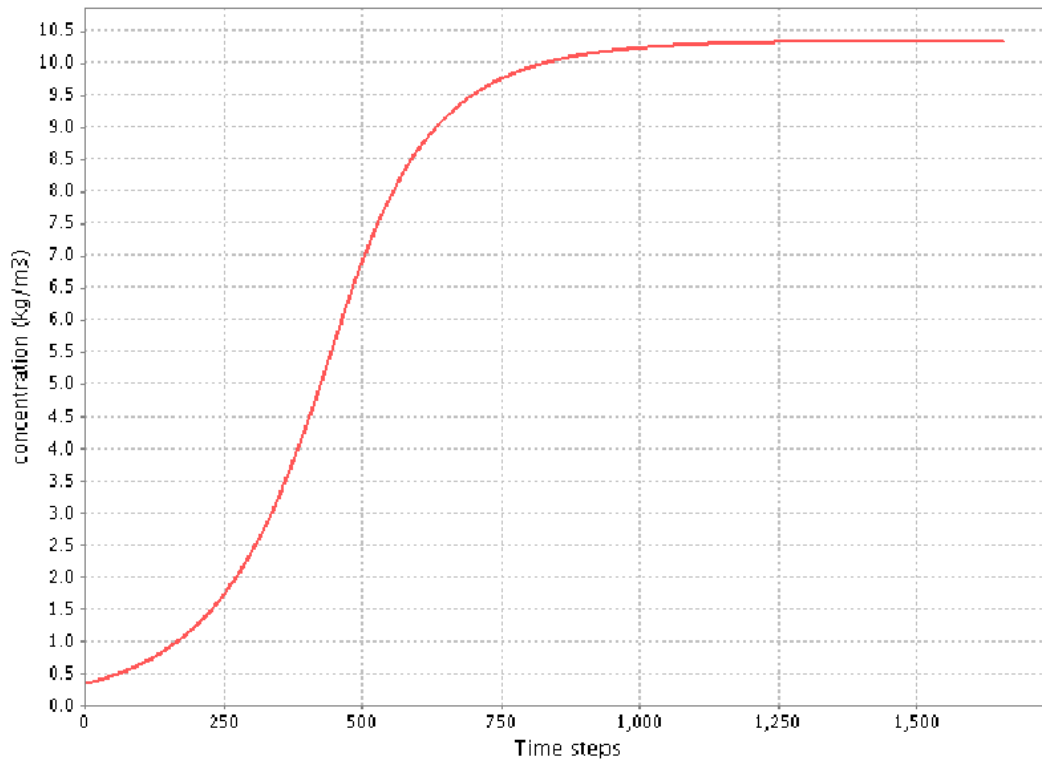


Figure 6. Mean biomass concentration

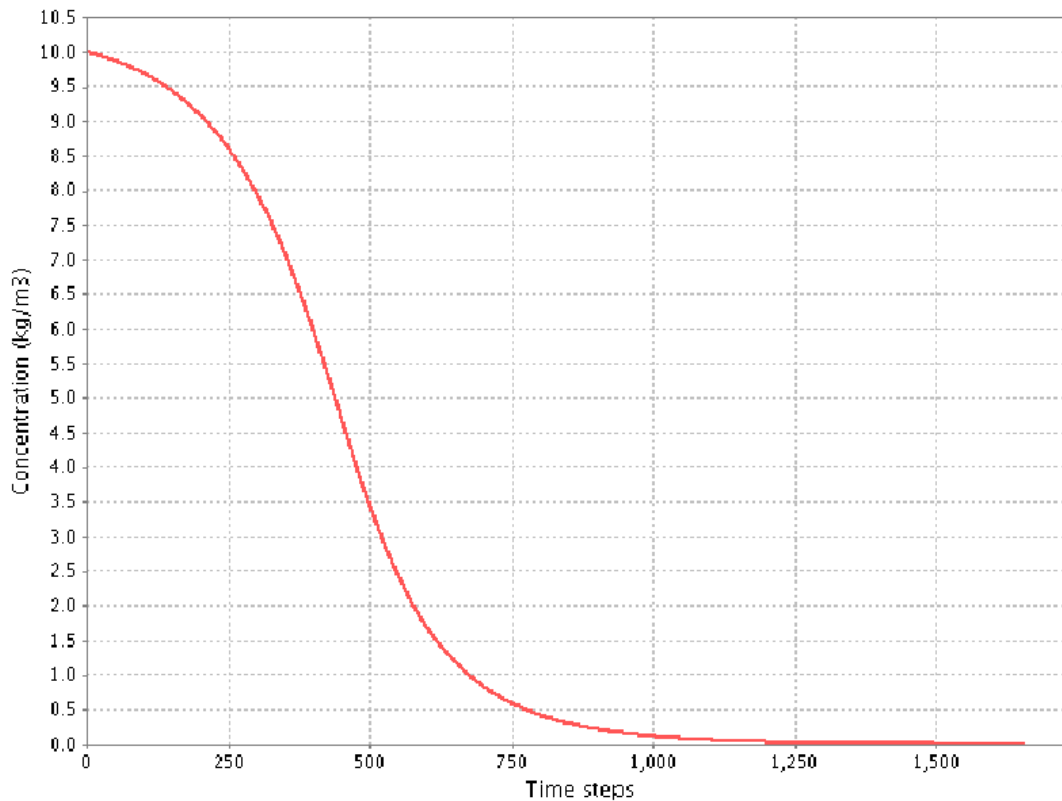


Figure 7 : Mean substrate concentration

As the bacterial cells consume the substrate, the mean substrate concentration decreases (figure 7) progressively. The bacterial cells transform the substrate into new cells which increase the mean biomass concentration in the system (figure 6). The mean biomass concentration reaches a „plateau“ which corresponds to the consumption of the total initial amount of substrate.

Figure 8 shows the variation of the number of bacterial cells in the system. The jumps in the number of cells correspond to the division events. The size of the cell increases until it reaches a critical mass at which the cells divide. The initial division events are synchronous, meaning that all the cells divide at the same instant. This is mainly due to the initial homogeneity of the substrate distribution in the system. The synchrony of the division events is progressively lost as the bacterial cells start forming colonies. In a given colony the cells located on the colony boundary have a better access to the substrate than the cells located in the centre of the colony. This affects their respective growth rates and explains, with the substrate concentration heterogeneity, the lost of synchrony of the division events.

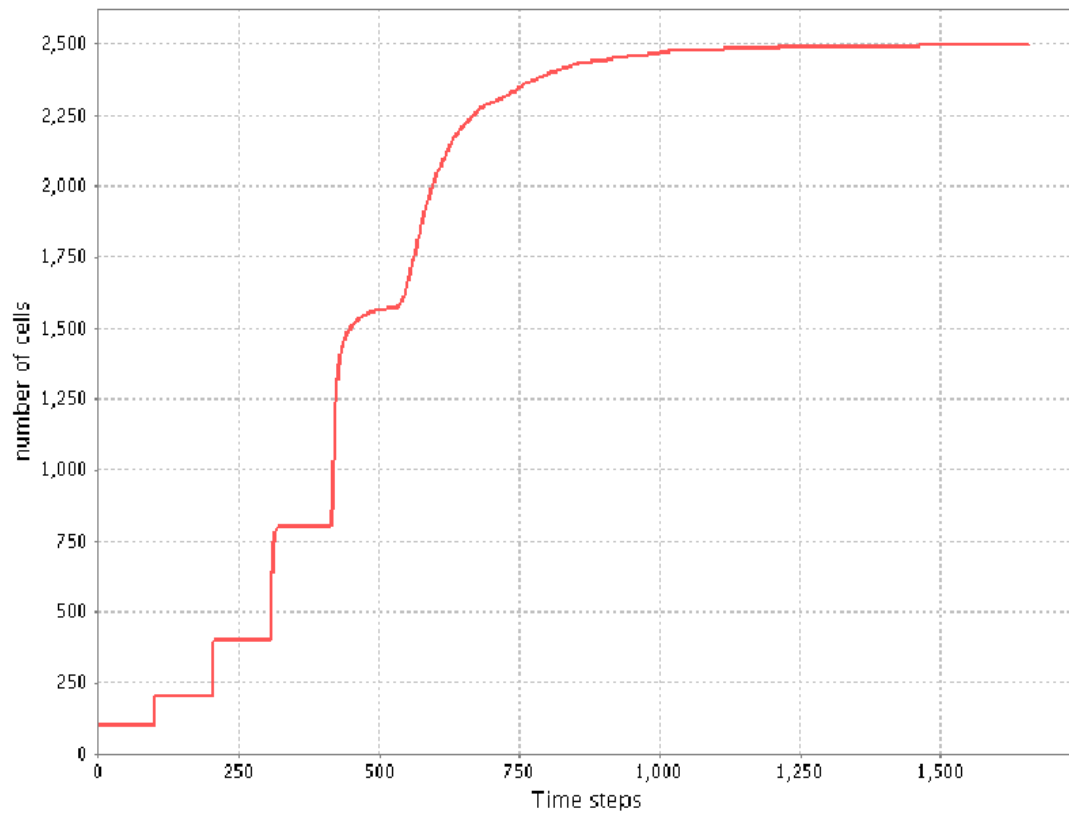


Figure 8: Total number of bacterial cells

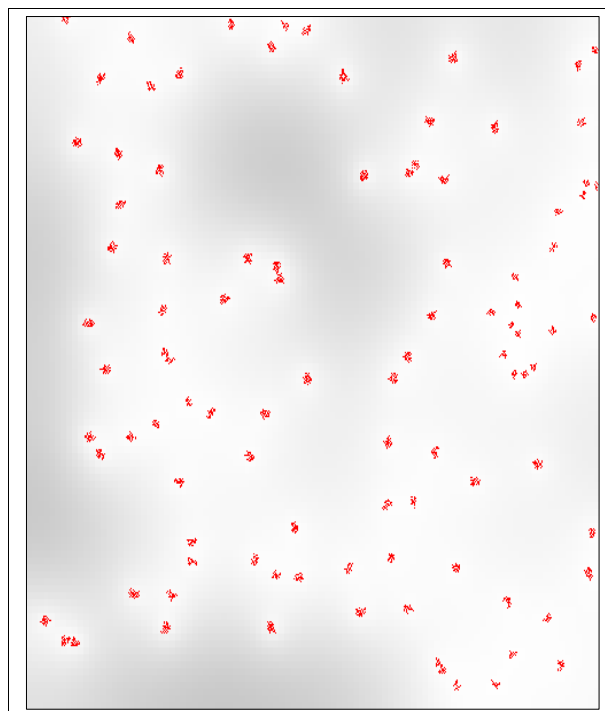


Figure 9: Snapshot of the simulated domain

## **Conclusion**

We developed an individual-based model to investigate the effect of local bacteria interactions on the global biofilm structure and dynamics. We implemented the model in Java programming language using the Mason framework. In this document we presented the simulator and the underlying model using the ODD framework. We included a simple simulation example and a listing of the parameters file. Future report will include additional simulation examples and a more detailed discussion about the characterization and control of the formed biofilm patterns.

## Chapter 2: Installation

### Prerequisites

To run the java program you need first to install:

1- Java virtual machine (preferably Sun's JRE (Java Runtime Environment 5 or later) downloadable at :

<http://java.sun.com/javase/downloads/index.jsp>

**(Java Runtime Environment (JRE) 6u1)**

2- Java3D API (Sun Microsystems) downloadable at :

<http://java.sun.com/products/java-media/3D/download.html>

**Bacson** can be run on the Linux or Windows XP. **Bacson (version 1.0)** is provided as a zip file named: **Bacson\_version1.0.zip**.

### Installation

Unzip the file to a new directory named **Bacson**. The directory now contains:

<b>&lt;lib&gt;</b>	Subdirectory	contains some libraries used by Bacson
<b>&lt;src&gt;</b>	Subdirectory	contains the Bacson source code
<b>BacsonGUI.jar</b>	Executable file	a java executable file. Runs Bacson with the Graphical User Interface
<b>Bacson.jar</b>		a java executable file. Runs Bacson without the Graphical User interface
<b>defaultParameters.properties</b>	Parameters file	a text file that can be modified using any text editing software

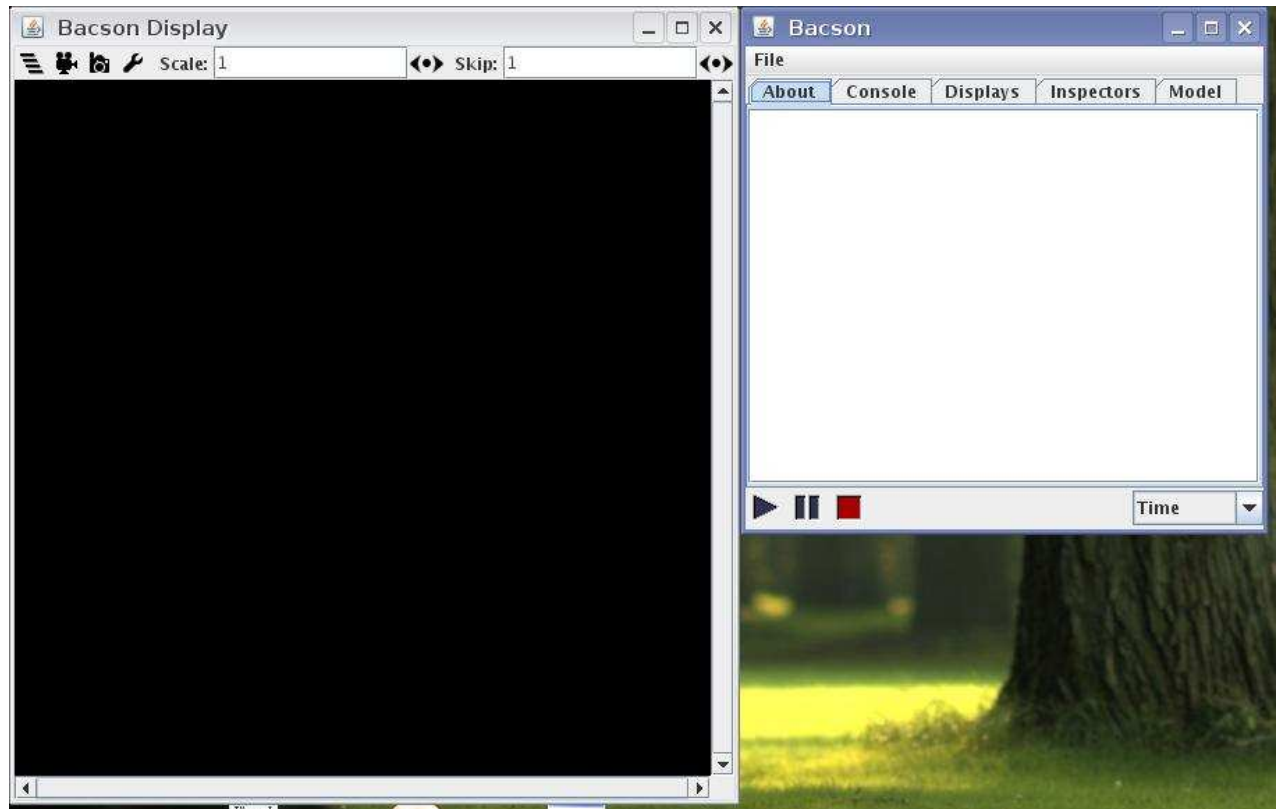
### Run the program

#### Run Bacson with graphical user interface

There are two possibilities:

- Possibility 1 : Double-click the **BacsonGUI.jar** file
- Possibility 2 : in a MS-DOS console or Linux console
  - Change the directory to the directory where you unzipped the Bacson zip file
  - use the following command : **java -jar "BacsonGUI.jar"**

Both possibilities should give the following result:



**Figure 10.** Bacson graphical user interface

To run the program with the default parameters set (specified in the **defaultParameters.properties** file) just click the play button.

#### Run Bacson without graphical user interface<sup>8</sup>

In a MS-DOS console or Linux console:

- Change the directory to the directory where you unzipped the Bacson zip file
- Use the following command: **java -jar "Bacson.jar"**

---

<sup>8</sup> Not included in the current zip file

## Chapter 3: Bacson User guide

### Using the Graphical user interface (GUI)

The Bacson GUI is formed by two main windows: the console window and the display window (see figure 10).

The console allows the user to modify the parameters, start and stop simulations, and view simulation results.

The display window offers a visual representation of the modeled spatial domain, the bacterial cells (red dots) and the food and products concentration (color gradient)

#### The console window

The console window contains 5 tabs called: **About**, **Console**, **Display**, **Inspectors** and **Model** (figure 1). The most important is the **Model** tab which allows the user to modify the model parameters and visualize the main results.

The **Model** tab gives access to three sub tabs (figure 11) called:

- **Preferences:** general parameters of the simulator
- **Parameters:** main model parameters
- **States:** main aggregated state variables

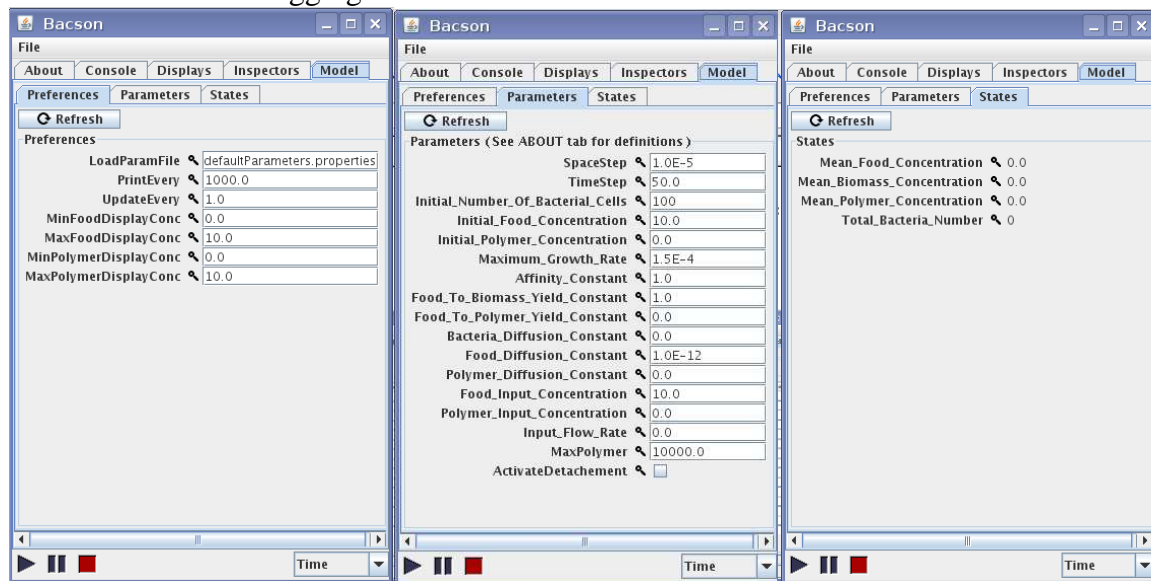


Figure 11: Console windows: sub tabs of the **Model** tab

## Model parameters

The following table summarizes the main parameters which can be modified using the **model** tab:

Console window -->Model tab -->Preferences tab		
Parameter	Description	Default value
<i>LoadParamFile</i>	Name of the parameter file to be loaded	defaultParameters.properties
<i>PrintEvery</i>	Period (number of time steps) at which the results are printed to output files	1000
<i>UpdateEvery</i>	Period(number of time steps) at which the results are updated in the “States” sub tab.	1
<i>MinFoodDisplayConc</i> <i>MaxFoodDisplayConc</i>	The food concentration is displayed with a color gradient going from White for <i>MinFoodDisplayConc</i> to Black for <i>MaxFoodDisplayConc</i> . Both are expressed in kg/m3	0.0 10.0
<i>MinPolymerDisplayConc</i> <i>MaxPolymerDisplayConc</i>	The polymer concentration is displayed with a color gradient going from white for <i>MinPolymerDisplayConc</i> to Orange for <i>MaxPolymerDisplayConc</i> . Both are expressed in kg/m3	0.0 10.0

Console window --> Model tab --> Parameters tab			
Parameter	Description	Symbol	Default value
<i>SpaceStep</i>	The spatial domain is discretized into small squares with size SpaceStep (in m). This discretisation is used for solving the food and product diffusion equations.	$\Delta x = \Delta y$	10E-6
<i>TimeStep</i>	The time step (in s) of the simulation	$\Delta t$	50.0
<i>Initial_Number_Of_Bacterial_Cells</i>	Initial number of bacterial cells.		100
<i>Initial_Food_Concentration</i>	Initial uniform food concentration (in kg/m3)		10.0
<i>Initial_Polymer_Concentration</i>	Initial uniform polymer (product) concentration (in kg/m3)		0.0
<i>Maximum_Growth_Rate</i>	Monod kinetic parameter : Maximum growth rate of the bacterial cell, expressed in s-1	$\mu_{max}$	1.5E-4
<i>Affinity_Constant</i>	Monod kinetic parameter : Affinity factor, expressed in kg/m3	$k_s$	1.0
<i>Food_To_Biomass_Yield_Constant</i>	Conversion factor food to biomass	$Y_s$	1.0
<i>Food_To_Polymer_Yield_Constant</i>	Conversion factor food to polymer	$Y_p = 1 - Y_s$	0.0
<i>Bacteria_Diffusion_Constant</i>	Bacteria diffusion constant (Brownian mobility) in m2/s	$D_{b0}$	0.0
<i>Food_Diffusion_Constant</i>	Food diffusion constant in m2/s	$D_s$	1.0E-12
<i>Polymer_Diffusion_Constant</i>	Product diffusion constant in m2/s	$D_p$	1.0E-16
<i>Food_Input_Concentration</i>	Food Input concentration in kg/m3	$s_{in}$	10.0
<i>Polymer_Input_Concentration</i>	Polymer input concentration in kg/m3	$p_{in}$	0.0

<b><i>Input_Flow_Rate</i></b>	Input flow rate in m <sup>3</sup> /s	Q	0.0
<b><i>MaxPolymer</i></b>	This parameter is used for the biofilm detachment process (in kg/m <sup>3</sup> )	p <sub>max</sub>	10000
<b><i>ActivateDetachment</i></b>	activate/desactivate the biofilm detachment process		False
<b><i>ActivateDivisonProcess</i></b>	Let the bacterial cells divide once their mass reach 2m <sub>r</sub> = 6.2830 E-15 kg (constant)		True
<b><i>ActivateShovingRelaxationProcess</i></b>	Perform the shoving relaxation process according to the sub model described in chapter 1.		True
<b><i>Beta</i></b>	Used to parametrize the effect of the polymer on the bacteria motion (equation 3) expressed in m <sup>3</sup> /kg	β	10

## Model outputs

The simulator creates an output folder called: **resultsTimeInMilliseconds**. This folder is located in the Bacson folder and contains the following output files:

- Profiles : Horizontal profile of the food and polymer concentration at mid domain height
- TimeSeries : Time series of the main aggregated state variables (mean food and product concentrations, mean biomass concentration and total number of bacterial cells)
- FoodMap : the matrix of spatial substrate concentration
- PolymerMap : the matrix of spatial polymer concentrations
- BacteriaMap : array of the coordinates of the bacterial cells
- logFile : record of the CPU time spent by the simulator in each simulated process

The GUI allows the user to take snapshots by clicking the photo icon on the display window (see figure 12)

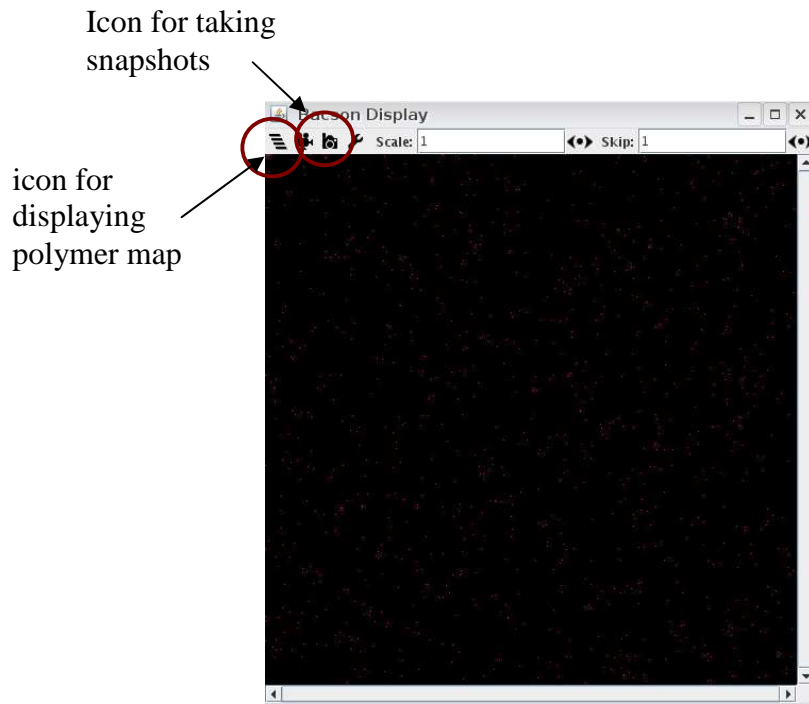


Figure 12. Display window

## Additional options

It is possible to display the polymer concentration map as a 3D colored map (figure 13). The option can be activated as follows:

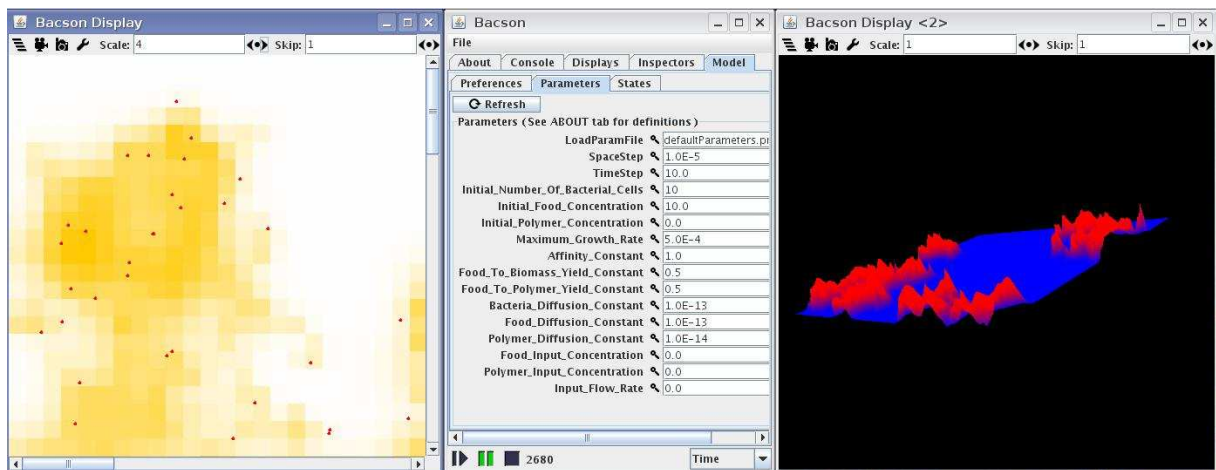


Figure 13. Snapshot of the Bacson display window.

Console window --> Display tab --> show all

The 3D map can be zoomed in and out and rotated using the mouse roller and left button.

## Appendix A

## Default parameters file

# INITIAL CONDITIONS

#-----

# Initial number of bacterial cells

numBactIni=100

# Initial substrate concentration in kg/m<sup>3</sup>

Sini=10

# Initial polymer concentration in kg/m<sup>3</sup>

Pini=0

# set intX to -1 if you want the initial x-pos of the bacteria to be randomly selected

intX=-1

# set intY to -1 if you want the initial y-pos of the bacteria to be randomly selected

intY=-1

# Diffusion constants

#-----

# Bacteria diffusion constant in m<sup>2</sup>/s (Df=0 for immobile cells)

Df=0.0

# Substrate diffusion constant in m<sup>2</sup>/s

Ds=1.0E-12

# Polymer diffusion constant in m<sup>2</sup>/s

Dp=1.0E-16

# Kinetic constants

#-----

# Maximum growth rate of the bacteria in s<sup>-1</sup>

muMax=1.5E-4

# Affinity constant in kg/m<sup>3</sup>

ks=1.0

#Biomass yield in kg of biomass per kg of consumed substrate

yx=1.0

#Polymer yield in kg of polymer per kg of consumed substrate

yp=0.0

# General

#-----

```
#Time step in s
TimeStep=50.0

# Spatial discretization step of the domain in m
SpaceStep=10E-6

# Feeding
# -----
#Input food concentration in kg/m3
InputFoodConc = 10.0

# Input polymer concentration in kg/m3
InputPolymerConc=0.0

# Flow rate in m3/s (note: 1 μm3/s = 1.0E-18 m3/s)
InputFlowRate=0.0

# Detachment
# -----
# Parameter used for the calculation of the detachment probability
maxPolymer=10000

# Boundary conditions for food and polymers
# -----
# 1 : Neumann ; 0 : Dirichlet
northBorderType=1
southBorderType=1
eastBorderType=1
westBorderType=1

# if neuman specify the flow
northFlux=0.0
southFlux=0.0
eastFlux=0.0
westFlux=0.0

# if Dirichlet specify the concentration
northConc=-1
southConc=-1
eastConc=-1
westConc=-1

# Beta factor : parametrize the reduction of mobility of the cells in the presence of polymer
beta=10

#activate the bacteria divison process
activateBacteriaDivision = true

# activate the shoving relaxation process
activateShovingRelaxation = true
```