

# User Manual

Daniel Griffin

## Requirements

Requirements to run the program:

- A Java Development Kit (JDK) version 8.0+, including JavaFX 8 (this comes as standard with Oracle JDK 8, but may not be installed as standard with OpenJDK)
- Apache Ant
- A JAR file of the la4j library version 0.6.0, this can be downloaded from <https://mvnrepository.com/artifact/org.la4j/la4j/0.6.0>

## How to run the program

To run the program:

- Clone the repository
- Create a folder called lib in the main directory of the repository, and save the la4j library JAR file within this folder
- In the main directory of the repository type "ant run"

Other Ant commands:

- "ant compile": Compiles all the source files
- "ant jar": Creates a jar file of the program
- "ant clean": Removes the compiled source code and the jar file

## Creating an input file

An input file is simply a text file, where each parameter you wish to change from its default value is listed, along with the value it should take. The name of the parameter must be followed by a colon, after which the value the parameter should take is written. An example input file is shown below.

```
grid height: 80
grid width: 80
cell height: 5
cell width: 5
rate of diffusion: 0.4
nutrient for sustenance: 10
nutrient for growth: 60
threshold for cell division: 2600
crowding function: 0, 40, 40, 40, 30, 20, 10, 0, 0
number of timesteps for cell division: 8
boundary condition: reflecting
initial nutrient pattern: default
probability of cell division: 0.5
```

All the possible parameters that can be changed are displayed. The value specified for each in this example is its default value. The meaning of each parameter and the values it can take are as follows:

- grid height – (a positive integer): The overall height of the grid.
- grid width – (a positive integer): The overall width of the grid.
- cell height – (a positive integer): The height of an individual grid cell.
- cell width – (a positive integer): The width of an individual grid cell.
- rate of diffusion – ( a real number between 0 and 1 inclusive): The rate at which nutrient diffuses.
- nutrient for sustenance – (an integer between 0 and 100 inclusive): The amount of nutrient a bacterium cell requires to survive in a single time step.
- nutrient for growth – (an integer between 0 and 100 inclusive): The amount of nutrient a bacterium cell requires for cell division to occur.
- threshold for cell division – (a non negative integer): The value which the product of the crowding function and the nutrient in a given cell should reach, if there is to be a chance for cell division to take place.
- crowding function – (9 non negative integers separated by commas): A value which is used to estimate the likelihood of cell division occurring based on the number of neighbouring cells. For example, in the input file given as an illustration above, the crowding function has 9 values. The first gives a value for when there are no neighbouring cells around the cell in question, the second value for when there is one neighbouring cell and so on.
- number of timesteps for cell division – (a positive integer): Number of timesteps after which cell division can occur. In the case of the example input file, this would mean cell division can occur every 8 timesteps, whilst nutrient diffusion and consumption always happen every time step.

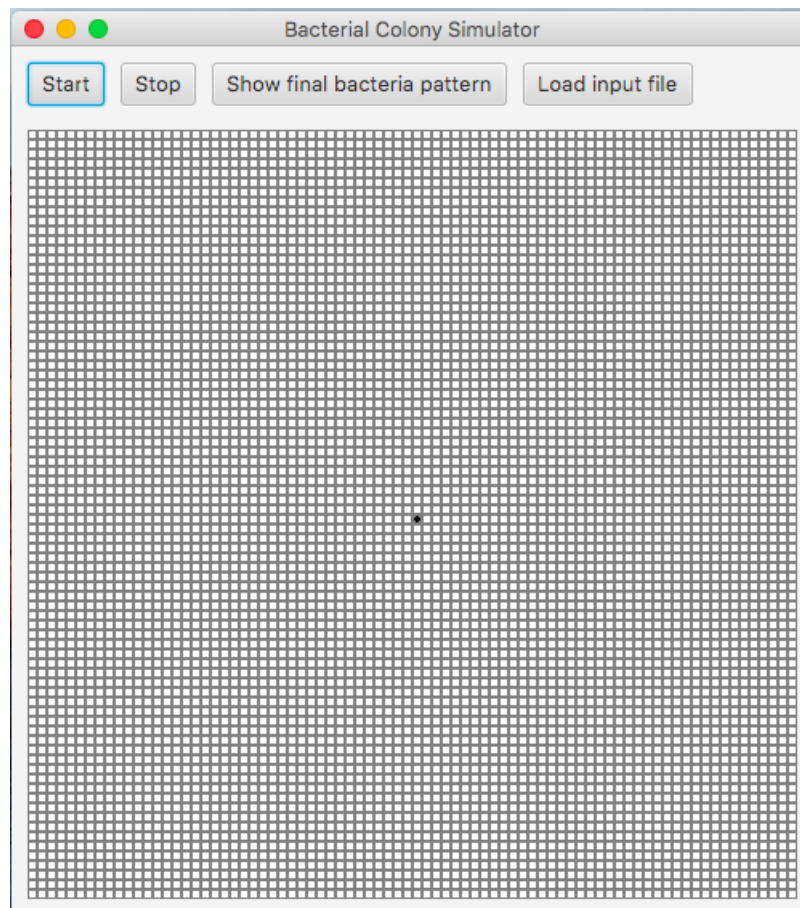
- boundary condition – ("periodic", "reflecting" or "absorbent"): Specifies the boundary of the cellular automata.
- initial nutrient pattern – ("default", "random", "absorbingmiddle"): Used to specify the initial nutrient in each cell. For *default* this is 100 in every cell, for *random* it is a random amount of nutrient in every cell (between 0 and 100), lastly, *absorbingmiddle* refers to a configuration in which the middle row can not contain any nutrient and absorbs nutrient from neighbouring cells (like the absorbent boundary).
- probability of cell division – (real number between 0 and 1 inclusive): A probability that specifies the likelihood of cell division occurring should the threshold for cell division be reached.

## Running a simulation

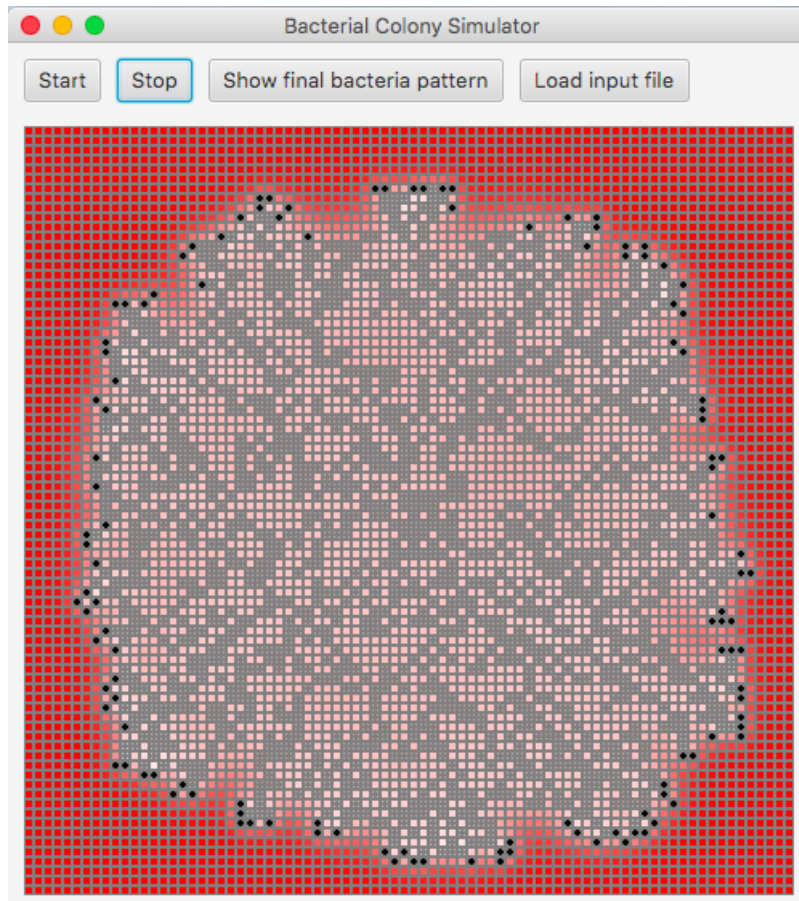
When the program first loads the screen shot on the following page illustrates what the program will look like. The centre cell is automatically set to alive. However, by clicking on any cell the dead/alive state of that cell can be toggled.

At the top of the program is four buttons, these are:

- Start: Which starts the simulation running, and resumes the simulation after it has been stopped.
- Stop: Stops/pauses the simulation.
- Show final bacteria pattern: Once clicked the simulation cannot be resumed. It shows the final morphology in a more visually clear way than is shown when the simulation is running. This is illustrated further on in the section.
- Load input file: When clicked, the simulation – if one is running – is paused, and a input file can be selected. If no errors appear after selecting the input file, then it has been loaded (see previous section for how to create input files).



When the simulation is running the screen will resemble that which is displayed on the following page. Each cell has a redness level that runs from white to red. Where white represent no nutrient in a cell, and red represents a maximum amount of nutrient in a cell, namely a value of 100. Values in between have colours on the white-red scale in proportion to their nutrient level. Within a grid cell there can also be a circle. This can take two colours, black and grey, the former represents an alive bacterium cell, the latter a dead bacterium cell. If no bacteria has ever existed in that grid space no circle will be shown.



When the *Show final bacteria pattern* button is clicked the screen will resemble that which is shown on the next page. All bacteria that were at one stage alive are shown, with nothing else (grid lines or nutrient) displayed. At this stage the *Start* and *Stop* buttons are hidden. After clicking *Load input file* and selecting a valid file, the *Start* and *Stop* buttons will become visible again for the next simulation to be run.

