This article is written by students. It may contain omissions and imperfections, reported as far as possible by our reviewers in the editing notes.

Bacteria attacks! - a study of the evolution of bacteria cultures

Year 2022-2023

Students:

Gavrilă Cristian, Șipoș Sony, Voicu Teofil, students in the 11th grade

Penea Andrei, Neagu Sofian, Bărăgan Lorena, Pruneanu Andrei, Moldovan Bianca, Roman Anamaria, students in the 10th grade

Teachers: Văcărețu Ariana

Schools: Colegiul Național „Emil Racoviță”

Researcher: Dr. Cătălin Țurcaș - Universitatea „Babeș-Bolyai”

1. Introduction

The study of bacteria population and evolution is an important and interesting field with large applications.

This line of research advances our knowledge on the bacterial ecosystem, as well as help in the development of human health, the medical industry and biotechnological applications.

Mathematically modelling the processes in a bacteria population leads to the possibility of predicting the biological behaviour of certain systems, which can help in experimental designs and biological research.

2. The problem

In a bacterial culture, some bacteria organisms are producing a toxic substance that kills other bacteria. The change in population is modelled by a system of differential equations, where the growth rate is proportional to the existing population $N(t)$ and decreases at a rate proportional to the concentration of the toxic substance $T(t)$.

The equations are:

$$\frac{dN(t)}{dt} = k \cdot N(t) \cdot (1 - a \cdot T(t))$$

$$\frac{dT(t)}{dt} = r \cdot N(t)$$

where:

$N(t)$ - the population of bacteria at a time t

$T(t)$ - the concentration of the toxic substance at a time t

k - fixed parameter that determines the rate of population growth

a - fixed parameter that determines the impact of the toxic substance on population growth

r - constant rate at which the toxic substance is produced per organism

Our purpose is to analyse the evolution of the population for any initial population and parameters, synthesise our findings into conclusions and pose questions for future research.

3. Our solution

We have found that there are two ways of approaching the problem: a theoretical one and a computational, more experimental one. Although the mathematical approach would be closer to a real situation, we only used it for general information. For the actual solving, we settled on using the computational one, as we found it to be more flexible and suitable to our project.

To closely match a real-life situation, a good way to approximate the solution is through discrete calculus, an explanation of which is found in Appendix A.

After running the simulation multiple times, we arrived at the conclusion that no matter what the initial parameters are, the population will always reach 0, implying a constant quantity of toxins.

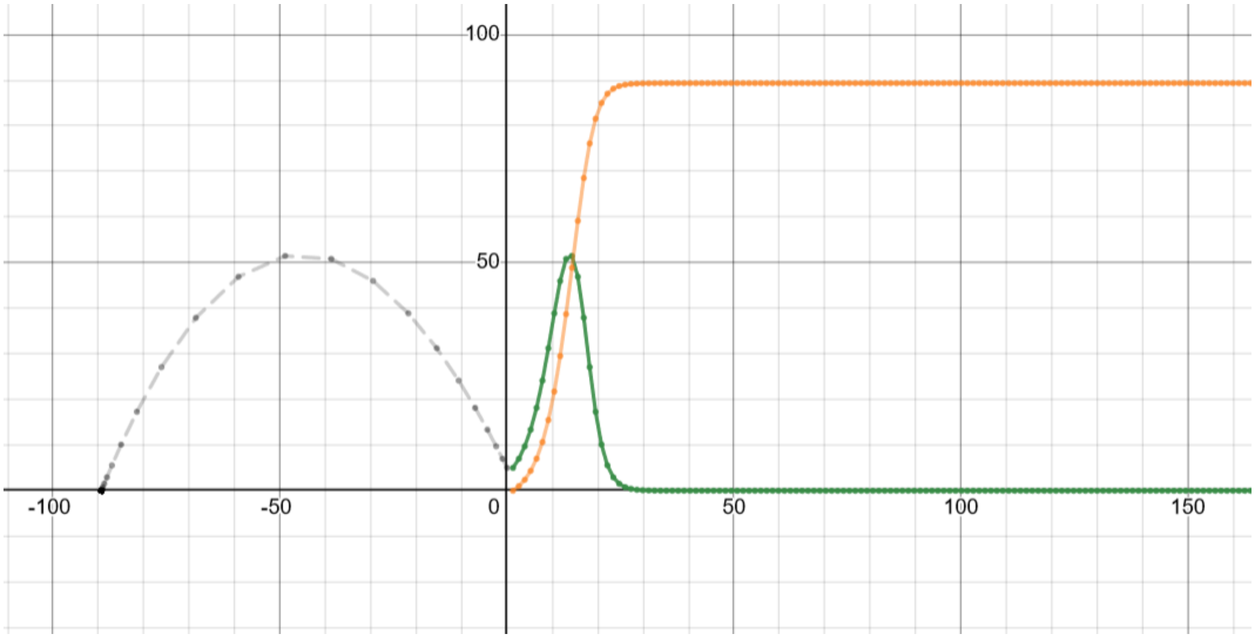


Fig. 1: Graph for the initial problem

The graph in figure 1 represents one of the simulations. The green line is the number of bacteria and the orange line is the toxin quantity, both as functions of time. The graph on the left is a parametric representation of the two graphs (see Appendix C).

This is indeed not a very interesting result, so we have decided to change the problem itself. In a real situation, toxins would not last forever due to evaporation and chemical decay, so we introduced a new term: b, the rate of toxin decay. Thus, the equations now become:

$$\frac{dN(t)}{dt} = k \cdot N(t) \cdot (1 - a \cdot T(t))$$

$$\frac{dT(t)}{dt} = r \cdot N(t) - b \cdot T(t)$$

This way, the toxins decay at a rate proportional to the current toxin quantity, so if the concentration is large, the decay will also be somewhat proportional.

With this out of the way, the solution is now much more fascinating.

A stable point for the system of differential equations is reached for $\frac{dN}{dt}$ and $\frac{dT}{dt}$ equal to 0, so the values for T and N are

$$T = 1/a \text{ and } N = T \cdot b/r,$$

so there exists a solution that is stable. If the reader is interested in learning more about stability in ordinary differential equations, please consult the references chapter.
Indeed, if we generate a graph of the population evolution, for certain values of the parameters the population reaches a stable state in which the number of toxins and the number of bacteria are balanced. This is best represented by the parametric graph.

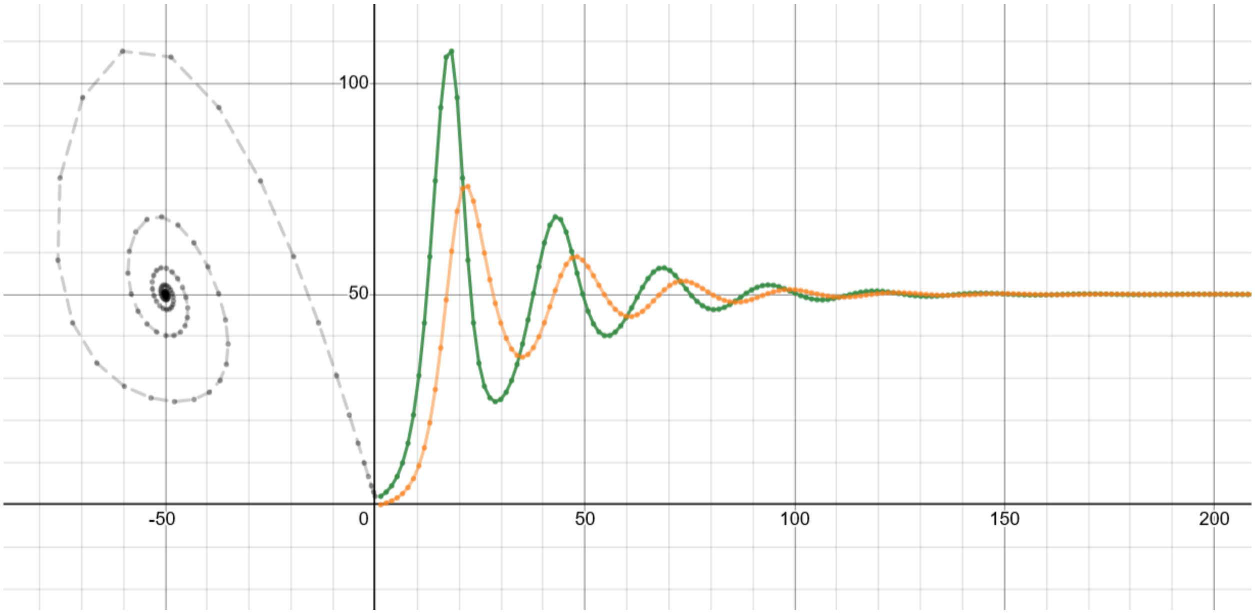


Fig. 2: Graph of a numerical solution for the new problem

4. Conclusion

Our model of the problem indicates that, given the formulas in the text, the bacteria population always dies. Modifying the initial equations by introducing an additional factor, we can observe a totally different evolution: in most cases the bacteria and the toxins reach an equilibrium. Further improvements can be made to the problem by finding a mathematical solution for the differential equation. It is also a good idea to write a program that can run tests on its own and see if there are more special cases where the population doesn’t reach a steady state.

5. Acknowledgements

We would like to thank Dr. Cătălin Țurcaș from Universitatea Babeș Bolyai and Mihai Lazăr from Univeristatea Tehnica Cluj-Napoca for assisting us over the course of this project.

6. Appendix

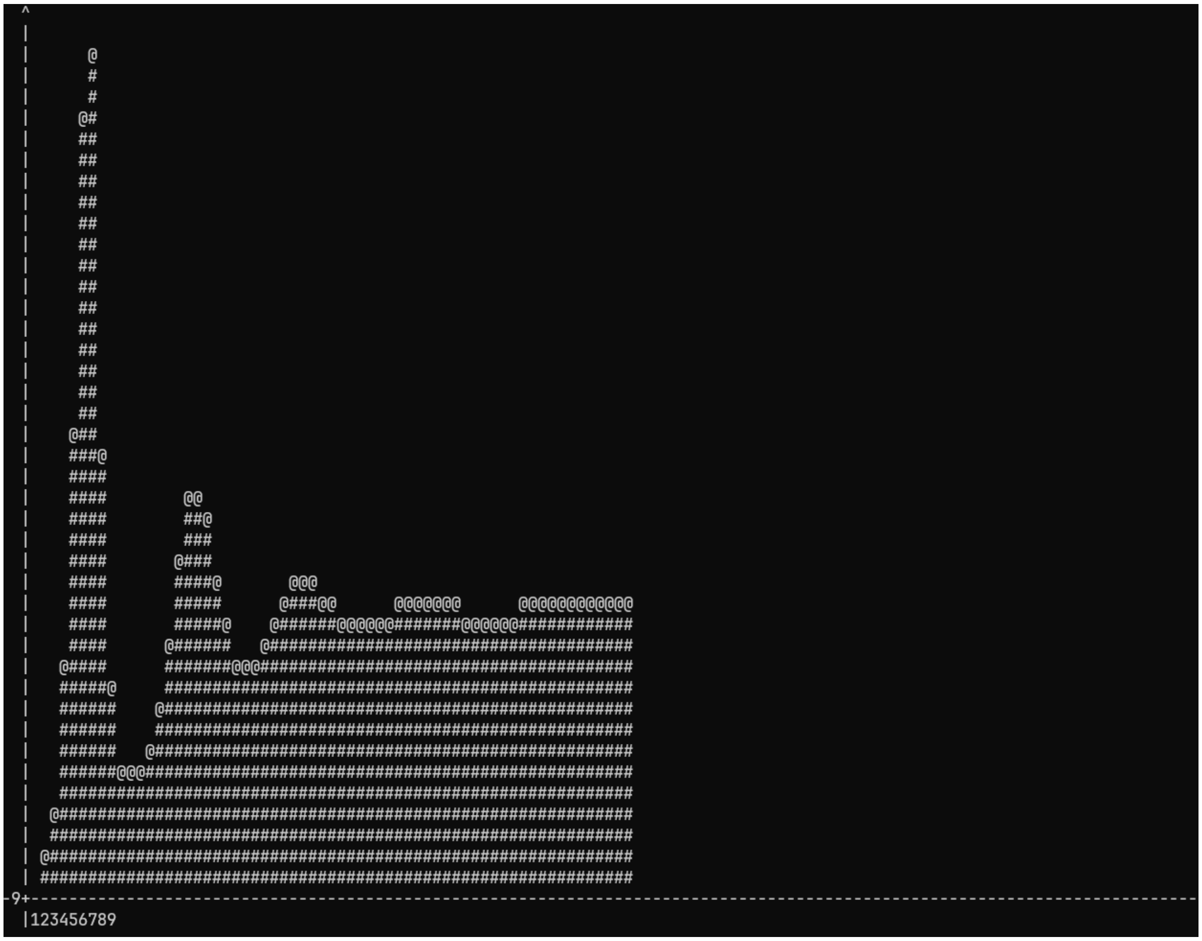
A. Discrete calculus

Discrete calculus is a mathematical concept that is used to approximate mathematical functions. Instead of needing to find an exact formula for a function, an incremental and iterative method is used to calculate the desired information about a function like its value, derivative and so on at each step of the process. The smaller the step size is, the more calculations are done and the better the accuracy is, so a computer program is perfect for these kinds of tasks. The following appendices contain various implementations of this algorithm.

B. Graph plotter in C++

The graph represents the number of bacteria in the population as a function of time. It was created using ASCII render in a Windows Command Prompt. The program computes all the points on the graph and displays them with an approximation directly influenced by the size of the screen (see Fig. 3). This method is far from being optimal, but is still a great way of visualizing the evolution of the bacteria population. The values were calculated through discrete (numerical) calculus (see Appendix A).

This article is written by students. It may contain omissions and imperfections, reported as far as possible by our reviewers in the editing notes.



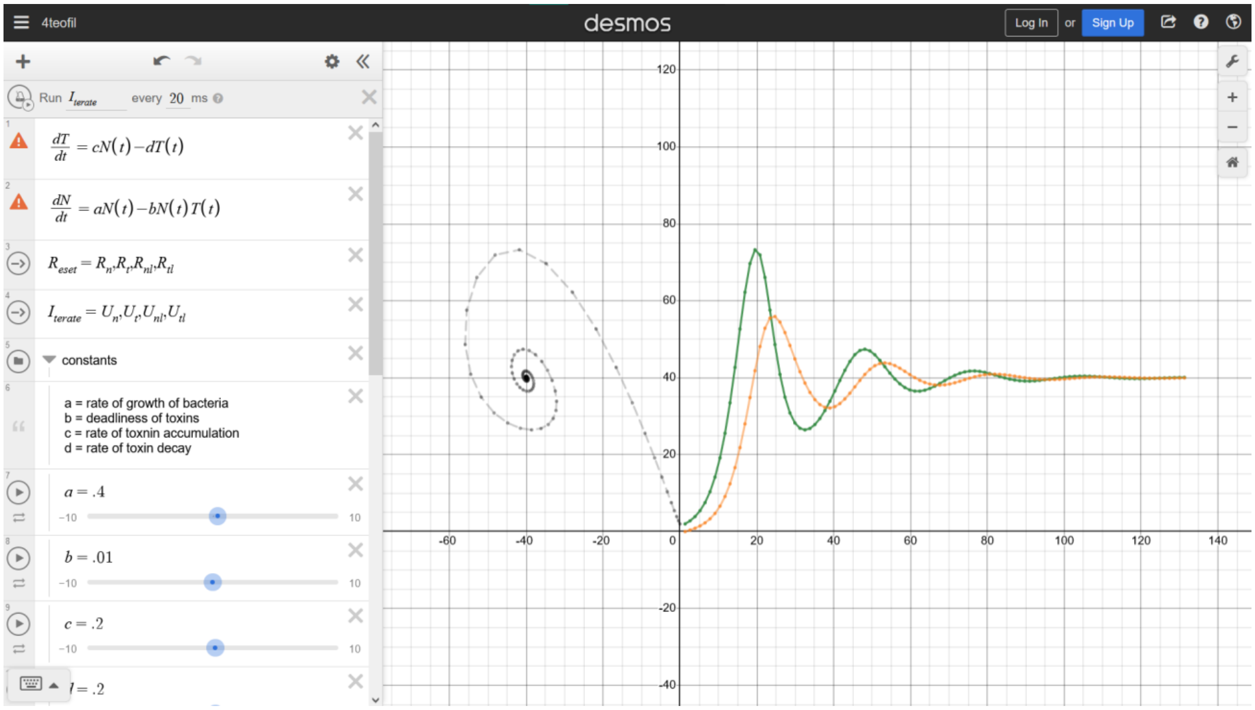
```
  ^
  |
  |      @
  |      #
  |      #
  |     @#
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |     ##
  |    @##
  |    ###@
  |    ####
  |    ####        @@
  |    ####        ##@
  |    ####        ###
  |    ####       @###
  |    ####       ####@      @@@
  |    ####       #####      @###@@      @@@@@@@     @@@@@@@@@@@@
  |    ####       #####@    @######@@@@@@######@@@@@@############
  |    ####      @######   @#####################################
  |   @####      #######@@@######################################
  |   #####@     ################################################
  |   ######    @################################################
  |   ######    #################################################
  |   ######   @#################################################
  |   ######@@@##################################################
  |   ###########################################################
  |  @###########################################################
  |  ############################################################
  | @############################################################
  | #############################################################
-9+---------------------------------------------------------------------------------------------------------------------------
  |123456789
```

Fig. 3: C++ graph in command prompt

The code can be found at the following link: https://codeshare.io/xvOYgj

C. Desmos

After finishing the C++ program, with the help of Mihai Lazăr we created a Desmos graph, which enabled a much more accurate representation of the stages of the bacteria population. With green and orange are represented the bacteria population and the toxin quantity, respectively (Fig. 4). The graph on the left represents the values in a parametric form N(T) (N on the y-axis and T on the x-axis). For the values, numerical calculus was used again.



Run I_{terate} every 20 ms

1 $\frac{dT}{dt} = cN(t) - dT(t)$

2 $\frac{dN}{dt} = aN(t) - bN(t)T(t)$

3 $R_{eset} = R_n, R_t, R_{nl}, R_{tl}$

4 $I_{terate} = U_n, U_t, U_{nl}, U_{tl}$

5 constants

6 a = rate of growth of bacteria
b = deadliness of toxins
c = rate of toxnin accumulation
d = rate of toxin decay

7 $a = .4$ −10 10

8 $b = .01$ −10 10

9 $c = .2$ −10 10

$l = .2$

Fig. 4: Desmos graph

Desmos graph: https://www.desmos.com/calculator/tiexm1f6w4

D. Graph plotter in Python

After we spent some hours analysing the earlier generated graph in Desmos, we decided to find another alternative, combining the advantages of the previously tried ones. The Python library **matplotlib** allows a fast way of plotting a graph, based on two arrays containing the coordinates of the points to be represented. The mathematical concept was the same as with the previous two versions, so it was easy to just translate it to python. The next step was to take advantage of the library mentioned earlier and generate the graph using the data sets. The result is an aesthetically pleasing and fairly accurate graph which can be easily replotted in order to analyse different sets of parameters (Fig. 5).
The significance of the lines is the same as previously, with the slight change of rearranging the parametric graph on the right of the y-axis.

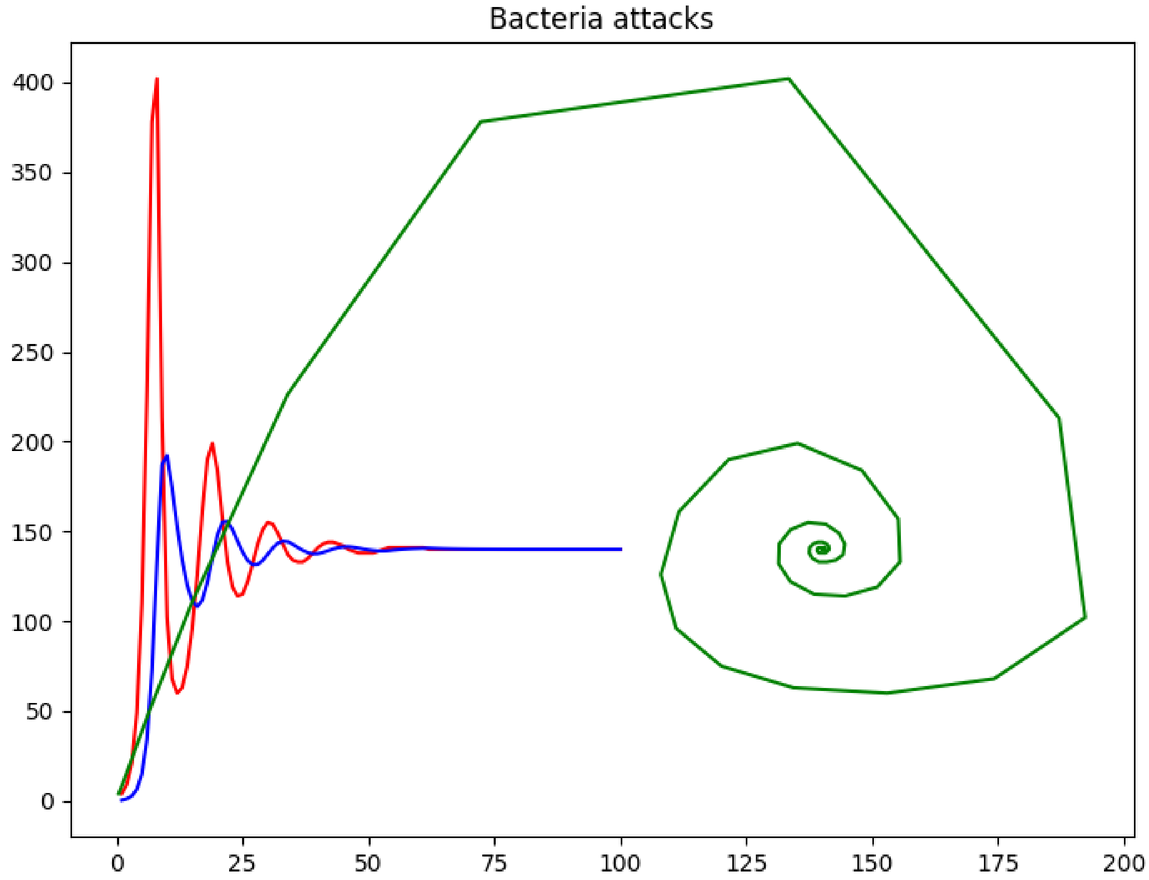


Fig. 5: Python graph

Python code: https://codeshare.io/nz0Yq4

7. References

- Robinson, J. C. (2004), *An introduction to ordinary differential equations,* Cambridge University Press
- **Matplotlib Python library explained** (https://www.geeksforgeeks.org/matplotlib-tutorial/)