Proceedings of The National Conference
On Undergraduate Research (NCUR) 2006
The University of North Carolina at Asheville
Asheville, North Carolina
April 6 – 8, 2006

Agent Based Model for the Production Mechanism and Control of Blood Cells in the Human Body

José Pimentel
Department of Mathematics
University of Puerto Rico at Humacao
Humacao, Puerto Rico

Faculty Advisor: Elio Ramos

Abstract

Traditionally, delayed feedback systems are formulated mathematically using Delay Differential Equations (DDE). In the following paper a computational model is presented for a delayed feedback system, namely the mechanism of blood cells production and control in the human body, using an agent based simulation. In the agent based approach the model starts with an initial population of mature blood cells (neutrophils) that evolve on time by the combined effect of production of young cells and random decay of mature cells. In this context the maturation period from young to mature is associated with a parameter that introduces a delay in the dynamics of the system. The model has been compared with a continuous DDE model due to Mackey & Glass[3] and is able to reproduce the general behavior obtained in more complex approaches. Furthermore, the model has been applied for the formulation, in simple terms, of several blood diseases.

Keywords: Blood cells, Neutrophils, agent-based model, simulation, computer model.

1. Introduction

Blood cells are continuously flowing through the blood. They are produced in the bone marrow in a process called hematopoiesis were stem cells become a blood cell depending on the need of the body. There are three main group cells named red cells, white cells and platelets. An optimum concentration of these cells in the human body is extremely important for a good health. The red cells are in charge of delivering oxygen through the body, the white cells (neutrophils) are responsible of defending the body against diseases and the platelets that are responsible of the coagulation of the blood. In a healthy person the production of the blood cells is regulated by the needs of the human body. If the concentration of cells is high the production mechanism diminishes, meanwhile, if the concentration of the cells is low the production mechanism increases[1].

In certain medical conditions this production mechanism can be affected by anomalies in the maturation period of the blood cells. If the maturation period is shorter or longer than normal the production mechanism will be unable to react as expected. Is for this reason that many mathematical models that pretend to model the circulatory system in human beings consider the maturation period as a key element. One of the best known mathematical models for the production of blood cells was proponed by Mackey and Glass[3]. This model was formulated as a continuous Delay Differential Equation (DDE) where the delay parameter was associated with the maturation period of the blood cells. Using this approach the authors were able to model several blood diseases.

In this paper an alternative approach is presented for modeling the production mechanism of blood cells in the human body. In this alternative approach, instead of solving a DDE, an agent based modeling technique is used. In this approach it is possible to model the behavior of individual cells and associate them with an aging (maturation) parameter. In addition, it is possible to include the effect of random decay (death) of blood cells. In each time step, new cells will be produced based on the number of mature cells.

The next section will describe the agent based modeling approach and its applications to the problem of modeling the blood production mechanism. Then the implementations of the computational model, using the agent based modeling language NetLogo, will be presented. Also a comparison of the results with the standard solutions of the Mackey-Glass DDE will be presented. Finally, some examples of diseases that can be modeled using our approach will be shown.

2. Computational Method

This simplified model for the production of blood cells assumes that the concentration of mature blood cells depends on the maturation period, the death of cells, and a production function. The maturation period (τ) is defined as the time that takes for a young cell to reach maturity. The death of blood cells (α) is related with the random death of blood cells that is occurring continuously in the circulatory system. The production function determines how many new young cells will be produced based in the current concentration of mature cells. A typical production function[2] is the following:

$$P(x) = \frac{\beta x}{1 + x^{10}} \tag{1}$$

where x is the concentration of mature blood cells and beta (β) is a free parameter. As is shown in the plot of P(x) in Figure (1), if the concentration of mature cells is extremely high the production of new cells is low, on the other hand, if the concentration of mature cells is extremely low the production is low. Only for intermediate concentrations the production is high.

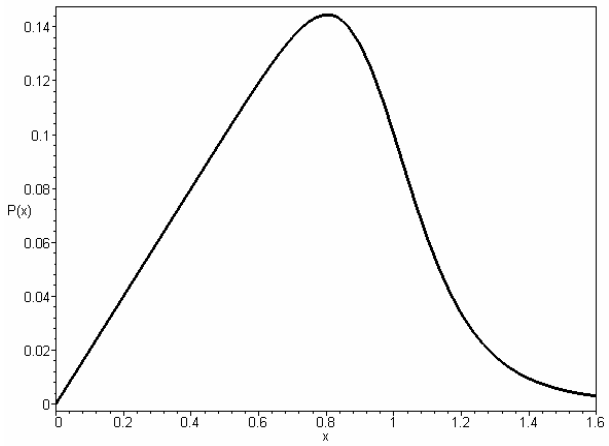

Figure 1: Blood cells production function

With this information it is possible to develop a computational model for the process of blood cells production using the method of agents. The method of agents[4], also known as agent-based or individual-based model, is a simulation technique where the complexity of the environment is obtained by the local interaction of each agent in the population. Unlike the standard mathematical approaches (i.e. differential equations) this technique considers the spatial dimension explicitly and quantities like concentrations and densities are calculated by the number of agent of a certain type in a region. In the computational model the agents involved will be the amount of neutrophils in the blood stream and a definition of certain rules

determine the conditions for the production and decay of these agents. More details about the computational procedure will be presented in the next section.

2.1. algorithm

The basic entities in the computational model are blood cells. Each blood cell is characterized by two parameters: the age (τ) and the state. When new cells are produced the age parameter is 0 and the state is "young". As the simulation advance in time the new cells get older (the age parameter increase) until they reach maturity and the state changes to "mature".

Step 1: Define an initial population of mature cells (age = τ and state="mature") and a time parameter with an initial value t = 0.

Step 2: Repeat step 3-6 until t reach a maximum time

Step 3 Compute the concentration of mature cells

Step 4 Evaluate the production function at the concentration given in Step 3. From this evaluation compute the amount of young cells (age = 0 and state = "young") that need to be produced.

Step 5 From the number of mature cells a certain proportion alpha will die.

Step 6 Increase the time and age parameter and verify if the state of each cell remain the same.

2.2. implementation of model

The model was developed using the agent-based programming environment NetLogo[5]. NetLogo was created in the Center for Connected Learning and Computer-based Modeling in Northwestern University in Evanston, Illinois. NetLogo consists of a two dimensional world containing two type of generic agents: turtles and patches. Turtles are the main entities that will be involved in the simulation. For each one NetLogo provides programming primitives to define its behavior under different circumstances. The turtles can be programmed to move around the two dimensional world and step-on on different patches and either change its properties or the properties of the patches.

The implementations of the blood cells production mechanism with NetLogo start with the following declarations:

```
breeds [cells]
cells-own [state clock]
globals [time number concentration positions]
```

The `breeds` primitive is used to declare a type of turtle named `cells`. For each cell, associations with the `state` and `clock` parameters were made. State can be either "young" or "mature" and clock is the age parameter. Then several global variables that will be used along the simulation were defined. Then a `setup` procedure is declared:

```
to setup
  ca
  set-default-shape cells "circle"
  setup-patches
  setup-cells
  setup-plot
end
```

The `setup` procedure is a user defined equivalent to the main function in programming languages like C/C++. The `clear-all` keyword is used to reset and clear all the variables in the simulation environment and the `set-default-shape` keyword is used to define cells with a certain shape. The `setup-patches` procedure is used to define the location of the bone marrow and the walls of the simulated arteries.

```
to go
  ask turtles
  [
    move-turtles
    mature-turtles
  ]
   decay-turtles
   set number (count turtles with [state = "mature"])
   produce-turtles
   set concentration ((count turtles with [state = "mature"]) / area)
   set time (time + 0.5)
   do-plot
end
```

To `go` is the main procedure of the code because in it has all function that creates the dynamics of the simulation. All functions will be defined later in the code. The variable `number` will count all the mature cells and return the value, variable `concentration` will count all the cells that have mature and divide it by the `area` that in this case is 100. The variable `time` will increment in 0.5 in each cycle.

```
to setup-turtles
  set number (initial-concentration * area)
  create-custom-cells number [
                                set clock tau
                                set state  "mature"
                                set color red
                                random-pos
                              ]
end
```

`Setup-cells` will create the initials conditions of the cells. The `time` variable will be 0. The variable `number` will be pre-determinate by a constant named `initial-concentration`, but this concentration needs to be multiplied by the `area` because concentration of cells is not equal to quantity of cells. Therefore to be able to see the cells the concentration needs to be converted into a real value. Next, the creation of the initial cells with the `create-custom-cells` command of number cells. The initial conditions of the cells are customize between the brackets. `Random-pos` is a procedure that determines the initial location of the cells. The `setup` function contains a procedure called `setup-patches`. This procedure creates the marrow bone and the arterial walls in the simulation.

```
to move-cells
  fd (random-float 1.0)
  set clock (clock + 1)
  if ycor > 2
  [
   rt 5
  ]
  if ycor < -2
  [
   lt 5
  ]
end
```

The `move-cells` procedure is used to define the movement of the cells. In the simulation the movement is from left to right with small random rotation when the cells reach the top to bottom boundaries (arteries).

```
to-report newborn [ cells ]
  locals [ x ]
  set x (cells / area)
  report (area) * ( (beta * x)  / (1 + (x ^ 10) ))
end
```

The `newborn` reporter computes how many young cells are produced based in the amount of mature cells. Unlike procedures, reporters are able to return values. This reporter is based on the function defined in Equation (1).

```
to produce-cells
  create-custom-cells newborn [ number ]
  [
    set state "young"
    set color green
    setxy -16 0
    set clock 0
    set heading (90 + random-normal 0.0 10.0)
  ]
end
```

The `produce-cells` procedure is used to create and define the properties of young cells. For visualization purposes initial cells are randomly oriented in a cone.

```
to mature-turtles
  if clock >= tau and state = "young"
  [
    set state "mature"
    set color red
  ]
end
```

The `mature-cells` procedure is used to define the transition from "`young`" to "`mature`" and the change in the visual appearance of the cells.

```
to decay-turtles
  locals[death]
   set death  int (count turtles with [state = "mature"] * alpha)
   ask random-n-of death (turtles with [state = "mature"])
   [
     die
   ]
end
```

The `decay-cells` procedure is used to model the random decay of mature cells in each time step.

2.3 interface of the model

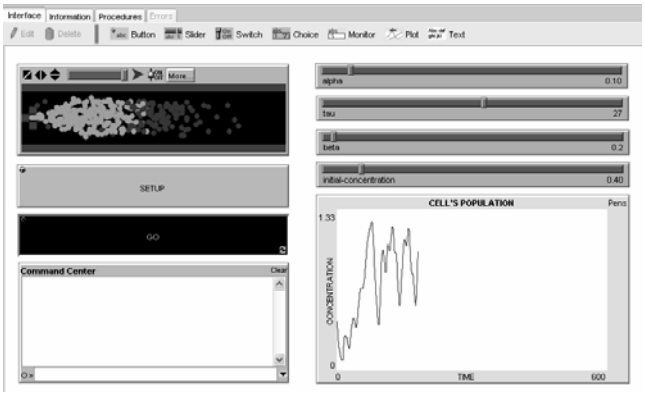

Figure 2

A simple graphical interface was developed in NetLogo. The interface (see Figure 2) consists of a graphical viewer, two bottoms, three sliders, and a plot window. The graphical viewer is where the simulated cells are visualized. The `setup` and `go` bottoms are used to initialize and run the simulation. With the sliders it is possible to specify the initial concentration, the decay (α) and production (β) parameter and the maturation period (τ).

2.4. example of the model while running

Figure 3: Dynamics of the cells with our computational model.

3. Results

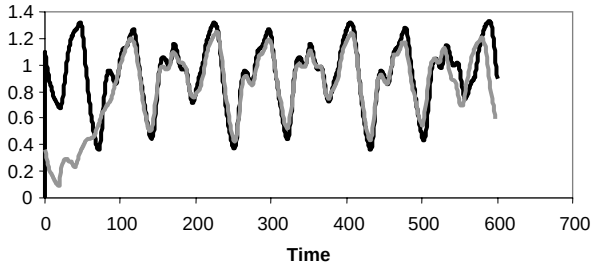

Figure 4: Comparison of computational model (grey line) and the continuous (black line) model

As a starting test-case, a comparison of the computational model with the continuous Mackey-Glass[3] model was made (Figure 4). The objective with this comparison is to show that besides the simplicity of the agent based model it is able to reproduce essential features (e.g. periodicity) of the continuous model. In general the solution of the Mackey-Glass DDE requires advanced numerical techniques. Furthermore the agent based model can be easily extended to more complex and interested scenarios.

Applications of the computational model were incorporated using the simulated environment to model three types of blood diseases. The first disease is Neutropenia were the neutrophils count drops under the normal count (Figure 5).

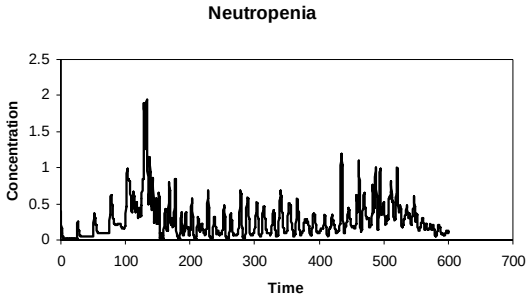

Figure 5: Neutropenia

Then a consideration of Agranulocytosis was made. Agranulocytosis is when white blood cells (especially neutrophils) are substantially absent in the blood stream (see Figure 6).

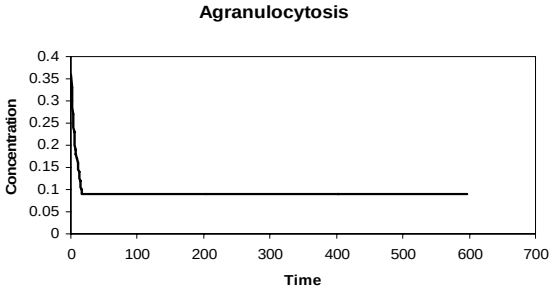

Figure 6: Agranulocytosis

Finally, Leukcytosis was simulated in the computational model. Leukcytosis is a disease were the count of white blood cells is higher than normal (see Figure 7).

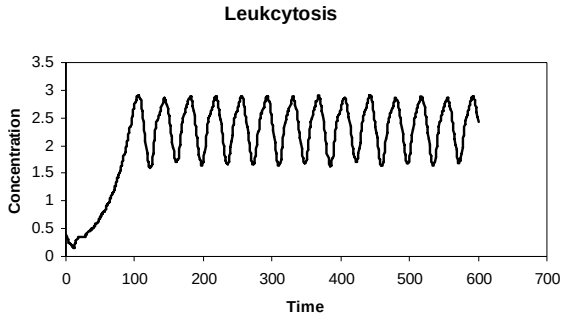

Figure 7: Leukcytosis

4. Conclusions

A computational model was developed, using the agent-based technique, to simulate the production mechanism of blood cells. Unlike the standard mathematical approach, commonly found in the literature, the blood cells were considered as individuals. For each individual it is associated an aging (maturation) property with a mechanism to produce and decay cells. The combined effects result in a model that exhibits delayed feedback dynamics and is comparable to more complex approaches.

5. Acknowledgements

This research has been sponsored by the National Security Agency (NSA) under the proposal #1498230-04-C-0486.

6. References

Book

1. Campbell N., Reece J. 2004. Biology (7th edition) Benjamin Cummings Publishing.
2. Mackey, Glass 1988. From Clocks to Chaos: The rhythms of Life. Princeton University Press.
3. Mackey, M.C., and Glass, L. 1977. Oscillation and Chaos in Physiological Control Systems. Science 197: 287-89.
4. Resnick, M. 2000.Turtles, Termites, and Traffic Jams. MIT Press.
5. Tissue, Seth, Wilensky U. 2004. NetLogo: A simple Environment to Modeling Complexity. International Conference on Complex Systems, Boston.

Journals

6. C. Colijn, M.C. Mackey. "A mathematical model of hematopoiesis: Cyclical neutropenia", part II, J. Theor. Biol. (2005), *237*, 133–146.
7. Drobnjak, A.C. Fowler & M.C. Mackey. "Oscillations in a maturation model of blood cell production", J. Math. Biol. (2004, submitted).