Internal Report 2012-14 September 2012

Universiteit Leiden
Opleiding Informatica

3D Visualization
of Mycobacterium
infection on Zebrafish

Xander Croes

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Contents

1 Introduction

Research on infectious diseases is difficult and is performed on animal models. Performing the research on animal models has the advantage of not being in humans, therefor causing less ethical issues, and making it possible to perform the research on large amounts of animals. Disadvantages of performing research this way are that the research is experimental and costs considerable time and resources. In light of these disadvantages *in silico* research is being used more recently.

An infectious disease with an enormous impact is tuberculosis. Tuberculosis, caused by *Mycobacterium tuberculosis (Mtb)*, kills millions of adult humans annually [1]. Because tuberculosis is not yet comletely understood, there is still a lot of research being done on this disease. In our approach we will continue this research by providing an *in silico* simulation and visualization tool for the infection process.

This chapter is organised as follows, first we address the zebrafish animal model, then we address the infection process, after that we address computational modelling, followed by the research question and finally the scope of the project.

1.1 Zebrafish

Zebrafish have become a well established animal model to perform research on. They are easy to maintain in large numbers, breed quickly and it is easy to genetically manipulate them. Moreover zebrafish have an immune system very similar to the immune system of humans, therfor they are very usefull in understanding infection processes. Currently, biologists study the early infection process of *Mycobacterium marinum (Mm)* on zebrafish embryos to understand the early infection process of *Mtb*. They use embryos because embryos have a fully developed innate immune system, but the adaptive immune system is not funtional yet. *Mm* is genetically closely related to *Mtb* and causes a disease in animals like fish similar to the disease caused by *Mtb* in humans [1].

1.2 Infection process

The infection process of *Mm* in zebrafish embryos, also seen in figure 1, is as follows: Biologists first inject the zebrafish embryo with *Mm*. The innate immune system detects the infection and sends immune cells such as macrophages to the infection site. The macrophages migrate to the infection site and when they reach the infection site they take up the bacteria in a process called phagocytosis. phago-lysosomal fusion inside the macrophages is supposed to kill the bacteria, but the bacteria are somehow able to resist and survive. The bacteria then infect the macrophage and proliferate. The amount of bacteria grows exponentially while the infected macrophage migrates to deeper tissues. After a while the infected macrophage is killed and other (uninfected) macrophages are attracted by the dead macrophage. The healthy macrophages aggregate around the dead macrophage and take up the bacteria of the dead macrophage. Again, instead of being killed the bacteria survive and infect the healthy macrophages. The aggregated macrophages now start to form structures called granulomas. After a while infected macrophages can get released from the granuloma in a process called dissemination. The infection process is then repeated, migrating the infected macrophage to deeper tissues [3].

1.3 Computational modeling

Researchers have been using computational models to understand complex biological processes. Advantages of a computational approach are the low cost in terms of time and resources and the ease of performing 'what if' scenarios. The main disadvantage is the difficulty of making a good representation of the processes.

Segovia-Juarez et al. defined an agent based model of an *Mtb* infection in alveolar tissue. They tried to display the pathogenesis of tuberculosis using a cellular automaton, however this was only a 2D view of the process [4].

Carvalho et. al made a petrinet model of the early Mycobacterial infection process in zebrafish embryos [2]. Although petrinets can generate very usefull data to visualize the infection process, petrinets cannot perform the visualization themselves. For this a tool is needed, which uses the output of the petrinet as input. In this project we continue the work of Carvalho et. al and make this tool.

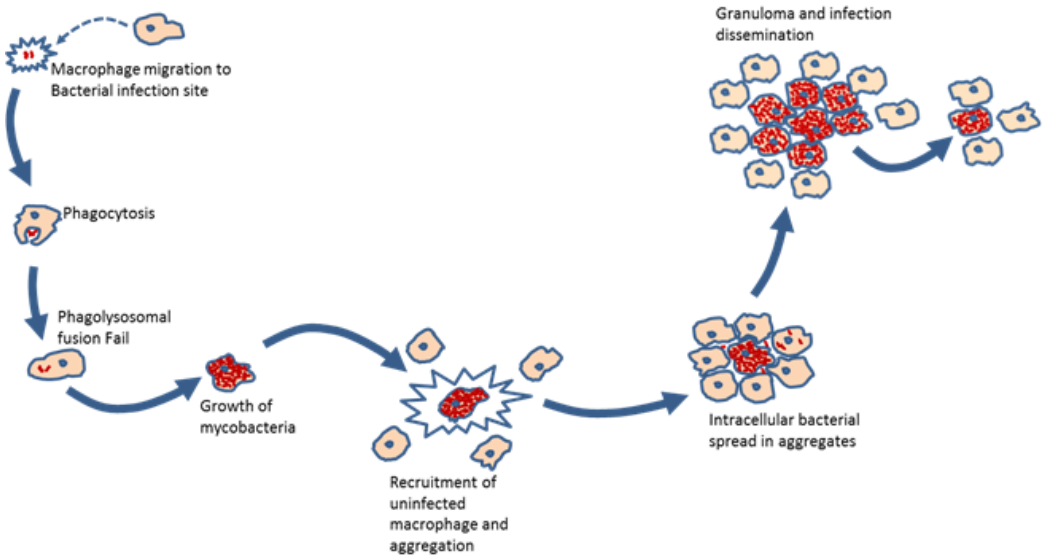

Figure 1: The infection process of *Mycobacterium Marinum* on zebrafish.

1.4 Research question

Our research question is: Is it possible to make a 3D visualization tool of the early infection process of *Mycobacterium Marinum* in a zebrafish embryo? If this is possible, then what do we need to do to visualize the infection process and how do we visualize it?

1.5 Scope

In this project we intend to define a prototype of a zebrafish 3D model to simulate and visualize the early Mycobacterial infection process. The infection process consists of: Introducing a bacteria in a 3D space environment that represents the zebrafish embryo. The immune response of the innate immune system to the bacterial infection and migration of macrophages to the bacterial infection site. Phagocytosis of the bacteria by the macrophage. Infection of the macrophage and bacterial proliferation inside the infected macrophage. And granuloma formation.

2 Material & methods

2.1 Hardware components

The program was developped and tested on an Acer Aspire 5520G laptop with an AMD Turion 64 x2 Mobile technology TL-58 cpu, NVIDIA Geforce 8600M GS graphicscard and 3 GB DDR2 memory.

2.2 Software components

The operating system used to develop the program was gentoo with kernel version 3.5.0. The program has been written in object oriented prgramming language C++. We used the OpenGL [5] and freeglut [6] graphical libraries. We used these libraries because we want to have very accurate control over our graphical representation. Furthermore these libraries are portable, which means they can very easliy be used in different operating systems.

2.3 Methods

We divided the program into separated components using a *model, view, controller* method [7]. The model contains the data structures and their information, the view handles the visualization part and the controller handles the infection process. Furthermore we used *object oriented* programming [8], which makes interaction and separation of different biological aspects very easy.

3 Analysis

In this section we perform a detailed analysis on what we need to model to perform a simulation of the infection process of mycobacterium on zebrafish. We first analyse what data structures are needed and then we analyse what subroutines are needed.

3.1 Data structures

We first need to define a space in which we can perform the simulation. We used a cartesian coordinate system, using x, y and z axis, on which we can visualize any 3D object. For easy reference to points in this space we created a struct, containing the x, y and z coordinates. Furthermore from the biological infection process we immediately discover four different objects for which we can make a data structure: the zebrafish, bacteria, macrophages and granulomas. In Figure 2 the basic structure of the program is depicted.

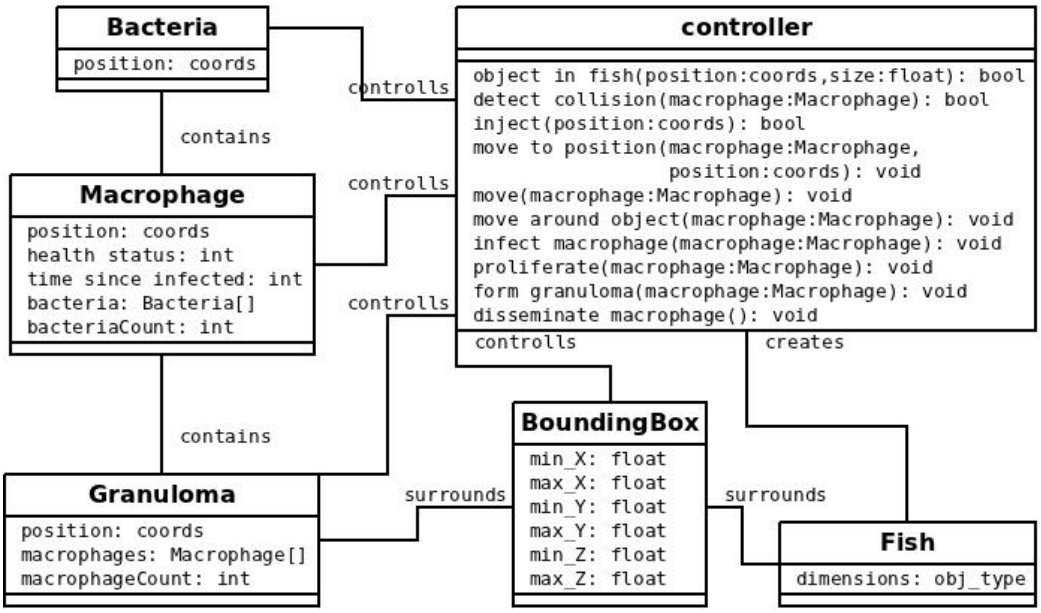

Figure 2: Program structure.

3.1.1 Zebrafish

To model the zebrafish we need to have an object with a specific size and shape. We Also need to limit the simulation space to the zebrafish body, because it makes no sense to track bacteria or macrophages outside the fish body. Furthermore there is no formation of granuloma outiside the fish body. To limit the simulation space we created boundingbox objects. These boundingboxes are basically rectangular objects, containing the minimum and maximum values of x,y and z. Because the fish body itself is not rectangular at all we used a *divide and conquer* method and split the fish body up into several boundingboxes, each surrounding a small part of the zebrafish body.

3.1.2 Bacteria

To model bacteria we created a small red sphere. This structure needs to have a specific size (radius) and position. We define the position by using the center of the sphere. The combination of radius and center can be used to make sure the bacteria is inside the zebrafish body.

3.1.3 Macrophage

To model macrophages we created a medium green sphere. This structure needs to have a specific size (radius) and position. We define the position by using the center of the sphere. The combination of radius and center can be used to make sure the macrophage is inside the zebrafish body. Furthermore we know from the biological process that macrophages can be either healthy, infected or dead. For this we added a health status to the structure. When a macrophage becomes infected, in a way the bacteria infecting the macrophage become part of it. Therefor we added a list containing the bacteria infecting the macrophage. To visualize the macrophage in an orderly fashion when bacteria are infecting it, we defined fixed positions within the macrophage, where infecting bacteria can be visualized. In order to perform proliferation we need to have some sort of notion of time. We added this in the structure by raising a counter every step that the macrophage is infected.

3.1.4 Granuloma

To model granuloma we created a big orange sphere. This structure needs to have a specific size (radius) and position. We define the position by using

the center of the sphere. The combination of radius and center can be used to make sure the granuloma is inside the zebrafish body. Because granulomas are an aggregation of immune cells like macrophages we keep a list of macrophages, that are contained in the granuloma, in this stucture. To visualize the granuloma in an orderly fashion, we defined fixed positions within the granuloma, where macrophages contained in this granuloma can be visualized. To prevent collisions between granulomas and macrophages we included a boundingbox, surrounding the granuloma, in the object.

3.2 Subroutines

From investigating the biological infection process we discover several main subprocesses: injection, detection, proliferation, granuloma formation, dissemination and migration. In figure 3 the main flow of the program is depicted.

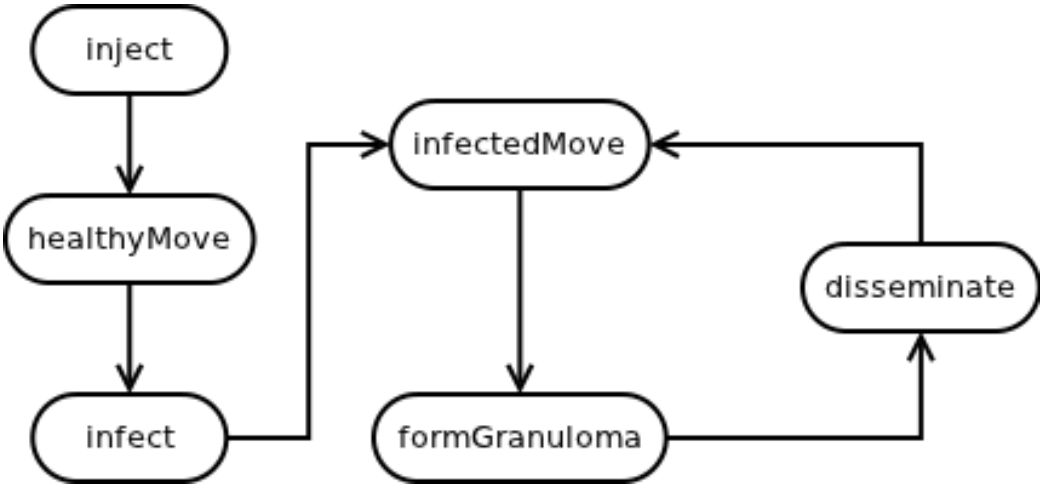

Figure 3: Program flow.

3.2.1 Injection

In the biological process we select a position to inject bacteria in the fish by jabbing an injection needle in the zebrafish body. We can see the position of the initial infection site as a set of coordinates [x,y,z]. In our model we need to select those coordinates also. We can do this by selecting a position with

the mouse. We let the user select the x and y coordinates with a rightclick of the mouse. The z coordinate is randomly generated because we use a 2D screen and input device, but when technology permits it (3D screen/goggles and input device) this could/should also be picked by the user. When all the coordinates are inside the fish body we create a new bacteria at the selected point.

3.2.2 Detection

In the biological process the immune system detects the infection and sends immune cells to the infection site. To model this detection we pass the location of the infection to one of the movement subroutines (Section 3.2.6).

3.2.3 Proliferation

In the biological process proliferation occurs when the macrophage is infected. During proliferation the amount of bacteria that are infecting a macrophage multiply. To model this we used the counter and health status defined in the macrophage structure to increase the amount of bacteria inside the macrophage. Because of the sizes of the structures we used to model the macrophage and bacteria, there is a limitation on the amount of bacteria that can infect the macrophage at the same time. Currently it's only possible to have a maximum of six bacteria infecting the macrophage at the same time.

3.2.4 Granuloma formation

In the biological process granuloma formation occurs when the macrophage has died. Granuloma formation consists of recruiting healthy macrophages and aggregating those healthy macrophages around the dead macrophage. To model this we first created new healthy macrophages and then pass the position of the dead macrophage to one of the movement subroutines to move the healthy macrophages towards the dead macrophage.

3.2.5 Dissemination

In the biological process dissemination occurs seemingly random on fully matured granulomas. During dissemination an infected macrophage leaves the granuloma and starts moving in the fish body again. To model this we

select a random granuloma and random position just outside the granuloma and create a new infected macrophage at the generated position. There is one bacteria infecting this macrophage.

3.2.6 Migration

In the biological process during migration the macrophage moves around in the zebrafish body. To model this we increase or decrease the x, y and z coordinates of the position of the macrophage. We can identify three different movement patterns, for which we created different subroutines:

The macrophage moves towards a specific position. This occurs when the immune system detects an infection that is not contained by living immune cells or granulomas. We model this by passing the position in the subroutine parameters and moving the macrophage in the direction of the position.

The macrophage moves around in the fish body. This occurs when a macrophage is infected. We model this by first checking if the macrophage is infected and if it is moving it along the inside of the edges of the bounding-boxes containing the zebrafish.

The macrophage moves around an object. This occurs when a granuloma is in the path of a macrophage. To model this we try to find the shortest path arround the blockage and then move the macrophage along this path.

4 Results

To visualize the infection process we used the entire zebrafish body. This means we had to upscale the bacteria, macrophages and granuloma, because normally those cannot be seen when looking at the entire zebrafish. We created 5 classes for the visualization: Fish, BoundingBox, Bacteria, Macrophage and Granuloma.

4.1 Zebrafish visualization

We visualize the zebrafish with the Fish class. This class uses a 3ds loader which loads a class of 3D object, Zfish.3ds. This is a file that contains a 3D image of a zebrafish embryo. This image is then loaded onto a wireframe representing the fish body (Figure 4).

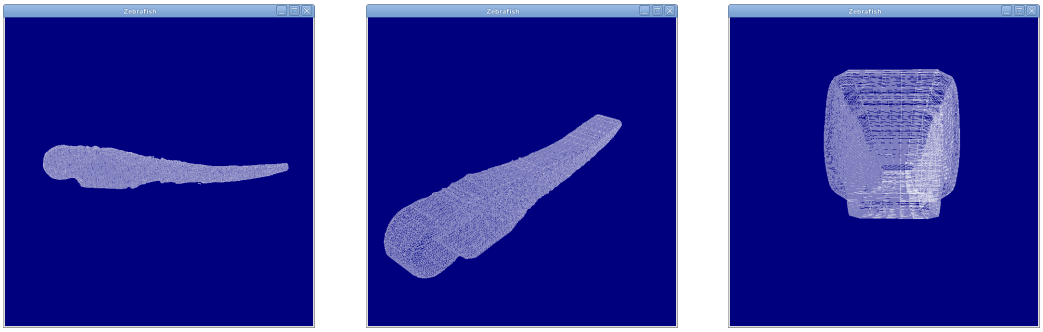

Figure 4: Different views of the zebrafish representation.

4.2 Simulation space

We limit the simulation space with the BoundingBox class. These boundingboxes can be visualized as yellow wireframe cubes on the fish body (Figure 5). The boundingboxes respect the coordinates of the zebrafish body and make sure macrophages stay inside the fish body. Furthermore, boundingboxes are used in collision detection. When used in collision detection a boundingbox can be visualized as a yellow wireframe cube around a granuloma (Figure 6). When a collision is detected, the macrophage will move around the blocking object. In Figure 7 we can see how the collision detection works and Algorithm 1 shows how we implemented the collision detection.

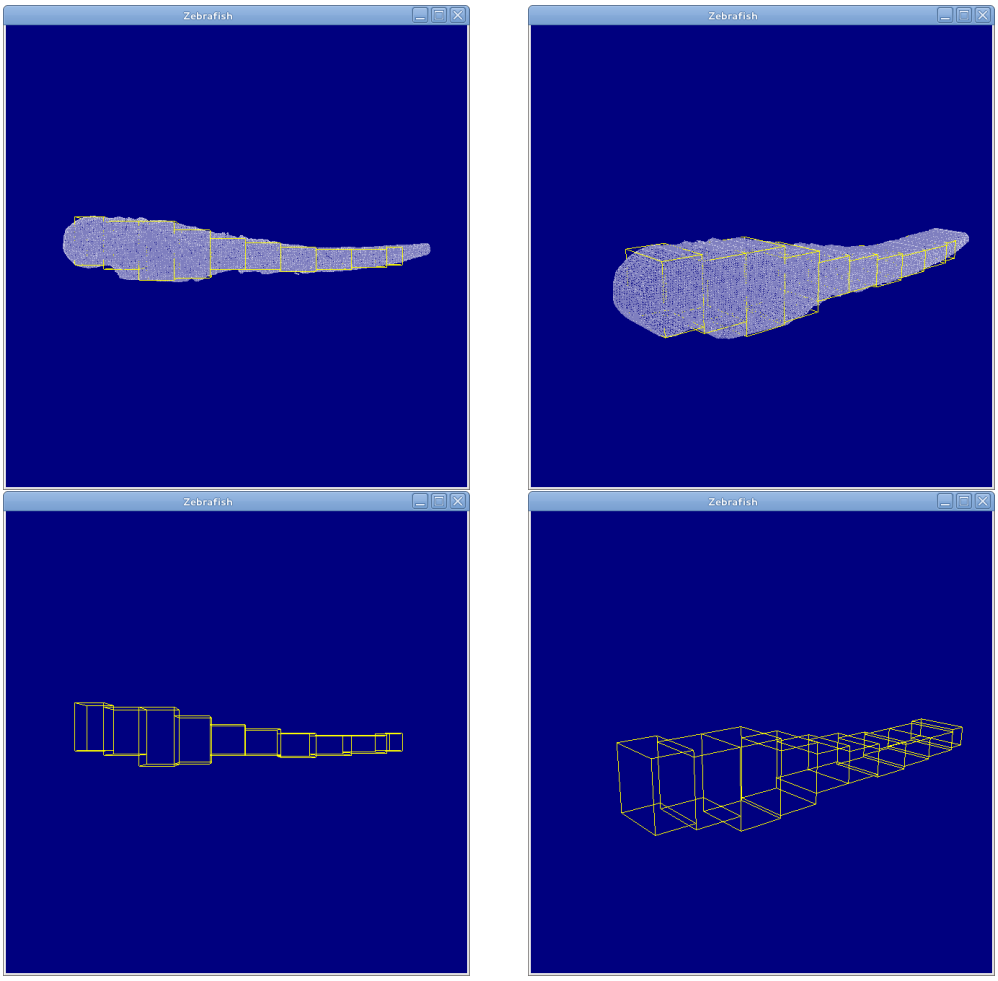

Figure 5: The boundingboxes on the fish body

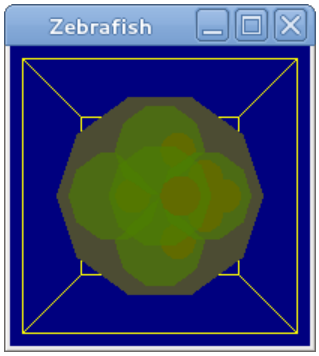

Figure 6: Boundingbox
around granuloma.

```
Procedure: detectCollision (pos, radius)
begin
    minX := pos.x - radius
    maxX := pos.x + radius
    minY := pos.y - radius
    maxY := pos.y + radius
    minZ := pos.z - radius
    maxZ := pos.z + radius

    for each granuloma do
        granBox := granuloma.bbox
        if the interval of minX to maxX overlaps with x coordinates of
        granBox then
            if the interval of minY to maxY overlaps with y
            coordinates of granBox then
                if the interval of minZ to maxZ overlaps with z
                coordinates of granBox then
                    return true;
                fi
            fi
        fi
    end
    return false;
end
```

Algorithm 1: Collision detection algorithm.

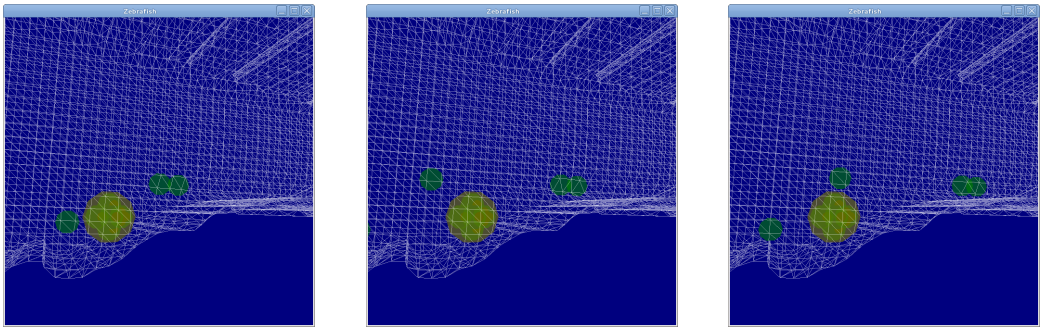

Figure 7: Collision detection at work.

4.3 Bacteria visualization

Bacteria are visualized as small red spheres. These spheres can be located either inside the fish body or inside macrophages (Figure 8). The center of these spheres lies at the coordinates contained in the class objects.

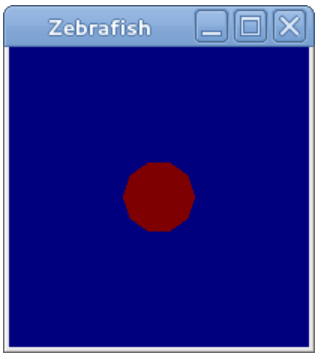

Figure 8: Visualization of a bacteria.

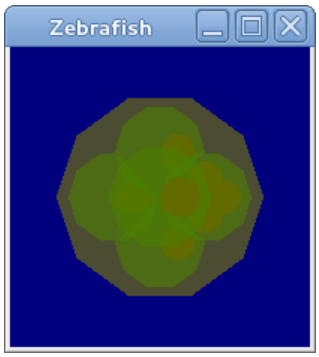

Figure 9: Visualization of a granuloma.

4.4 Granuloma visualization

Granulomas are visualized as big orange spheres. These spheres are located inside the fish body (Figure 9). The center of these spheres lies at the coordinates contained in the class objects. Furthermore the spheres are transparent enough to show the macrophages they contain. For the formation of granulomas we used Algorithm 2.

```
formed := false
positioned := 0

create new granuloma at position of dead macrophage
for  i := 2 to amount of positions in granuloma do
    create a new macrophage
end
while  not formed do
    for  each macrophage do
        if  macrophage has reached granuloma then
            positioned := positioned + 1
        fi
        else
            move macrophage to granuloma
        fi
    end
    if  positioned = amount of positions in granuloma then
        form granuloma
        formed := true
    fi
od
```

Algorithm 2: Granuloma formation algorithm.

4.5 Macrophage visualization

Macrophages are visualized as medium green spheres. These spheres are located either inside the fish body or inside granulomas (Figure 10). The center of these spheres lies at the coordinates contained in the class objects. Furthermore the spheres are transparent enough to show the bacteria they contain.

4.6 Simulation visualization

After running the tool for a while we get a result like that in Figure 11.

Figure 10: Visualization of a macrophage in several different stages.

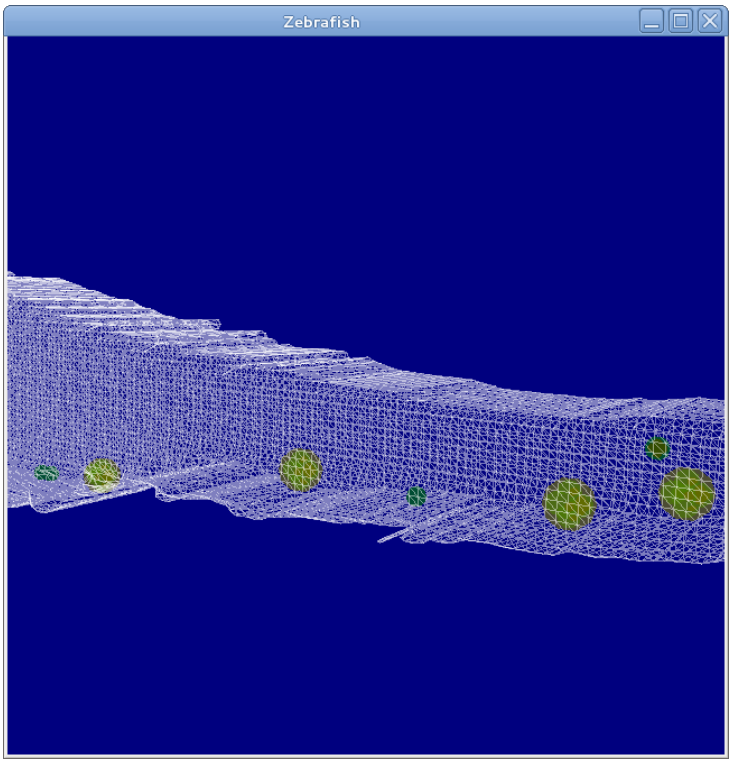

Figure 11: Result of running the tool.

Name	Description	Value
BACT_SIZE	Radius of the sphere representing the bacteria	0.02
MACR_SIZE	Radius of the sphere representing the macrophage	0.05
GRAN_SIZE	Radius of the sphere representing the granuloma	0.11
MAX_BACT	Maximum amount of bacteria that each macrophage can take up	6
MAX_MACR	Maximum amount of macrophages that each granuloma can contain	6
MOVE_SPEED	Movement speed of the macrophages (roughly every 25 ms)	0.01

Figure 12: Table with the constant values we used in the tool.

5 Conclusion

We have defined a prototype to visualize the infection process in a zebrafish 3D model, which means it is possible to make a tool to visualize the infection process. We did this by making components for every subprocess of the infection process. This was a very effective way to program, because we could test all components individually and make sure all functionality was working as intended.

5.1 Discussion

With the petrinet model it was hard to understand what happened during the infection process. We made a nice tool to visualize the information that petrinet models, and even other models, generate. It would be possible to create different subroutines to read different information formats, but this is not desirable because of the amount of programming code this requires. Therefor the output from the petrinet models should be in a specific format.

5.2 Future work

The program can be extended by including interaction between proteins that regulate the infection process, such as signalling, bacteria spread and dissemination. We can also include time lapse, changing the simulation space of the 3D zebrafish model as time passes to represent the aging/growing of the zebrafish. We may also include non linear simulation, making it possible to have multiple infection processes running at the same time. Finally we could add functionality to read inputfiles generated by image analysis tools and/or modeling tools, such as petri nets.

References

[1] Prouty, M.G., et al.: Zebrafish-Mycobacterium marinum model for mycobacterial pathogenesis. FEMS Microbiology Letters 225, 177-182 (2003)

[2] Carvalho, R.V., Davids, W., Meijer, A.H., Verbeek, F.J.: Spatio-Temporal Modeling and Simulation of Mycobacterium Pathogenesis Using Petri Nets. (2012)

[3] Lesley, R., Ramakrishnan, L.: Insights into early mycobacterial pathogenesis from the zebrash. Current Opinion in Microbiology 11(3), 277283 (2008)

[4] Segovia-Juarez, J.L., et al.: Identifying control mechanisms of granuloma formation during M. tuberculosis infection using an agent-based model. Journal of Theoretical Biology 231(3), 357376 (2004)

[5] http://www.opengl.org/

[6] http://freeglut.sourceforge.net/docs/api.php

[7] http://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller

[8] http://en.wikipedia.org/wiki/Object-oriented_programming