University College London

Simulation of Cancer Tumour Therapy by Using Agent-Based Modelling

MSci Project 2004/2005

Anastasia Semenova
Project Supervisor: Chris Clack

Abstract

Increasingly macromolecules, such as antibodies, are being used in the treatment of cancer. The aim of this project is to assist researchers in the development of their hypothesis for this treatment by building an application that models tumour interstitial is structured and how it may affect the antibody delivery to cancerous cells. A model was developed to simulate these interactions using agent-based modelling techniques, and a graphical interface projected this model into 3D animation. The system was tested against current research findings and showed favourable results.

1 Introduction

Cancer is a group of diseases characterized by an uncontrolled growth and spread of abnormal cells. It can be caused by both external and internal factors. Causal factors may act together or in sequence to initiate or promote cancer. In Britain, the chances of getting cancer in one's lifetime are as high as one in three[1]. There are many different treatments for cancer, and novel cancer therapies utilise macromolecular agents, such as antibodies. The agents in this molecular medicine are significantly larger compared to conventional therapies and success of these new therapies could ultimately depend on how well an agent can penetrate the interstitial matrix of the tumour (also know as the extra-cellular matrix). Modelling processes could aid in proposing and designing new strategies to improve the delivery of macromolecular agents to tumours [5].

The main aim of this project was to build a simulator with an agent based model to be used by researches of the UCL department of Biochemistry and Oncology to help them formulate their hypothesis about drug delivery. The model needed include a 3D graphical animation of the agent model, so that researchers could better understand the processes inside of the tumour. The model was limited to modelling the path of antibodies as they travel from the vasculature inside a tumour to a tumour cell through the extra-cellular matrix.

Normal cells divide and grow in an orderly fashion, compared with cancer cells which divide and grow out of control, disrupting normal body tissues and organ functions. The extra-cellular matrix, a network of proteins, holds together normal cells, but is disrupted by cancerous cells. Cancerous cell secrete chemicals, such as protease, into their surrounding environment in order to achieve angiogenesis and metastasis. Angiogenesis recruits blood vessels so that a tumour can nourish itself and grow. Metastasis is a process by which cancerous cells break away from the primary tumour, and are carried by the blood stream or lymphocyte system to form new colonies elsewhere. For metastasis in particular, that cancerous cell must break down its surrounding structured environment (ECM) to be able to that it can break

[1] http://www.cancerresearchuk.org/aboutcancer/whatiscancer/

loose. Cancer is classified by the part of the body in which it originates; different cancers are entirely different diseases, and respond to different treatment.

Conventional treatment of cancer does not distinguish between healthy and tumour tissue, the advent of immunotherapy (using antibodies and other macromolecules) allows a more localised and selective targeting of tumours. Antibodies in our immune system are produced to combat disease, they latch onto antigens that appear on diseased cells and the immune system destroys the couple. Manufactured antibodies can be made to prefer to latch onto antigens that are found on cancerous cells. Attaching toxins to the antibodies enables the delivery of a lethal dose of chemical straight into the heart of the tumour. The only difficulty is that it is awkward for large macromolecules such as antibodies to penetrate the tumour interstitial. On-going research is striving to solve that problem.

This report begins with an introduction to the biological elements that are being modelled in this project, focusing on the relationships between antibodies, cancer and the extra-cellular matrix (ECM). This is followed by a reviews of previous work in the field of tumour ECM, details of the hindrance to drug delivery in cancer cell and an analysis of the agents that were included in the model. Details of the model development and system design are subsequently discussed, together with various design choices and a detailed explanation of the model. The next section gives an overview of the implementation of the system, the detailing integration of the model and the GUI. The penultimate section evaluates the system, discussing brief experiment and the success of the system. Concluding remarks and suggestions for further work make up the final section.

2 Background

2.1 Biology

Antibodies and Antigens

Antibodies are proteins produced by the body that bind to specific antigens that have stimulated the immune system. Once they have bound, the antigen can be destroyed. Antibodies are the mammalian defence against invaders, namely the antigens that the invaders produce. All mammals have B lymphocytes that are each capable of producing a single type of antibody. The antibody is then able to bind to a specific chemical structure that is found on an antigen molecule. When antigens are present in the body, the specialised B lymphocyte that recognises the antigen, is stimulated to produce and secrete antibodies. The B lymphocyte clones the spleen or lymph cells that can produce the desired antibody, and these cells then generate that antibody. Due to the cloning process involved, these antibodies are called monoclonal antibodies. It is possible to manufacture antibodies artificially by cloning. Specially cultured cells can be used as "factories" for the production of antibodies: "…monoclonal antibodies are specific for cell-surface proteins expressed by certain types of tumour cells; chemical complexes of such monoclonal antibodies with toxic drugs or simply the antibodies themselves have been developed for cancer chemotherapy."[1]

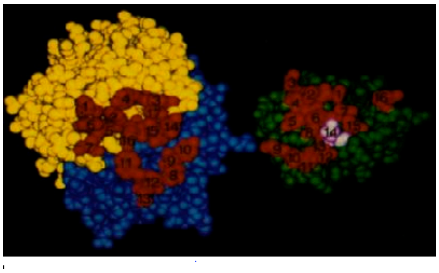


Figure 2.1 The binding surface (epitope) of the antibody and the antigen. The yellow and blue detail is the antibody. The antibody-antigen complex fit closely together.[2]

Proteins are designed to bind to molecules, from small ions to large and complex molecules. They have two properties that characterise their interactions:

- Specificity: the ability to bind to a specific molecule in the presence of others
- Affinity: the strength of that binding

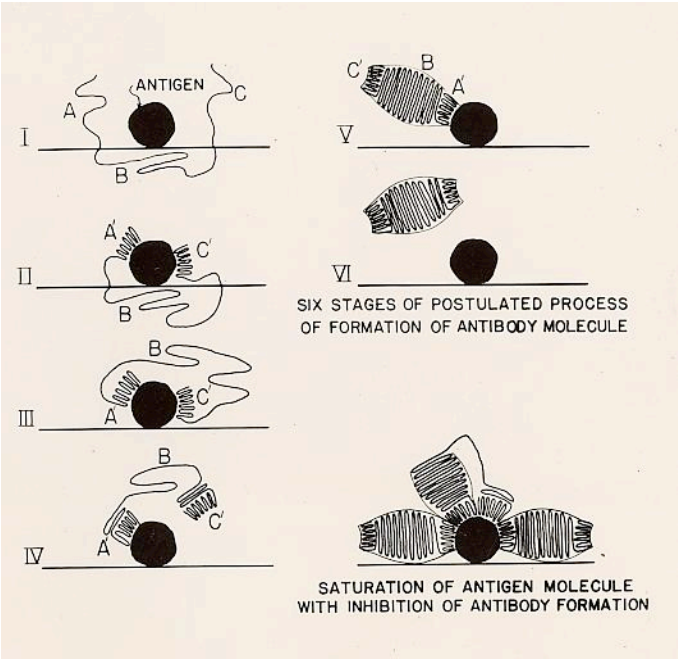


Fig 2.2 Six stages of postulated process of formation of an antibody.[3]

These two properties are highly developed in the antibodies. They have a binding site which is complementary to the antigen in shape and chemical composition. The presence of an antigen stimulates the body to produce a vast quantity of antibodies, some of which bind to different regions of the antigen. This forms an antibody-antigen complex that initiates a protective reaction in cells of the immune system.

This reaction cannot be achieved artificially e.g. for the purposes of cancer therapy. Introducing antibodies that are foreign to the body does not stimulate a natural protective reaction from the immune system. However: “antibodies with a high affinity for tumour-associated antigens target selectively to tumours when given intravenously; raidiolabelling the antibody enables their use in therapy or imaging.”[4] The artificially introduced antibody is able to bind directly to the antigen of a cancerous cell and is therefore able to deliver radiation or toxins accurately. The exploitation of tumour-located antibodies offers a means for exquisite activation of prodrugs to active drugs in tumour cells, leading to their targeted sterilisation without effect upon normal tissues.

However, the issues of scale and distribution present a problem. The antibodies need to be injected either into the blood stream or directly into the tumour. They require close proximity to the cancerous cells in order to bind to the antigens. The hindrance is the extracellular matrix (ECM) which all cells are surrounded by.

ECM

The extracellular matrix is a complex meshwork of fibres that hold the cells stably in a structure. This structure of cells makes up tissue. The ECM is made up of: "…a jelly of proteins and polysaccharides … Cells themselves produce and secrete these materials, thus creating their own immediate environment. "[5]

Out of the materials that the cells secrete the most abundant is collagen (about 80% of all protein in collagen). Each filament of collagen is shaped like a rod and is roughly 30 – 60 times as long as an antibody. The rods bind to form long triple helix structures that make the structure for surrounding cells. In a tumour, the cancer cells release a chemical that breaks down this structure. Around the cancerous cells the ECM is unorganised and the percentage of collagen is higher than normal. Studies have shown that this most hinders the permutation of molecules, such as antibodies, through the ECM to the cell.[6] This makes travelling through a tumour interstitial very difficult for larger molecules.

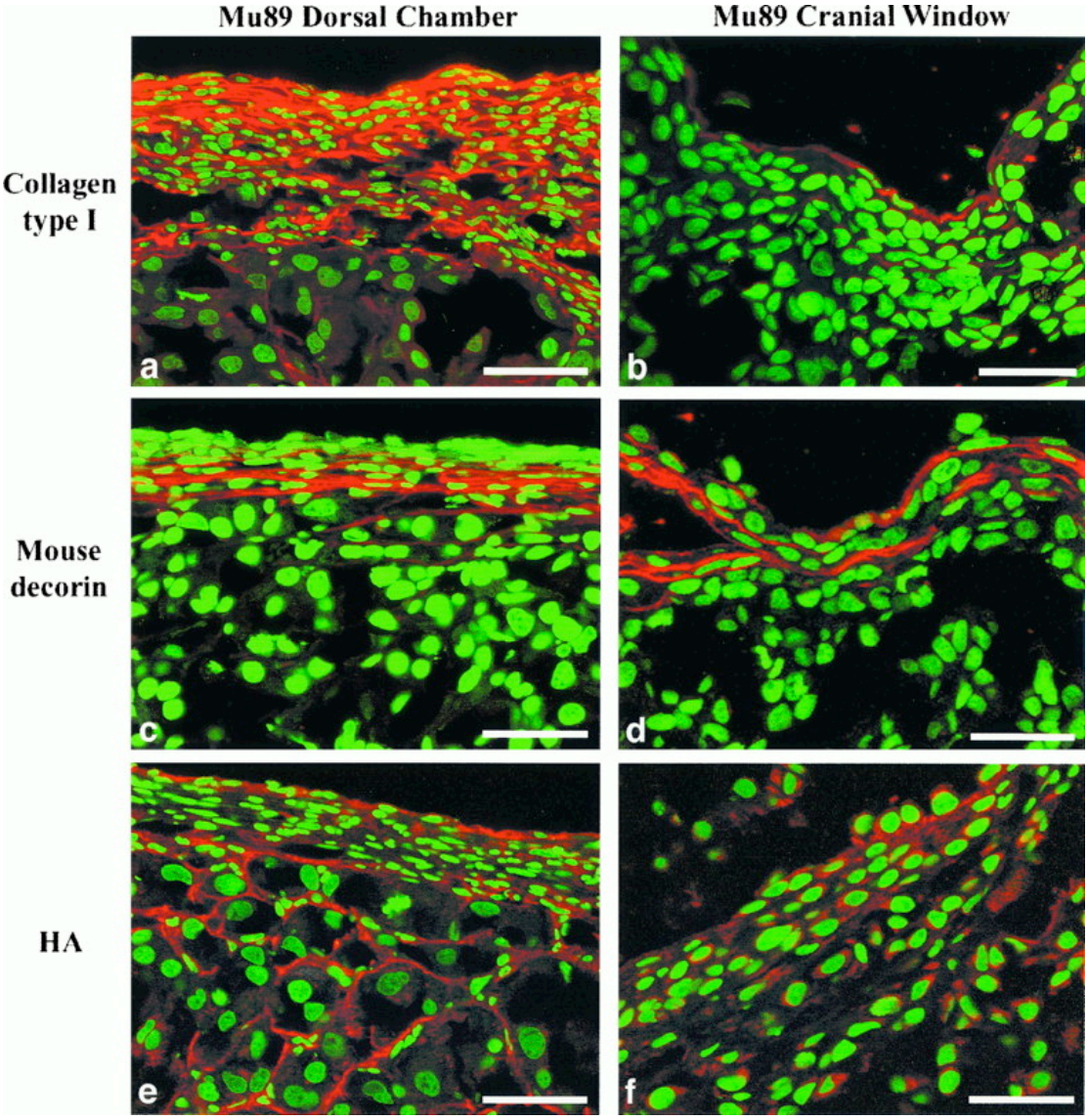


Fig 2.3 Immunostaining for collagen type I (*a* and *b*) and decorin (*c* and *d*), and labeling for HA (*e* and *f*) in DC (*a*, *c*, and *e*) and CW (*b*, *d*, and *f*) tumors. Collagen type I occupies a greater area of the periphery in DC than in CW tumors. In both DC and CW tumors the decorin staining is restricted to the periphery of the tumor. HA staining is intense in the center of Mu89 in the DC, whereas in the periphery the staining is weak. (Bar = 100 μm.) (http://www.pnas.org/cgi/content-nw/full/98/8/4628/F3)

Another component of the ECM is Hyaluronin. Hyaluronin is a highly hydrated GAG and is a major component of the ECM of migrating cells[2], such as cancer cells. Hyaluronin is charged and is able to bind to water molecules to create a gel-like

[2] Lodish et al. Molecular Cell Biology, 5th edition

matter. Osmotic pressure is created because hyaluronin can also bind to the acidic group, raising the ion concentration. Although hyaluronin poses no hindrance to smaller molecules, it is thought that larger macro molecules are impeded by it.

Inside a tumour, the ECM forms a saturated gel-like medium, made up of elastic fibres and collagen [3]. The main transportation force is dictated by diffusion, and although molecules may be dragged through the interstitial space by convectional fluid flow, research has found this unlikely to be the case inside a tumour. The cancerous cell also excretes antigens into the ECM, which makes it more likely that an antibody will bind to one outside of the cancerous cell. Ideally, the antigen needs to bind to an antigen inside the cell or on the cell membrane. Having considered this, the trajectory of an antibody is only a hypothesis, as analysis at this microscopic level is almost impossible.

2.2 Agent-Based Modeling

Complex Systems

"A complex system that involves numerous interacting agents whose aggregate behaviours are to be understood. Such aggregate activity is non-linear, hence it cannot be derived from the summation of individual components behaviour."[7]

The field of artificial life has shown that it is possible to achieve complex behaviour by employing simple rules and simple agents. For example, swarms – where individuals seemingly act in unison. They turn together, they flow around obstacles, they seem to move as one, all of which indicates that there is some controller who commands the movement of the swarm. However, swarming behaviour displayed in flocks of birds and schools of fish can be recreated by agents following three simple rules. The three rules below were created by Reynold in 1987, and were used for the first simulations of flocking behaviour: (1) avoid collisions with others, (2) match velocity of others, and (3) move towards the group centre. This technique has been used to model insect colonies and immune responses. It will be used in this project. The approach is called Agent-Based Modelling. The model considers the action of agents inside of an environment.

Agents and their Enviroment

Agents can be simple as a sub-routine or as complex as a conscious entity, intuitively they should posses a degree of autonomy. They can engage with the external environment without direct external control. They are also identifiable and hold their own identity, as this will be important in a system with many agents.

The agents abide by rules, and all the interactions between them and other agents, and the environment are pre-defined.

The environment is the space that the agents interact in, it can issue cues to the agents and confine them to a space.

Agent-Based Modelling in biology

Interesting and important aspects of a biological system are driven by local interactions which might involve interaction between molecules, cells or individual organisms. Previous approaches to modelling biological systems have been based around the use of differential equations to describe the possible behaviour at a global level. Thus reaction kinetics describes the behaviour of a system of chemicals from the point of view of the overall system but does not distinguish any local characteristics of the reactions. It is not possible to model the specific interactions of a molecule using this approach, for example, the collagen in the ECM, in which it may get trapped or bound off it [12]. By contrast, agent based modelling can closely model specific interactions - see for example [10] and 11].

3 Analysis

3.1 Problem Domain

It was very important to define the scope of the problem, so that the best suited solution could be delivered. By collaborating with researchers from the UCL

departments of Oncology and Biochemistry some key features of the systems were established:

1. The system should model the path of an antibody through the ECM as it travels from the vasculature to the tumour cell.
2. The system should be designed in a way that allows extendibility, the addition of new type of agents, interaction rules etc.
3. The system should model some staple agents, like the tumour cell and the antibody.
4. The system should be able to display model dynamics graphically.

The system should model the path of an antibody through the ECM as it travels from the vasculature to the tumour cell.

It was decided to model only the passage of the antibody from the vasculature to the tumour cell. This would include the movement of the antibody through the ECM; the model would be restricted to a focused area, as if it was under a microscope. Biological complexity still prohibits the analysis of transport hindrance within the ECM (or IM in a tumour) (Ramanujan *et al*., 2002), but computational modelling could assist. Various mathematical models of different aspect of drug delivery have provided statistical answers to key issues. However, the deeper understanding, needed to design the drugs, can only come from close scrutiny. Tumour-specific antibodies localize in the radiosensitive areas of the tumour, whereas non-specific antibodies can penetrate into the necrosis, a dead part of the tumour, where the dose is wasted. This results in the observed biological effect being different from the predicted effect of conventional dose estimates (Flynn *et al.*, 2001). In building an agent based model of this activity it is possible to come to a novel solution to account for these discrepancies.

The system should be designed in a way that allows extendibility, addition of new types of agents and interaction rules. It must also be able to modify parameters on existing agents.

Scientific knowledge of molecular biology is expanding; where scientists had previously thought that GAG molecules were most responsible for hindering

macromolecule transportation inside a tumour, in recent years it has been shown that actually collagen fibrils play a greater role and are more obtrusive than the GAGs [4]. Some research suggests that collagen is the main hindrance. However, other research suggests that although collagen significantly impedes diffusion, it alone cannot account for all of the hindrance [8]. The application has to be able to accommodate different theories and developments in scientific knowledge. Tumours are very different in construction and have widely different properties depending on many factors. For example, tumour site: tumours can be categorised into slow-diffusing (if planted into the dorsal chamber of a mouse) and fast-diffusing groups (if planted into the cranial window of the little beast), characterized by high and low collagen (of type I) levels. This indicates that not only tumour type, but tumour location in the host is vital information for drug delivery [7]. Based on the current clinical discoveries, to ensure the sustainability of the system, it has to be very flexible.

The system should model some staple agents, like the tumour cell and the antibody

.

The model would need to contain some biological agents that would be omnipresent and robust. For the purposes of this project, these will be assumed to be the antibody, the tumour cell, the vasculature (vein) from which the antibody will arrive.

The system should be able to display model dynamics graphically.

The graphical simulation of the model was proposed by the researchers of the UCL Oncology and Biochemistry department. They were interested in the visualisation of model dynamics in order to test new hypothesis and ideas.

3.2 Modelling the right elements – a review of research

3.2.1 Complexity

A tumour is a very complex object to model, and even when simplified to a small part consisting of a vasculature and several cells the complexity remains high. A study of the current research was essential to select the likely contenders for modelling, as the whole aim was to develop the system capable of mimicking the results of clinical data.

3.2.2 What the research shows

Fluid movement in a tumour is minimal

The movement of fluids, molecules and particles inside a tumour plays an important role in tumour growth, angiogenesis, metastasis and therapy. The fluid flow in tumours is severely retarded by the absence of lymphatic vessels, which take the flow of fluid from capillaries in normal tissue. This results in an increase in interstitial fluid pressure and in little convection. The main transport mechanism is, in fact, passive diffusion [7].

The structure of the ECM inside a tumour is more prohibitive to diffusion of larger macromolecules such as antibodies.

The success of novel treatments of cancer relies on large agents, such as antibodies, penetrating inside the tumour through the interstitum (the ECM). The ECM of all tissues is composed of fibrillar and non-fibrillar collagen, glycosaminoglycans (GAGs) and proteoglycans (the latter binding to collagen dictates the inter ECM spacing). In the past, resistance to flow had been attributed to one particular GAG, hyaluronin, whose charged ends attract water molecules to make the environment around itself gel-like. However, recent studies have found that collagen content of tumours has an inverse correlation with macromolecule diffusion. There are scientific results that show the success of delivery of larger particles to be dependant on tumour site and ECM composition structure. Diffusion in tumours of macromolecules decreases with size even more than could be expected from solution data due to the existence of cellular obstacles and matrix molecules [7]

The restrictive role of GAG molecules, in particular Hyaluronin, may have been over estimated

GAG molecules are known for the gelling-like qualities they bring with them. They need these to lubricate joints; indeed studies have clearly demonstrated that HA impedes fluid flow in tissues ([9],[6]). There is *in vitro* data that shows that in gels with both hyaluronin acid and collagen resistance to diffusion is greater than in gels where only one of the two is present. However, a study that looked directly at four different tumour types could not explain the modest differences in hyaluronin and total GAG to the observed variation of tissue functional properties. The study proposed that macromolecular access to the tumour was dominated by collagen assembly and was comparatively insensitive to GAG content variation. Hence, tumours rich in hyaluronin are not necessarily resistant to fluid and macromolecule penetration unless the hyaluronin is stabilized by a collagen [5]. Furthermore, some scientist hypothesise that the swelling potential of hyaluronin increases the pore size between ECM molecules and actually facilitates diffusion [7].

Collagen networks and bundles hinder diffusion

Netti et al [4] were the first to report that in contradiction to an existing paradigm collagen and not the GAGs contributed most to the hindrance of diffusion, and extensive collagen networks were found in the most penetration-resistant tumors. Testing 'rigid' tumors found that diffusitivity went up twofold after the tumor was treated with collagenase treatment, designed to enzymatically degrade the collagen network [4]. There have been many studies that have concluded that collagen presence and structure is an important factor in transport hindrance. Prepared gels have been measured for permeability, and have inversely correlated with collagen content, and although the results do not match to tumors permeability exactly, it can still be concluded that collagen can account for most of the diffusional hindrance [8]. High collagen content environments produce a major barrier to molecular diffusion, especially for larger molecules.

Antibody affinity facilitates retention of the antibody in the tumor

Antibodies that have a higher affinity for antigens have more chance of locking onto an antigen and staying there. The most effective antibodies have good clearance form the blood and good localization to the tumour [3]. Tumour-specific antibodies localize in the radiosensitive parts of the tumour, non-specific antibodies are able to penetrate into the necrosis, the dead part of the tumour, where the dose is wasted. The observed biological effect can be very different to the predicted effect from conventional dose estimates [2]. Choosing the right antibody for the job is paramount to successful treatment, for example, the triviant antibody (it has three arms with which to latch onto antigens) can be retained at higher concentrations and for longer, while the monovalent antibodies (these have just the one arm) have an increased mobility throughout the tumour.

3.3 Chosen agents

The initial desire to model everything from the effect of the blood pumping through veins to the water molecules was abandoned when the scope of the project was realised. The main agents to be modelled are outlined below. The rationale for them was taken directly from the most up-to-date research.

3.3.1 Antibodies

The main point of the model is to see what could potentially be happening to the antibodies on their way to the cell in order to improve of their design. As mentioned above, both tumours and antibodies can vary wildly. Trying to match the right antibody to the right tumour is important. The antibody needs to have several parameters, which can be tuned to modify the antibody's behaviour.

3.3.2 Tumour Cells and Antigens

Modelling the tumour cells alone would miss a crucial interaction between the antibody and the antigen. Antibodies can only lock onto antigens, so it is important to understand where the antigens may be located. Research shows that most of the antigens are on the cells surface, however some are floating freely in the ECM and a

few could be inside the cell. It should be noted, that antibodies do have a small chance of going through the cell membrane and getting inside the cell, providing the most effective way of treatment.

3.3.3 Extra-cellular matrix (ECM)

In modelling the ECM, the first problem to consider was the time scales on which different agents operate. The movement of even a slowly diffusing antibody, for example, would be far greater than any changes in the ECM. Therefore, the following simplification was adopted: only antibodies were considered as mobile, other agents were considered as static. The complexity of the model would increase significantly if all of the agents were allowed to be mobile. Furthermore, the results of the simulations most likely would not be affected noticeably. Considering the modern view on the subject of GAG molecules and their lack of retarding properties, it was decided not to include Hyaluronin in this model.

Collagen

Collagen is the main structural molecule in the ECM, and is arguably the most effected by the tumour. The normal structure of collagen is of long structured filaments of collagen molecules that hold the environment together. The tumour introduces a chemical that breaks this structure up into haywire. This, coupled with the cells' own production of collagen, increases the collagen concentration around cells. The cells act in this fashion to be mobile, but gain more than that, the path of an antigen when it is close to the tumour cell becomes disruptive.

4 Model Development and System Design

4.1 The Agent Base Model

The model that needed to be produced had to fulfil several criteria. It needed to be simple enough to run at a standard Workstation at reasonable speed. At the same time it had to capture the right level of details of the biological system. The three main biological elements that had to be modelled, were the *antibodies,* the *tumour cells* and some elements of the *extra-cellular matrix*. All biological elements were modelled as agents that inhabit or make up the environment.

4.1.1 Environment

The initial approach for modelling the environment was to use a cellular automaton. However, this approach was not suited to the problem. The vast number of static agents compared with the relatively few mobile ones indicated that a better approach needed to be considered, not least because of the vast scale of the problem (*see section 4.3.1* Consideration of Speed and Memory). Instead, the environmental space would be a plain 3D space that could be referenced by coordinates. The “background” environment of a tumour would be mostly fluid with an absence of convection due to the interstitial structure. This meant that the environment would be only populated by agents that will be described below.

The model space was limited to a cube ($10^3 \times 10^3 \times 10^3$ units, each unit is equal the size of 1 antibody (*see below for exact size*)), note that this restriction was necessary in order to capture the movement of the antibody graphically and was *not* a restriction on the potential scale of the agent model.

There is one element of this space that models an aspect of the biological setting: one of the six sides of the cube represents the vasculature from were the antibody agents start their walk.

4.1.2 Agents

The agents that were modelled are as follows and the reasoning for the choices has been discussed in Chapter 3. All of the agents needed to be completely self-efficient, keeping track of all of the information that might be needed about them as well as knowing all of the rules for their interactions. This fact was essential in making the model not only flexible to the introduction of new agents, modifications to old one, but also allowed for a flexibility in the total number of agents modelled. The agents that were included in this model are antibodies, tumour cells with antigens and collagen. They were split in groups of dynamic agents and static agents, and are described below.

Note: units of measure are Angstrom (A) = $10^{-10} meter$*, Nanometer (nm) =*$10^{-9} meter$*, Micrometer (um) =*$10^{-6} meter$*.* The "units" measure that is referred to below is exactly the size of a single antibody: a cube, whose side is 45 A.

4.1.3 Dynamic Agent

<u>Antibodies</u>

The smallest of the agents at just 45 Angstrom, the antibodies that are being used in the research of the UCL department of Oncology and Biochemistry have a shape that could be approximated to a cube. Their minuteness in comparison with most of the other agents in the model made them the measure in which all of the other agents would be measured.

Movement:

The transport of macromolecules in the tumour interstitial is primarily diffusion, as mentioned earlier. The following equation taken from Netti and Jain's Interstitial Transport in Solid Tumours[4] describes this movement:

$$J = -D\nabla c + cR_F K\nabla p \qquad (1.0)$$

Where J is the mass flux, D is the diffusion coefficient of the molecule in the interstitial, c is interstitial concentration, R_F is the drag coefficient, and $K\nabla p$ is the interstitial fluid velocity. However, a lot of these variables were unknown for the tumour type that was being modelled and the a simplified equation, 1.1, was used instead to model antibody movement through the tumour interstitial. At each discrete time-step the movement of antibodies is a sum of movement due to a pressure force and diffusion. Depending on what values the coefficients are set to, the antibody is able to move due to diffusion alone, from the pressure force, or the combination of both. The antibodies have two parameters describing step-sizes S_f and S_d for each type of movement. By changing parameters S_f and S_d both movement with a pressure force and diffusion can be modelled.. The following equation is formally describing the movement of the antibody

$$x_{i+1} = x_i + S_f v_x + S_d \Delta \qquad (1.1)$$

where :

x_i - is the location coordinates of the antibody at time *I,*

v - is the normalised vector which determines the direction due to a pressure force,

S_f -the step size resulted from a pressure force,

Δ - a normalised random vector,

S_d - the step size resulting from diffusion.

Interaction rules:

When structuring the interactions rules of the antibody the following things were considered,

i. The final aim of the antibody is to bind with an antigen.

ii. The interactions of the antibody with the tumour ECM.
iii. The behaviour of the antibody near tumour cells.
iv. The actions of the antibody when approaching a boundary of the space.
v. The appearance of antibodies from the vasculature.

i. Antibody – Antigen interaction

The binding interaction between antibody and antigen would be determined by the antibody's affinity for the antigen. This affinity would be modelled as a distance around the antigen at which the antibody suspends movement as described above and moves only in the direction of the antigen. For example, with the distance at 10 units, if the route of the antibody came with in 10 units of an antigen, it would move only towards the antigen.

ii. Antibody – Collagen interaction

In the ECM, the antibody could encounter collagen structures; it is known that collagen impedes diffusion by its large network of impenetrable fibres, and so it would be logical to model this interaction by not allowing antibodies to freely pass through collagen fibres.

iii. Antibody – Cell interaction

The antibody does not get attracted to cells but it can pass through to the inside of a cell, with a small probability, via the cell membrane. This interaction can be modelled by using a parameter to determine whether the antibody will enter the cell or not.

iv. Antibody – Border interaction

The limited space that the simulation considers is bounded by planes of coordinates. Once an antibody moved outside of these coordinates it reached its final state for that simulation. It was debated whether to bounce the

antibodies off the border back in, however as this would not make for a realistic model, it was not approved.

v. Antibody – Vasculature (vein) interaction

All antibodies will appear from the vasculature (the veins that carries the blood around a tumour) plane of the special environment. A random coordinate on that plane will be the start position of each antibody. There is a small probability that the antibody would eventually get taken up into the blood stream again (when the concentration of antibodies in the blood falls lower than in the tumour), this was modelled with the use of a parameter check much like the cell interaction.

4.1.4 Static Agents

Tumor Cell

The largest agent (the diameter of the cell is about 15-20 um, or thousands of units), was modelled as a sphere that could not fully fit into the designated space of the model.

The size and location were determined by its centre and the diameter. Each cell contains antigens, most of them would be distributed on the cell surface at random. To work out each coordinate for an antigen a random vector was taken, normalised, and multiplied by the radius.

Antigens

As antibodies are able to form a unity with an antigen, antigens were assumed to be the same size as an antibody and roughly the same shape. They are located on the cell surface membrane, their distribution is random (*see* 4.14 Tumour Cell) The interaction with an antibody would be initiated by the antibody, so the antigen needed not to know any other details. When attached to an antibody the two agents would occupy the same location in space.

The ECM: Collagen

From the ECM only the collagen network would be modelled. The modelling of hyaluronin was considered, however in the light of research showing the negligible effect it has in comparison with collagen and once again considering the scale restriction posed by the graphical display (only a part of the hyaluronin could be considered), it was left out.

Collagen was modelled as cylindrical rods, of the dimensions 10 units(5nm) diameter, and 30 units(aprox.15nm) length. The structure of the collagen around the cells was taken to be unorganised, as is common in tumours. The locations of the rods were therefore random and in both aspects of vector direction and location.

4.2 System design

4.2.1 Overview

Two different aspects needed to be incorporated into the application: the simulation of the agent model and the animation of this model and rendering it to screen. The simulation of the agent model needed to be flexible and extendable, however the graphical display was restricted by notions of clarity of picture, scale, and by the number of pixels available on an average screen.

A simplified model of the system can be seen in fig. 4.1. The user sets up the initial parameters and starts the application. The simulators' job is threefold: to calculate from the parameters the environment and the initial positions of mobile agents. The simulator then needs to iterate between calculating the locations of agents and resolving their interactions for each time-step in the simulation and updating the display to show this simulation to the user. It repeats the later two steps until the end of the run of the simulation. This system architecture does not leave much room for flexibility, for example re-running a simulation would be impossible as new calculations would be made each time. Also the possible lag that larger simulation would create would distort the animation on the display.

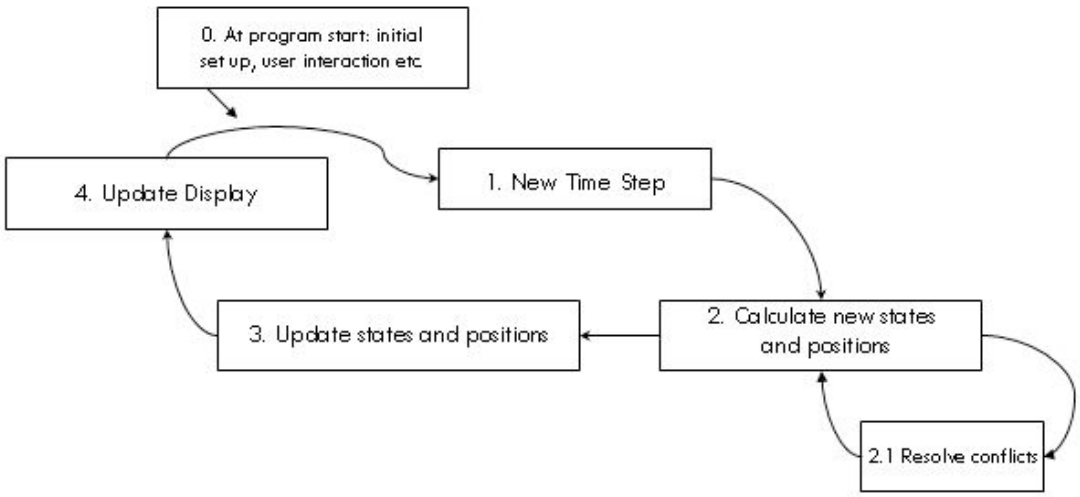
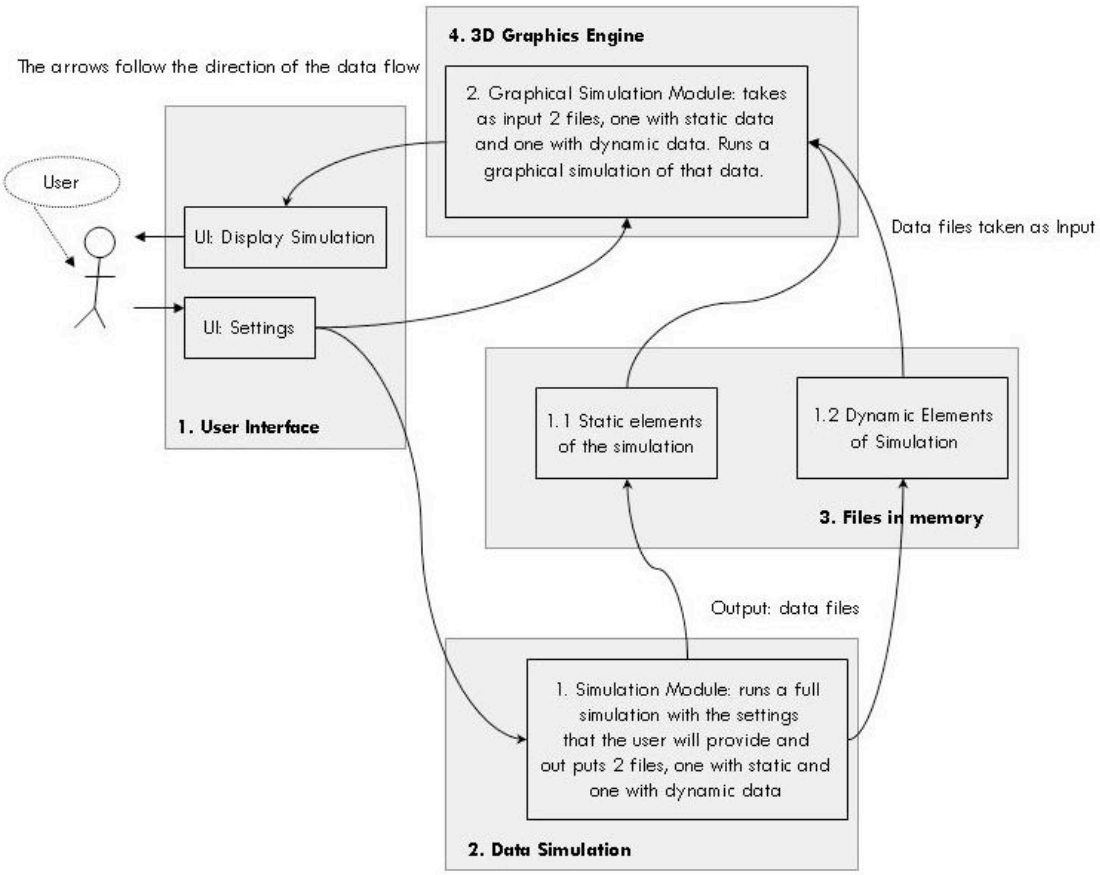


Figure 4.1 Naïve System Design

The solution to this rigidity was to split the system into two modules: one part would generate all of the agent model data and the other would display it. This would allow users to save model data and re-run simulations. Multiple data simulations could be generated to discover statistical properties of the model. This architecture of the system can be seen in *fig.4.2* Taking the work of simulating the data out of the cycle, as can be seen in *fig4.1*, allows for a more complex model and a higher number of agents. If the cyclical approach was applied, the graphical simulation would be impossible to watch – the delay in simulating each time step would be too long. This system design introduces the possibility of parallel computation as multiple runs can be simulated on different computers. This design was eventually implemented.



4.2 Final System Design

4.3 Components of the System

4.3.1 Data Simulator

The data simulator is of paramount importance to the application. All of the data for a complete simulation was calculated by it and written to file. When designing the simulator the most important considerations were speed, robustness and flexibility.

The speed of the simulation, although not critical to the success of it, requires discussion. A simulation that takes a close to infinite time will be of no use, and inaccurate. Such an application is likely to be the result of poor design. Keeping in only the essential elements initially and optimising the solution at each increment of complexity was the only approach that would yield the best effects.

Robustness of the model was crucial as the laws of physics had to be adhered to; the agents could not occupy the same space at the same time (unless it was part of the intention), and had to follow the rules of the model accurately.

The flexibility of the system's agents had to be incorporated into the design. It was foreseeable that the model would need to be extended and emerging factors from new research would need to be added to the model. New agents, old agents with differing parameters, new interaction rules and different spatial dimensions: all of these aspects had to be considered. Some of the aspects were implemented directly into the application (changing parameters f exiting agents), others had to be considered in the design of the system.

Consideration of Speed and Memory

Building a 3-D cellular automata had the potential to be the fastest computationally. This meant cells could be regulated by a baser motion of reactions and be truer to the biology. However, it very soon became clear that the active space would need to be at least a 1024 x1024 x1024 cell grid with some objects being thousands of times bigger than others. This means that even to store 1 byte of information per cell a gigabyte of memory would be used.

The natural solution to this would be to make the cell size larger and loose some of the detail of the model. This takes away from the simplicity; one cell would be able to contain more than one agent, yet some agents would still span over more than one cell. To keep track of all of these agents they would all need to be listed and checked. This approach would result in detailed lists of all agents that were in the model. As it seemed the advantages of large cells were significant, the final solution did not incorporate a matrix data structure.

The alternative was to have arrays of agents occupying, at each moment, locations in space. At each time step an iteration of all the mobile agents would be made, and consequently of all the static agents would apply the appropriate rules of interaction and logic (i.e. agent has not fallen out of the simulated area, etc.). This works quicker than the construction of the large grid that contains multiple agents. This approach also has the advantage of scalability and flexibility, and was the used to implement the model.

4.4 The Simulator and Agents

To ensure that the addition of new agents and manipulation of old agents would be as simple as possible the structure used for the simulator made was made to have one main class that would oversee all of the agent classes.

This controlling class would contain lists of agent objects and manipulate them using these lists. The main methods for this simulator would be to:

- Initialise the static environment.
- Simulate all of the mobile data, resolving conflicts.
- Write all of the data necessary to reproduce the simulation to a file.

For this to happen with fewest possible disruptions to the model, all of the agents needed to be self-sufficient. This means methods for self-generation, interactions and even reading and saving itself to the file would be hidden from the simulator inside the agents. This need-to-know detail of method construction ensured the resourcefulness of the model. As the overseeing class it could just call upon appropriate methods, of appropriate agents with hardly any need for restructuring.

4.4.1 File Structure: Dynamic and Static Data

When considering the two different types of data of mobile and immobile agents it was clear that there was a need to separate the two sets of data. Firstly, to enable the user to simulate multiple runs over the same environment. Secondly, to save on the awkwardness of generating and saving multiple simulations and saving the similar environment ones. The dynamic data file would make sense when considered with the static data, so they are linked.

It was likely that a user would want to run several simulations over one environment (where the environment is meant as the static agents of the system and the space they are in). There needed to be a distinction between static data and dynamic data. This would allow for a more efficient use of time. Simulations that could take many hours could be run overnight, and the five minute graphical visualisation of the data can be watched in the morning.

4.4.2 User Interface and 3D Graphic Engine

The user interface would be very important to the usability of the system and will allow users to understand the dynamics of the system. Changing parameters and loading files could be done manually. Considering that this program will be mainly used by biologists, the user-friendly interface has been used. A simple UI was created providing the ability to input and change parameters of the model, and to enable the user to clearly see the graphic simulation. The easiest solution was to create a simple dialog that would look similar to Windows interfaces. The central application interface would be relatively bare, just enabling access to the main functionalities of the program.

The two main elements of the user interface would still be the windows, where the simulation is viewed and the parameter input dialog.

4.4.2.1 Controls

The input side of the interface needed to be simple, fool-proof and to the greatest extent intuitive and un-intrusive. These qualities were achieved by creating a bar at the top of the program that enabled users to generate data, run the simulation as graphics, and a several buttons that enables rotation and re-runs.

When generating the simulation data, it was necessary to obtain some parameters from the users. This information has default values to ensure that there is less annoyance for users, and this data is based on the research considered. Limiting input windows (i.e. you can only enter numbers in a text box etc), will ensure that user error is kept to a minimum.

4.4.2.2 *Graphics* 3D

The design of the graphical output considered the complexity of the simulation.

Agent	***Graphical Representation***

Antibody	The antibody is represented as a rotating small cube (although bigger than its true size). The cube would leave a trail of yellow dots indicating visited locations. On contact with an antigen the colour of the antibody would change.
Cell	The cell is represented by an opaque sphere; this is due to the fact that most of the interactions happen between the antigen and the antibodies. This makes them clearer to see.
Antigen	Antigen represented as a cube similar size to the antibody, but different colour. This was mainly to ease the understanding of the model and make the interactions more clear.
Collagen	Cylinder-like objects were represented as opaque cylinders of different sizes.

5 Implementation

5.1 Development Tools

The programming language C# was chosen to develop the model, as it implements positive qualities from a generation of languages before it. Below is a summarisation of the key features of C#:

Language	Mimicked Features
Java	Object-orientation, Interfaces, Exeptions, threads, Namespaces, Strong typing, Garbage collection and Dynamis loading of code
C++	Operator overloading ,Pointer arithmetic in unsafe code, Some syntactic details (as java: Object-orientation (single inheritance), Interfaces, Exceptions, Threads, Namespaces)
New C# features	Reference and output parameters, Delegates , Indexers, Boxing / unboxing

As it is well adapted to the object-orientated paradigm it was very well suited to write an agent based model. The graphical simulation was implemented using openGL. The OpenGL graphics system is a software interface to graphics hardware (GL stands for Graphics Library). It enables programs to produce 3-D colour images of moving objects. An openGL C# wrapper was used to enable the openGL functions with C#[8] .

5.2 System Overview

Two modules would be required to implement this application; the GUI and an Agent Based Simulation Model. The Simulation Model module includes a number of classes, each of them implements different object of the model, e.g. the antibody, antigen. There is a container class, *Simulator*, that manages all simulated object and computes the simulation. The GUI would handle the user input and visualise the simulated data. The system is briefly shown in fig 5.1.

Environment, the Agents and the Model

The model that was designed does not have a specific environment, there is no a 3-dimentional matrix that holds the locations of the agents or agents themselves. Instead there are lists of agents of similar types, and an overseeing class that keeps the consistency of the model.

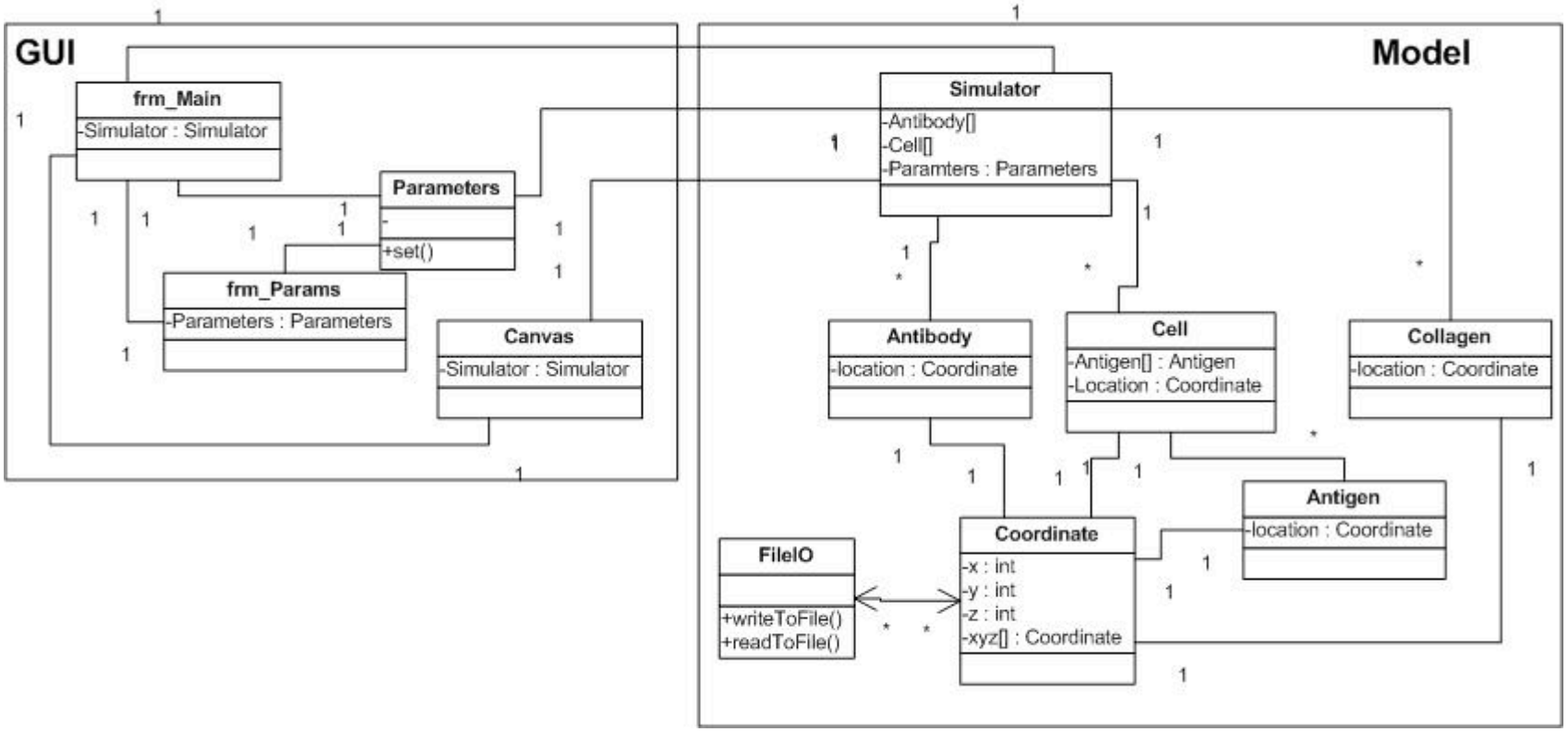


Fig. 5.1 A brief overview of the main classes in the system as it was implemented

The Simulator: Controller and Agents

The *Simulator* class is the manager of the model. All of the agents are independent, except where it is essential in the biology for them to be independent, for example a tumour cell will always have antigens. They are not aware of any of the other agent objects or interactions, as mentioned above (5.4); there is an overseeing class that controls all of the agents and computes the essential. The class simulates both the static environment agents and the dynamic agents in a similar way. The static agents are simulated first and their details are reordered, over them the dynamic agents are simulated. This implementation ensures the flexibility of the model as a whole for the addition of new agents and scaling the number of agents. The *Parameter* class is created for the passing of specific agent parameters and initialises the *Simulator*.

The Agents

The Simulator excepts agents to have certain methods in order to manage them.

The agents keep their exact location, size, and provide methods for all of the possible interactions they may encounter and are well aware of their state. Each agent object has a method that enables it to save itself to a file and to be able to initialise itself from a file, taking as a parameter a file name which to read and write from. Below are some details of agent specifications.

Cells

From the model we know that a cell is a static object. It was implemented as a sphere with a fixed diameter, location and an array of antigens. The antigens need to be subject to scrutiny and so the *Cell* provides a method for testing whether a particular object has fallen within the affinity distance of one of its antigens. Each *Cell* can check the distance between itself and the another agent. Only when the *Cell* finds an agent near itself, then it needs to check the agent location against its own antigens.

Antigens

Antigens can be static or mobile depending on the model; (whether they are a carcinoembryonic antigen, a cell membrane antigen, or one that is floating in the ECM). This model considered only the more frequently occurring a carcinoembryonic antigen. Antigens are simple agents that have a fixed location, size and affinity distance, which are all initialised during their creation

Antibody

The *Antibody* is initialised by its start coordinates, two step sizes and a vector. It moves freely through the space according to the formula 1.1 (Chapter 4). The two step sizes allow the incorporation of different models of movements: diffusion and forced movement.

The antibody is the only agent in the system whose location changes during the simulation, therefore it needs to be aware of the location of other agents. At each time step the *Simulator* is responsible for giving the *Antibody* a new location which is valid (there is no interference with other agents and the rules of the model are adhered to). The antibody has a final location which determines its final state, either reaching a border or reaching an antigen.

Collagen

Collagen is a static agent implemented in the shape of a cylinder. It has a start and end point, as well as a diameter. *Collagen* is created after the *Cell* objects have been created as it cannot be found inside the cells.

5.3 File Input/Output:

The model uses two kinds of information, static data and dynamic data. Static data holds information on all of the objects, whose location never changes; dynamic data hold information about all of the objects whose location does change. As each agent is able to read/write to file, there is no complex code for saving data in the

Simulator class. From the code below, one can see that the antibody needs to save its ID, Status, Number of Visited points, diameter and step size, followed by an array of the coordinates that it has visited. This type of structure is inappropriate for collagen, which only has one location and different variables. The information that is needed to recreate each agent is very specific, if the reading and writing to file was handled by the *Simulator* class, or any other one class, then every time a new agent was added or an old agent modified, complicated code in both classes would be modified.

Code from saveToFile(string m_fileName, bool overwrite) method from the *Antibody* class

```
//save antibody and its walk to file

//antibody information

tw.WriteLine(this.id+" "+(int) this.Status+" "
+this.location.XYZ.GetLength(0) +" "+this.diameter +" " +this.stepSize);

//the walk coordinates

for(int i = 0; i<location.XYZ.GetLength(0); i++)
{
tw.WriteLine(this.location.XYZ[i][0]+" "
+this.location.XYZ[i][1]+" "+this.location.XYZ[i][2]);
}
```

The code in the Simulator would stay the same for saving an antibody regardless of which variables change in the Antibody class (example code below). This is the same for new agents. The Simulator only needs to know that the agent has a “saveToFile” method.

Code from the saveDynamicData(string m_fileName, string staticFileName)

method from the *Simulator* class.

```
// save antibodies info
for (int i = 0; i < antibodies.Length; i++)
{
antibodies[i].saveToFile(m_fileName, true);
}
```

Reading from file is handled in a similar ubiquitous approach. The *Simulator* reads the file to find out what agents types there are in a simulation and how many of each agent there is. All the Simulator has to do is call an agents constructor with the specific filename and the current index of a counter, and the rest of the work for initializing the agent with the right values is done by that agent.

Code from the readStaticData(string m_fileName) method from the *Simulator* class.

```
// create new cells from file
cells = new Cell[numCells];
for (int i = 0; i < numCells; i++)
{
cells[i] = new Cell(m_fileName, i);
}
```

5.4 The GUI

The GUI was implemented as a windows application.

The main screen is the start up of the application and is responsible for creating all the essential objects for both the graphics engine and the simulation: *Simulator, Parameters* and *Canvas*. Its main responsibility is to present the simulated data as a 3D Model and provide access to the main menus for user interaction. From this screen the user can call up two other forms. A parameter setting form, *frmParams*, where the input data to the *Simulator* is initialized and a file is selected for saving the simulated data. The other form, *frmStats*, displays a summary of the simulation.

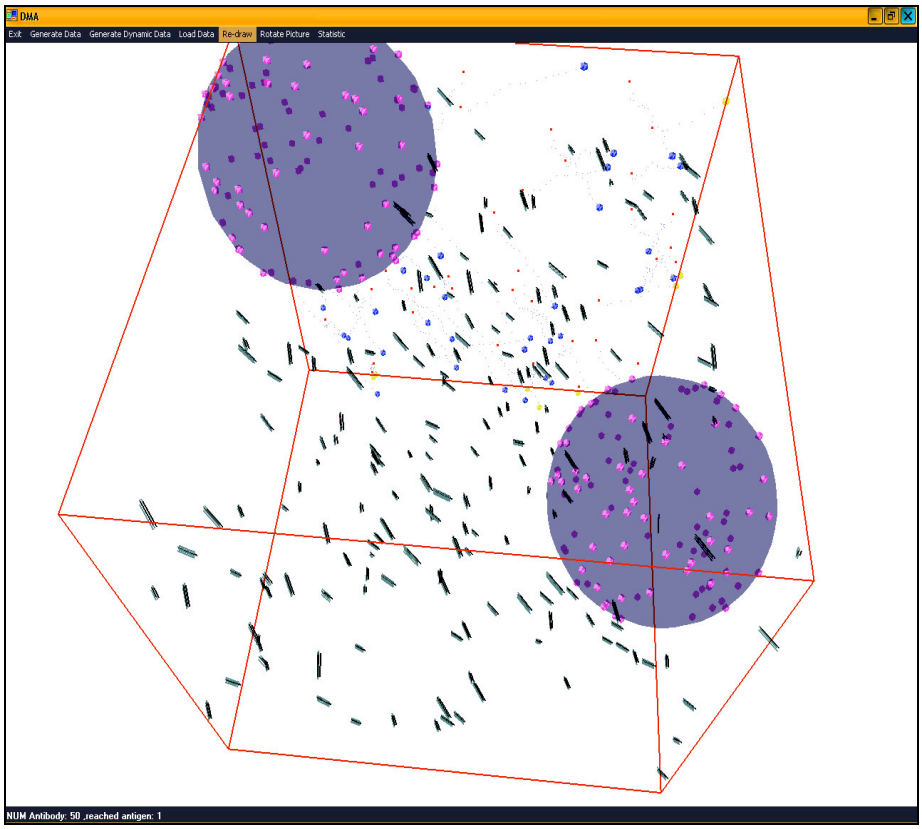


Fig 6.X Screen shot of the application – main screen (the colours have been reversed for clarity)

For more screen shots of the application see Apendix

5.5 Testing the System

The testing of the code for sanity was done throughout the implementation with the use of TestDriven.NET. This software integrated with the .NET environment and facilitated testing on the fly.

When generating a model that looks chaotic, it is hard to be able to judge its correctness, the only way of doing this is testing it against real data. This will be explored in the next chapter (Evaluation). The other way of testing the simulated data was to look at the graphical simulation, this was gave immediate results that could be checked for correctness by eye.

The implementation of the UI was tested for robustness. People unfamiliar with the system were asked to try to run a simulation, save the data, and then load the simulation again. This showed up some usability issues that were corrected.

5.6 Extendibility of the System

Implementation of this system allows high flexibility; the complexity of the model can be increased incrementally without the need to make changes to the existing agents. New types of agents (e.g. other molecules in the ECM) can be added seamlessly into the system, the only classes that would need to be modified would be the *Simulator*, *Parameters* and *Canvas* which were written with a view to extendibility.

6 EVALUATION

The purpose of the evaluation was to see if the modelling was successful and as true as it could be to results gathered from *in vitro* and *in vito* studies.

6.1 Experiments and results

The experiments that were conducted needed to test some statistical properties of the simulation. Runs of fifty types of each simulation were made and statistical analysis was conducted on the data. The null hypothesis posed were that:

- 1. There was no relation between the affinity of the antibody and the final number of antibodies that managed to attach to antigens.
- 2. The was no relation between the success of the antibody reaching an antigen and the mode of transportation (a pure pressure force, just diffusion or the two combined.)
- 3.There was no relation between the amount of collagen in a tumour and the antibodies success in reaching an antigen

The methods for conducting the testes were to run multiple simulations changing key parameters and seeing the statistical properties of the data.

These were the results for the experiments conducted for hypothesis 1:

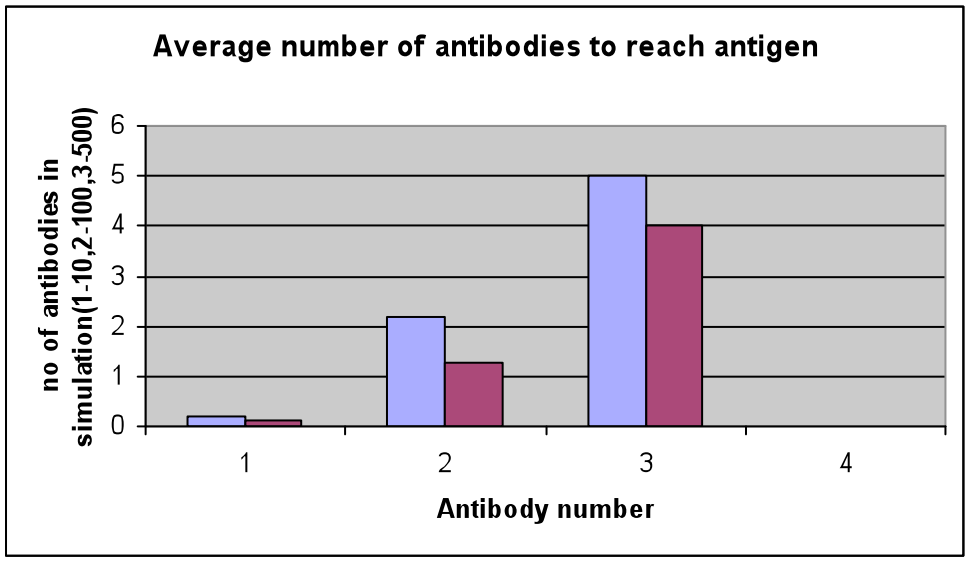
		Number of antibodies that reached an antigen by the end of the simulation		
Affinity	no. of antibodies	Mean	Standard Deviation	% of antibodies to reach an antigen
10	10	0.20	0.42	2.0
10	100	2.20	2.15	2.2
10	500	5.00	4.45	1.0
20	10	0.55	0.93	5.5
20	100	5.40	2.32	5.4
20	500	19.00	11.70	3.8
50	10	1.18	1.17	11.8
50	100	14.40	6.48	14.4
50	500	83.40	41.11	16.7

Fig. 6.1 results of statistically analysing the affinity of an antibody

There is a correlation that as affinity of an anybody increases so does the percentage of antibodies that make it to an antigen. An affinity of 10 results in 1/100 making it to an antigen, but an increase the affinity to 50 and the figure is close to 1/5. These results were to be expected by common sense, however the confirmation of them and the percentage increase were essential. The model that is simulated in this respect follows the literature on the subject and this is reassuring.

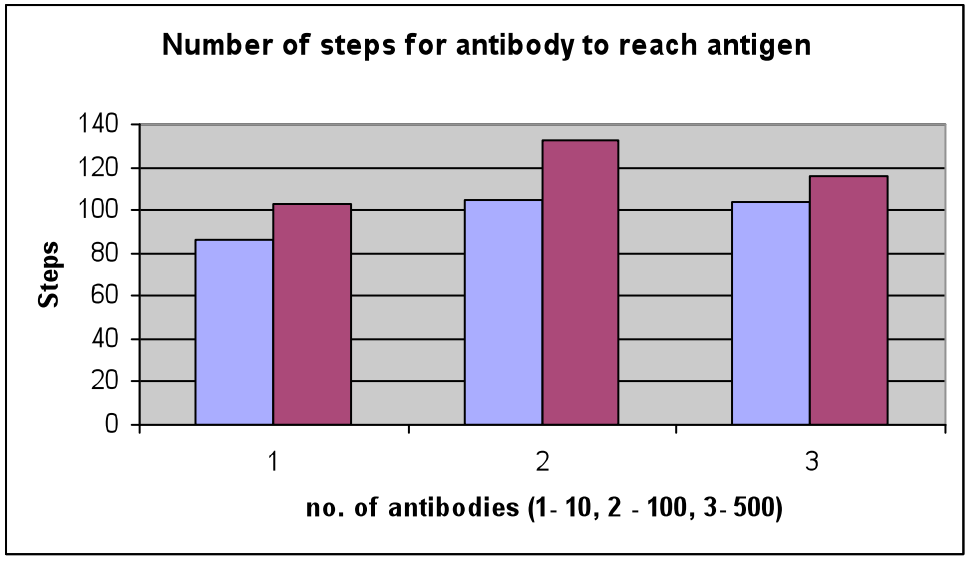
There were no clear results to disprove hypothesis 2. The averages were similar for both movements. Indeed, it seemed that with a pressure force to guide the antibody it had less success in attaching to an antibody.

Hypothesis 3 results were encouraging, the simulation with no collagen have more success in antibodies binding to antigens.



Note that the dark purple columns represent 4% collagen simulation, and the light blue represents 0% collagen in simulation.

There was another interesting result from the comparison of collagen concentration. The simulation with the greater collagen concentration produced results where antibodies generally took more steps in order to come to a final state that in a simulation with no collagen. This suggests that although collagen might be hindering in macromolecule transportation through the ECM, its presence might retain the antibodies in the ECM of a tumour for longer.

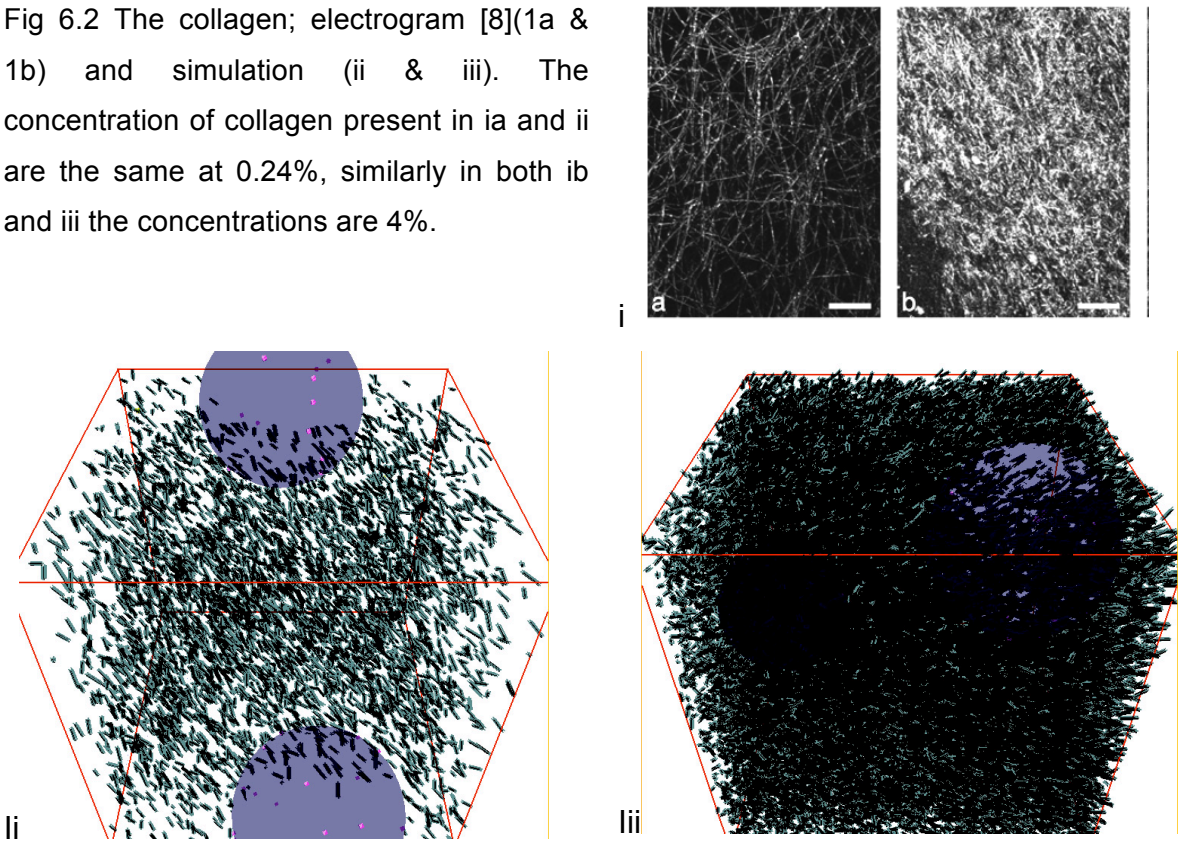


Note that the dark purple columns represent 4% collagen simulation, and the light blue represents 0% collagen in simulation.

6.2 Collagen Simulation

The success of the collagen simulation can be seen by eye. The structures produced by the simulator closely match the electrograms of the real thing. From fig 6.1 one can clearly see the correlation in density between the ia and ii diagrams, as well as the ib and iii diagrams. As this was the best real data that was come across about collagen, this seem an adequate test for the simulation model.

Fig 6.2 The collagen; electrogram [8](1a & 1b) and simulation (ii & iii). The concentration of collagen present in ia and ii are the same at 0.24%, similarly in both ib and iii the concentrations are 4%.



Overall the results of the experiments were encouraging, as they matched most of the recent research in the field. This leads to the conclusion that the model produced for the simulation of the movement of antibodies thought the ECM to a cancerous cell was sound.

7 Conclusions and Further Work

A model was developed that describes the interactions between the antibody and antigens in the environment of a tumour. This model was realised as a computer application that had a convenient user interface with changeable parameters. The behaviour of the model was validated against clinical data and demonstrated good qualitative agreement. The model sensitivity to model parameters was explored.

7.1 Further work

The scope for further work in this model is almost infinite and the model was built to incorporate that. Some features that could be added in the next release of the model would be to add more complexity of agents. For example, building up the complexity of the ECM by the addition of hyaluronin and even possibly charged particles to see what effect they have on the model.

More radically it had been noted [5] that tumour interstitial not a passive fibre-matrix gel, but a living material undergoing dynamic remodelling of shape. Consequently the transport parameters available change in time and space. There are techniques that enable measurement of this in vivo (multiphton in vivo microscopy), but it would be interesting if the modelling world can throw light on to the problem. The suggestion being to model the tumour ECM as a dynamic object with no static agents at all.

Acknowledgments

I would like to thank all of the people who have helped the research and implementation of this project, in particular my supervisor Chris Clack for his guidance and advice and Dr Sylvia Nagl and Amos Folarin for their expertise in the fields of oncology and biochemistry.

References

[1] Bert, J. L. & Pearce, R. H. (1990) in Handbook of Physiology: The Cardiovascular System, eds. Berne, R. M. & Sperelakis, N. (Am. Physiol. Soc., Bethesda, MD),. **4**: 521–547.

[2] Flynn AA, Pedley RB, Green AJ, Boxer GM, Boden R, Begent RHJ (2001) Optimizing radioimmunotherapy by matching dose distribution with tumor structure using 3D reconstructions of serial images. *Cancer Biotherapy & Badiopharmaceuticals,* **16** (5): 391-400

[3] Flynn AA, Green AJ, Pedley RB, Boxer GM, Dearling J, Watson R, Boden R, Begent RHJ (2002) A model-based approach for the optimization of radioimmunotherapy through antibody design and radionuclide selection. *Cancer,* **94** (4): 1249-1257 *Suppl. S*

[4] Netti PA, Berk DA, Swartz MA, Grodzinsky AJ, Jain RK (2000) Role of Extracellular Matrix Assembly in Interstitial Transport in Solid Tumours *Cancer Research,* **60** (9): 2497-2503

[5] Netti,P.A. & Jain, R.K. (2003) Interstitial Transport in solid tumours. In Cancer Modelling and Simulation, Chapman & Hall, 51-74.

[6] Parameswaran, S., Brown, L. V., Ibbott, G. S. and Lai-Fook, S. J. (1999) Microvasc. Res. 58, 114–127

[7] Pluen A, Boucher Y, Ramanujan S, McKee TD, Gohongi T, di Tomaso E, Brown EB, Izumi Y, Campbell RB, Berk DA, Jain RK (2001) Role of tumour–host interactions in interstitial diffusion of macromolecules: Cranial vs. subcutaneous tumours. *PNAS,* **98** (8): 4628-4633

[8] Ramanujan S, Pluen A, McKee TD, Brown EB, Boucher Y, Jain RK. (2002) Diffusion and convection in collagen gels: Implications for transport in the tumour interstitium . *Biophysical Journal,* **83** (3): 1650-1660

[9] Boucher, Y., Baxter, L. T. & Jain, R. K. (1990) Interstitial pressure gradients in tissue-isolated and subcutaneous tumours: Implications for therapy *Cancer Res.* **50**, 4478-4484

[10] Bentley, K. and Clack, C. (2004) The artificial cytoskeleton for lifetime adaptation of morphology. Proceedings 9th international conference on the Simulation and Synthesis of Living Systems (ALIFE IX), 13-16.

[11] Bentley, K. and Clack, C. (2005) Morphological plasticity: environmentally driven morphogenesis. *Lecture Notes in Computer Science* **3630**, 118-127.

[12] Rod Smallwood and Mike Holcombe, AGENT-BASED MODELLING IN BIOLOGY:FROM MOLECULES TO POPULATION

8 APPENDICES

A. User Manual

Installing .Net

Unfortunately the C# compiler and/or .Net environment needs to be installed onto your computer before the code will run. If you do not have it installed please go to the .Net folder and follow the instructions for installing .Net.

Running the application

The application is called DMA.exe(in the DMA/release/bin folder) and once the .Net environment is installed all that is required if a double click on the icon and the application should run.

The application itself consists of a screen with a menu bar at the top and a black canvas with the outlines of a cube on it. The menu function (left to right) are as follows:

Exit – exits the application

Generate Data – take you to a parameters screen, where the parameters for the simulation are set – Please be patient, some simulations take time the white screen of the application doesn't mean it has crashed.

Generate dynamic data – this function will generate data that in only dynamic and request the loading from file of a static

data. The simulation of both sorts of data will have to be done at least once.

Load Data – this load previously generated data files. There is an option to load a static data file, which will have a associated dynamic data file. Similarly you can load a dynamic data file which will have a static data file associated with it.

Re-Draw – will redraws the simulation from the beginning, using the currently loaded data

Rotate Picture – rotates the perspective of the cube. This can be done on the X axis, the Y-axis or both of them together. Reset brings the view back to the original view point.

Statistics – a screen that will showcase the basic stats of the run

What the animation means:

Brownish spheres – cancerous cells
Green cubes – antigens
White pink rods – Collagen fibres
Yellow cubes – moving antibodies, moving antibodies leave a trail behind them of their past coordinates, these are yellow dots. When an antibody hits the boundary of the cube(and therefore the simulation) its colour changes to *blue*. If it hit's an antigen, the antigen and antibody merge into one and the colour of them becomes *red*.
Turquoise points represent the start of the antibodies path.

B. System Manual

To extend the code one will need to have a C# editor, there are some that are available for free off the internet, or there is the Microsoft environment .Net which is not so free.

The system has been built in order to simplify any extensions that could be necessary.
The 3 main classes that would need to be changed when adding new agents would be Simulator, Canvas and Parameters (if the agents requires different parameter settings). The main class of the application frm_Main, this deals with all of the initialisation. The Simulator class deals with all of the aspects of simulation. The Canvas class deals with all of the rendering of the animation to screen.
The new agent should provide methods for all of its interactions, these include saving to and reading from file.

The wrapped for the openGL with C# is referenced with wglave5. To run on a c# compiler it must be specified that unsafe code is allowed.