PUC

ISSN 0103-9741

Monografias em Ciência da Computação
n° 17/08

An Agent-based Framework for Stem Cell Behavior Modeling and Simulation

Diêgo Bispo Conceição

Maíra Athanázio de Cerqueira Gatti

Carlos José Pereira de Lucena

Departamento de Informática

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO

RUA MARQUÊS DE SÃO VICENTE, 225 - CEP 22451-900

RIO DE JANEIRO - BRASIL

Monografias em Ciência da Computação, No. 17/08 ISSN: 0103-9741
Editor: Prof. Carlos José Pereira de Lucena April, 2008

An Agent-based Framework for Stem Cell Behavior Modeling and Simulation

Diêgo Bispo Conceição, Maíra Athanázio de Cerqueira Gatti, and Carlos José Pereira de Lucena

Departamento de Informática
Rua Marques de São Vicente 225, 4º andar RDC
Rio de Janeiro – RJ – Brasil

diego@les.inf.puc-rio.br, {mgatti, lucena}@inf.puc-rio.br

Abstract. This paper presents an agent-based framework for the modeling and simulation of stem cell behavior. It covers a simplified cell life cycle model and stem cell proliferation and differentiation process. We instantiated the framework for the neurons cell differentiation process. We believe that *in silico* stem cell therapy is an important step for *in vitro* stem cell therapy research, reducing costs and avoiding some ethical issues.

Keywords: Multi-agent Systems, Frameworks, Computational Biology, Stem Cell, Simulation.

Resumo. Este artigo apresenta um framework para a modelagem e simulação do comportamento de células-tronco. Ele cobre um modelo do ciclo de vida celular simplificado e dos processos de proliferação e diferenciação. Acredita-se que a terapia *in silico* de células tronco é um importante passo para a pesquisa de terapia *in vitro,* reduzindo custos e evitando questões éticas.

Palavras-chave: Sistemas Multiagentes, Frameworks, Biologia Computacional, Célula-tronco, Simulação.

In charge of publications

Rosane Teles Lins Castilho
Assessoria de Biblioteca, Documentação e Informação
PUC-Rio Departamento de Informática
Rua Marquês de São Vicente, 225 - Gávea
22453-900 Rio de Janeiro RJ Brasil
Tel. +55 21 3114-1516 Fax: +55 21 3114-1530
E-mail: bib-di@inf.puc-rio.br

I. Introduction

The systems biology community is building increasingly complex models and simulations of cells and other biological entities. This community is beginning to look at alternatives to traditional representations, such as those provided by ordinary differential equations (ODE). Stem cells play a prominent role in biology and life sciences. Their importance is growing more and more, not only in basic research fields such as cell or developmental biology, but also in medicine and clinical research. The main reason underlying this broad interest in stem cells is their capacity to reconstitute functional tissues after disturbance or injury. They are able to produce a huge number of differentiated, functional cells and, at the same time, they maintain or even re-establish their own population.

A stem cell is a primitive cell that can either self-renew (reproduce itself) or give rise to more specialized cell types. The new perspective on stem cell systems as networks of different cell types and their interactions implies that stem ness should not be treated as an explicit cellular property, but as the result of a dynamic self-organization process. The microenvironment interactions and their specific effects on proliferation and differentiation have to be embedded in the concept.

Nowadays, stem cells are cultivated in the lab in order to differentiate into a specific mature cell. Today it is hard to predict the stem cell behavior under some substances. Moreover, all the infrastructure necessary to maintain a stem cell culture is very expensive and many stem cells are wasted if the injected substance does not lead the culture to the desired mature cell. Thus, stem cell simulation is a powerful tool for reducing such costs and accelerating the stem cell therapy process.

There have been several attempts to build formal models (for instance, ordinary differential equations [5] and cellular automata [6] so that predictions can be made about how and why stem cells behave either individually or collectively. Moreover, the model has to consistently explain the broad variety of experimental observations. It is intended to link these macroscopic phenomena to underlying (latent) microscopic mechanisms. To include experimental observations, which describe individual cell behavior, into the analysis, the model must be able to describe single cell as well as cell population behavior.

With regard to the medicine point of view, if the model is sufficiently detailed and accurate, it serves as a reference, a guide for interpreting experimental results and a powerful means of suggesting new hypotheses. Moreover, the simulation lets physicians test experimentally unfeasible scenarios and can potentially reduce experimental costs and time (experiments in vitro can last weeks; the same experiments can take just a few minutes if they are done in silico). Besides, such a technique avoids ethical problems about the use of embryonic stem cells.

Models are simplifications, abstractions. They are a crucial means of relating theory to reality and there is wide agreement that we must at least dimension them to relate to that reality and the theory on which they are based. On the other hand, agent-based models (ABM) or multi-agent models (MASs) usually deal with objects that demonstrate some sort of dynamic behavior.

Usually agents or objects react either passively to their environment or to other agents, or actively to their environment or other agents. In this sense, agents usually move in spatial systems like cities, which means motion or movement. Usually, there are many agents — not one or two but n. When there are many agents, they react to each other through time and their collective behavior can be unpredictable, surprising, hence novel and emergent. In this way, this style of modeling is quite consistent with the sciences of complexity.

An AB simulation is a simulation with many intelligent agents interacting among themselves and with the environment. In a typical AB simulation of social behavior, the agents are the individuals that take rational decisions based on their neighbors' decisions. Very interesting social phenomena have been recently investigated, such as, for example, cooperation [7], social instability [8], and crowd modeling [9].

The great advantage of this modeling technique is that the emergent phenomena can be modeled through very simple rules governing the behavior of each agent. The global effect resulting from the interaction of the individuals is often unpredictable. More specifically, an agent is a high-level software abstraction that provides a convenient and powerful way to describe a complex software entity in terms of its behavior within a contextual computational environment [10]. Furthermore, the dynamic structures present in biological systems can be intuitively represented and efficiently implemented in agent-oriented simulators [11] [12].

However, in order to achieve reuse we need a general framework for the modeling and simulation of the intra-cellular processes and the stem cell proliferation and differentiation, i.e., a set of classes, their relationships and common behavior that can be reused and which speed up the development of different cell and different cell differentiation processes. This paper presents the first results achieved towards the construction of this agent-based stem cell general framework.

Outline

This paper is organized as follow: Section 2 describes the domain analysis (detailing the stem-cell self-organization, the processes, scope and the conceptual model developed). Section 3 highlights the related work. Section 4 presents the description of the framework developed with its hop spots and the instantiation. Section 5 describes the results for the framework instance. And finally, Section 6 concludes the paper and presents future work

II. The Stem-Cell Self-Organization Description

An important entity with an active influence in the process is the Niche. The Niche is a specialized cellular environment, which provides stem cells with the support needed for self-renewal, and contains the cells and proteins that constitute the extra cellular environment. The Niche has regulatory mechanisms in order to save stem cells from depletion and to protect the host from over-exuberant stem-cell proliferation [14]. The niche is stem cells' habitat, as the environment is the agents' habitat.

To ensure self-renewal and differentiation, stem cells undergo two types of cell division (Figure 1): **symmetric division** gives rise to two identical daughter cells, both endowed with stem cell properties; **asymmetric division**, on the other hand, produces only one stem cell, a progenitor cell and a precursor cell with limited self-renewal potential [15] [16] [18]. Progenitors can go through several rounds of cell division before terminally differentiating into a mature cell with no self-renewal potential [19].

II.i. The Cell Conceptual Model

In order to simplify the model we had to consider only the most active components during the cell life-cycle and the mitosis division (which is the stem cell division during the self renewal process). By active components we mean components that more directly contribute and influence the cellular differentiation during the cycle.

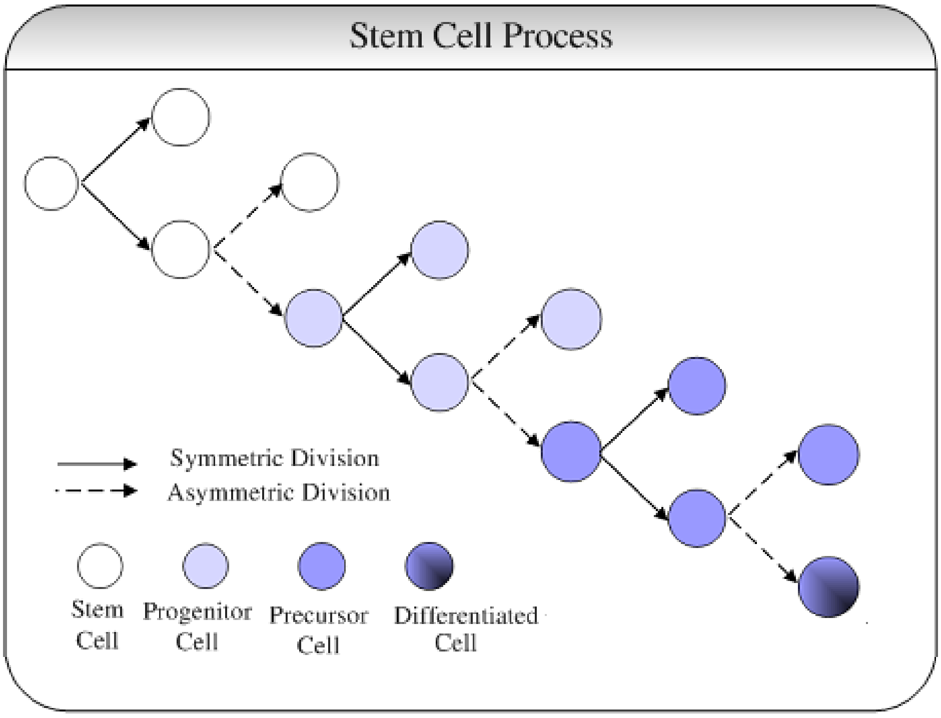

Fig. 1. Stem Cells Process Descprition

The Centrosome is a Microtubule Organizing Center (MTOC) that has a pair of Centrioles, and three kinds of Microtubules (Astral, Polar and Kinetochore). The microtubules are nucleated structures capable of growing and shrinking in order to generate force, segregate the chromosomes correctly during cell division and move organelles and cell structures to new locations.

Before the start of the Mitosis phase, the genetic material in the nucleus is in a loosely bundled coil called Chromatin. The Chromatin condenses together into the Chromosome. There are 46 Chromosomes, which condensate into two Chromatids during the Mitosis phase.

Each Chromatid has a Centromere, which in turn has a Kinetochore (which is the point where the kinetochore microtubules are attached to the chromosome).

Besides those elements, the cell modeled is composed of: cell membrane, nuclear membrane, nuclei, substances, protein, and organelles. During the cell life cycle, many proteins are synthesized, which determines if the cell may continue the division, or not.

Each cell can have one or more neighbor cells. The precursor cell and progenitor cell are cells that can proliferate. For instance, a neuron is a type of cell with no potentiality (i.e, not able to proliferate). The cells can go to 11 states; the state can be a phase of the mitosis or cell life cycle process or a dead state.

II.ii. The Cell Life Cycle Scope

During the cell life cycle, there is a mechanism called Checkpoint, which monitors the cell. For each phase, Gap 0, Gap 1, Synthesis, Gap 2, and so on, the Checkpoint checks if some events have been started or ended in order to prevent the cell from entering an undesirable state of error.

Considering the cell life cycle, the cell may stay in the G0 phase while it does not receive any signal for starting the cycle. The signal may be a protein, or any other substance. At the end of the cycle considering the mitosis division, there are two new cells in the G0 phase.

It is possible to change this course depending on which signals the cell receives. Basically, during the G1 phase, the cell receives a signal to: start division, start specialization in the case of being a stem cell, or start differentiation in the case of being a specialized cell.

Furthermore, any cell (stem cell, specialized or differentiated) may go into apoptosis. Basically, if any internal process leads the cell to a bad state, or if the niche is overpopulated, the cell kills itself. Table 1 details all the events and the checkpoint monitoring specified for each cell life cycle phase

Table 1. Cell Life-Cycle Phases, Events and Checkpoints

Phase	Events	Checkpoint
Gap 0	Start Cell Life-Cycle	Check if the Chromatins are ok.
Gap 1	Synthesize Cyclin D	Check if Cyclin D and E were synthesized into the Nuclei, i.e, if there are the cdk2-CycD and cdk2-CycA complex. Check if pRb protein is inactive and E2F is active after cyc D,E synthesis. Check if the cell is prepared to: divide or differentiate.
	Synthesize Cyclin E	
	Synthesize Substances	
	Increase Cell Metabolic Rate	
Synthesis	Synthesize Cyclin A	Check if Cyclin A was synthesized, i.e, if there is the cdk2-cycA complex in the nuclei. Check if there are Chromatins replicated, if the Centrosome was replicated and if the MFP is deactivated.
	Start Chromatin Replication	
	Replicate Centrosome	
Gap 2	Finish Chromatin Replication	Check if all the Chromatins were replicated, if Cyclin M was syn-

	Destroy pRb protein and E2F	thesized and if MPF is activated (cdk2-cycB).
	Synthesize Cyclin M (=B)	
	MPF is activated	
	Destroy cyclins: D, E, A	
Prophase	Condense Chromatin into Chromosomes	Check if all the Chromatins and their replicas were condensate into Chromosomes.
	Create Repulsive Forces (for Astral and Polar Microtubules)	
	Destroy cyclin B	
Prometaphase	Create Repulsive Forces (for Kinetochore Microtubules)	Check if all the Kinetochore Microtubules were attached to each Chromosome, and if the Nuclear Membrane was dissolved. Check if: APC inactive MCC = MAD2/BUB1/Cdc20
	Dissolve Nuclear Membrane	
	Attach Kinetochore To Microtubule	
	Attach MAD2 to BUB1	
Metaphase	Move Chromosome To Equatorial Plane	Check for each Chromosome its position in the cell. Check if APC is active.
	Detach MAD2 to BUB1 -> detach securing from separase-securin complex	
Anaphase	Separate Chromatids (through the separase)	Check if all the Chromatids were separated.
	Demount Mitotic Spindle	
Telophase	Shrinking Microtubules	Check if the Nuclear Membranes were created, and if all the Chromosomes were unfold back to Chromatin Check if all cdk are inactives.
	Create Nuclear Membranes	
	Start unfold Chromosomes Back To Chromatin	
Cytokinesis	Finish unfold Chromosomes Back To Chromatin	-

	Constrict Cell Membrane
	{
	Miosina RLC fosforilada
	Se cdk ativos:
	Inibe ATPase ->
	actina inativa ->
	nao separa
	Se cdk inativos
	ATPase -> quebra ATP ->
	Actina ativa -> contrai membrana celular.
	}	

	Create Abscission (division)	
	Create Nuclear Membranes	
	Reorganize and Disappear Non-Kinetochore Microtubules	

III. The Agent-Oriented Framework

In this section we describe the agent-oriented framework developed for stem cell behavior modeling and simulation. In cooperation with stem cell researchers, we developed the conceptual model of the entities and processes described in Section 3 and considered it as first-order abstraction that simplifies the intracellular processes and stem cell behavior.

We used some AUML Diagrams [2] to help us in the designing of some structures and dynamics, such as the agent use case diagram, agent class diagram and agent interaction diagram. We developed our framework on top of MASON, which is a fast discrete-event multi-agent simulation library core in Java, designed to be the foundation for large custom-purpose Java simulations, and also to provide more than enough functionality for many lightweight simulation needs. MASON contains both a model library and an optional suite of visualization tools in 2D and 3D [3].

Subsections 4.1, 4.2 and 4.3 present the main framework diagrams and the frozen spots, while subsection 4.4 describes the hot spots and how to instantiate the framework. As already mentioned, we instantiated the framework to be able to generate mature neuron cells.

III.i. Use Case Diagram

Figure 2, placed at the end of this paper in order to save space, shows the AUML agent use case. The actors are agents. The agents are the types of cell, the proteins, DNA and complex. The cells can be a proliferative cell or non-proliferative cell. The stem, progenitor and precursor cells are proliferative, while the neuron cells are non-proliferative. CDK, Reitinoic Acid, Cyclin and LIF are proteins.

The Cdk-Cyc Complex is the join between a CDK and a Cyclin protein. The cells can die or start a cell cycle phase. Each cell cycle phase may only be stated if some events occurred. The DNA which is an adaptive agent interacts with the proteins during the molecular pathways regulation.

III.ii. Class Diagram

Figure 3, also placed at the end of this paper, shows the framework's classes diagram. This diagram does not show all attributes and methods in order to be comprehensive regarding its structure. The classes inside the stipple line are Mason's classes [3]. The others classes are classes of the framework.

The framework uses four design patterns [4]. The pattern Prototype is used to implement the stem cell's substances because this facilitates the objects' clone process. This pattern is represented by the *PrototypeSubstances* class. The pattern State is used to implement the states the cell can be in. The *State* class represents this pattern. The pattern Strategy is used to implement the cell's asymmetric division strategy. Each type of cell has one distinct asymmetric division strategy and the class that represents this pattern is *AssimetricDivisionStrategy*. The singleton is used to implement the niche, because the niche is unique. The classes that represent this pattern are *NicheImpl* and *GuiNiche*.

III.iii. How to instantiate the framework

In this section we describe the first framework's instance developed for the neuron cells differentiation process. There are other hot spots in the framework as the proteins and so on, however we just describe the most important ones in this paper in order to have a better idea of how instantiating it.

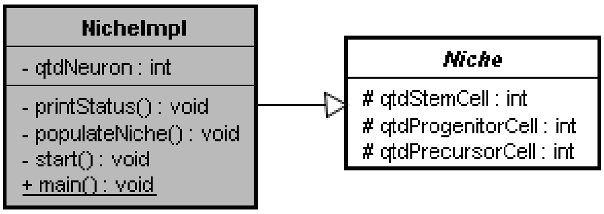

Fig. 4 - Niche Hot Spot

The first hot spot that must be implemented is the environment where the cells represented by the agents live, called a niche, as already mentioned. Figure 4 is shown in partial view to save space. The *Neuron* class extended the Cell class and it was necessary to implement the method *step()*. Each agent (cells and proteins) must implement this method because it is handled by the simulator. Besides the *main(), start() and populateNiche()* methods, for instance, it is necessary to implement the *regulateNiche()* method, which must contain the environment's constraints according to the niche to be simulated.

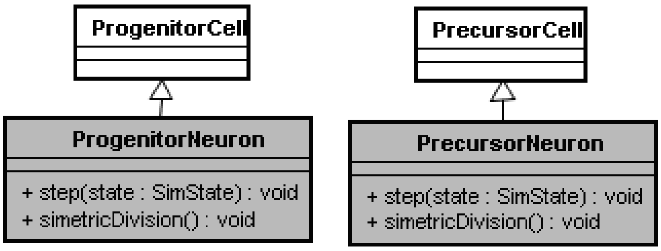
ProgenitorCell
PrecursorCell
ProgenitorNeuron
+ step(state : SimState) : void
+ simetricDivision() : void
PrecursorNeuron
+ step(state : SimState) : void
+ simetricDivision() : void

Fig. 5. Progenitor and Precursor Hot Spots

Other important hot spots are the progenitor and precursor cells, which must be instantiated depending on the process for a specific type of cell to be simulated. It is necessary to add features and behaviors depending on the type of cells to be instantiated (figure 5).

The *ProgenitorNeuron* class extended the *ProgenitorCell* class and the *PrecursorNeuron* class extended the *PrecursorCell* class. The methods *step* and *simetricDivision* needed to be implemented for both classes extended.

The *Division2ProgenitorNeuron*, *Division2PrecursorNeuron* and *Division2Neuron* classes are those, respectively, responsible for implementing the strategy for asymmetric divisions from stem cells, progenitor neurons and precursor neurons cells and they instantiate the *AssimetricDivisionStrategy* class. The abstract *assimetricDivison()* method needed to be implemented for these classes.

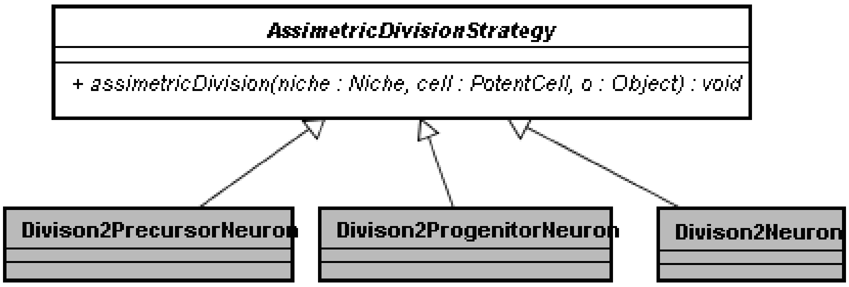
AssimetricDivisionStrategy
+ assimetricDivision(niche : Niche, cell : PotentCell, o : Object) : void
Divison2PrecursorNeuron
Divison2ProgenitorNeuron
Divison2Neuron

Fig. 6. Asymmetric Division Hot Spot

IV. The results

In order to execute the instantiation for the generation of neuron cells,a computer with the following features was used: Intel(R) Core (TM) 2 CPU 6300@ 1.86 GHz, 2GB of RAM.

Figure 6 shows the result of the simulation with the visualization module. In 4 minutes there were 153 stem cells, 661 progenitor cells, 1,539 precursor cells and 462 neurons running in the simulator.

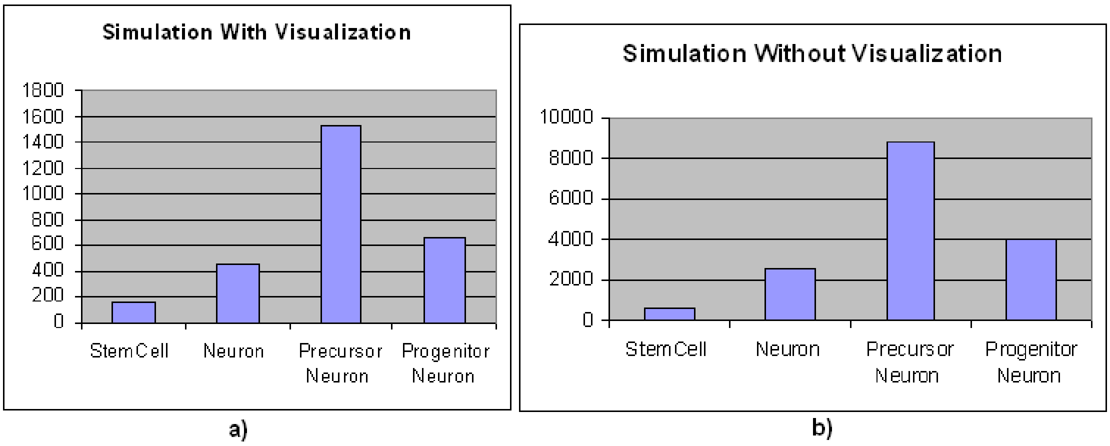

Fig. 6. Number of cells living in the niche a) with the visualization layer, and b) without the visualization layer

V. Related work

In order to support the claim that the agent approach is more suitable than other modeling approaches, existing approaches have been taken and recast in the agent-based modeling and simulation framework, which has demonstrated a number of clear advantages of the agent approach over existing approaches [12].

To start with, mathematical models [6] [5] do not allow expressing partial information about a system, i.e. to formally describe open systems, as an open MAS does.

Moreover, depending on the system's complexity, there would be an explosion of differential equations to model it, with more than 50 equations to model a subsystem, for example. Another drawback is the absence of an abstraction for the models. Physicians have to deeply understand mathematical methods in order to model the system, while MAS can provide the right level of abstraction for that.

Concerning the conceptual comparison, MASs are not just probabilistic dependent as the Monte Carlo methods [20]. More than reproducing the emergent behavior, agent-based simulation can provide advanced mechanisms, such as learning and adaptation that, as far as we know, is not possible to implement through Monte Carlo simulation. Those mechanisms not only make the model more complete but also allow the optimization of self-organization, for instance.

To date [21], [11], [12] have produced formal and mutually consistent specifications of the leading of some predictive models of stem cell behavior within their agent framework. They have also produced simulations and visualizations of these models. In their approach, each stem cell is implemented as an agent. They modeled and simulated stem cells in a dynamic environment with division and determined capabilities (stem cells which have reached their cycle phase and which are surrounded by stem cells become determined). However, the stem cell behavior modeled was too simple, unfeasible and not adequate or evolvable to an adequate model for the physicians' experiments.

Gatti et al. [1] developed a simplied stem cell conceptual based-agent model and a stem cell behavior simulator based on a multi-agent system. It was possible visualize the emergent behavior by means of a 2D visualization component. This prototype also provided a 2D cell visualization in a higher level of detail(macro level) by clicking over the cell. This system did not use any framework, middleware or agent-base platform. The proposed solution overloaded the CPU even with small numbers of entities. The visualization froze up many times as a result.

VI. Conclusions and future work

The stem cell researchers' collaborators were very excited with the first results. Basically, they observed in the visualization of the neuron instance the first emergent phenomenon that is similar to the emergent phenomenon *in vitro*: the differentiated cells are located at the colony's extremity while the specialized and stem cells are located at the colony's center. The great advantage of this modeling technique is that the emergent phenomena can be modeled through very simple rules governing the behavior of each agent, we can reuse all the modeled already developed and also can model learning, adaptive behavior and open systems.

The number of cells running together and the time of execution in the simulation were satisfactory. Of course we need to increase this number and to achieve this goal we are distributing the framework and application in grid architecture with ten processors. If we have around 20,000 cells we can reach more refined self-organizing mechanisms that might occurs in those kid of systems.

Moreover, the next steps are a review of a set of tests on the proposed model, intending to adjust it for the demand of the biomedical domain. Other activities are to produce reports about Macro and Micro levels to support the comprehension of the information visualized during the simulation process, and identify with the specialists of the domain the trustworthiness of the results achieved through this first instance, in order to allow an adaptation and bias adequate to the reality of the research of the domain. Study of each protein and substances and how these relate must be carried out and reproduced in this application.

Also, as a future work, we want to optimize the emerging behavior generated by the self-organizing stem-cell represented by agents. The goal will be to define macros properties and, starting from local interactions, to integrate specialized online search planner [22] to optimize the behavior so that the macros properties can be satisfied. Hence the simulator will allow more specialists' interactions through the simulation environment with more reliability regarding the in vitro process, and also increasing the tool usability and dependability, as well as facilitating the validation process.

ACKNOWLEDGMENTS. We wish to acknowledge the team of collaborators in this project, entitled LANDIC, which is the Neurogenesis and Cell Differentiation Lab, included the stem cell researcher Stevens Rehen. The team helped us in the conceptual model validation, with some discussions around the research hypothesis, and will still work with us in the future works.

References

[1] Gatti, M., de Vasconcellos, J.E., Lucena, C.J.P.; An Agent Oriented Software Engineering Approach for the Adult Stem-Cell Modeling, Simulation and Visualization. Workshop SEAS, João Pessoa, 2007.

[2] Gatti, M., von Staa, A., Lucena, C.; *AUML-BP: A Basic Agent Oriented Software Development Process Model Using AUML*; Monografias em Ciência da Computação, Departamento de Informática, PUC-Rio, No. 21/07, 25 pg., 2007.

[3] Luke, S., Cioffi-Revilla, C., Panait, L. and Sullivan, K. (2004), 'MASON: A New Multi-Agent Simulation Toolkit', SwarmFest 2004, Eighth Annual Swarm Users/Researchers Conference, University of Michigan, Ann Arbor, Michigan USA.

[4] Gamma, E., Helm, R., Johnson, R., Vlissides, J. "Design Patterns: Elements of Reusable Object-Oriented Software". Reading, MA: Addison Wesley, 1995.

[5] Lei, J., Mackey, M.C. (2007). "Stochastic differential delay equation, moment stability, and application to hematopoietic stem cell regulation system," SIAM J. Appl. Math. (2007), 67(2), 387–407.

[6] Agur, Z., Daniel, Y. and Ginosar, Y. (2002). The universal properties of stem cells as pinpointed by a simple discrete model. Mathematical Biology, 44:79–86, 2002.

[7] Axelrod, R. (1997) The Complexity of Cooperation (Princeton Univ. Press, Princeton).

[8] Epstein, E. M. (2002) Proc. Natl. Acad. Sci. USA 99, 7243–7250

[9] Helbing, D., Farkas, I. & Vicsek, T. (2000) Nature 407, 487–490.

[10] Jennings, N., R. (2000). On Agent-Based Software Engineering. Artificial Intelligence Journal, 117 (2) 277-296, 2000.

[11] d'Inverno, M. and Prophet, J. (2004). Modelling, simulation and visualisation of adult stem cells. In P. Gonzalez, E. Merelli, and A. Omicini, editors, Fourth International Workshop on Network Tools and Applications, NETTAB, pages 105–116, 2004.

[12] d'Inverno, M. and Saunders, R. (2005). Agent-based modeling of stem cell organisation in a niche. In Engineering Self-Organising Systems, volume 3464 of LNAI. Springer, 2005.

[13] d'Inverno, M., Theise, N. D. and Prophet, J. (2005). Mathematical modeling of stem cells: a complexity primer for the stem cell biologist. In C. Potten, J. Watson, R. Clarke, and A. Renehan, ed., Tissue Stem Cells: Biology and Applications. Marcel Dekker.

[14] Scadden, D. T. (2006). The stem-cell niche as an entity of action. Nature 441, 1075-1079 (29 June 2006) | doi:10.1038/nature04957; Published online 28 June 2006.

[15] Vogel, H., Niewisch, H., and Matioli, G.; J. Theor. Biol., 22(2):249-270, 1969.

[16] Loeffler, M. and Grossmann, B.; J. Theor. Biol., 150(2):175-191, 1991.

[17] Loeffler, M. and Roeder, I.. Cells Tissues Organs, 171(1):8-26, 2002.

[18] Lord, B.I.; in Stem cells, Cambridge Academic Press, 1997, pp 401-422.

[19] Wichmann, H. E. and Loeffler, M.; Mathematical modeling of cell proliferation: Stem cell regulation in hemopoiesis, CRC Press, Boca Raton, 1985.

[20] Metropolis, N. and Ulam, S. (1949). The Monte Carlo Method. J. Amer. Stat. Assoc. 44, 335-341, 1949.

[21] Theise, N. D. and d'Inverno, M. (2003). Understanding cell lineages as complex adaptive systems. Blood, Cells, Molecules and Diseases, 32:17–20, 2003.

[22] Weiss, G.; Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence. (ed). MIT Press, 2001. Chapter 4, pp. 165-199.

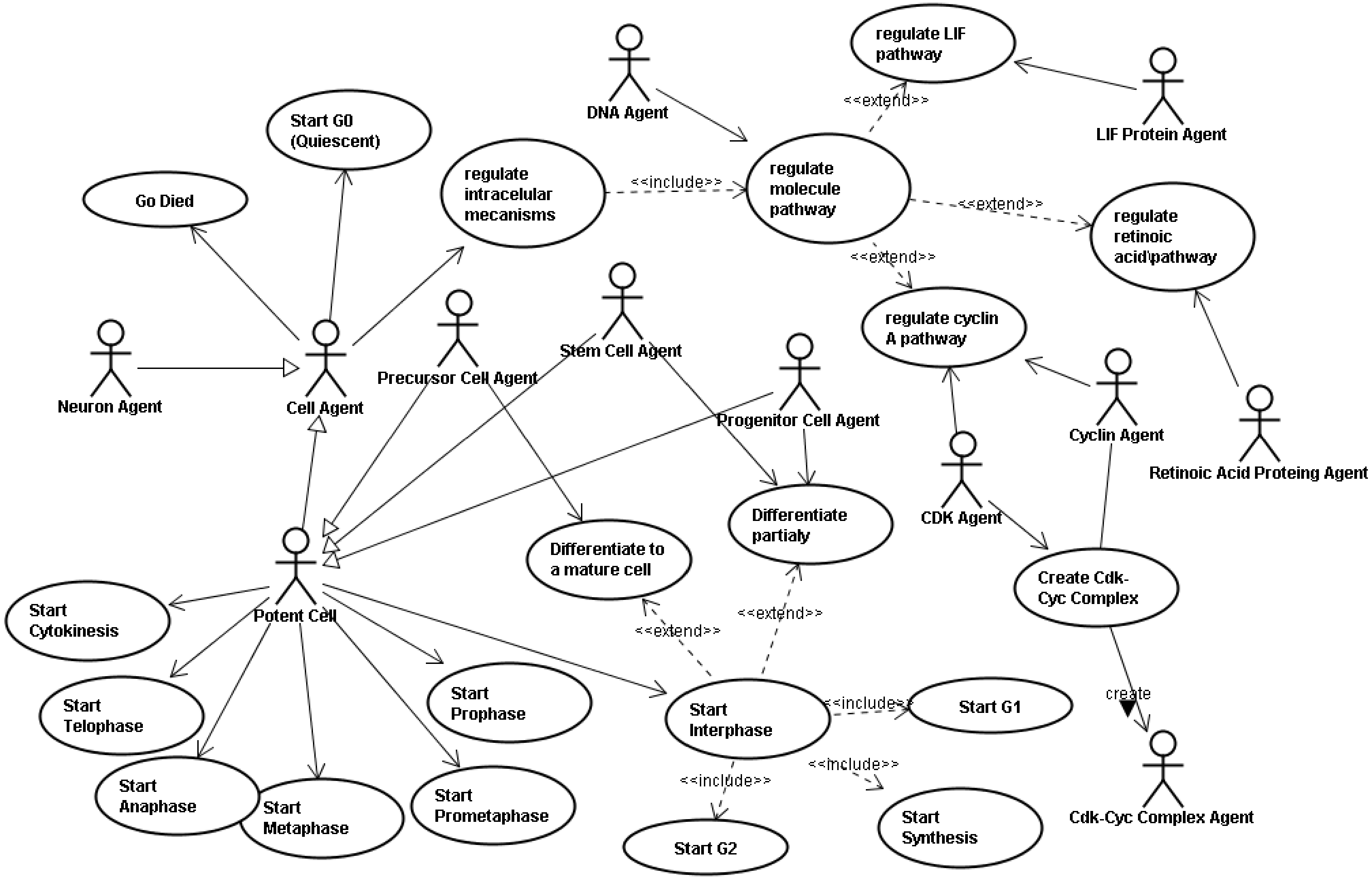

Figure 2 - AUML Use Case Diagram for the Stem Cell Framework

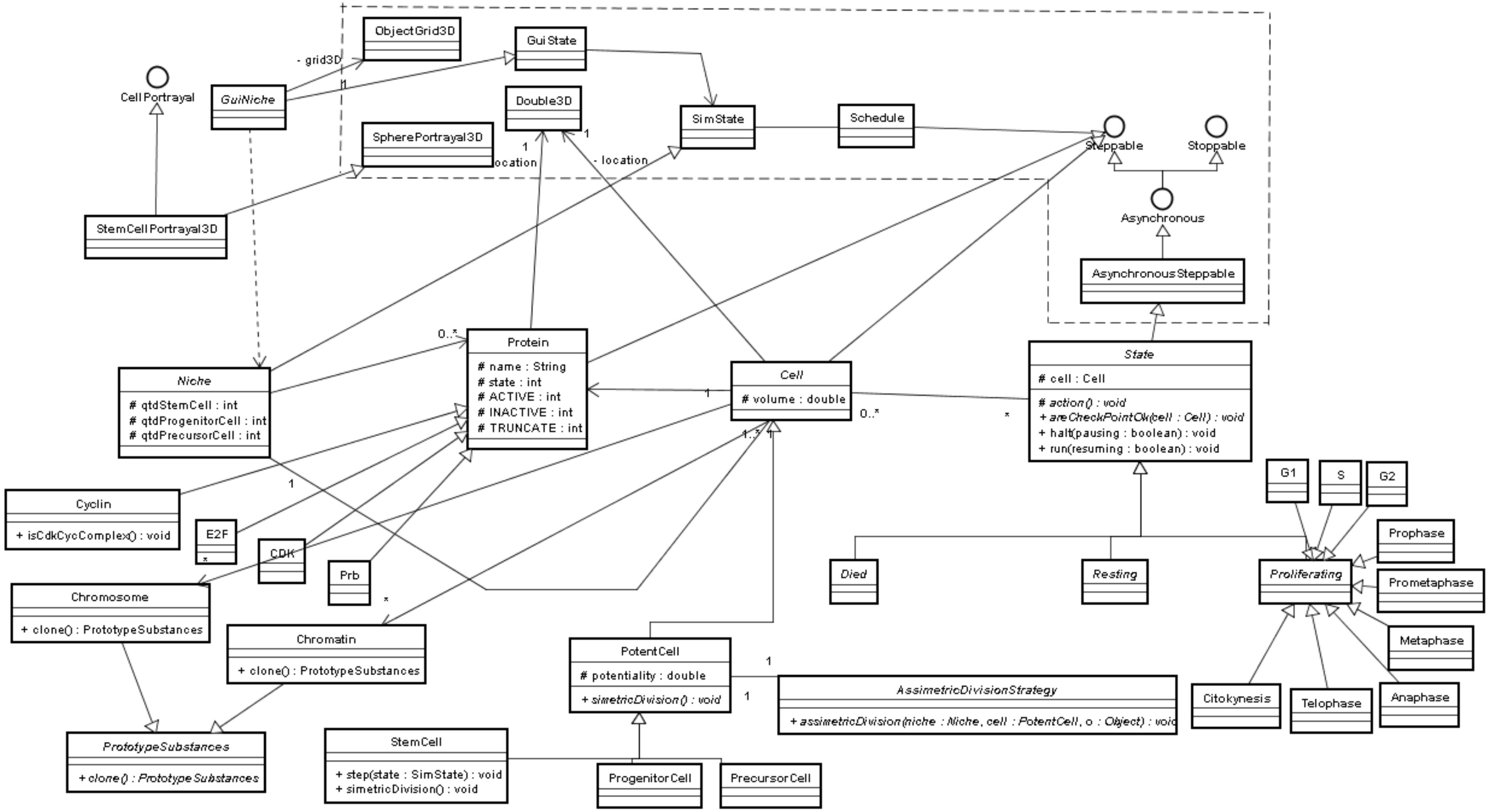

Figure 3 - Framework Diagram Class (partial view)