Support for Agent Based Simulation of Biomolecular Systems

Harika Korukonda and Carla Purdy
School of Electronic and Computing Systems, University of Cincinnati
contact: carla.purdy@uc.edu

Abstract

We describe our work developing GenericSystem, which contains the basic classes and functions necessary to model and simulate a typical biomolecular system. GenericSystem is user-friendly and extensible. It consists of well-defined functions which can be readily customized to reduce development time for a biomolecular simulation. Five systems-- bioluminescence in *Vibrio fischeri* bacteria, skin regulatory system, phage lambda in *E. coli*, epithelial cells growth cycle and Wnt signaling pathway--have been successfully implemented using the GenericSystem tool. We show some results for one of these systems and describe the process to translate a typical differential equation-based system description into a GenericSystem model.

Introduction

For molecular systems, traditional modeling techniques are known as equation based modeling (EBM) [2] and include ordinary differential Equations (ODE), partial differential equations (PDE) [3], stochastic differential equations (SDE), Petri nets [4] and Pi-calculus [5]. The main disadvantage of these approaches is their inability to consider the spatial dynamics and heterogeneity of the system. Petri nets and Pi-calculus are graph based techniques. Graph-based techniques have several additional shortcomings such as basic modeling constructs which are quite primitive and hence they fail to model complex systems successfully. Also spatial representation of the system can get very complex with respect to time and effort. Another main disadvantage is the inefficiency of representing priorities or ordering of events which is essential in systems modeling. Because of these shortcomings, graph-based modeling techniques are not well-developed and hence rarely used. The other class of models comprising of ODE, PDE and SDE form the equation based modeling class of techniques. This gamut of approaches essentially represents the system by identifying the system variables. These variables are integrated and sets of equations relating these variables are formed. Evaluation of these equations forms the basis of EBM. The fact that this approach has been in use for several decades essentially showcases its ability to model a system satisfactorily. But as the complexity of the system increases, this approach starts to fail. This happens mainly because the equations involved become too complex to handle. A clear comparison between agent based modeling and the equation based approach is given in [2].

The disadvantages of modeling using traditional techniques include: the spatial dynamics of the systems cannot be modeled; systems with both continuous and discrete behavior cannot be modeled; the high complexity and stochasticity of the system cannot be taken into account; and most of these methods tend to aggregate the values during modeling, which may lead to incorrect results [14]. ABM, on the other hand, has many advantages: ABM describes a system in a way which is closest to it in reality [1]; randomness is applied in the most appropriate way instead of just adding a noise term to an equation; ABM captures emergence phenomena of the system which are the result of interactions between individual entities of the system; and ABM is flexible, i.e., it provides a natural framework for tuning the complexity of the agents. The behavior, degree of rationality, ability to learn and evolve and rules of interactions are adjustable [1]; the levels of agglomeration can be varied. i.e., dealing with single agents and groups of agents simultaneously becomes easy; the interactions can be changed dynamically, since they are defined at the agent level; and positive and negative feedback can be modeled. For systems in which activities describe the system better than processes and in which stochasticity applies to an agent's behavior, ABM is often the most appropriate way of modeling [1]. It can also be applied to problems where the population is heterogeneous or the topology of the interactions is heterogeneous and complex. There are several situations when ABM is the only resort. When the behavior of individual entities of a system cannot be clearly defined through aggregate transition rates, ABM is especially useful. As the individual behavior grows in complexity, the complexity of differential equations modeling them also grows exponentially and thus becomes unmanageable. ABM has no such overhead and has proved successful in modeling several complex systems [7].

Project Goals

The goals of this project were:

- To design GenericSystem, a generic easy-to-use simulation model using the agent-based modeling technique which can efficiently model many of the commonly found biological systems.
- To implement GenericSystem using MASON [9,10] and make the right use of the advantages available in the tool. MASON was chosen as the basis for our system after a thorough analysis of the available tools. A summary of many of the tools we considered is available in [8].

- To incorporate as many features as possible into the generic system so that it can successfully be used to model systems with entities of various complex shapes.
- To provide a procedure for transforming a biomolecular system modeled traditionally into an ABM version using our tool
- To provide case studies of specific system models, including examples previously developed individually in our lab (bioluminescence in *Vibrio fischeri* [12], skin cell regulation (normal and wound conditions) [13] and phage-lambda in *E. coli* acting as a biological inverter [11]).
- To provide an example of translating a differential equations-based simulation to an ABM simulation in GenericSystem, based on the Wnt signaling pathway [16,17].

.

GenericSystem

GenericSystem was designed using AUML, an agent based extension of UML [15]. There are three main classes of agents, Stationary, Mobile, and Vibrating. Users can extend these classes or add new classes which are derived from the base class Agent. Initial subclasses included in the system are Rectangular Sheet. Rectangular Box, Spherical, Cylindrical, Sticky Rectangular Sheet (Stationary), Sphere, Dumbbell, Rectangular Box, Rectangular Sheet (Mobile), and Rectangular Box, Rectangular Sheet, U- Shaped (Vibrating). Figure 1 shows an AUML diagram for a GenericSystem class. Notice how the diagram facilitates modeling communication between the agent and its environment.

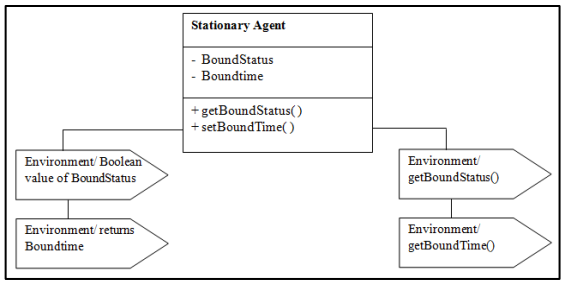


Figure 1. AUML diagram of Agent class.

Experimental Setup and Base Parameters of Simulations

The computer used to run the experiments has the 1.6GHz Intel Core 2 Duo processor. The operating system used is a 32bit Windows 7 OS. The RAM installed in the computer is 2GB.

The software versions used in the work are:

- MASON version 14
- Java ™ SDK 6 update 11
- Java 3D version 1.5.1
- Eclipse platform version 3.4.1 (IDE for Java)

Model design parameters are:

- Size : Generic unit which can be interpreted as the user wishes. It can be specified by using the scale function of display class.
- Velocity: Can be interpreted as generic unit of time or generic unit of length as per user's convenience.
- Container : The large rectangular box which containes all the simulations elements. It can be interpreted as the entire simulation space where the reactions are taking place. It is assumed that all the reactions take place within the container.

For all the systems to be implemented, the chemical reactions and their reaction rates have been found from the literature. The lifetimes and binding times of the molecules are calculated from the respective reaction rates. In [6] a relation between rate constants and the reactions times has been established. The inverse of the rate constant is considered as the reaction time measure. This is because any two given reactions with the same initial concentration of reactants proceed with velocities that have the same ratio as their reaction rate constant ratio. If v1 and v2 are reaction velocities of reactions with rate constants K1 and K2, then the relation between them defined in [6] is

$$v1/K1 = v2/K2..$$

Building a GenericSystem Model

We illustrate the use of GenericSystem through the model of the Wnt signaling pathway and some simulations of its behavior. The Wnt xignaling pathway describes a set of proteins most commonly known for their effect on embryogenesis and cancer tumors. A protein called β-catenin acts as a transcriptional coactivator for cancer causing tumor cells. Other important proteins which form part of the Wnt signaling pathway are APC and Axin, which is required for degradation of β-catenin. The reactions taking place in the Wnt pathway are shown in Figure 2.

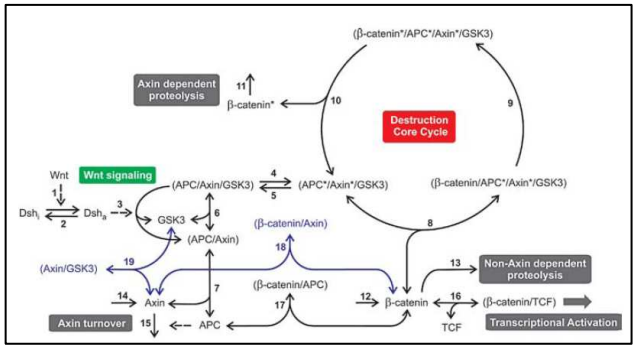
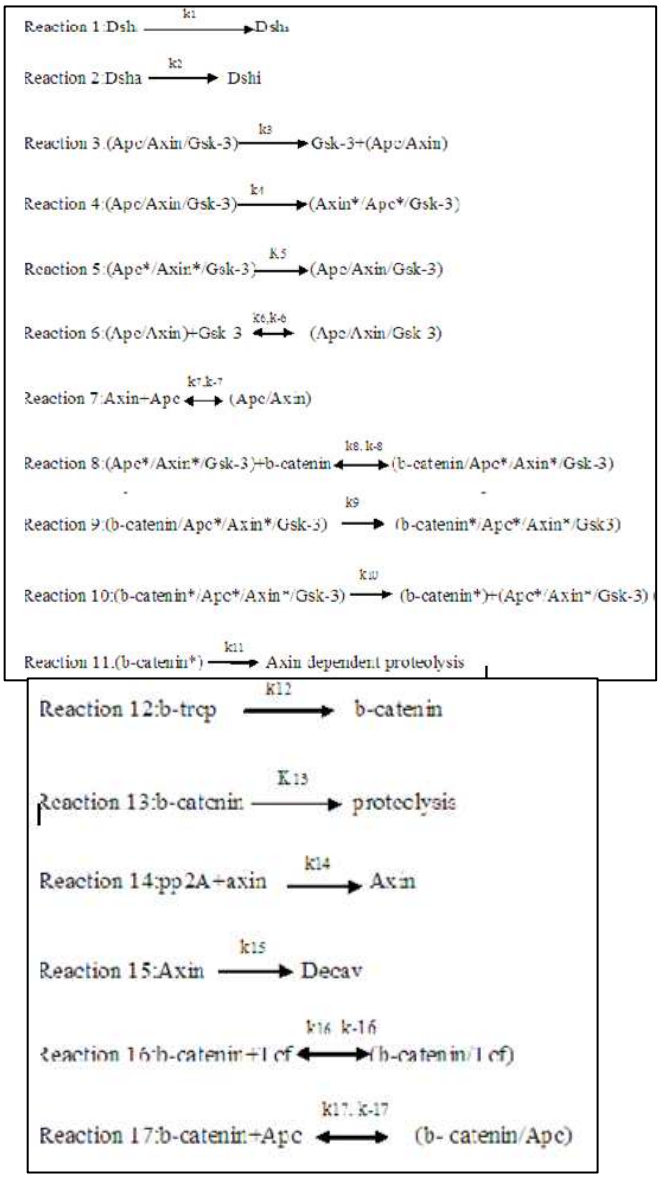


Figure 2. Reactions in Wnt pathway [16,17].

This forms a transcriptional regulation of Wnt genes. When Wnt signal is absent, APC directly associates with TCF/LEF binding site on Wnt target genes and mediates exchange between coactivator and corepresser complex proteins. This represses concentration of β-catenin. Also

when Wnt signal is absent, APC transports β-catenin from the nucleus to the destruction complex where it phosphorylates and is recognized by β-TrCP. This also results in further degradation of β-catenin protein. On the other hand, when Wnt signal is present, the phosphorylation of β-catenin is inhibited, leading to its dissociation from the Axin-assembled destruction complex [18]. The stabilized β-catenin reaches the nucleus and binds to the TCF/LEF resulting in activation of Wnt target genes. The chemical reactions taking place in the system are shown in Figure 3 and Figure 4. A differential equation based model of the Wnt pathway was previously developed in our lab [19]. Here we use the GenericSystem procedure to translate that model into an ABM model.



Figures 3 and 4. Wnt equations.

Now we categorize the molecules to match agent types available in the GenericSystem. The agents chosen to implement the system molecules are provided in Table 1.

Molecule	Agent
Dshi	Mobile Spherical Agent
Dsha	Mobile Spherical Agent
apc*/axin*/gsk3	Mobile Spherical Agent
apc/axin/gsk3	Mobile Spherical Agent
Gsk3	Mobile Spherical Agent
Apc/axin	Mobile Spherical Agent
Apc	Mobile Spherical Agent
β-catenin/apc*	Mobile Spherical Agent
β-catenin */apc*/axin*/gsk3	Mobile Spherical Agent
β-catenin *	Mobile Spherical Agent
β-catenin	Mobile Spherical Agent
Axin	Mobile Spherical Agent
Tcf	Stationary Rectangular Box
β-catenin/tcf	Mobile Spherical Agent
β-catenin/apc	Mobile Spherical Agent
TCF/LEF Binding site	Stationary Rectangular Box

Table 1. Wnt molecules and corresponding agents.

The chemical reactions are interpreted as collisions between the corresponding molecules. The rate of reaction is modeled as the velocity of corresponding molecules. The functions used are listed in Table 2.

Chemical Reaction	Function of GenericSystem
Bonding to another molecule	Attach()
Unbond from another molecule	Detach()
Grow in size	GrowInSize()
Decay	DeleteAgent()
React with another agent	DetectCollision()
Move freely	Move()
React and form new molecule	DeleteAgent(), CreateAgent()

Table 2. Agent functions and chemical reactions.

The number of molecules and their lifetimes are based on the concentrations given in [16]. The rate constants Ki are given in Figure 5.

Rate	K1	K2	K3	K4	K5	K6	k7
value	0.182	0.0182	0.05	0.267	0.133	0.0909	0.0909
Rate	K8	K9	K10	K11	K12	K13	k14
Value	1000	12000	0.01	0.5	206	206	0.417
Rate	K15	K16	K17				
Value	0.423	0.000257	0.0000822				

Figure 5. Rate constants for Wnt equations.

The concentrations of molecules in this experiment are spread over a large range starting from 0.00049 to 100. Hence directly relating the concentration to the number of molecules is not possible in this case. So the number of molecules are selected according to their concentration levels. Taking the number of molecules to the limiting number, i.e., the maximum allowed by the tool, is not advisable since the movement of the molecules is hindered and this reduces the rate of reactions. Hence the maximum number is taken to be 100 and accordingly other molecule numbers are selected. Molecules which are very low in concentration are assigned number 1. A series of experiments are run with various combinations of numbers of molecules. The numbers of molecules in the base case and their lifetimes are given in Table 3 and Table 4.

Molecule	Number
Dshi	100
Dsha	0
apc*/axin*/gsk 3	5
apc/axin/gsk3	3
Gsk3	1
Apc/axin	3
Apc	100
β-catenin/apc*	1
β-catenin */apc*/axin*/g	1
β-catenin *	1
β-catenin	20
Axin	2

Table 3. Numbers of molecules.

The lifetimes of the molecules that decay are calculated by taking the rate of decay and relating it the total simulation time. Table 4 tabulates the lifetimes of the molecules.

Molecule	Lifetime
Dshi	1000
Dsha	10000
Axin	500
β-catenin*	300
β-catenin	100

Table 4. Molecule lifetimes.

Snapshots of simulation

The system was successfully modeled using GenericSystem. Snapshots of the simulation are shown in Figurea 6, 7, 8, and 9.

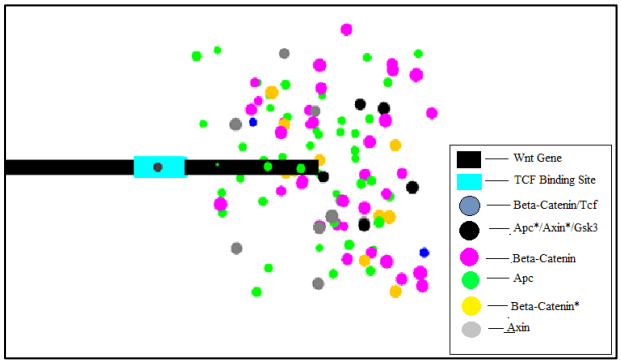


Figure 6. System at time step 500.

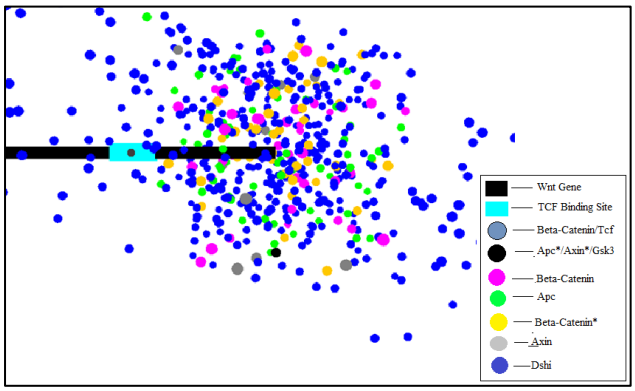


Figure 7. System at time step 900.

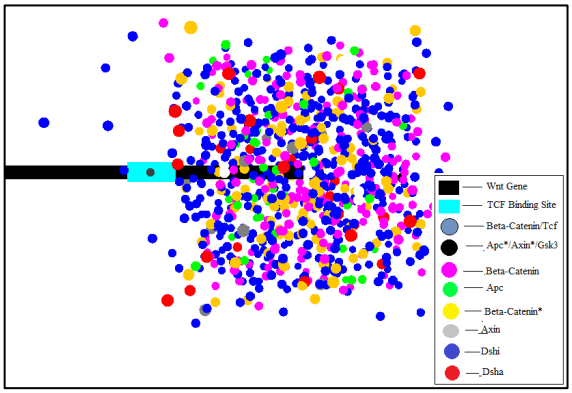


Figure 8. System at time step 1100.

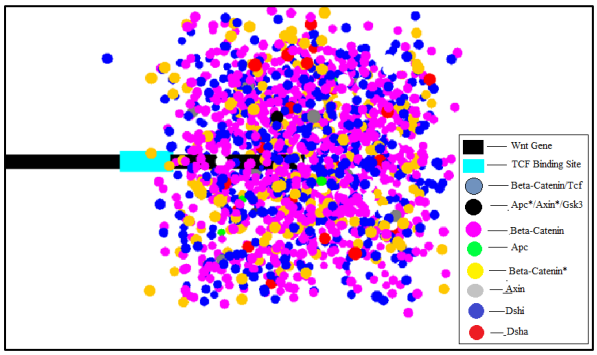


Figure 9. System at time stop 7000.

Description of Results and Comparison to Literature

The numbers of the molecules β-catenin and Apc*/Axin*/gsk3 have been observed throughout the simulation. As Wnt changes from 0 to 1, the number of β-catenin molecules increases from 20 to 500. The number of Apc*/Axin*/gsk3 molecules decreases from 5 to 1. These results are plotted in Figures 10 and 11

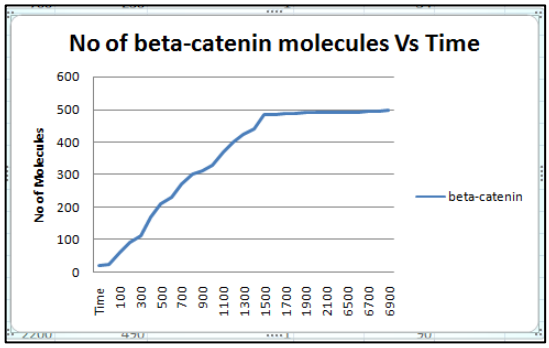


Figure 10. β-catenin molecules vs time.

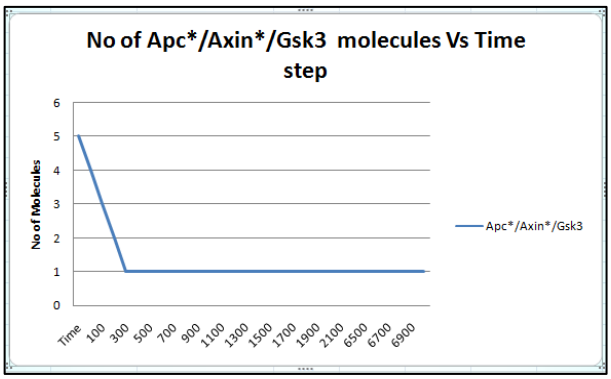


Figure 11. Apc*/Axin*/gsk3 molecules vs time.

The results obtained by the differential equations method in [19] are shown in Figures 12 and 13. We see that in this case the ABM simulation results match well with the results in [19].

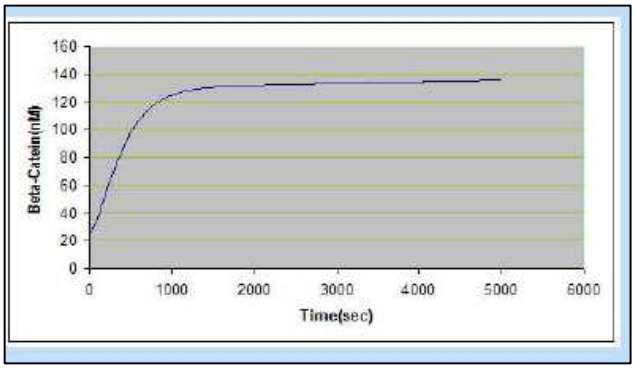


Figure 12. Concentration of β-catenin vs time [19].

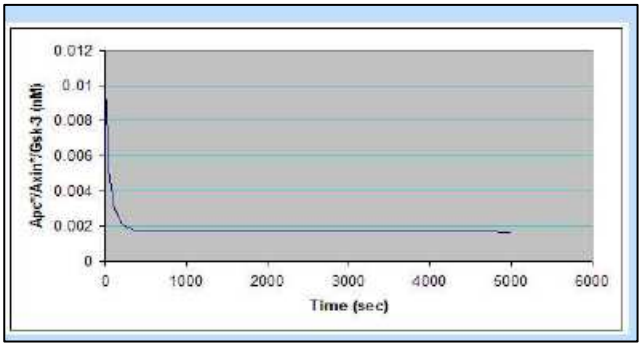


Figure 13. Concentration of Apc*/Axin*/Gsk3 vs time [19].

Conclusions and Future Work

Here we have compared our ABM model with a differential equations model. In the case we considered, we see that qualitatively we are getting the same behavior. Similar results were obtained for the other systems mentioned above. Details about these systems can be found in [21]. The agent based model and stochastic

modeling were compared by Karkutla in [20]. Karkutla also showed that ABM can be used to simulate nonhomogeneous systems, which cannot be simulated accurately by either stochastic or differential equations, and he demonstrated that for cases where both ABM and stochastic simulation can be applied, the results also compare well quantitatively.

GenericSystem can use the graphical display feature of MASON to produce animations of the models under consideration. Using this feature, we are working on developing realistic animations of a version of the skin cell example. We are also working on extending the system to model more complex dynamic behavior, for example DNA self-assembly and nanotube growth. A tool that could simulate such phenomena in a cost-effective way would be very useful in supporting virtual experiments involving novel materials for future generation computer elements. Accurate modeling of fine-grained dynamic behavior will require additional computational resources. Thus another question we are studying is how to accurately characterize the relative costs of ABM simulation versus stochastic or differential equation simulations. ABM methods are most effective for fine-grained behavior and low concentrations of molecules. A method to quantify this statement fpr a given example would be very useful.

References

1. E. Bonabeau, Agent-based modeling: Methods and techniques for simulating human systems, *Proceedings of the National Academy of Sciences*, vol. 99, May 2002, pp. 7280-7287.
2. H. Van Dyke Parunak, Robert Savit, and Rick L. Riolo, Agent-based modeling vs. equation-based modeling: a case study and users' guide, *Proceedings of the First International Workshop on Multi-Agent Systems and Agent-Based Simulation*, July 1998 , pp. 10-25.
3. James W. Haefner, *Modeling Biological Systems: Principles And Applications*, Springer, 1996, pp. 129-132
4. Mor Peleg, Daniel Rubin, and Russ B. Altman, Using Petri net tools to study properties and dynamics of biological systems, *Journal of American Medical Information Association*, 12(2), Mar–Apr 2005, pp. 181–199.
5. A. Regev, W. Silverman, and E. Shapiro, Representation and simulation of biochemical processes using the pi-calculus process algebra, *Pacific Symposium on Biocomputing*, 6, 2001, pp. 459–470.
6. S. Khan , R. Makkena , F. McGeary , K. Decker , W. Gillis and C. Schmidt, A multi-agent system for the quantitative simulation of biological networks, *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, Melbourne, Australia, July 14-18, 2003.
7. http://www.dcs.shef.ac.uk/~rod /Integrative_Systems_Biology.html. Accessed 06/02/2009
8. I. Politopoulos, *Review and Analysis Of Agent-Based Models In Biology*, 2007.
9. http://cs.gmu.edu/~eclab/projects/mason/ Accessed 05/10/2009.
10. S. Luke, C. Cioffi-Revilla, L. Panait, and K. Sullivan MASON: a new multi-agent simulation toolkit, *SwarmFest Workshop*, 2004
11. V. Vallurupalli and C.Purdy, "Agent based modeling and simulation of biomolecular reactions", *Scalable Computing: Practice and Experience (SCPE)*, 8(2), 2006, pp. 185-196.
12. Krupa Sagar Mylavarapu , Agent based model of bioluminescence in vibrio fischeri, Master's Thesis, University of Cincinnati, Ohio, Oct 2008.
13. Balakumar Rajendran, 3D agent based model of cell growth, Master's Thesis, University of Cincinnati, Ohio, Dec 2008
14. Paul Davidsson, Agent based social simulation: a computer science view, *Journal of Artificial Societies and Social Simulation* 5 (1), 2002.
15. B. Bauer, J. P. Muller, and J. Odell, Agent UML: A formalism for specifying multiagent interaction, *Agent-Oriented Software Engineering*, Paolo Ciancarini and Michael Wooldridge ed., Springer-Verlag, Berlin, 2001, pp. 91-103.
16. E. Lee, A. Salic, R. Kruger, R. Heinrich, and M.W. Kirschner, The roles of APC and Axin derived from experimental and theoretical analysis of the Wnt pathway, *PLoS Biol 1(1)*, 2003, pp. 116-132.
17. Wnt Signaling pathway, http://en.wikipedia.org/wiki /Wnt_signalling_pathway, accessed 10/15/2010.
18. Yue Xiong and Yojiro Kotake, No exit strategy? no problem: APC inhibits β-catenin inside the nucleus, *Genes Dev. 20*, March 15, 2006, pp. 637-642
19. Sravanthi Mailavaram, Database for the study of biological pathways, with Wnt signaling pathway use case, Master's Thesis, University of Cincinnati, Ohio, Nov 2008.
20. Raja Karkutla, Agent based and stochastic simulations for non-homogeneous systems, Master's Thesis, University of Cincinnati, Ohio, March 2010.
21. Harika Korukonda , A generic agent based modeling tool for simulating biomolecular systems, Master's Thesis, University of Cincinnati, Ohio, November 2010.