High Performance Computing for Tumor Propagation Agent-based Model

Ghazal Tashakor, Emilio Luque, Remo Suppi

Universitat Autònoma de Barcelona,
Department Computer Architecture & Operating Systems,
School of Engineering, Campus Bellaterra,
Cerdanyola del Vallès, Barcelona, 08193, Spain
Ghazal.Tashakor@caos.uab.cat, { Emilio.Luque, Remo.Suppi }@uab.cat

Abstract. Agent based modeling (ABM) and High Performance Computing (HPC) techniques are very popular in investigation and understanding cellular and molecular systems. The complex nature of these systems and the demand for emulation and comprehension at different levels in these models creates the expectation for new effective simulation strategies and tools. The present paper peruses the foresaid demands and the approaches for developing simulation in tumor model and its interactions using ABM and HPC. ABM allows the analysis of the actions and interactions of autonomous agents (cells in this case) to evaluate their effects on the system as a whole in order to re-create and predict the appearance of a complex phenomenon. This is a parametric model and it is necessary to explore the data model space to determine which combinations of adjustments cause the behaviors which are of interest. In this case, HPC is a useful tool to perform experiments in acceptable time.
Keywords: Agent Based Models (ABM), High Performance Simulation, Hpcnetlogo, HPC.

1 Introduction

Presumptive and dynamic modeling techniques in different simulators to generate complex biological models are promising research tools in oncology, but the need for computational techniques for large scale parameters and factors in complex biological models has not yet been addressed. Here in this paper, we perused a simulation model which has been created based on the agent-based system in Netlogo by Wilensky to emulate the development and progression of tumor and metastasis [1].

This model demonstrates the growth of a tumor and how it prevents chemical treatment. The tumor mass is recognized as a self-organized and self-regulated model of complexity and contains not only inherent properties of tumor cells (e.g., stem cells, growth, and migration) but also its interactions with both the immune and the vascular system. All types of self-organized immune systems rely on immune cells' capability to produce signaling molecules (e.g., cytokines, antibodies) in reaction to the environmental conflicts, which consequently are recognized and lead to the

dynamic changes of their factors and actions [2]. Besides, the blood vessel development, or angiogenesis, is controlled by the multiple molecules including vascular endothelial growth factor and endothelins. The tumor cells discharge both cytokines and angiogenic factors which have strong influence on both the local immune system and angiogenic responses of the immune and vascular systems [3].

The sum of the interactions between the tumor, the immune and the vascular local systems determines both the success and the rate of tumor progression. The complexity of these interactions during tumor progression creates more complex interactions that can be modeled and simulated using agent-based systems (ABS) to provide a dynamic representation of the tumor development to extract the emergent information and knowledge of the system.

Cancer research is generating a large amount of knowledge using information provided by the genome. This knowledge can be used, with the support of the computational techniques, in addition to those described in the laboratory and clinics that can be complex and slow, to create parametric models based on principles/rules at the cell level that allow us to understand the emergent behavior of the tumor growth/propagation at system level [4, 8].

The high-grade tumor (World Health Organization grade III & IV) expresses a cellular hierarchy with the self-renewing called tumorigenic cancer stem cells (CSCs) which is an attractive model to explain many aspects of tumor behavior and its different growth conditions[8].

Most researches and observation of the cancer stem cell growth has been focused on the discovery of mutations and variations of tumor suppressor genes with recessive loss of the function. Consequently, the design of the computational models, especially multi-agent models and using high performance computing resources and techniques, will allow the increase of the knowledge on this subject and will accelerate the research and facilitate the clinical experiments. Although it is a promising technique, it is very important to consider that the computational models must be validated to represent the real system and this may be a complex task, but once this step is done, it can be used to understand how CSCs emerge.

This paper is organized as follows; Section 2 describes the research about general tumor and cancer models. Section 3 presents the main concepts about the extended model used to demonstrate the possibilities of ABM and HPC. In section 4, our experimental environment, named HPCNetlogo, is described which is designed to facilitate the scientists doing the HPC experimentation without the complexity of high performance environment. Section 5 shows the results of the experimentation from HPC point of view (speedup, efficiency and scalability of the model) for different parametric executions to explore some parts of the possibilities of the model data space. Finally, section 6 presents the conclusion and open lines of this work.

2 State of the Art

There is a large set of references where their authors have used probabilistic or mathematical models to simulate the growth of active cancer cells using different

phases and cell cycle durations. These models can be classified as oriented toward the vasculature without considering the tumor mass or others that determine the expansion of the tumor in the presence of non-evolutionary angiogenesis. [9]

However, models have evolved and the tumor mass can be considered as a self-organized and self-regulated system that includes, in addition to the properties of the tumor cells (stem cells, growth, migration, etc.), interactions with the system and angiogenesis. This model, which is considered very complex, considers the self-organization of the immune system (to analyze the immune cells that produce signaling molecules) or the evolution of angiogenesis and how they produce dynamic changes in the system. [6, 8, 10-12]

Agent based modeling (ABM) is a discrete-based modeling that offers many possibilities and advantages over other methods for studying and simulating cancer development [6]. ABM allows modelers to study the largest influences on tumor behavior by changing factors and parameters and generating repetitions to obtain data stability (statistically). There are several types of ABM techniques which have been used in cancer research such as lattice-based/free/gas, cellular potts, and sub-cellular elements modelling methods [2, 6].

Wilensky's tumor model [1] in the NetLogo library is used as the base for the present case study. This model is designed as a self-organized ABM model that illustrates the growth of a tumor and how it resists chemical treatment. In this case the model consists of two kinds of cells: stem cells and transitory cells and includes controls to kill transitory cells that are younger, to kill a stem cell or moving cells or visual information about the total number of living cells or their trajectory.

Considering the complexity of interactions between tumor cells and vascularization during tumor progression, the ABM model provides a more dynamic representation of the tumor development. However, since the number of parameters and agents required to give a realistic approach increases, it is necessary to use high-performance computing to make an exploration of the model data space in the acceptable time. In addition, since the model includes stochastic variables, it is necessary to perform a large set of simulations to obtain statistically acceptable values which the HPC needs.

3 ABM Tumor Model

The design of the agent-based tumor model of the present work is based on the Wilensky model [1] and considers the premises stated by Hallmarks of Cancer [5].

In this model two kinds of cells are considered: stem cells and transitory cells. Each cell is represented by an agent and tumor cells can breed, move, or die where the time is measured in simulation steps (ticks). Initial tumor cells are blue and the cells change their color depending on their age (different colors from red palette). A stem cell can divide either asymmetrically or symmetrically during the mitosis. In this case study, we considered malignant tumors where a symmetric mitosis is followed. In this process the stem cell divides symmetrically in two stem cells and the first child of this division remains static, but the second cell moves to distant organs to generate a metastasis. In this model, we considered only metastasis (and not invasion) as the

mechanism of the spreading cancer cells, reaching it to an organ different from the original tumor using blood or lymphatic paths. In the model, the metastasis is red and made of cells that die young. As the tumor propagate and get larger, stem cells reproduce and die younger.

Moreover, the model includes some considerations about hallmarks listed in [4] with the acquired capabilities of cancer:

1. Self-sufficiency in growth signals
2. Insensitivity to antigrowth signals
3. Evading apoptosis
4. Enabling replicative immortality
5. Sustained angiogenesis
6. Cells invasion and metastasis

In order to represent this behavior, the model includes parameters for modeling the number of initial stem cells, the growth factor in order to model the self-sufficiency of growth signals and insensibility to anti-growth signals, three values to represent the apoptosis grade (normal, low, very low) and a replication factor in order to model the replicative immortality and angiogenesis capacity of the tumor.

The simulated model represents, with different parameters, the original stem cells, the transitory cells, and the metastasis process. The visualization of the tumor grows exponentially and the metastasis process is showed as new tumors are allocated outside the original location, evolving in an independent location. Figure 1 shows the simulation evolution of 6 original stem cells (center) with a random position metastasis in new independent tumors with growth-factor=1.25, replication-factor=high and apoptosis=normal. In this case, the evolution from the 6 stem cells reaches, in steady state, a range of 23,000 to 29,000 cells (agents).

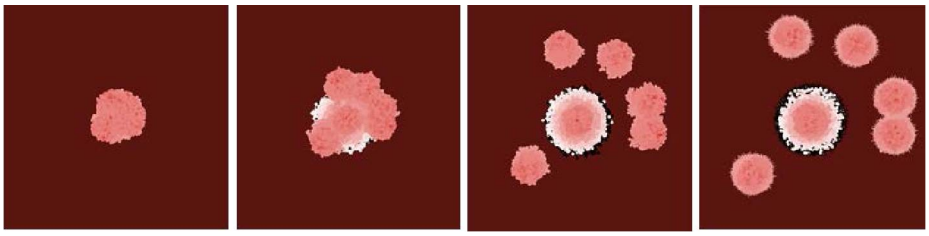

Fig. 1. Stem cells evolution and metastasis visualization (from 6 original stem cells to 23,000-29,000 in a steady state).

Figure 2 shows the steady state of a tumor metastasis visualization with 6 stem cells and the growth-factor=1.75, replication-factor=high, and apoptosis=low. As can be seen, the growth of metastasis is more aggressive and through reducing apoptosis, there is a greater number of cells that do not die, amounting to near 200,000 cells (agents).

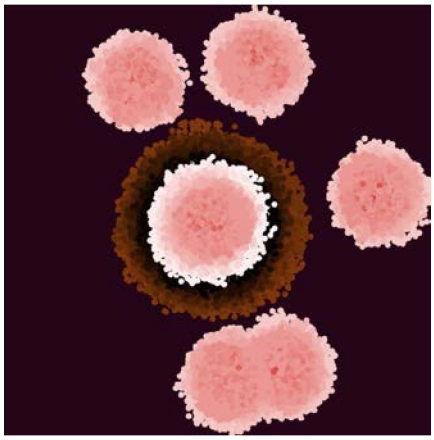

Fig. 2. 6 stem cells evolution and metastasis visualization with grow-factor=1.75, apoptosis=low and replication-factor=high (near 200,000 cells in steady state).

4 High Performance Approaches for Tumor ABM Model

In agent based model, each cell is represented as an agent with rules, parameters and interactions with the environment to reproduce the emergent growth behavior of the tumor and metastasis. In this model, all parameters, rules and interactions are parametrized and it is easy to adjust a rule or a specific constant for describing different tumor behaviors. The main problem of this type of simulation is the higher number of parameter combinations that is necessary to explore the data space of the model.

Netlogo [1] includes the Behavior Space tool that allows the exploration of the model data space using parametric executions in varying settings of the model and for recording the results of each model run. This process (*parameter sweeping*) permits us to explore the model's space of possible behaviors and determine which combinations of settings cause the behaviors of interest. The main problem of these executions is that the Behavioral Space only supports multithreading, so its performance is limited to the number of cores/threads at the local infrastructure.

The solution for this problem is to execute the parametric simulations using a HPC cluster in order to reduce the necessary time to explore a determinate model data space. This is an interesting idea but, considering HPC cluster complexity, it can be complicated for scientists to run these models in an easy way within an HPC cluster.

In order to provide an easy to use and user-friendly solution for non-technical users, we have developed a workflow, named HpcNetlogo [13] to deploy the simulation of ABM models on a HPC cluster. The execution of this workflow is unassisted and the initial user interaction is through an interface based on the web technology. In the local environment, the user configures his model, performs the

local tests, sets up the Behavior Space and, through a service, schedule his model, and indicates where and how he wants it to be run.

There is a Frontend service that analyzes the user-specified model and generates all configuration files required for the proposed simulation scenarios, creates the initialization files for the high-performance cluster queue system and then sends them to Backend. The backend executes the experiment and when it finishes, the user can see the result through the same web interface. HpcNetlogo is open source (GPLV3) and can be adapted for different cluster/queue environments (see more information in [13]). Figure 3 shows the diagram of the architecture and its web interface.

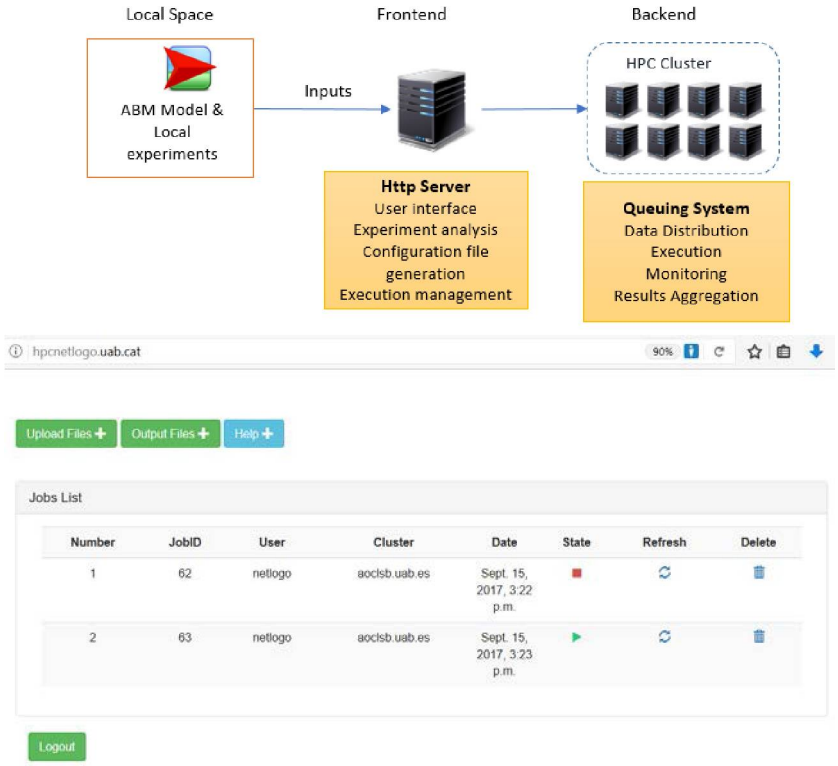

Fig. 3. HPCNetlogo Architecture and Web Interface

5 Results and Discussion

With the objective of analyzing the behavior of the developed model on an HPC cluster, a set of simulations with different parameters and values have been performed

to explore, from the point of view of performance, the behavior of the model (speedup, efficiency and scalability).

These experiments are the proof of concept considering HPC performance where the scalability, number of agents and performance of the model are analyzed but not the simulation result as growth factor, apoptosis, angiogenesis or other interesting values for an oncologist or researcher in tumor propagation (however these are available as simulation results).

The figure 4 shows the speed-up and efficiency for these executions. In this figure, the speed-up is close to lineal in the lower number of cores and gets reduced towards the higher number of cores. Although the results of speedup and efficiency (about 80%) seems good, but the main issue is the memory contention problems at the Java runtime (Netlogo) for higher core configurations and the I/O from network file system (NFS).

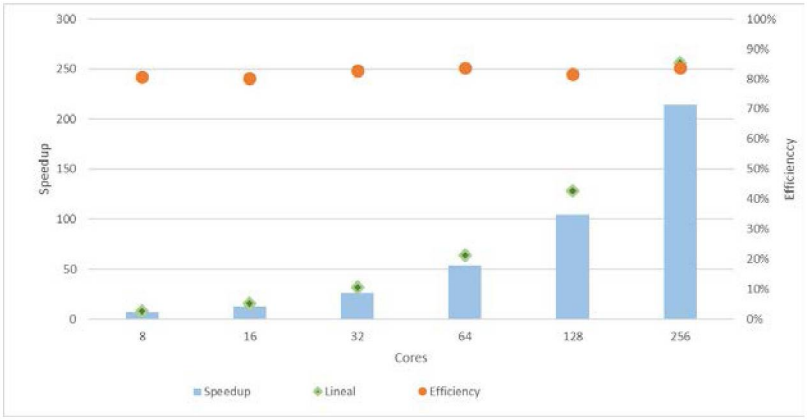

Fig. 4. Speedup and Efficiency for 8 to 256 Cores

Figure 5 shows how scalability effects of the model over the speedup and efficiency considering a grow-factor equals to 2.0 and the speedup in compare with the values of grow-factor equals to 1.25. This growth-factor, in this case study, implies that for the value of 1.25 there are about to 13,000 agents and for the value of 2, 00 there are about to 110,000 agents. As can be observed, the speedup and efficiency are affected by the grow-factor equals to 2.00 but it keeps on showing the values performance.

Figure 6 shows the impact of the shared resources at the Java runtime (NFS, memory) during the model execution for the larger number of agents: 30 initial stem cells with a growth-factor equals to 2.00 that arrives to 361,000 agents (cells). In this case, the figure show the distribution of the execution time in a box and whisker plot. As can be observed, the model is executed in a fixed number of time units (1,000) at the eight cores and runs about 2,234 seconds (average) but on 64 cores it will be run about 4,417 seconds (average). In order to solve this problem, it is necessary to change technology (java runtime) and evolve towards high performance distributed simulation environments as Care HPS [14].

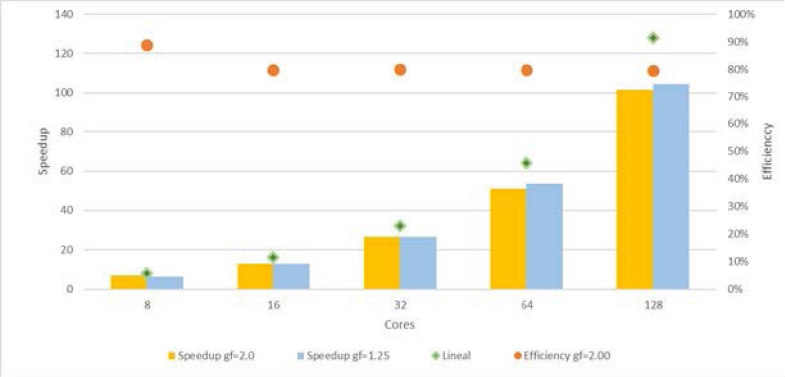

Fig. 5. Speedup and Efficiency for 13,000 and 110,000 Agents (cells)

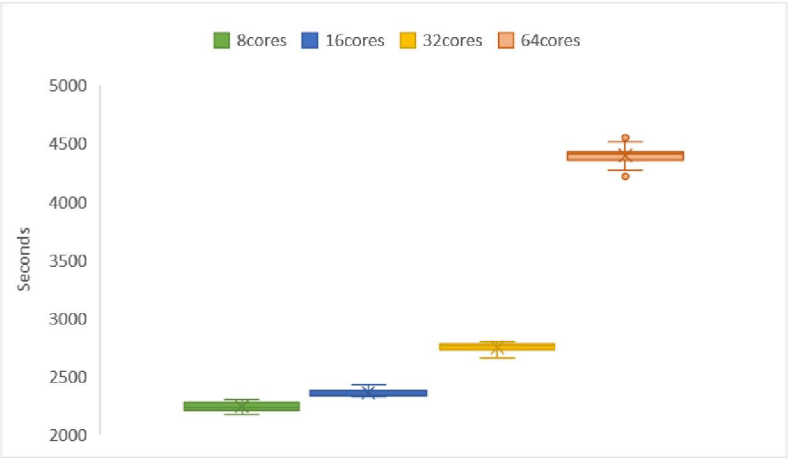

Fig. 6. Box Diagram of Execution Time for 361,000 Agents

Figure 7 shows the scalability of the model for different number of stem cells (from 6 to 60) and to values of grow-factor (1.25 and 1.75). As can be observed the scalability of the model is lineal. In this experiment, the simulation shows that in the steady state (1.000 simulation steps) for grow-factor of 1.25 and 6 initial stem cells the tumor arrived 6.6k agents (cells) and for 60 initial stem cells the tumor arrived to 122k agents (cells). For growth-factor of 1.75 and 6 initial stem cells, the number of agents (cells) was 76k and for 60 initial stem cells the number of agents (cells) augments to 404k.

It is necessary to remark that employing HPC is extremely necessary because this type of analysis is time consuming as can be seen. At these proofs of concepts only the minimum number of repetitions to obtain stable statistics data which have been executed and only 1000 simulation steps have been performed. It is important to

consider that in other configurations/parameters may be necessary to increase the number of simulation steps to justify the necessity of employing HPC for this type of experimentation.

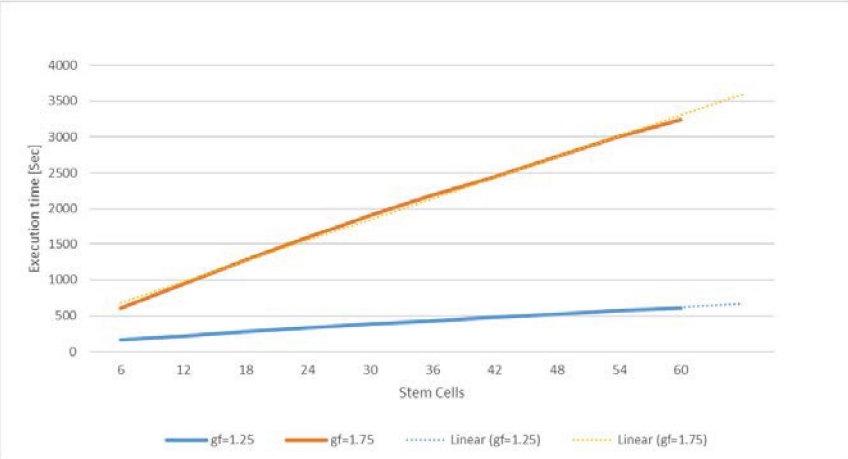

Fig. 7. Simulation runtime (average) for growth factors 1.25, 1.75 and from 6 to 60 stem cells

6 Conclusion

In this paper, we introduced a new tumor propagation model based on Wilensky model and considering the premises stated by Hallmarks of Cancer and its analysis using HPC cluster. For this experimentation, we used an easy-to-use HPC frontend, developed by the authors and named HpcNetlogo, that is an environment that allows the scientist to execute Netlogo model (in this case study) in an HPC cluster using a web interface and the Behavior Space tool included in Netlogo.

As it could be observed, the experiment shows good values for the speed-up and efficiency. However the drawback imposed by the Java Virtual Machine and access to shared resource is bigger when a high number of agents are used. These limitations could be addressed using distributed (non-parametric) simulation environments such as Care HPS [14] or Repast HPC [15] to reduce the impact of the shared resources access and to execute it with a large number of agents.

The future researches will be oriented toward:

- Detailed validation of the tumor propagation model: it is necessary to compare the results of the proposed model with the real data in order to validate the grow factor, replication rate and apoptosis factor in order to adjust this values to real ones.
- Implementation of tumor propagation model in Care HPS in order to analyze the performance values using a distributed simulation environment and its scalability for higher number of agents (cells).

– Time limitations prohibited more extensive tests to show performance & scalability value for different combinations of the main parameters. It is necessary to explore a bigger data space of the model with gradual increase in agent creation in order to analyze the impact of these in the simulation execution time.

Acknowledgments. This research has been supported by the MINECO Spain under contract TIN2014-53172-P.

7 References

1. Wilensky,U.: NetLogo Tumor model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. Project Integrated Simulation & Model Environment. National Science Foundation REC-9814682 and REC-0126227 (1998).
2. Dréau, Didier, et al.: An agent-based model of solid tumor progression. Bioinformatics and Computational Biology. LNCS, vol. 5462, pp. 187--198. Springer, Heidelberg (2009)
3. Knowles, J., Loizidou, M., Taylor, I.: Endothelin-1 and angiogenesis in cancer. Current Vascular Pharmacology, vol. 3, pp. 309--314. Bentham Science (2005)
4. Hanahan, D., Weinberg, R.: Hallmarks of Cancer: The Next Generation. Cell, vol. 4(5), pp. 646--674. Elsevier (2011)
5. Loeb, L.: Mutator phenotype may be required for multistep carcinogenesis. Cancer Research, vol 51(12), pp. 3073--3079. AACR (1991)
6. Wang, Z. et al.: Simulating Cancer Growth with Multiscale Agent-Based Modeling. Seminars in cancer biology, vol 30, pp. 70--8. Elsevier (2015)
7. Pérez-Rodríguez, G., et al.: High performance computing for three-dimensional Agent based molecular models. Journal of Molecular Graphics and Modelling, vol. 68 pp. 68—77. Elsevier (2016)
8. Lathia, J. D., et al.: Direct *In Vivo* evidence for tumor propagation by glioblastoma cancer stem cells. PLoS ONE, vol 6(9), e24807. PLOS (2011)
9. Kohandel, M., Kardar, M., Milosevic, M., Sivaloganathan, S.: Dynamics of tumor growth and combination of anti-angiogenic and cytotoxic therapies. Physics in Medicine & Biology, vol 52(13). IOP Publishing (2007)
10. De Pillis, L.G., Gua, W., Radunskayab A.E.: Mixed immunotherapy and chemotherapy of tumors: modeling, applications and biological interpretations. Journal of Theoretical Biology, vol. 238(4), pp. 841—862. Elselvier (2006)
11. Mallet, D., De Pillis, L.G.: A cellular automata model of tumor–immune system interactions. Journal of Theoretical Biology, vol. 239(3), pp. 334--350. Elselvier (2006)
12. Chaplain, M., McDougall, S., Anderson A: Mathematical modeling of tumor-induced angiogenesis. Annual Review of Biomedical Engineering, vol 8, pp. 233—257. Annual Reviews (2006)
13. Tashakor, G., Suppi, R.: HpcNetlogo. Frontend for the concurrent execution of Netlogo experiments using SGE HPC cluster. https://github.com/hpcnetlogo/hpcnetlogo. (2017)
14. Borges, F., Gutierrez-Milla, A., Luque, E., Suppi, R.: Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling. Future Generation Computer Systems, vol. 68, pp. 59--73. Elsevier (2017)
15. Collier, N., North, M.: Parallel agent-based simulation with Repast for High Performance Computing. Simulation, vol 89(10). pp. 1215—1235. Sage Journals (2012)