

Interactive visualization tool for planning cancer treatment

R Wcisło¹, W Dzwinel^{1,2*}, P Gosztyła¹, DA Yuen³, W Czech¹

¹AGH University of Science and Technology, Institute of Computer Science, Al. Mickiewicza 30, 30-059 Krakow, Poland, wcislo@agh.edu.pl, dzwinel@agh.edu.pl.

²Cracow University of Technology, Institute of Teleinformatics, Podchorążych 1, 30-084 Kraków, Poland

³University of Minnesota. Minnesota Supercomputing Institute, Minneapolis, MN55455-0219, USA, daveyuen@gmail.com.

Abstract. We discuss the components and main requirements of the interactive visualization and simulation system intended for better understanding the dynamics of solid tumor proliferation. The heterogeneous Complex Automata, discrete-continuum model is used as the simulation engine. It combines Cellular Automata paradigm, particle dynamics and continuum approaches to model mechanical interactions of tumor with the rest of tissue. We show that to provide interactivity, the system has to be efficiently implemented on workstations with multiple cores CPUs controlled by OpenMP interface and/or empowered by GPGPU accelerators. Currently, the computational power of modern CPU and GPU processors enable to simulate the tumors of a few millimeters in diameter in its both avascular and angiogenic phases. To validate the results of simulation with real tumors, we plan to integrate the tumor modeling simulator with the *Graph Investigator* tool. Then one can validate the simulation results on the base of topological similarity between the tumor vascular networks obtained from its direct observation and simulation. The interactive visualization system can have both educational and research aspects. It can be used as a tool for clinicians and oncologists for educational purposes and, in the nearest future, in medical *in silico* labs doing research in anticancer drug design and/or in planning cancer treatment.

Keywords Tumor growth, interactive visualization, complex automata model, parallel computations, complex networks, CPU vs. GPGPU

Article Outline

1. Introduction
 - 1.1. Motivation
 - 1.2. Modeling domain
2. Computational models of solid tumor growth
3. Particle model as a framework of the interactive visualization system
 - 3.1. Model description
 - 3.2. Directions of the model development
4. Parallel implementation of particle model
 - 4.1. Algorithms and data structures
 - 4.2. Speedups and exemplary results
 - 4.3. Possible improvements
5. Implementation
6. Validation tools
7. Summary and conclusions
8. Bibliography

* Corresponding author

1. Introduction

1.1 Motivation

It is widely believed that cancer is responsible for about 20% of deaths in developed countries [Jemal et. al., 2010]. Therefore, it is assumed to be one out of major killers in the developed world. Despite the enormous financial effort that has been devoted to research on cancer phenomenon, the most important aspects remain obscure for clinicians and experimentalists. This is the reason that many of the currently used therapeutic strategies become not entirely effective.

A better understanding of tumor formation and its proliferation can be expected from a computer (i.e., quantitative) modeling. Computer models can ultimately improve the overall clinical outcome by predicting the results of specific types of medical treatment administered at specific regions of interest and time checkpoints. During the last decade a great variety of tumor models covering various morphological and functional aspects of tumor growth has been developed. These advances have been recently reviewed [Lowengrub, 2010] with a focus on the classification of mathematical tools and computational algorithms.

As shown in [Chaplain, 2000], *in silico* experiments can play the role of angiogenesis assays. In [Stephanou et al., 2005], the Authors describe a computational tumor modeling framework to compare dosing schedules based on simulated therapeutic response. However, till now, despite the existence of scores of various computational models of cancer growth dynamics, a ready-to-use interactive tool for clinical application remains a dream of the future. This situation is unjustified knowing that both technologically and methodologically there is not any obstacles to create, at least, a toy interactive visualization system which might help in education of young oncologists and elucidate the fundamental mechanisms of cancer progression. The system can evolve further to a serious tools aimed to improve the therapeutic techniques currently used, to stimulate the development of new strategies and speed-up the anticancer drug design process.

In this paper we discuss the main components of a novel interactive visualization tool for simulation of cancer growth dynamics. We present preliminary results we have obtained towards development of a framework of such the system. The paper is focused on the computational model, visualization interface and implementation issues, which are the main components of such the interactive visualization system.

We present a framework of parallel 3-D model of tumor growth, which bases on Complex Automata paradigm, which combines particle dynamics with Cellular Automata and continuum models. To speed up calculations and to make on-line interactions possible we show the ways to optimize the code for multi-core CPUs and GPGPU rather than for massively parallel systems consisting of many heterogeneous computational nodes.

1.2 Modeling domain

A serious obstacle that must be overcome to simulate cancer dynamics is the intrinsic multiple scale nature of tumor growth. It involves processes occurring over a variety of time and length scales: from the tissue scale – e.g. vascular remodeling - to intracellular processes – e.g., progression through the cell-cycle.

The structure of a solid and vascularized tumor is very dynamic and heterogeneous. Its development occurs in five phases: oncogenesis (carcinogenesis), avascular growth, angiogenesis, vascular growth and metastasis. Oncogenesis is a biological process occurring on molecular level and characterized by serious disruptions in DNA reparation system mainly due to numerous mutations. Consequently, the escape of mutated cell from apoptosis is observed. The avascular stage is the earliest stage the tumor develops in the absence of blood supply. The tumor grows up to a maximum size which is limited by the amount of diffusing nutrients and oxygen reaching the tumor surface. In the third stage, hypoxic cells of this avascular tumor mass produce and release substances called tumor angiogenic factors (TAFs). They diffuse throughout the surrounding tissue, and, hitting vasculature, trigger a cascade of events which eventually lead to vascularisation of the tumor. These phases of tumor growth are depicted in Figure 1. In vascular stage, tumor has access to virtually unlimited resources, so it can proliferate beyond any limits. Moreover, in this growth phase tumor acquires a means of transport for cells that penetrate into the vasculature and form metastases in any part of the host organism.

The interactive system we propose comprises only two scales of tumor growth: avascular, angiogenesis and partly vascular phases (see Figure 1). The size of tumor simulated on-line, depends on the available computational power, 2-D or 3-D version of the model and its level of details. Approximately, taking into account the power of nowadays workstations, the maximum size of tumor which can be modeled using our approach does not exceed the diameter of 1 cm. This scale of interest is extremely important. Thus, once the tumor becomes vascularized, therapeutic prognoses worsen dramatically. On the other hand, in that scale, the system can be used in investigations of the role the various tumor angiogenic factors play and for defining the targets of antiangiogenic therapies.

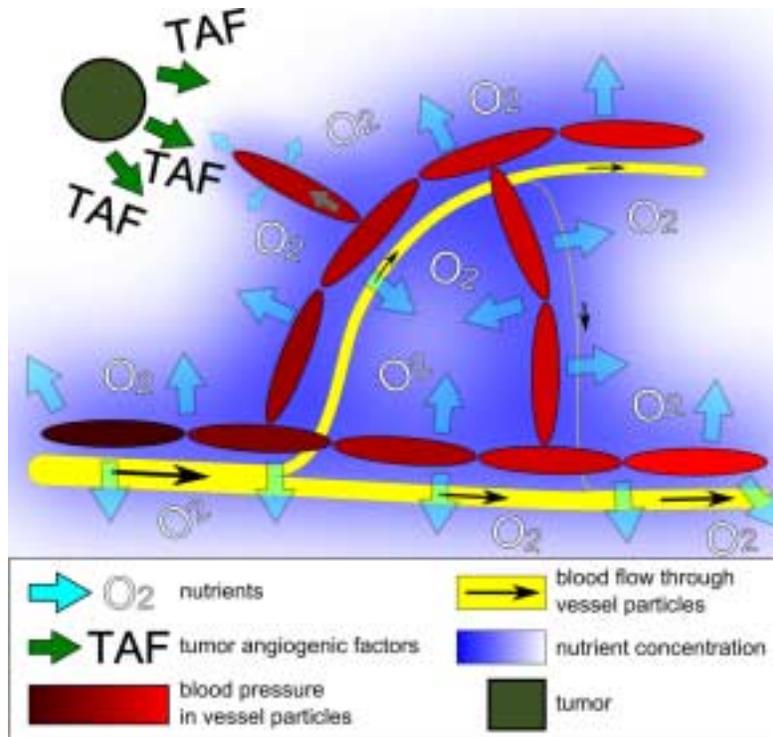


Figure 1 Angiogenic phase of tumor growth.

2. Computational models of solid tumor growth

The computational paradigm plays a crucial role in development of *in silico* implementation of cancer model. In the length scales exceeding centimeters, continuum methods are acceptable for modeling tumor dynamics. Continuum model parameters can be relatively easy to obtain, analyze and control. They may also be accessible through laboratory experimentation. However, despite they can provide significant insight into the relative role that different process play in the tumor formation they fail completely in predicting tumor microstructure. This may be important deficiency when studying the effect of genetic, cellular and microenvironment characteristics on overall tumor behavior. Within continuum models, it is not possible to capture such the important events as repeated sprout branching and the overall dendritic structure of the vascular network, compartmentalization of the tumor tissue and remodeling processes due to mechanical interactions [Lowengrub et al., 2010].

Whereas continuum models describe cell populations by means of continuous fields, discrete models deal with the dynamics of agents. The agents usually represent individual cells which are tracked and updated according to a specific set of biophysical rules taken from a discrete and finite space of states and evolve in discrete space and time. Unlike in the continuum models, discrete models can follow individual cells and can reveal more details about cell dynamics and its interaction with the tissue. There are, for example, percolation based models [Szczerba et al., 2008], Eden models [Alcorn et al., 2005; Lee et al., 2006], random walk and diffusion limited aggregation (DLA) models [Amyot et al., 2006], cell based models [Bauer et al., 2007], lattice gas models and cellular automata [Dormann and Deutsch, 2002; Moreira and Deutsch, 2002].

Discrete approach is particularly useful for studying spatial scales of tumor dynamics below 1 cm such as carcinogenesis, natural selection, genetic instability, interactions of individual cells with each other and the microenvironment. Analyses of cell population dynamics have also been employed in order to study biological characteristics applying to all cells in a particular population, such as response to therapy and in studies of immunology.

According to [Lowengrub et al., 2010], there are two main types of discrete models, i.e., lattice based and lattice-free. The former describes the dynamics of discrete tumor cells as automata on a grid whose states are governed by a set of deterministic or probabilistic rules. One of the most popular lattice based paradigm used for modeling tumor growth are cellular automata (CA) (see the critical overview [Dormann and Deutsch, 2002]). Cellular automata deal with the dynamics of discrete elements populating the nodes of structural grid. The elements take their state from a discrete - finite or infinite - space of states and evolve in discrete space and time. The dynamics of the elements is defined in terms of local, either deterministic or probabilistic, rules. To describe four aspects of tumor growth, namely, avascular growth, vascular growth, invasion, and angiogenesis, the classical CA models should be supplied with additional graph structures defined on the top of a regular grid [Topa, 2008]. The mechanical interactions between the tumor and healthy tissue are only partly addressed in [Mansury et al., 2002].

The lattice-free approaches describe the motion of discrete cells in arbitrary locations and their interactions. It is possible to translate detailed biological processes (e.g. cell life-cycle) into rules influencing both the cell motion and their mutual interactions. However, the computational cost increases faster than linearly with the number of modeled cells, limiting these methods to the spatial and temporal scales defined by the achievable computational power. Moreover, while these models are capable of describing biophysical processes in significant detail, it may be

nontrivial to obtain reliable measurements of model parameters through experiments e.g., parameters of cell interaction rules.

An important research domain involves the development of hybrid continuum–discrete models for tumor growth. These models have the potential to combine the best properties of both continuum and discrete approaches. They may provide more realistic coupling of biophysical processes across a wide range of length and time scales. Discrete models are usually hybridized with continuum approaches in the sense that both substrate concentrations and blood flow are computed using continuum approaches (i.e. oxygen, glucose, matrix-degrading enzymes concentration by solving reaction-diffusion equation and blood flow by integrating hydrodynamic equations) while the cell-based components such as migrating EC tip cells are discrete.

The oldest hybrid model, simulating two-dimensional spatial distribution of sprouts, was presented by Stokes and Lauffenburger [1991]. It uses a classic Folkman formulation [Folkman 1971; Folkman and Hochberg, 1973]. The tumor is located at the top center of a box of finite volume and the capillary at the bottom. The evolution of molecular species is governed by discretized reaction-diffusion equations. A stochastic differential equation simulates the migration of endothelial tip cells employing a particle dynamics in TAF concentration field. This approach is the basis of many other hybrid models (e.g., [Chaplain, 2000; Godde and Kurz, 2001; Preziosi, 2003; Alacorn et al., 2005; Amyot et al., 2006; Milde et al., 2008; Dormann and Deutsch, 2002; Moreira and Deutsch, 2002; Topa 2008]) which differ in:

- geometrical properties of the simulation such as: dimensionality (2-D, 3-D), type of discretization of space and time (on-grid, gridless), structure of vascular network (rigid, structured, unstructured) *etc.*
- modeling accuracy – defined e.g. by the number of angiogenic factors and other subprocesses included in the model,
- methodology of simulation of the process of vascularization and tumor growth (stochastic, deterministic, cellular automata, lattice-gas, DLA *etc.*),

The accuracy of the computational model depends on the proper choice of processes and multiple scales crucial for tumor growth. Because of the complexity of cancer evolution, many models to date focus on single key sub-processes, disregarding their interactions with others. Many attempts assume either a static tumor and concentrate on dynamic vascularization in the absence of tumor growth or a static network topology. Some of them use dynamic network describing hydrodynamics of blood flow while neglecting its interaction with concentration fields and tissue components [Godde and Kurz, 2001]. There are also many models which describe only avascular phase of tumor dynamics [Dormann and Deutsch, 2002]. The usefulness of these models is strongly constrained. Truly important are the models, which track coupled tumor growth and tumor-induced neovascularization using a discrete approach for both.

These hybridized models are the basis for the development of the multiple-scale approaches which - apart from the tissue scale evolution - include processes from cellular and molecular scales. They represent the most advanced simulation methodologies. Multiple scale models incorporate e.g. cellular heterogeneity, intercellular phenomena, more complex mechanical laws to describe the response of the tissue to external forces, blood hydrodynamics and vessel remodeling. Advanced multi-scale models of tumor progression are presented, e.g., in papers by Bellamo *et al.* [2003] and by Alacorn *et al.* [2005]. Rigid geometric constraints which disable realistic visualization and are the sources of many serious artifacts belongs to the principal weaknesses of these models [Wcislo et al., 2009].

For challenging high computational demand of discrete approaches in modeling spatial scales exceeding 1 cm, coupled continuum–discrete descriptions of tumor cells are realized. It is the second type of hybridized models in which the tumor itself is described using both continuum and discrete components to reduce computational complexity of discrete approach. This hybrid modeling is very important as it affords the possibility of seamlessly up-scaling from the cell-scale to the tumor and tissue scales. For example, a greater part of the system can be simulated using continuum approach while some critical regions can be modeled using discrete approach, as in [Kim et al., 2007; Stolarska et al., 2009]. They simulate the motion of separate (discrete) cells in the outer proliferating rim of an avascular tumor while they use continuum description of cell dynamics (i.e., density of cells in various states) in the inner quiescent and necrotic regions of the tumor. On the other hand, both the continuum and discrete representations of tumor cells can be employed simultaneously throughout space, subject to mass and momentum conservation laws which incorporate interactions among the discrete and continuous fields [Bearer et al., 2009]. Very detailed overview of the recent continuum, discrete, hybrid and multi-scale models of tumor proliferation, containing almost 600 references, are presented in Lowengrub et al. [2010].

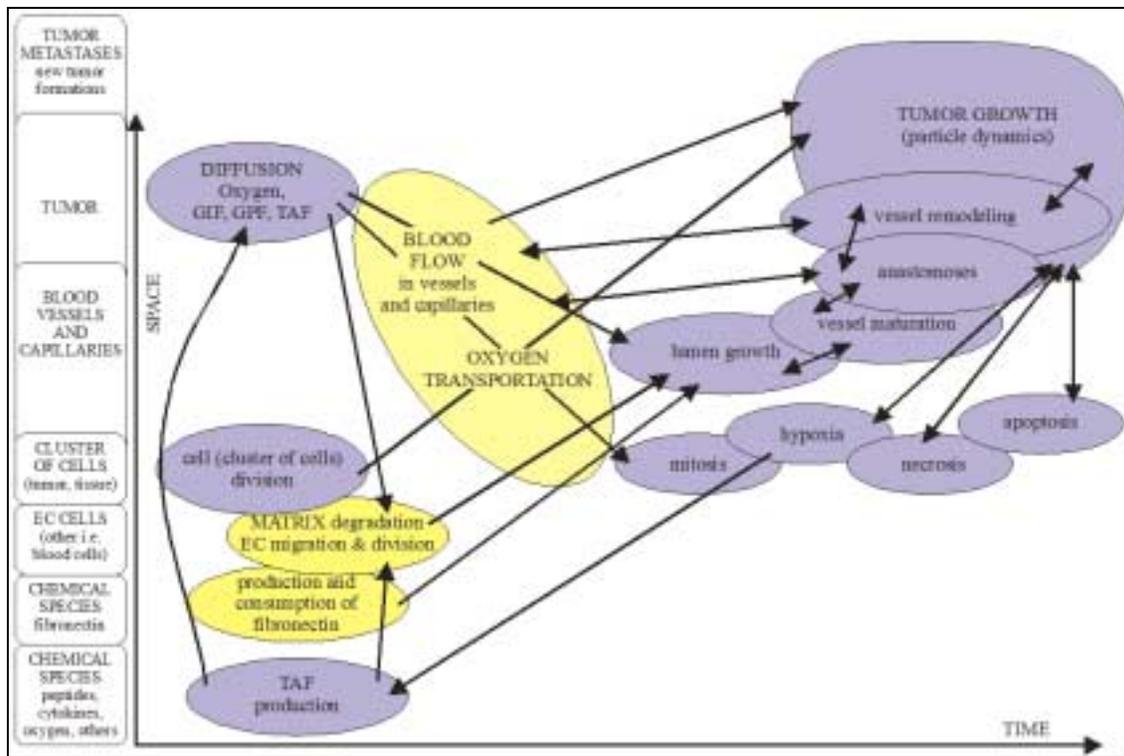


Figure 2 Main processes from various spatio-temporal scales involved in tumor proliferation (from [Wcislo et al. 2009]).

In macroscopic (> 1 cm) and mesoscopic (> 1 mm) scales tumor growth is a purely mechanical phenomenon. Just mechanical interactions influences the most the structure of vasculature, blood flow, tumor shape and decides about its directional progression. Due to the lack of computational framework, existing computational paradigms are not able to reproduce adequately this basic process. This is a serious obstacle in creation of a truly multi-scale model of tumor dynamics, which enables not only addition of novel components representing the chemical and biochemical inter and intracellular processes but allows for realistic tumor visualization in macroscopic scale as well. Moreover, the lack of computational metaphor which enables realistic visualization of

tumor dynamics disables the possibility of creation interactive tumor growth simulator which allows for its dynamics be observed and modified directly by oncologist. We propose here the complex automata paradigm (CxA) based on particle dynamics as a computational framework of the mechanistic tumor model.

In Figure 2 we present the main processes from various spatio-temporal scales involved in tumor growth. The microscopic processes such as cell motility and cell cycle are discrete. The macroscopic scale refers to phenomena which are typical for continuum systems such as diffusion (of oxygen and TAF), overall tumor condensation and blood flow. In macroscopic models, microscopic phases can be approximated by coarse grained models as long as the methodology of multi-scale simulation and adequate computational resources are lacking. The arrows in Figure 2 show the relationships between these processes. The particle model presented in the following sections refers only to the processes shaded in blue.

Complex automata (CxA) are a generalization of cellular automata paradigm. They represent a scalable hierarchical aggregation of CA and agent-based models [Hoextra et al. 2007]. Globally, CxA can behave either as the classical CA nodes on a structural lattice or as interacting particles whose dynamics is described, e.g., by the Newtonian laws of motion or other stochastic laws.

In [Weislo et al., 2008, 2009, 2010a,b] it was shown that the complex automata paradigm employing both particle dynamics and cellular automata rules can be used as a robust modeling framework, e.g., for developing realistic models of tumor growth as a result of emergent behavior of many interacting cells. This framework represents the spatio-temporal scales involving mechanical interactions between growing tumor, normal tissue and expanding vascular network. This framework can be easily extended by including both fine grained processes responsible for tumor creation/proliferation and be coupled with tissue scale processes modeled by continuum reaction-diffusion and blood hydrodynamics equations. In the following paragraph we present briefly the assumptions of our CxA particle model.

3. Particle model as a framework of the interactive visualization system

3.1 Model description

As shown in Figure 3, in our Complex Automata model a fragment of tissue and vasculature is made of particles. The particle is defined as $\Lambda_N = \{O_i: O(\mathbf{r}_i, \mathbf{v}_i, \mathbf{a}_i), i=1, \dots, N\}$ where: i is the particle index; N - the number of particles, $\mathbf{r}_i, \mathbf{v}_i, \mathbf{a}_i$ - particle position, velocity and attributes, respectively. We assume additionally that:

- Each particle represents a single cell with a fragment of ECM (extracellular matrix).
- The vector of attributes \mathbf{a}_i is defined by:
 - the particle type $\{tumor\ cell\ (TC),\ normal\ cell\ (NC),\ endothelial\ cell\ (EC)\}$,
 - cell life-cycle phase shown in Figure 4 $\{newly\ formed,\ mature,\ in\ hypoxia,\ after\ hypoxia,\ apoptosis,\ necrosis\}$,
 - other parameters such as: cell size, cell age, *hypoxia* time, concentrations of k =TAF (tumor angiogenic factor), or O_2 (and other diffusive substances) and total pressure exerted on particle i from the rest of tumor body and tissue mass.
- The particle system is closed in 3-D computational box under a constant external pressure.
- The vessel is constructed of tube-like “particles” (called EC-tubes) made of two particles connected by a spring (see Figure 4).
- We define three types of interactions: particle-particle, particle-tube, and tube-tube.

- The forces between particles mimic both mechanical repulsion and attraction due to cell adhesiveness and depletion interactions cause by both ECM matrix and the cell itself.
- We postulate the particle-particle conservative interaction potential $\Omega(d_{ij})$ defined as follows:

$$\Omega(d_{ij}) = \begin{cases} a_1 d_{ij}^2, & \text{for } d_{ij} < 0 \\ a_2 d_{ij}^2, & \text{for } 0 < d_{ij} < d_{cut} \\ a_2 d_{cut}^2, & \text{for } d_{ij} \geq d_{cut} \end{cases} \quad \text{where } a_1 > a_2 \quad d_{ij} = |\mathbf{r}_{ij}| - (r_i + r_j) \quad (1)$$

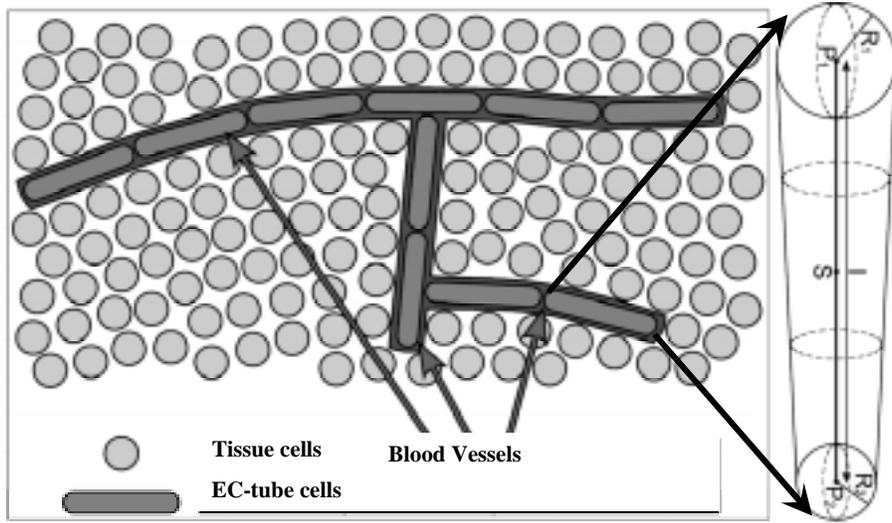


Figure 3 Tissue particles and tube-like EC-tube particles made of two spherical “vessel particles”.

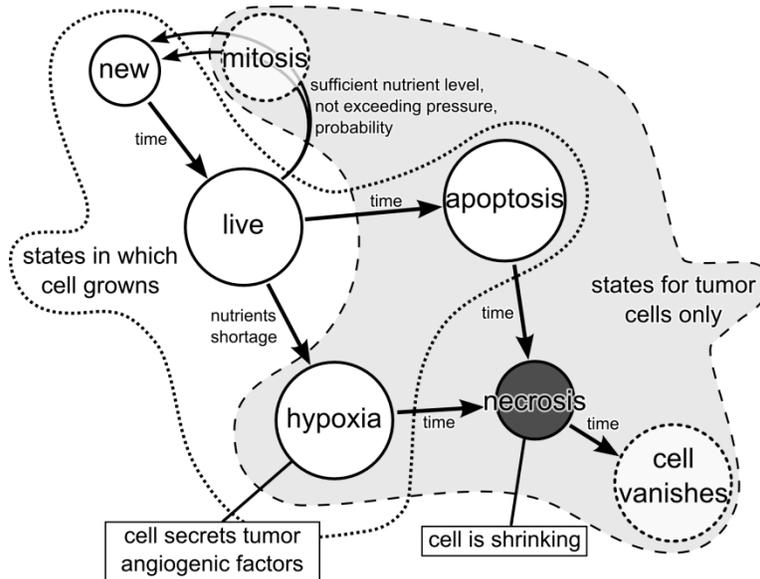


Figure 4 Cell life cycle and various cell stages.

where $|\mathbf{r}_{ij}|$ is the distance between particles while r_i and r_j are their radii. Additional viscosity force, proportional to the particle velocity, simulates dissipative character of interactions. The

cells of all kinds (tumor, normal and EC-tubes) evolve in discrete time according to Newtonian dynamics in the continuum diffusion fields of TAF and nutrients. The concentration fields are updated every time-step. We assume that both the concentrations and hydrodynamic quantities are in steady state in the time-scale defined by the time-step of numerical integration of equations of motion. This assumption is justified because diffusion of oxygen and TAFs through the tissue is many orders of magnitude faster than the process of tumor growth. On the other hand, the blood circulation is slower than diffusion but still faster than cell-life cycle. Therefore we used fast approximation procedures for both calculation of blood flow rates in capillaries and solving reaction-diffusion equation.

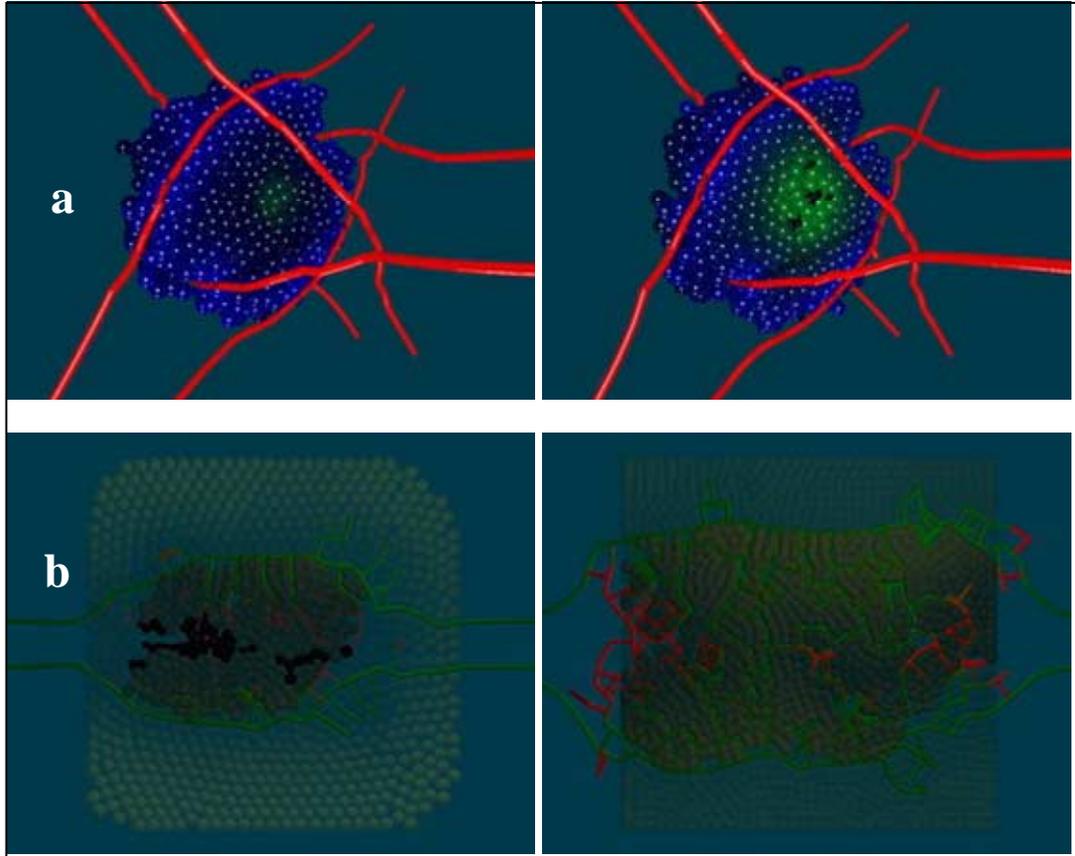


Figure 5 Snapshots from particle based simulations of dynamics of a) avascular tumor (necrotic center is displayed) b) vascularization of tumor influenced by pericytes (functional vessels are in green, the red ones are capillaries under regression)

As shown in Figure 5 (see <http://www.icsr.agh.edu.pl/~wcislo/Angiogeneza/index.html>) we can observe and control both the phase of avascular tumor dynamics and its proliferation after angiogenic switch.

For example, the section of the tumor spheroid, shown in Figure 5a, displays a layered structure. We can observe formation of a core zone composed mainly of necrotic material surrounded by a layer of quiescent tumor cells and an outer ring of proliferating tumor cells. It is crucial to understand the processes, which are responsible for the growth of a layered and saturating tumor.

Expanding tumor involves even more factors such as blood dynamics in newly formed vessels their regression/maturation and remodeling due to complex mutual interactions of TAF/nutrients/pericytes and growing tissue/tumor pressure. Particle model allows for attacking many important problems influencing angiogenic switch e.g., if intravenous infusion of pericytes grown ex vivo will stabilize angiogenesis and slow down tumor growth (e.g., [Darland et al, 2003]). The snapshots from pilot simulations using the CxA particle model are presented in Figure 5b.

The newly formed blood vessels due to the process of angiogenesis become functional when they form anastomoses. It allows for blood flow due to pressure gradient on its ends. Otherwise, we assume that the functional capillaries but without pericyte support dissolve. Moreover, nonfunctional and immature vessels without blood flow undergo the process of regression. The vessel maturation is controlled by the density of pericytes. The varying degrees of pericyte recruitment indicate differences in the functional status of the tumor vasculature. In simulation presented in Figure 5b we assume very simplistic model of vessel maturation. The regression time depends on the local density of EC tubes assuming constant concentration of pericytes. If the density is too high regression time is shorter. This model should be corrected calculating pericyte concentration using the continuum models mentioned earlier.

3.2 Directions of the model development

For characterization of e.g. melanoma cell lines, the model should simulate the migration of cancer cells in normoxic and hypoxic conditions. We show in [Wcislo et al. 2009] that the inward motion of cells from the rim of 3-D tumor globule to the necrotic center is a purely mechanical effect caused by pressure drop from the surface towards this center. However, the mechanical effects allowing for cancer cell motility outward the tumor is more sophisticated phenomena (e.g. [Motsumoto et. al., 2008]).

To find the mechanical and biological factors enabling tumor cell motility, the tumor mass dynamics should be modeled in a greater precision than it was done in [Wcislo et al., 2009]. In this model we simulate the tumor as a collection of soft spheres. However, for many types of tumor, the mass can be treated rather as a physical system with properties on the border between the solid and the liquid. Therefore, to simulate better the influence of adhesiveness of cells and viscosity for tumor fingering and compartmentalization we plan to use different model of cell interactions, similar to that in dissipative particle dynamics and fluid particle dynamics methods [Espanol, 1998; Dzwinel et al., 2002]

The tumor, normal tissue cells and EC-tubes will be defined by its mass, moment of inertia, translational and angular momenta. Two particles i and j interact with one another by a collision operator \mathbf{F}_{ij} defined as a sum of constituent forces, whose parameters are dependent on the type of interacting particles. The forces are central and non-central and consist of conservative \mathbf{F}_C (see Eq. (2)), dissipative \mathbf{F}_D and Brownian \mathbf{F}_B components and:

$$\mathbf{F}_{ij} = \mathbf{F}_C + \mathbf{F}_D + \mathbf{F}_B$$

$$\mathbf{F}_{ij} = -F(r_{ij}) \cdot \mathbf{e}_{ij} - \gamma \cdot [A(r_{ij})\mathbf{1} + B(r_{ij})\mathbf{e}_{ij}\mathbf{e}_{ij}] \circ \left[\mathbf{v}_{ij} + \frac{1}{2}\mathbf{r}_{ij} \times (\vec{\omega}_i + \vec{\omega}_j) \right] + \tilde{\mathbf{F}}_B(r_{ij}) \quad (2)$$

where: $F(r_{ij})$ – is a central conservative force, whose formula depends on the types of interacting particles, γ - is a scaling factor of dissipative forces corresponding to viscosity, $A(r_{ij})$ and $B(r_{ij})$ are the weighting functions, \mathbf{F}_B is a random component representing cell random motion. In original fluid particle method this factor is scaled by the temperature of the system and expressed in terms of the weighted Wiener increments. In our model it will define the dissipative properties of the tissue, i.e., its softness.

The value of \mathbf{F}_{ij} , is equal to 0 if the separation distance between two particles i and j , r_{ij} , exceeds a cut-off radius R_{cut} . The total force per each particle i is computed as a sum of forces interacting with particle i within the sphere of the radius equal to R_{cut} . The temporal evolution of the particle ensemble obeys the Newtonian equations of motion with rotation of the particles included,

$$\dot{\mathbf{P}}_i = \sum_{j:r_{ij}<R_{cut}} \mathbf{F}_{ij}(\mathbf{r}_i, \mathbf{v}_i, \omega_i), \dot{\mathbf{r}}_i = \mathbf{v}_i, \dot{\omega}_i = \frac{1}{I_i} \sum_{j:r_{ij}<R_{cut}} \mathbf{N}_{ij}(\mathbf{r}_i, \mathbf{v}_i, \omega_i), \mathbf{N}_{ij} = -\frac{1}{2} \mathbf{r}_{ij} \times \mathbf{F}_{ij} \quad (3)$$

where \mathbf{r}_i is the position of particle i , \mathbf{P}_i is its momentum and ω_i angular velocity. We assume that the interactions between spherical particles and EC-tube particles have a similar character.

To enable the simulations of tumor sizes above one centimeter the hybridization of two models – continuum approach and the CxA based framework - is the principal methodological goal for the future. It will consist of four steps:

1. Development of continuum mathematical model, which can be use for simulating tumor larger than 10 cm in diameter.
2. On the base of the wavelet solver [Vasilyev, 2003] - extraction of the ROI (regions of interest) in which the discrete model will be used.
3. Elaboration of bridging procedures between continuum and discrete models.
4. Further development of the discrete particle based model towards, so called, coarse grained particle models where a particle does not represent a single cell but a fragment of tissue.

In the rest of this paper we limit our consideration to the particle based framework presented in 3.1. Our goal is to provide a flexible and fast simulation tool for interactive visualization which could be used on small but strong stand-alone workstations by clinicians for both educational and/or research purposes. In the following paragraph we present the implementation issue, which are crucial to achieve this goal.

4. Parallel implementation of particle model

For modeling of tumors of realistic sizes, i.e., a few millimeters of diameter, the dynamics of 10^5 - 10^7 particles - normal, cancerous and EC-tube cells - have to be simulated by exploiting the power of nowadays multi-core CPUs, multi-processor systems and by using optimized N-body parallel codes. The box of the size 3.5 mm contains about 10^6 cancer cells – i.e. the approximate number of particles that can be simulated on a strong laptop in a reasonable computational time. To simulate larger tumors, the codes should be tuned to current parallel processors.

However, the CxA particle system is very different than standard particle ensembles such as in the short range molecular dynamics. Consequently, the process of code parallelization is more complicated [Wcislo et al., 2010a,b]. The cells can proliferate, change their size or annihilate. Moreover, they have additional attributes, which evolve according to the rules of CA and

influence the cells' dynamics. The attributes, in turn, depend on concentration fields of O_2 , TAF and other substances. This requires solving reaction-diffusion equations and calculating the intensity of blood flow in capillary vessels every time step.

4.1 Algorithms and data structures

Classical N -body codes, such as molecular dynamics (MD), simulate spatio-temporal evolution of a particle ensemble confined in a periodic cube by integrating numerically Newtonian equations of motion [Haile, 1992]. Single time-step consists of two consecutive procedures: computation of forces acting on each particle and moving them according to the total momentum calculated.

For short-range interactions, the forces can be computed using fast $O(N)$ method exploiting alternately Hockney or Verlet algorithms [16]. However, both the calculation of forces and approximate procedure used for solving diffusion equation, require finding all the pairs of particles in the nearest neighborhood. As shown in Figure 6a, the computational box is divided onto cubic sub-boxes with edges equal to the interaction range. The particle located in a given sub-box interacts with other particles located in this sub-box and in adjacent sub-boxes.

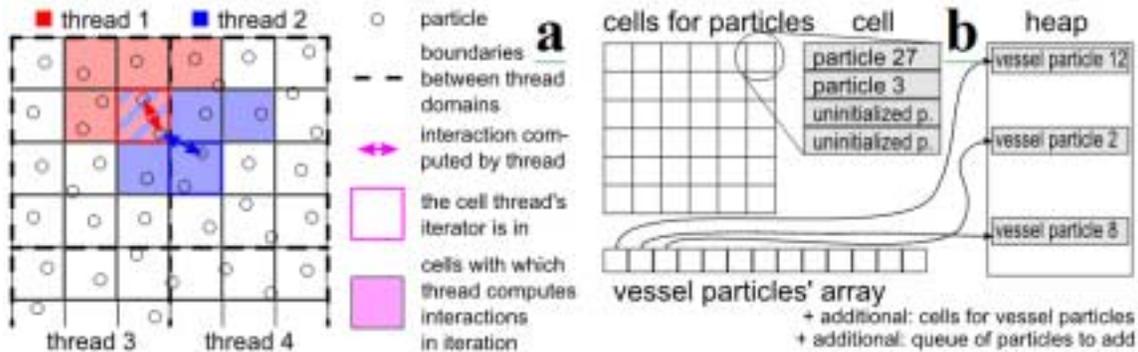


Figure 6 a) Domain decomposition used for forces calculation. b) Data structures storing spherical and vessel particles [Wcislo et al., 2010].

As shown in Figure 7, the computation of EC-tube particles interactions is the most critical component influencing computational efficiency. The length of EC-tube is considerably greater than its width. It involves considerably larger sizes of sub-boxes (Figure 6) than those used for spherical particles. Moreover, the tubes can grow exceeding the size of 5 sub-boxes used for forces calculation between spherical cells.

To solve this problem we propose using instead of one array of particle positions, two separate data structures \mathbf{P} and \mathbf{V} : \mathbf{P} for storing spherical particles and \mathbf{V} for EC-tubes, respectively (see Figure 6b). The \mathbf{P} data structure is represented by 3-D array of Hockney sub-boxes (Hockney cells) with tumor and normal particles. The \mathbf{V} is a data structure consisting of the array of pointers to records representing EC-tubes and the additional 3-D array of sub-boxes used to compute particle-tube and tube-tube interactions. The sub-boxes in this array correspond to respective sub-boxes in \mathbf{P} . Because vessel particle is long enough to cross several sub-boxes, it cannot be assigned to a single sub-box, as it is in Hockney algorithm. Instead, EC-tube is placed in a minimal cuboid composed of all the sub-boxes it crosses. This cuboid is enlarged then by one sub-box margin in each direction, covering the vessel particle together with its cut-off radius. We assume that the vessel particle belongs to all the sub-boxes forming this final cuboid.

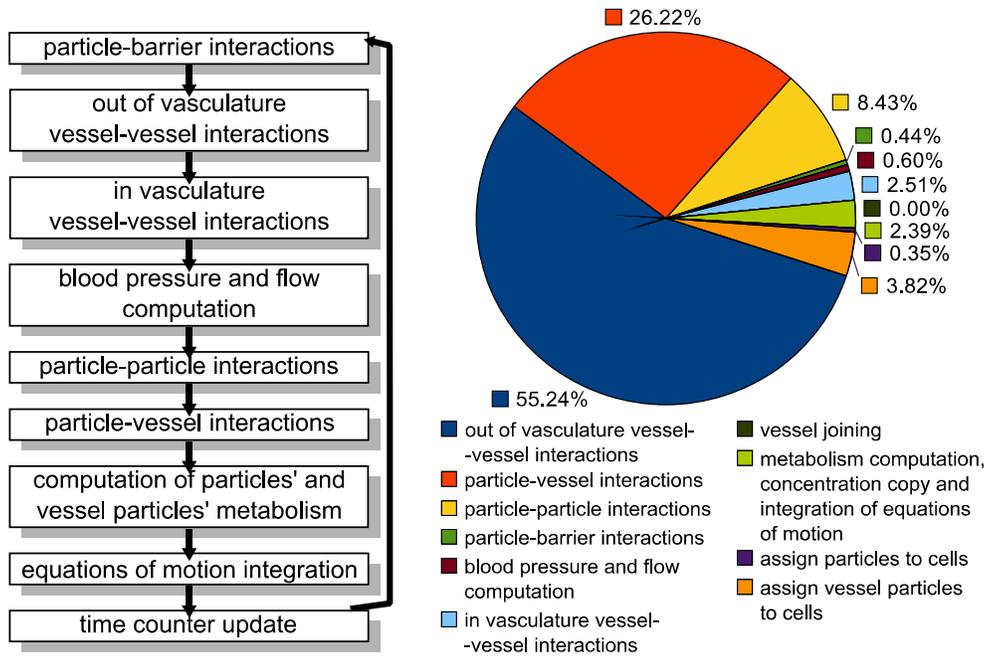


Figure 7 The main procedures invoked in a single time-step and diagram showing the shares of computational time used by various procedures of the model (the evolution of 10^6 particles was simulated).

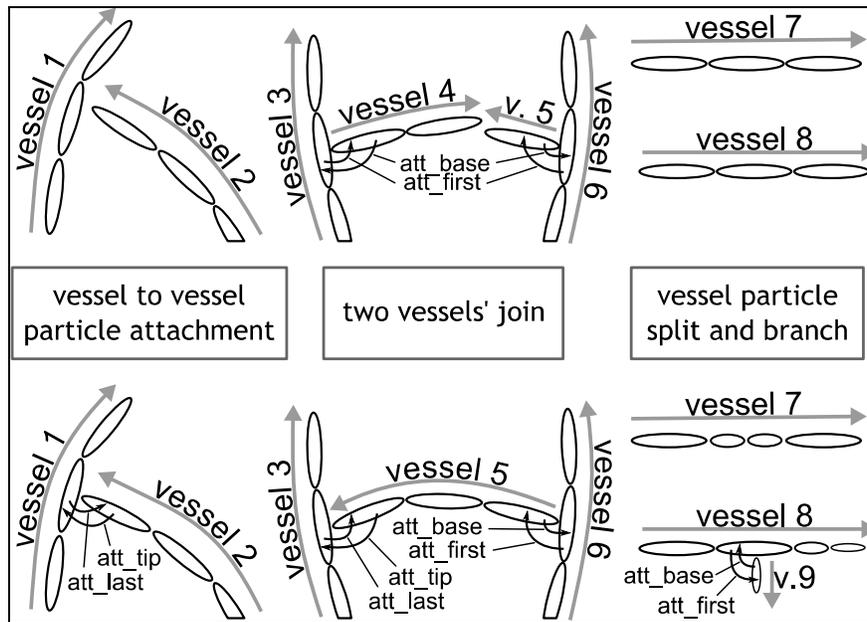


Figure 8 Vessel-vessel interactions and vessel growth rule.

Calculation of forces between particles is realized by three separate algorithms: particle-particle, particle-vessel and vessel-vessel interactions calculation. Particle-particle forces are calculated using standard Hockney algorithm [Hockney and Eastwood, 1981]. In case of particle-vessel computation, for each corresponding pair of sub-boxs cp , cv from \mathbf{P} and \mathbf{V} , respectively, particles

from cp are tested against vessel particles from cv . If the distance between the pair of particles is shorter than the cut-off radius, their mutual interaction is calculated.

The algorithm for vessel-vessel computations is constructed knowing that if two EC-tubes lie in a distance shorter than cut-off radius, there exists at least one sub-box in \mathbf{V} containing both particles. Therefore, all interacting pairs can be found by iterating throughout all sub-boxes and testing all-to-all distances. The problem is that a pair of particles representing two interacting EC-tubes can be found in many sub-boxs while it should be taken only once. To solve it, we introduce ternary relation R (see Figure 9), which eliminates redundant interactions:

$$Ec \times Ec \times C \supseteq R = \left\{ \begin{array}{l} (e_1, e_2, c) : e_1, e_2 \in Ec; c \in C; \\ \max(e_1.mx, e_2.mx) = c.x, \\ \max(e_1.my, e_2.my) = c.y, \\ \max(e_1.mz, e_2.mz) = c.z \end{array} \right. \quad (4)$$

where: Ec is the set of EC-tubes, C is the set of sub-boxs in mesh, $c.x, c.y, c.z$ are coordinates of sub-box c in 3-D array and

$$\begin{aligned} e.mx &= \min(\text{sub-box}(e.p1).x, \text{sub-box}(e.p2).x), \\ e.my &= \min(\text{sub-box}(e.p1).y, \text{sub-box}(e.p2).y), \\ e.mz &= \min(\text{sub-box}(e.p1).z, \text{sub-box}(e.p2).z), \end{aligned} \quad (5)$$

where: $e.p1$ and $e.p2$ are two ends of EC-tube e , $\text{sub-box}(p)$ is the sub-box to which point p belongs to.

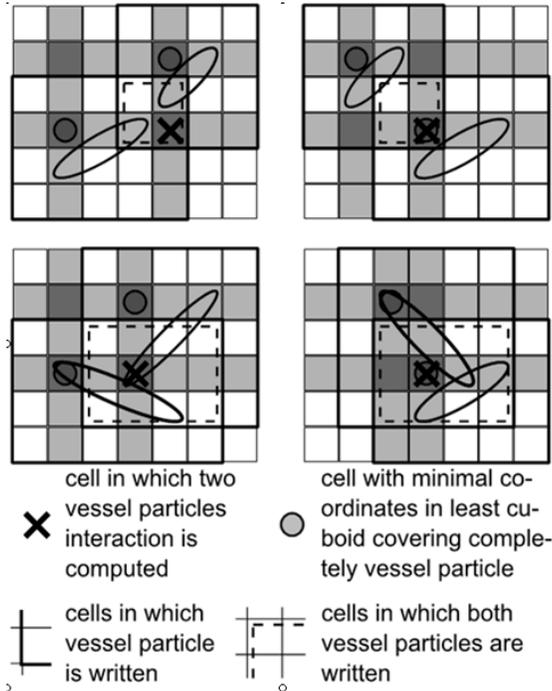


Figure 9 Graphical interpretation of relation (4) [Wcislo et al., 2010].

To reduce the number of cache misses, the sub-boxes in \mathbf{P} do not contain pointers to particle records but whole records instead. Each sub-box is represented then by an array of fixed number of objects (see Figure 6b). All the tumor and normal particles are allocated directly inside corresponding sub-boxes. This guarantees that particles are always properly ordered in memory according to their positions. However, we pay the price of greater memory consumption. This is

because the sub-boxes have various numbers of particles and many records are empty. As particles move, they change the sub-boxes they belong to. Therefore, the arrays **P** and **V** are updated after each time-step. In case of **V**, the sub-boxes are built from the beginning by using the array of EC-tube pointers located in **V**. Whereas for **P**, because particles are allocated inside the sub-boxes, changing location from one sub-box to the other means that the whole particle record must be moved to a different memory location.

This takes longer time in comparison to pointers operation in the standard approach. However, because of steady nature of particles dynamics in our model, such the situation does not occur too often. In fact, in our simulations the process of reordering particles requires less time than standard linked-list procedure. The reason is, that in the former, the particles which do not change their sub-boxes need only “read” operation of their coordinates from the memory, while in the latter, for all the particles there is an additional “write” operation.

During simulation, the number of spherical and EC-tube particles can both increase due to *mitosis* and decrease as the result of *apoptosis* and *necrosis*. Information of newly formed and dead particles must be added and removed from the data structures. As doing this directly could cause problems with synchronization, three intermediate data structures are employed: for newborn particles in **P**, for new vessel particles in **V** and for indexes of dying vessel particles in **V**. Removing objects from **P** is done directly as it is sub-box-local operation, which does not impair other threads operation and never cause data structures to be rebuilt. Moving object from intermediate structures to **P** and **V** and removing object from **V** is done sequentially between separate time-steps.

4.2 Speedups and exemplary results

Our parallel algorithm is constructed for a single shared memory node and is implemented in C++ with OpenMP interface. The timings were obtained for SGI Altix XE 1300 cluster consisting of 256 SGI Altix XE 300 nodes and SPARC Enterprise T5120. The single Altix node consists of two four-core processors Intel Xeon 2.66 with 16 GB of RAM allowing for maximum 8 threads executed in parallel. The SPARC computer consists of eight-core 1.2 GHz UltraSPARC T2 CPU capable of running in parallel eight threads per single core. It gives in total 64 threads per node executed concurrently on 32 GB of RAM.

We have employed domain decomposition both along one side of the computational box (each box slice was handled by one thread) and dividing the box onto sub-boxes of equal sizes (for 8 threads we have 2x2x2 grid of sub-boxes, while for 64, 4x4x4 grid of sub-boxes).

As shown in Figure 10, the preliminary timings obtained for our parallel code are very encouraging. We got speed-up of about 7 on 8 threads CPU and about 30 on 64 threads CPU simulating 10^6 particles. The timings could be better for more realistic vessel densities much lower than those considered in the test runs.

The snapshots from simulations of tumor vasculature progression obtained for timing tests are shown in Figure 11. The tests were performed for particle ensembles of various sizes. The initial scene consists of two straight parallel vessels, the cells representing normal tissue and a few cancerous cells located between the vessels.

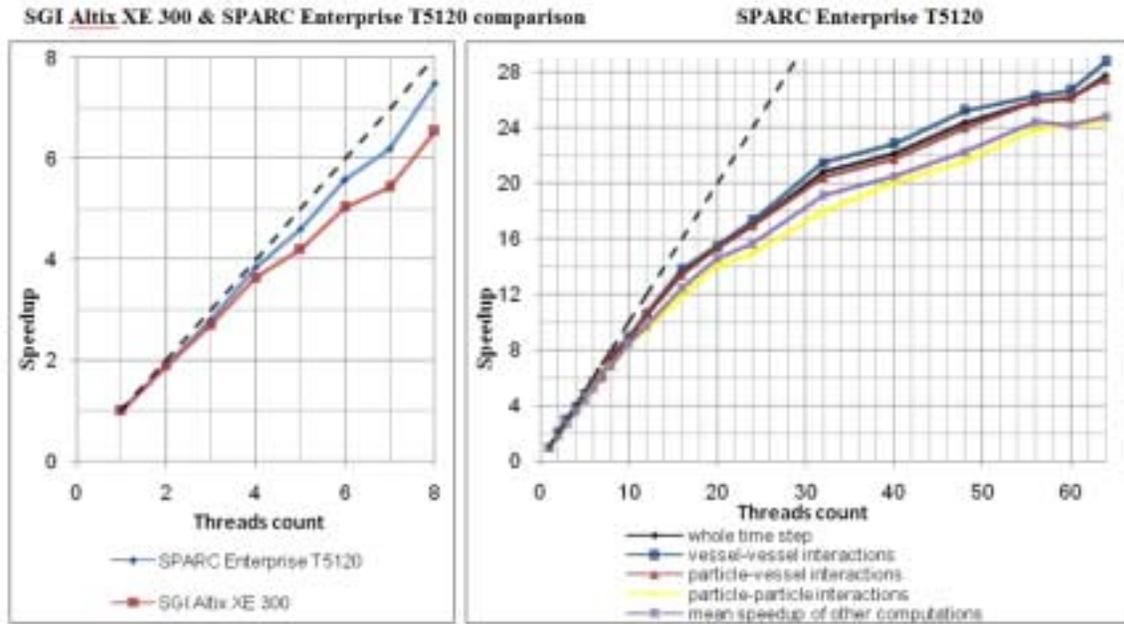


Figure 10 The speed-ups obtained for the main procedures of the tumor model during simulation of 10^6 particle ensembles on two test machines [Wcislo et al. 2010].

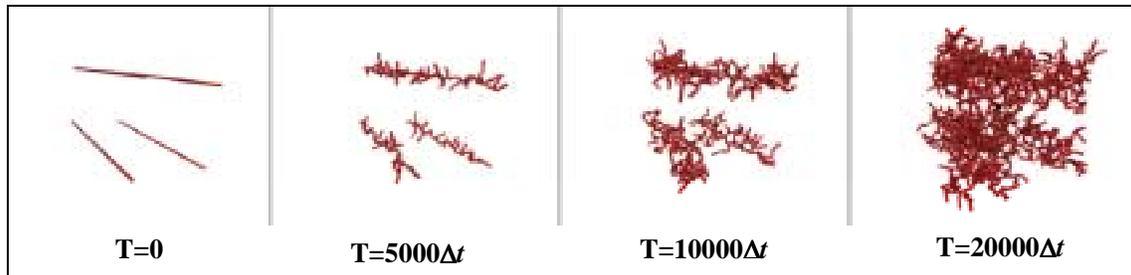


Figure 11 The snapshots showing the time evolution of tumor vasculature. The tissue particles are invisible.

Because of increasing TAFs concentration, secreted by the tumor cells in *hypoxia*, we can observe newborn capillaries sprouting out from the source vessels. The vasculature expands and is continually remodeled due to tumor growth dynamics. The sprouts can bifurcate and merge creating anastomoses. The blood flow is stimulated by pressure difference in anastomosing vessels. Only productive vessels have a chance to survive if the TAFs concentration is sufficiently high. Unproductive vessels disappear after some time. Well oxygenated cells are colored blue (dark gray) while the cells in hypoxia are marked by shades of green (light gray). Necrotic cells are black.

4.3 Possible improvements

To exploit the full power of multiprocessor system, the second level parallelism could be introduced based on message-passing MPI interface. However, it would make the code extremely complicated and rigid for improvements. This could also extend the time for implementation and tests. Moreover, running the code on the large number of CPUs is usually restricted by system administrators and consumes much time and money. So, having in mind the shift in modern chip technology towards production of multiple-core CPUs (empowered by GPU) we decided to meet

this trend implementing the code open for both future improvements in the model and technological progress.

Tuning the code for GPU architecture seems to be much better idea. We can expect that the computations can be considerably accelerated. We have implemented on Nvidia GPUs in CUDA environment the procedure of calculation of particle-particle interactions, which exploits brute force algorithm combined with Hockney's one. Proposed algorithm was tested on two CUDA-enabled devices: GeForce GT 330M and Tesla C1060. Their technical parameters are summarized in Table 1.

Table 1: The parameters of GPU devices

Device	GeForce GT 330M	Tesla C1060
Compute capability	1.2	1.3
Number of multiprocessors (MP)	6	30
Number of CUDA cores (8/MP)	48	240
Global memory	1023,3 MB	4095,8 MB
Clock frequency	1,26 GHz	1,296 GHz

Table 2: Speed-up calculated against CPU algorithm employing four threads and running on Intel i5 M 430.

Number of particles	GeForce GT 330M	Tesla C1060
128	6,42	31,37
3072	6,12	31,77
15360	6,04	31,21

In the algorithm used for forces calculations we divide the computational box on reference boxes in which we calculate interparticle distance arrays using Hockney algorithm. We apply data structure consisting of independent vectors representing particles (storing particle position, velocities, forces, and attributes) which number depends on the average number of particles placed inside a single reference box. We increase the size of reference boxes to check the algorithm stability, having in mind that the computation of the vessel-vessel interactions requires larger Hockney sub-boxes (see Figure 9). The computational time increases with square of the number of vectors, however, in our model, the number of particles in reference boxes will be not larger than 200.

As shown in Figure 12, GPU implementations of particle-particle procedure are order of magnitude faster than their multi-thread CPU counterparts implemented in OpenMP environment. In Table 2 we present the speed-ups obtained for the particle-particle procedure (in 3-D) implemented in CUDA implementation versus its MPI four-thread implementation on Intel i5 M 430. We can observe that the speed-up is very stable and does not depend on the number of the vectors processed, i.e., no degradation of computational efficiency is observed for increasing size of the cut-off radius. This result is very encouraging.

However, the comparisons are made for the same algorithm employing in both environments. The confrontation with the algorithm presented in Section 4 involves tuning its GPU version to the requirements of the whole model and real simulation conditions. We plan to implement soon this algorithm in our CxA model of tumor dynamics. We expect to obtain the GPU/CPU speed-up at least 10 comparing to the CPU algorithm presented in section 4.1. Such decrease of computational time will enable to perform interactive simulations and visualizations of same types of tumors (e.g. melanoma) in angiogenic phase. That is, the tumors of sizes of up to 1 cm in diameter can be interactively visualized.

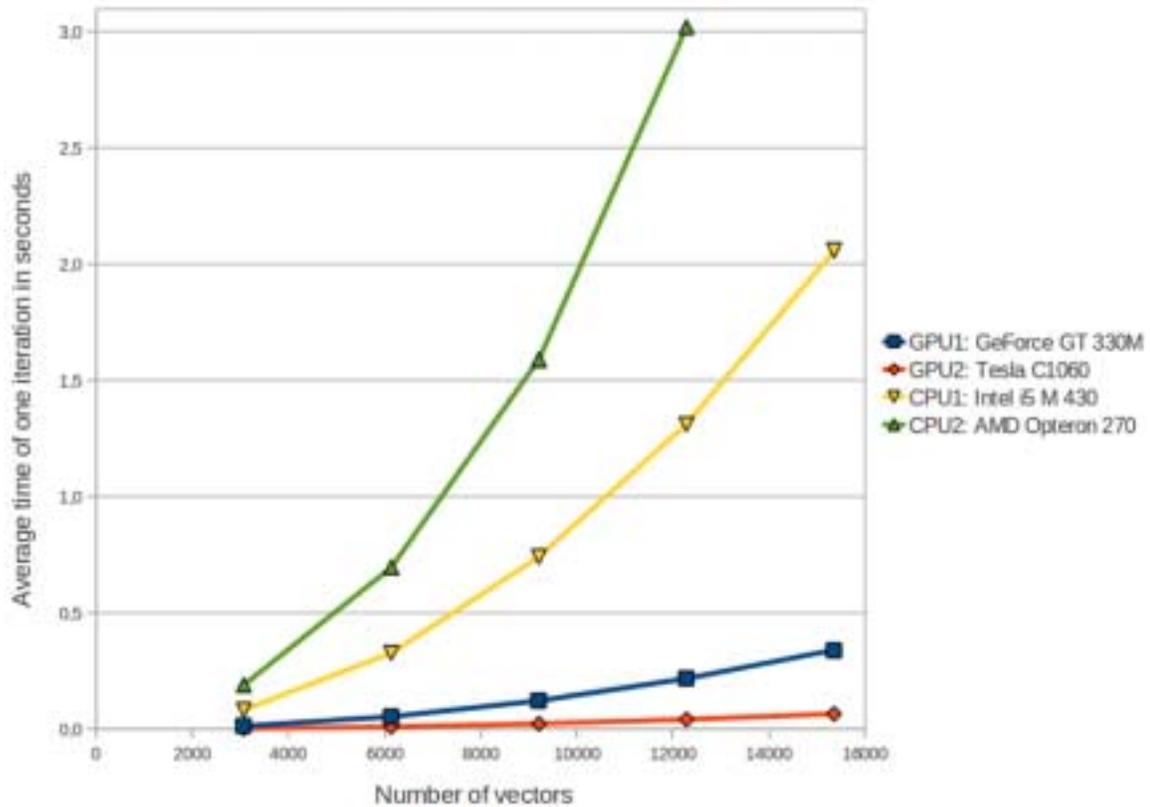


Figure 12 Timings obtained for two versions of particle-particle procedure: one optimized in CUDA and the second in OpenMP environment.

5. The system implementation

In Figure 13 we present our vision of a stand-alone system for interactive visualization of tumor dynamics. The system consists of the parallel implementation - both on CPU and GPU - of the CxA based tumor model and a flexible, user-friendly, interface (see Figure 14).

The simulation program allows several types of tissues to be simulated. Each type of tissue (soft tissue, bones etc.) can be described by the set of characteristic features such as hardness, density, average size of cells, the rate of diffusion of particular substances (oxygen, TAF), oxygen requirements, life span, resistance to oxygen deficiency, etc. In our system we use predefined sets of data for selected types of tissues, so it is not necessary to set all the parameters every time. These sets are prepared earlier by the specialists on the basis of biomedical data.

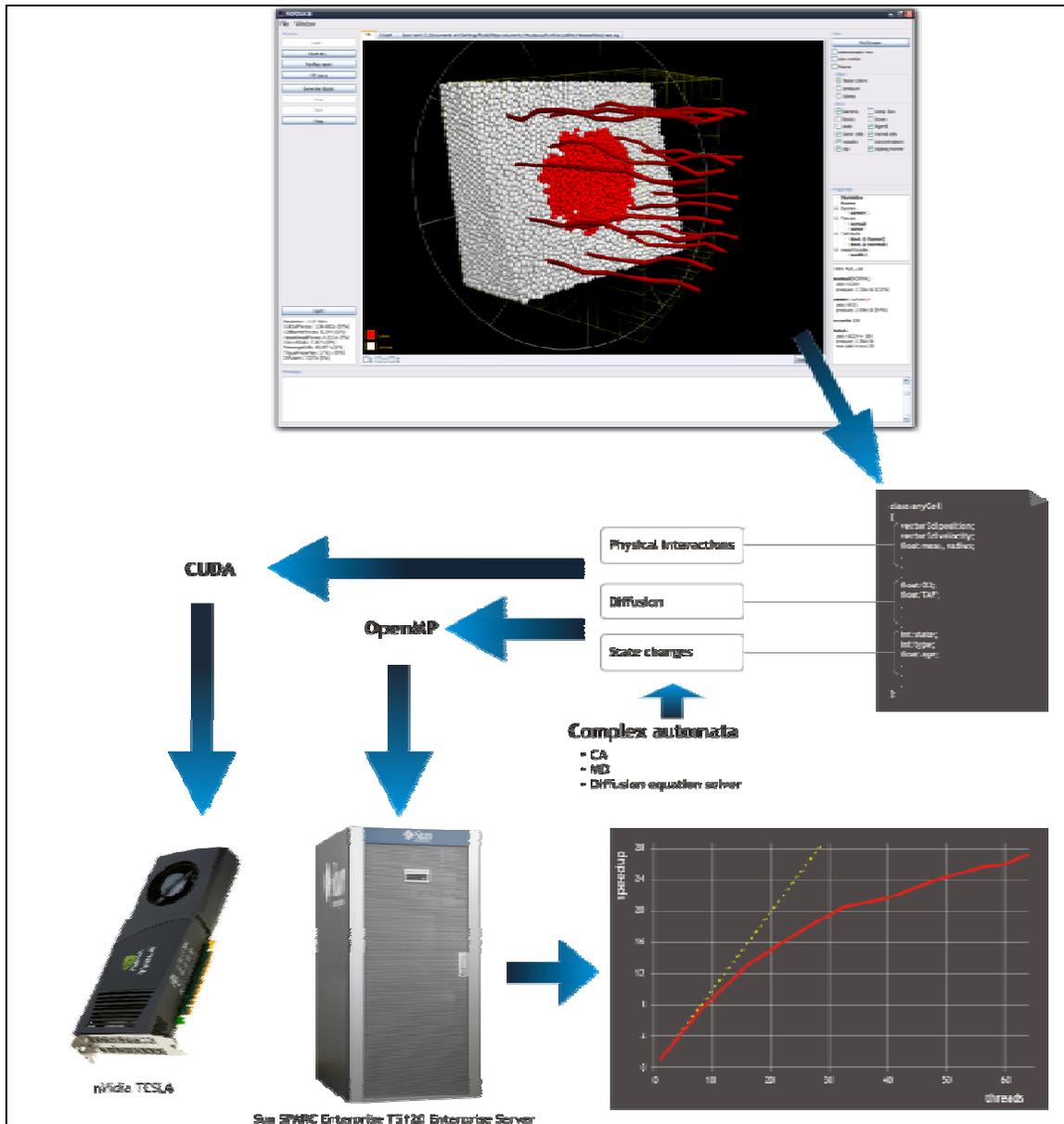


Figure 13 The concept of the interactive tool for visualization of tumor dynamics.

The interface, displayed in Figures 14 and 15, enables to position the tissues as well as adjust their size, location and space orientation facilitating the creation of initial simulation scene. For example, we display in Figure 14 a typical situation when a cancerous tissue is surrounded by healthy cells and the two are interwoven with the network of blood vessels. Such the operations are mostly performed with the help of a computer mouse and a few keyboard shortcuts and function keys. The interface allows to the user both rendering the scene and watching the simulation from various angles, zoom in and zoom out, rotate etc. Moreover, it makes possible to observe an arbitrary intersection of the evolving tumor and surrounding tissue. This way the interface allows for observing tissue interior for 3-D simulations.

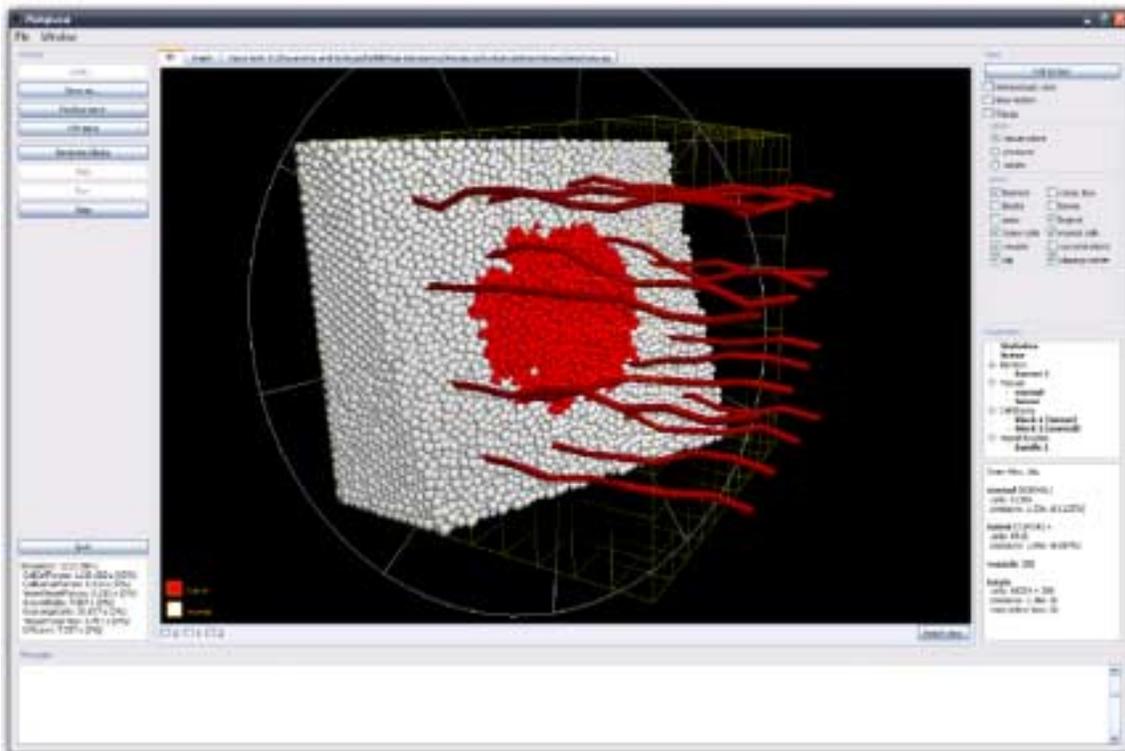


Figure 14 The screenshot of the interactive system interface.

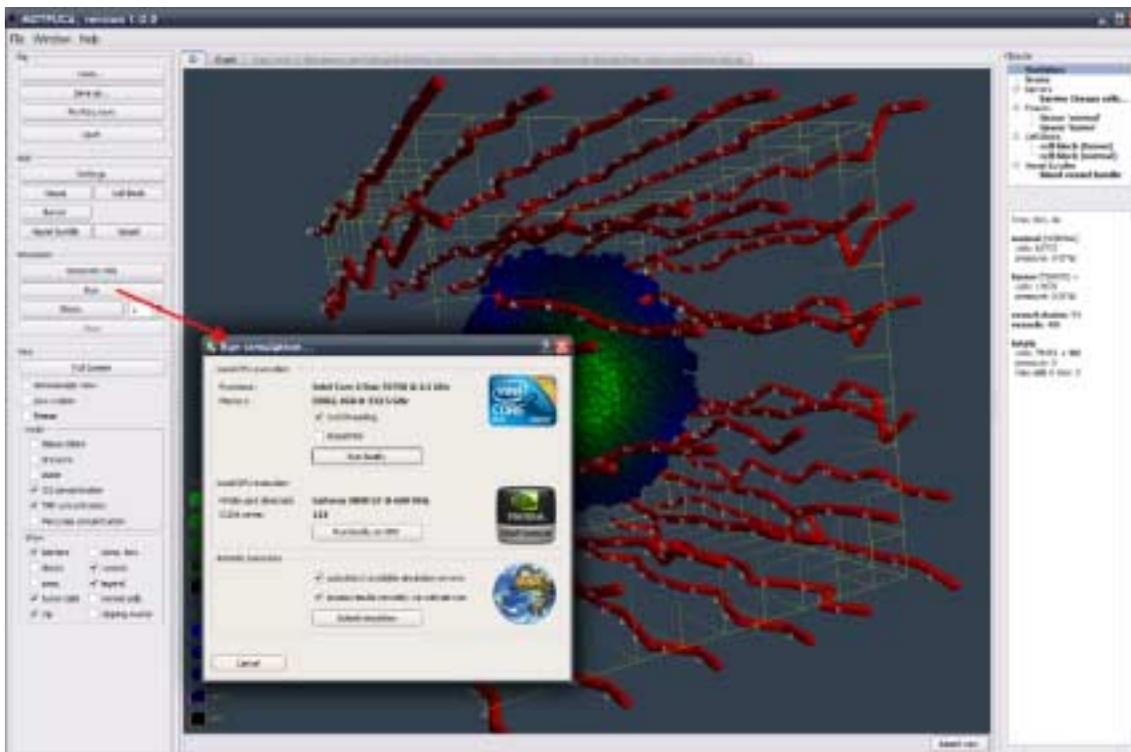


Figure 15 Different modes of system run: CPU OpenMP or GPU CUDA.

During both initiation stage and in the course of simulation, each model parameter can be modified interactively. Moreover, the simulation conditions can be changed anytime, e.g., some parts of tissue can be removed and/or added. This functionality is very important for examination the tumor dynamics after surgical intervention. The modified simulation scenario might also be saved as a template for the following simulations.

The modeling can be carried out in two or three dimensions. However, though 3-D simulations are more realistic, they are more demanding computationally. It can be easily estimated that assuming the same number of particles, 3-D simulation will be at least three times slower than its 2-D counterpart. As soon as the simulation scenario and initial conditions are defined, the system is ready to run. It might be run either on a multi-core CPU with the shared memory - then the simulation program uses the OpenMP libraries – and, in the nearest future, on GPU devices in CUDA environment. This warrants some degree of sustainability just in case when one technology appears to be superior over the other. As shown in Figure 15, the proper version can be selected at the beginning of modeling experiment.

Thanks to parallel realization, the system allows for interactive, fluent visualization of tumor dynamics for particle ensembles consisting of hundred of thousands of particles. If the simulation is performed on-line (Figures 14 and 15), the program displays information about particular tissues (e.g. the number of cells, the number of cells in various states, pressure, O₂ concentration). Larger models can be also simulated in batch mode, producing movies and diagrams.

6. Validation tool

The microvascular density (MVD) is the parameter describing structural properties of blood network used from years as an important descriptor in cancer therapy [Eberhard, et. al, 2000]. However, MVD value depends on space position, it is strongly case dependent and, moreover, it gives very poor knowledge about the topology of the vasculature. So, it cannot be used alone as a “fingerprint” describing current state of cancer dynamics. A quantitative understanding of the mechanism of tumor dynamics involves more sophisticated, universal, space and case invariant descriptors representing the most relevant topological features of tumor vasculature.

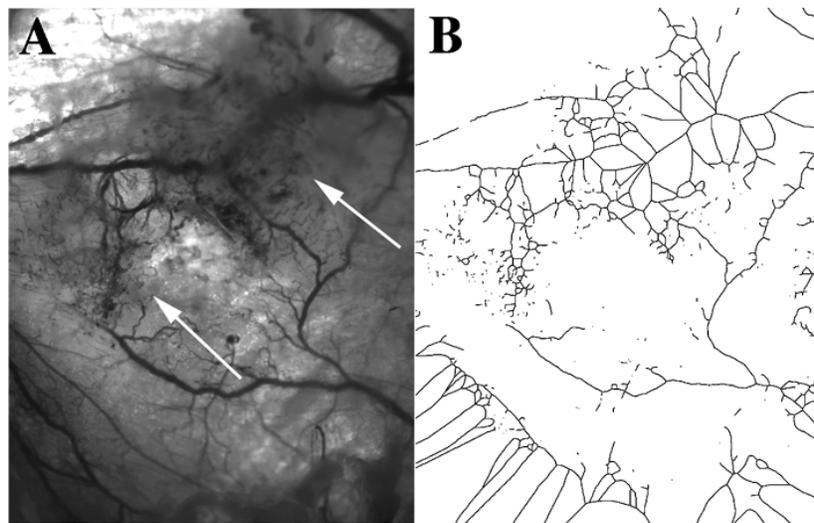


Figure 16 The result of skeletonization of a picture displaying a blood network surrounding tumor mass.



Figure 17 The result of skeletonization of a picture displaying a vascular network in healthy adipose tissue [Parish, 2003].

The knowledge about topology of biological networks can play a significant role in understanding processes taking place in tissue and organs (see e.g. [Albert and Barabasi, 2002; Girvan and Newman, 2002]). As shown in [Newman, 2003], investigation of network topology using graph descriptors can help in understanding the behavior of the entire system but also to evaluate models by comparing results of *in-silico* experiments with *in-vivo* ones. The structure of a network encodes a number of global and local information, which can be extracted by dedicated measures and used in further analysis. Quantitative evaluation of such the network properties as connectivity, symmetry, ability to transfer signals or to form node clusters can reveal qualitative features of underlying complex system. In addition, local information e.g. vertex centrality or vertex clustering coefficient enables to find sub-networks playing the most significant role in the whole system and allows for grouping functionally similar nodes. The analysis of vertex or edge-features distributions can also bring insight into system-level characteristics. Some general complex network descriptors are collected in Table 3.

Table 3 Selected graph descriptors available in *Graph Investigator* application. The descriptors denoted by asterisk can be computed efficiently using CUDA kernels.

Descriptor	Remarks
Randić Connectivity Index	$\chi(G) = \sum_{(u,v) \in E} (k_u \cdot k_v)^{-\frac{1}{2}}$, where k_i denotes the degree of a vertex i . Connectivity measure derived from chemical graph theory
General Connectivity Index	${}^l\chi(G) = \sum_{P_l \subseteq G} \left(\prod_{w \in V_{P_l}} k_w \right)^{-1/2}$, where P_l denotes path of length l and V_{P_l} all vertices that belong to this path
Zagreb Index M_1	$M_1 = \sum_{v \in V} (k_v)^2$
Zagreb Index M_2	$M_2 = \sum_{e \in E} w_e$, where w_e is a weight of edge e , for unweighted graphs $w_e = 1$

Modified Zagreb Index ${}^{(m)}M_1$	${}^{(m)}M_1 = \sum_{v \in V} (k_v)^{-2}$
Modified Zagreb Index ${}^{(m)}M_2$	${}^{(m)}M_2 = \sum_{e \in E} (w_e)^{-1}$
Total Adjacency Matrix	$A(G) = \sum_{v \in V} k_v = 2 E $
Modified Total Adjacency Matrix	${}^{(m)}A(G) = \sum_{v \in V} (k_v)^{-1}$
B Index Ξ^*	$B(G) = \sum_{v \in V} \frac{k_v}{d_v}$, where d_v is vertex distance for vertex v
Vertex distance	Sum of distances between v and all other vertices from a graph G . $d_u = \sum_{u \in V} d(u, v)$
Density of edges	$den(G) = \frac{2m}{n(n-1)}$, where $m = E $
Information of vertex degrees	$I_{vd} = \sum_{v \in V} (k_v \log_2 k_v)$ Reflects connectivity and topological complexity in terms of number of branches, cycles, cliques, etc.
Radius of graph Ξ^*	$r(G) = \min_{v \in V} \epsilon_v$, where ϵ_v is eccentricity of vertex v
Vertex eccentricity Ξ^*	Maximum distance between vertex v and any of the remaining graph vertices $\epsilon_v = \max_{u \in V} d(v, u)$
Graph diameter Ξ^*	$diam(G) = \max_{u, v \in V} d(u, v)$ Reflects density of graph connections, achieving its maximal value for paths and minimal for cliques.
Total Walk Count	Counts all paths of all lengths in the graph and depends on the size, cyclicity, and branching of the graph quantifying property called <i>labyrinthicity</i> .
Efficiency Ξ^*	$E(G) = \frac{1}{n(n-1)} \sum_{u, v \in V, u \neq v} \frac{1}{d(u, v)}$ Measures the traffic capacity of a network and reflects its parallel-type transfer ability.
Heat Content, Heat Content Coefficients Ξ^*	Parameterized descriptors based on heat kernel matrix.
Closeness Ξ^*	Mean distance to each other vertex
Betweenness	Measures relative importance of a vertex in

	a shortest-path transfer through graph edges.
B-Matrix descriptors \mathbb{B}^l	Graph B-matrix encodes information about distribution of l -order degrees of graph vertices. As l -order degree of a vertex v we understand the size of the set of all vertices located at distance l from a vertex v .

To scrutinize if the vascular networks generated in numerical simulations are really similar to the realistic ones we plane to validate our model using complex network descriptors [Newman, 2003]. This validation can be conducted on the basis of the comparison between realistic images of tumor vascular networks from confocal microscopy and computer experiments. As shown in Figures 16,17 the vasculature topology can be extracted using skeletonization filter and consequently transformed to a graph. Subsequently, the graph can be described by the feature vectors with statistical and/or algebraic descriptors of complex networks [Newman, 2003; Czech et al., 2011] as the feature vector components. Finally, pattern recognition and machine learning methods can be used for the vector classification and hypotheses formulation. This situation is sketched in Figure 18.

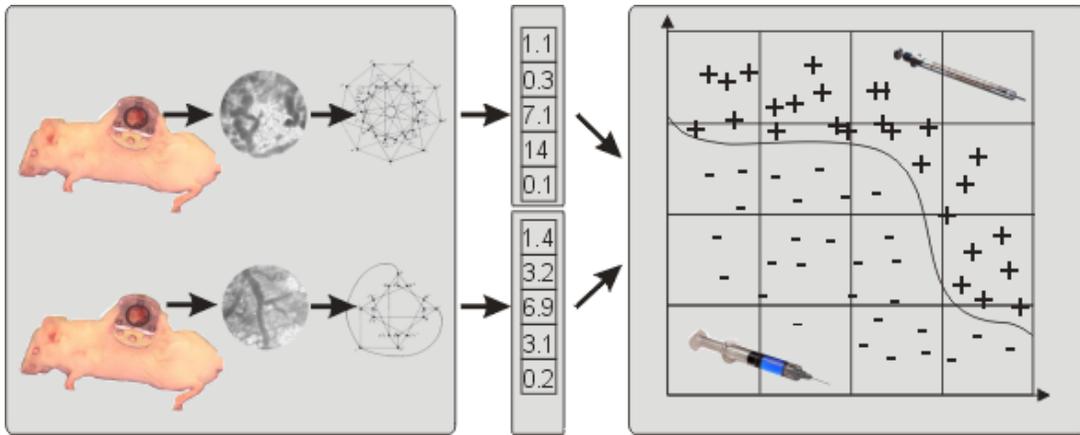


Figure 18 The diagram showing the process of feature vector generation on the basis of graph descriptors of tissue vasculature and hypotheses generation using machine learning tools.

Promising application of graph matching algorithms arises as far as inter-regional variation of vasculature in tumor is considered. Quantifying relations between vascular networks in different types of tumors can also bring valuable conclusions. For example, by comparing topologies of blood vessels in different metastatic tumors of the same type we might test hypothesis about their similarity.

We have recently developed [Czech et al., 2011] *Graph Investigator* - a robust programming package - which is capable of capturing topological features of networks with the use of various descriptors derived from graph theory. The screen shot of the interface is shown in Figure 20. The set of available graph descriptors includes over eighty statistical and algebraic measures. It allows for performing both inter-network comparisons and to analyze global and local structural properties of networks on the basis of diverse criteria. Furthermore, it allows for quantifying

inter-graph similarity by embedding graph patterns into low-dimensional space or distance measurement based on feature vectors.

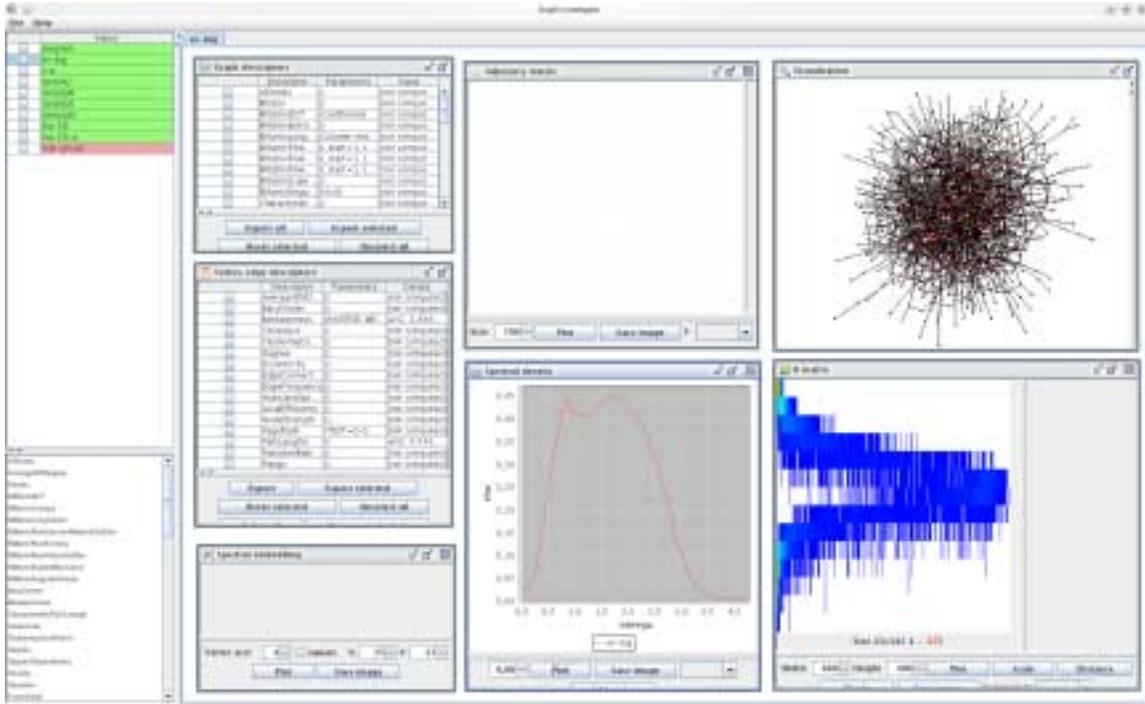


Figure 19 The main panel of *Graph Investigator*. The internal windows show visualization and computational modules of the application.

To supply the user with validation tool we plan to integrate *Graph Investigator* application with the tumor modeling simulator. The simulator is written in Java and much of descriptors are computed exploiting CUDA environment on GPGPU Nvidia devices. Recently, the efficient GPU implementations of BFS (Breadth-First Search) and all-shortest-paths algorithms were presented [Luo, 2010; Tran 2010]. We use CUDA implementation of BFS algorithm to obtain distance matrix for a graph that form the basis for computation of several graph descriptors such as efficiency or graph diameter. The CUDA kernels are invoked from *Graph Investigator* through *Java Native Interface*. The graph descriptors that can be computed with a help of GPU are marked in Table 3 with asterisk. Figure 20 displays comparison of computation times of graph diameter for CPU and GPU implementations. The test was performed on random Erdős-Rényi graphs with $p=0.001$ (probability of edge existence). The graph density grows with a graph size. The execution times presented in Figure 20 was averaged over 100 instances. For graphs of size 1000, the GPU implementation is 8 times faster.

Using both applications i.e. simulator and validation tool, it will be possible to observe, analyze and control on-line vasculature evolution using network descriptors. The multidimensional feature vector can be visualized using multidimensional scaling or principal component analysis tools. This way one can control qualitative changes in tumor dynamics [Topa and Dzwiniel, 2009] and match interactively modeling parameters to experimental results. The example of embedding

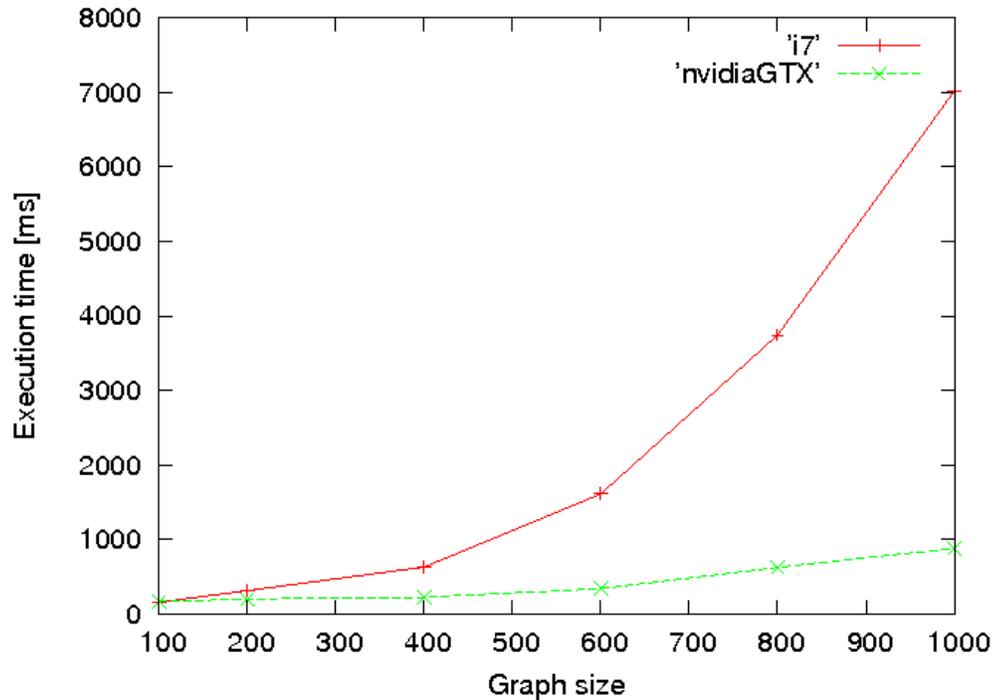


Figure 20 The comparison of computation times of graph diameter descriptor for Erdős-Rényi graphs with edge existence probability $p=0.001$ and sizes varying from 100 to 1000. The red line corresponds to CPU execution of BFS algorithm (Intel Core i7-920). The green line - CUDA implementation of BFS (Nvidia GTX 280).

6. Summary and conclusions

We have discussed a general vision of the system for interactive simulation and visualization of tumor dynamics and its ready-to-use components. It is intended as educational and research tool for oncologist and clinicians which could be deployed on a small but strong stand-alone workstations. In the future, such the tool could be used for planning cancer treatment.

The system is based on the concept of Complex Automata, which combines particle method and Cellular Automata modeling techniques. We show that the complex automata paradigm can be used as a framework for developing realistic models of tumor growth as a result of emergent behavior of many interacting cells. Successful and realistic reconstruction of mechanical interactions between proliferating tumor, healthy tissue, and evolving vascular network is the most important advantage of the CxA framework over other modeling paradigms.

The CxA model has many limitations. For example, it addresses a limited number of biological processes. Some of them, such the blood flow, received less attention than in other approaches. We did this intentionally to reduce the computational load. On the other hand, CxA model of tumor growth can be easily extended by implementing more precise sub-models of all the processes – known and unknown – responsible for tumor proliferation. However, the improvements by including more detailed processes and developing the particle model on its own (as it is proposed in section 3.2) can be undertaken provided that the efficient parallel version of the model will be implemented allowing for interactive visualization and simulation. We have demonstrated that due to parallelization of the model using OpenMP environment on modern

multiplecore CPUs we can improve computational efficiency almost one order of magnitude. We show that similar factor is achievable using GPGPU devices. This way, relatively large tumors can be modeled.

We expect that just technological progress simultaneously with improvements in parallel implementation of the model will allow for increasing of both the size of tumors simulated and the accuracy of the results obtained. This is the reason that the system is constantly being tuned to nowadays computational and visualization facilities allowing for considerable increase of simulation efficiency.

In the paper we also describe the interface that will be employed for setting up initial conditions and to assure interactive visualization of tumor dynamics. Such the user-friendly interface is particularly important for preparing the initial simulation scenario of tumor growth, which requires handling with hundreds of parameters.

The system will be developed in accord to both ICT technology development and the increasing knowledge about reasons and processes involved to cancer growth. The following versions of the system will be replenished, on the one hand, with more and more detailed microscopic and macroscopic processes and, on the other, its better numerical realization and computer implementation.

To supply the user with validation tool, we will integrate the tumor growth simulator with *Graph Investigator* - ready-to-use data mining tool for interactive analysis of topological structure of tumor vasculature. Such an integrated simulation/data analysis tool would allow to do research on cancer development undoubtedly more effectively due to faster search of parameters domain and hypothesis verification.

Acknowledgements

This research is financed by the Polish Ministry of Higher Education and Science, project N519 579338 and partially by AGH grant No. 11.11.120.777. We would like to thank Nvidia Company for support and donating Authors with Tesla C1060 GPU. The Authors are also very grateful to Professor Dr Arkadiusz Dudek MD PhD from University of Minnesota Medical School, Division of Hematology, Oncology, and Transplantation, Department of Medicine, and Mr Piotr Pawliczek from AGH Institute of Computer Science for cooperation in designing algorithms and preparing timings on GPU processors.

Bibliography

1. Alarcon T, Byrne H, Maini PK (2005) A multiple scale model for tumor growth. *Multiscale Model Simul* 3:440–475
2. Albert R, Barabasi A. (2002), *Statistical mechanics of complex networks*. *Reviews of Modern Physics*, 74, 47-97, 2002.
3. Amyot F, Small A, Gandjbakhche AH (2006) Stochastic modeling of tumor induced angiogenesis in a heterogeneous medium, the extracellular matrix. In: *Proc 28th IEEE EMBS Annual International Conference New York City, USA, 30 Aug- 3 Sept 2006*
4. Bauer AL, Jackson TL, Jiang YA (2007) Cell-based model exhibiting branching and anastomosis during tumor-induced angiogenesis. *Biophys J* 92:3105–3121
5. Bearer E L, Lowengrub J S, Chuang Y L, Frieboes H B, Jin F, Wise S M, Ferrari M, Agus DB and Cristini V 2009: Multiparameter computational modeling of tumor invasion *Cancer Res.* **69** 4493–501
6. Bellomo N, de Angelis E, Preziosi L (2003) Multiscale modeling and mathematical problems related to tumor evolution and medical therapy. *J Theor Med* 5:111–136
7. Chaplain MAJ (2000) Mathematical modelling of angiogenesis. *J Neuro-Oncol* 50:37–51
8. Czech, W., Goryczka S., Arodz, T., Dzwinel, W., Dudek, A.,Z., (2011) Exploring biological networks with Graph Investigator research application, accepted for publication in *Com[puting and Informatics*
9. Dormann S, Deutsch A (2002) Modeling of self-organized avascular tumor growth with a hybrid cellular automaton. *In Silico Biol* 2:393–406.
10. Darland DC, Massingham LJ, Smith SR, Piek E, Saint-Geniez M and D'Amore PA (2003) Pericyte production of cell-associated VEGF is differentiation-dependent and is associated with endothelial survival. *Developmental Biology* 264(1):275-288
11. Dzwinel W, Yuen DA, Boryczko K (2002) Mesoscopic Dynamics of Colloids Simulated with Dissipative Particle Dynamics and Fluid Particle Model. *JMol.Modeling*, 8:33-45
12. Eberhard A, Kahlert S, Goede V, Hemmerlein B, Plate KH, Augustin HG (2000) Heterogeneity of Angiogenesis and Blood Vessel Maturation in Human Tumors: Implications for Antiangiogenic Tumor Therapies. *Cancer Research* 60:1388.1393
13. Español P (1998) Fluid particle model. *Phys Rev E* 57:2930–2948
14. Folkman J (1971) Tumor angiogenesis: Therapeutic implications. *N Engl J Med* 285:1182–1186
15. Folkman J, Hochberg M (1973) Self-regulation of growth in three dimensions. *J Exp Med* 138:745–753

16. Girvan M., Newman MEJ. (2002) Community structure in social and biological networks, *Proceedings of the National Academy of Sciences*, vol. 99, page 7821
17. Godde R, Kurz H (2001) Structural and biophysical simulation of angiogenesis and vascular remodeling. *Dev Dyn* 220:387–401
18. Haile PM. *Molecular Dynamics Simulation*. Wiley: New York, 1992.
19. Hockney RW, Eastwood JW. *Computer Simulation Using Particles*. McGraw-Hill: New York, 1981.
20. Hoekstra AG, Lorenz E, Falcone LC, Chopard B (2007) Towards a complex automata framework for multi-scale modeling: Formalism and the scale separation map. *Lect Notes Comput Sci* 4487:1611–3349
21. Jemal A, Siegel R, Jiaquan Xu, Ward E, *Cancer Statistics 2010*, (2010) *CA Cancer J Clin*. 60:277-300
22. Kim Y, Stolarska MA and Othmer H G 2007 A hybrid model for tumor spheroid growth in vitro: I. Theoretical development and early results *Math. Methods Appl. Sci.* 17: 1773–98
23. Lee D-S, Rieger H, Bartha K (2006) Flow correlated percolation during vascular remodeling in growing tumors. *Phys Rev Let* 96:058104–1-4
24. Lowengrub J S, Frieboes H B, Jin F, Chuang Y-L, Li X, Macklin P, Wise S M and Cristini V (2010) Nonlinear modelling of cancer: bridging the gap between cells and tumours. *Nonlinearity* 23: R1–R91
25. Luo L., M. Wong, W. Hwu, An effective implementation of breadth-first search, *Proceedings of the 47th Design Automation Conference*, 2010.
26. Mansury Y, Kimura M, Lobo J and Deisboeck T S (2002) Emerging patterns in tumor systems: simulating the dynamics of multicellular clusters with an agent-based spatial agglomeration model. *J. Theor. Biol.* 219: 343–70
27. Milde F, Bergdorf M, Koumoutsakos PA (2008) Hybrid model of sprouting angiogenesis. *Lect Notes Comp Sci* 5102:167–176
28. Moreira J and Deutsch A (2002) Cellular automaton models of tumor development: A critical review. *Adv Complex Syst* 5:247–269
29. Matsumoto K, Nakamura T, Sakai K, Nakamura T, (2008) Hepatocyte growth factor and Met in tumor biology and therapeutic approach with NK4. *Proteomics*, 8, 3360–3370
30. Newman MEJ, (2003) *The Structure and Function of Complex Networks*, *SIAM Review*, 45/2,167–256
31. Parish C.R. *Cancer immunotherapy: the past, the present and the future*, *Immunology and cell biology*, 81, 2003.

32. Preziosi L (ed) (2003) Cancer modelling and simulation. Chapman & Hall/ CRC Mathematical Biology & Medicine 426
33. Stéphanou A, McDougall SR, Anderson ARA, Chaplain MAJ and Sherratt JA (2005) Mathematical Modelling of Flow in 2D and 3D Vascular Networks: Applications to Antiangiogenic and Chemotherapeutic Drug Strategies. *J Math Comput Model* 41:1137-1156
34. Stokes CL, Lauffenburger DA (1991) Analysis of the roles of microvessel endothelial cell random motility and chemotaxis in angiogenesis. *J Theor Biol* 152:377–403
35. Stolarska M A, Kim Y and Othmer H G (2009) Multiscale models of cell and tissue dynamics *Phil. Trans. R. Soc. A* 367 3525–53
36. Szczerba D, Lloyd BA, Bajka M, Szekely GA (2008) Multiphysics model of Myoma growth. *Lect Notes Comput Sci* 5102:187–196
37. Topa P., Dzwinel W., Yuen, D.A., (2006) A multiscale cellular automata model for simulating complex transportation systems, *Int. J. Modern Phys. C*, 17/10: 1437-60.
38. Topa, P., Dzwinel, W. (2009), Using network descriptors for comparison of vascular systems created by tumor-induced angiogenesis, *Theoretical and Applied Informatics*, 21/2, 83-94
39. Topa P (2008) Dynamically reorganising vascular networks modelled using cellular automata approach. *Lect Notes Comput Sci LNCS* 5191:494–499
40. Tran Q.N, Designing Efficient Many-Core Parallel Algorithms for All-Pairs Shortest-Paths Using CUDA, 2010 Seventh International Conference on Information Technology, 2010.
41. Vasilyev OV (2003) Solving multi-dimensional evolution problems with localized structures using second generation wavelets. *Int. J. Comp. Fluid Dyn.*, Special issue on High-resolution methods in Computational Fluid Dynamics, 17(2): 151–168
42. Wcislo R, Dzwinel, W (2008) Particle based model of tumor progression stimulated by the process of angiogenesis, *Lectures Notes in Computer Science, ICCS 2008, LNCS* 5102:177-186
43. Wcislo R, Dzwinel W, Yuen DA., Dudek AZ, (2009) A new model of tumor progression based on the concept of complex automata driven by particle dynamics. *J. Mol. Mod*, 15/12: 1517-1539.
44. Wcislo R, Dzwinel W (2010a) Particle Model of Tumor Growth and Its Parallel Implementation, *Lecture Notes in Computer Science, PPAM, Wrocław*, 13-16 September 2009, LNCS: 322-331
45. Wcislo R, Gosztyla P, Dzwinel W (2010b) N-body parallel model of tumor proliferation, *Proceedings of SCS Summer Computer Simulation Conference 2010*, July 11-14, 2010, Ottawa, Canada