Regenerative Systems - Challenges and Opportunities for Modeling, Simulation, and Visualization*

Adelinde M. Uhrmacher, Hans-Jörg Schulz,
Heidrun Schumann, Lars Schwabe, Dirk Timmermann
University of Rostock, 18059 Rostock, Germany,
{adelinde.uhrmacher,hans-joerg.schulz,
heidrun.schumann,lars.schwabe,dirk.timmermann}@uni-rostock.de

ABSTRACT

Regenerative systems are able to overcome significant perturbations, and maintain autonomously their functionality in dynamic and uncertain environments. More and more this ability of biological systems plays a role in designing technical systems, e.g., in sensor networks, as well. Important properties of regenerative systems are their dynamic structures and their operation on different spatial and temporal scales. Those propel the development of new modeling, simulation, and visualization methods. Among them, variants of the π-calculus formalism, a portfolio of Gillespie related spatial simulation algorithms, means for automatically configuring simulators, and the integrated visualization methods, that make use of innovative layouts and linked and coordinated views target challenges in analyzing biological regenerative systems. They provide a basis for analyzing regenerative systems in general by means of simulation.

1. INTRODUCTION

More and more computer systems are required to act independently, flexibly, and autonomously [18]. The agent metaphor has helped nurturing systems that work in a distributed and decentralized manner and adapt themselves to changes in their environment [54]. This ability of systems is often subsumed under the term self-X, e.g., self-organizing, self-optimizing, or self-healing. Self-healing or the ability to regenerate refers to the ability of a system to maintain its functionality in spite of significant perturbations from its environment and faults in its components. In the ideal case regeneration implies not only compensation but restoring functionalities. In contrast to other self-X concepts, it also bears in itself its goal and with it a measure of success. However, up to now it is still not clear how to achieve this ability and it appears natural to turn to biological systems for inspiration as regeneration appears as a salient feature of those.

VALUETOOLS 2009, October 20-22, Pisa, Italy

DOI 10.4108/ICST.VALUETOOLS2009.7907

Biologically inspired computing seeks to adopt specific mechanisms of biological systems to instill flexibility and autonomy, or the ability to regenerate, into the designed systems. It appears natural to use an experimental technique like computer simulation for a better understanding and evaluation of these approaches. Computer simulation, as dry-lab experimentation, is also on its way toward establishing itself as a method of scientific investigation in molecular and cell biology, which becomes particularly apparent in the field of systems biology [27]. As a highly dynamic field, systems biology is fast in adopting concepts from other fields, including computer science and engineering [46].

In traditional computer science, simulation is an established method for evaluating the performance, and gaining insight into the scalability and robustness of systems. However, despite its wide use, there is strong evidence that computer scientists should not only do more experiments but also do better ones, e.g., [51, 39]. Unlike biology, computer science does not see itself as an experimental branch of science, and most experiments constitute rather proofs of concept than thorough experimental evaluation studies. However, the more autonomous computer science systems become the more a shift in perception appears overdue.

Modeling and simulation forms a common and uniting ground advancing research on both technical and biological regenerative systems and facilitating the transfer of concepts between both. Important properties of regenerative systems are their variable structures and their operation on different temporal and spatial scales. Due to these properties, off-the-shelf modeling and simulation tools and methods will hardly suffice. This leads to new modeling and simulation methods, among which, due to the complexity of this type of systems, visualization methods play a central role [37].

In the following we will explore some of the challenges that regenerative systems provide for modeling, simulation, and visualization methods, and hopefully also scratch the potential these methods bear as a common platform for advancing work on regenerative systems in general. We will use methods developed for regenerative biological systems to illuminate the challenges and opportunities that regenerative systems offer. First let us shortly discuss aspects of regeneration by examples taken from biology and computer science respectively.

2. REGENERATIVE SYSTEMS

As already mentioned regeneration is a salient feature of biological systems. Variable structures and the operation on different temporal and spatial scales are central properties of these systems. Over the last decades these properties also gained importance in technical systems. The agent metaphor permeates many current software designs, and with this the need for variable structure modeling in realizing self-X systems. Distributed ad-hoc computing and sensor networks are examples, where also the second property of regenerative systems, i.e., the operation at different spatial and temporal scales, plays a major role. To achieve regeneration, whether in terms of restoring functionality or in terms of "simple" compensation, requires further features.

Technical systems are increasingly required to manage themselves fully autonomously, following high level objectives. This implies distributed computational environments that shall work with often limited resources of bandwidth, energy, and processing capability and shall be able to overcome perturbations and faults in their components. Biological systems have demonstrated to be efficient in dealing with many of these problems. Therefore, significant interest in identifying and applying suitable biological concepts exist, particularly in the area of sensor networks.

2.1 Biological cellular systems

Differentiation of cells is the process by which cells specialize. This implies that a cell's pattern of behavior, composition, and interaction typically changes. Signal transduction pathways are essential processes for converting a signal or stimulus into another, and for initiating and controlling the differentiation. Among those the canonical Wnt pathway is assumed to play a key role in differentiation of neural progenitor cells and thus for the regeneration of the brain (see 2.2).

The main outcome of the Wnt/β-catenin signaling pathway is an increase of the amount of β-catenin in response to Wnt molecules binding to receptors at the outside of the plasma membrane. In the absence of Wnt, a degradation complex efficiently reduces the amount of β-catenin. In the presence of Wnt, the degradation complex gets deactivated and consequently, due to its constant production, the amount of β-catenin increases. This affects the regulation and transcription of genes and accompanies the cells in their specialization (differentiation) process. Although formal models of the Wnt pathway exist, e.g., [30], many details of the Wnt signaling pathway are still not well understood. They motivate further work, e.g., analyzing the impact of cell-specificity, compartment specific dynamics, and cell cycle on the Wnt signaling pathway [33].

2.2 Biological neuronal networks

Biological neural networks are a prime example of systems with variable structure. Neurons, which are cells already undergone a cell differentiation process, communicate with each other via electrical discharges, so-called action potentials or 'spikes' (all-or-none binary signals). The connectivity between neurons depends on the history of these spikes. The communication between two neurons is realized via a synapse, and in terms of propagating the electrical signals the communication is uni-directional. One can distinguish between a presynaptic ('sending') and a postsynaptic ('receiving') neuron (Figure 1a). In the neocortex of mammals, a single neuron may make connections to approx. 10000 presynaptic neurons, where the signals integrated from the presynaptic neurons determine as to whether a spike is generated in the postsynaptic neuron. It has been hypothesized by Hebb [16] that the strength of a synapse between two neurons is increased, if the presynaptic neuron repeatedly contributes (in terms of sending spikes to it) in having the postsynaptic neuron emitting a spike. This way of altering the connections between neurons is referred to as *activity-dependent synaptic plasticity*.

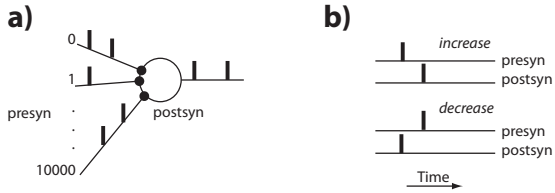


Figure 1: Illustration of neuronal communication and synaptic plasticity. a) A single postsynaptic neuron in the neocortex may integrate signals (all-or-none spikes, thick lines) from up to 10000 presynaptic neurons. b) The strength of a synaptic connection is increased if the presynaptic neuron sends a spike before a postsynaptic spike (upper row), whereas it is decreased if the presynaptic spike comes after the postsynaptic spike (bottom row).

Recently, it has been found experimentally [3] that the timing of spikes emitted by the pre- and postsynaptic neurons is crucial in shaping the connections between them (Figure 1b). In particular, it has been found that only when the presynaptic neuron sends a spike before a postsynaptic spike, the synapse is strengthened, whereas it is weakened when the presynaptic neuron sends a spike after a postsynaptic spike. Here, 'before' and 'after' refers to time-windows on the order of a few tens of milliseconds. This form of plasticity is referred to as *spike-time dependent plasticity* (see [36] for a review on models).

Activity-dependent synaptic plasticity in general, and spike-time dependent plasticity in particular, are believed to be the underlying mechanisms of learning, memory and the ontogenetic development of the nervous system. In the context of regenerative systems the latter is of primary interest. Biological neuronal systems are not regenerative in the sense of self-healing. Instead of generating new neurons to recover lost functions (like after a stroke), the remaining neurons dynamically reorganize by the mechanism of activity-dependent plasticity in order to compensate lost functions (like the motor control of limbs). During development, the nervous system establishes connections based on the spike activity of neurons, because not the detailed connectivity between neurons, but presumably only the corresponding algorithms to set it up are stored in the genome. In other words, the activity of biological neuronal networks, which is often driven by sensory stimulations from the environment, shapes the network connectivity, and the network connectivity shapes the network activity. During development these activity-dependent processes lead to a 'self-assembly' of the nervous system, and after perturbation they help to restore lost functions.

From a therapeutic perspective it is of interest to generate new neurons by controlling the process of cell differentiation and to make use of activity-dependent synaptic plasticity in order to integrate them into damaged parts of the nervous system, thereby helping the nervous system to become regenerative also in terms of self-

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907

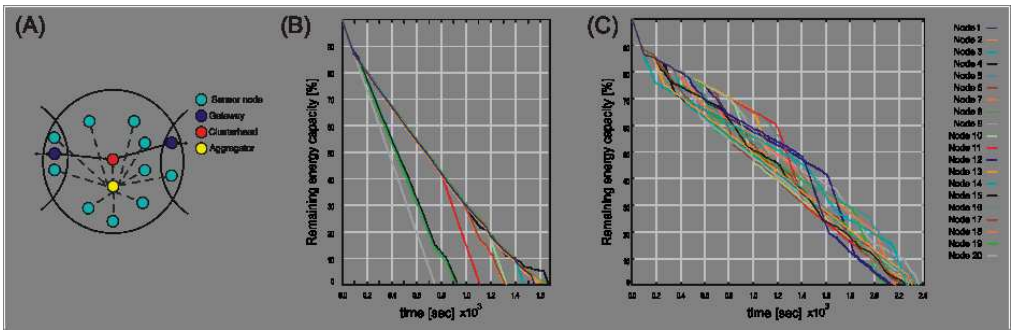


Figure 2: Prolongation of network lifetime by bio-inspired role changing [42]; (A) Example network cluster with different network roles, (B) Cluster energy consumption using reactive, centralized role change protocol, (C) Cluster energy consumption using proactive, bio-inspired and energy-aware role change protocol.

healing. However, given that spike-time dependent plasticity and the recovery of lost functions occur on vastly different spatial and temporal scales, new modeling and simulation concepts and techniques are needed, which can handle variable structures and different spatial and temporal scales.

2.3 Sensor networks

Wireless Sensor Networks (WSN) are composed of sensor nodes, which measure physical parameters in regions of interest, such as temperature, brightness, presence of objects etc., and send measured data to a data sink where the information is processed. As sensor nodes are left unattended after deployment, they are usually battery powered. Therefore, energy and corresponding computational power are most critical resources. One of the main components of energy consumption is wireless communication that is usually carried out by on-board radio transceivers. During radio signal propagation, the transmission power decreases proportionally to the square of distance or worse. For nodes with limited transmission power, transmission range as well as bandwidth and, hence, communication capabilities are also highly constrained. Implementation goals of large wireless sensor networks are quality of service, i.e., functionality and coverage of all relevant parameters, maximized lifetime despite limited batteries, and a well-balanced compromise between accuracy and effort. On the other hand, limited resources increase the probability of failures due to node errors, wireless communication perturbations, or directed attacks. Failures may yield wrong information about local physical values or may lead to completely unreachable and paralyzed network regions. Therefore, biologically and socially inspired principles have been thoroughly researched since some time to improve WSN behavior even under critical circumstances. Some principles with their technical applications impact will be briefly described.

After deployment and for reorganizing after faults, WSN benefit from self-organization by setting up a proper wireless interconnection structure, for example a flat mesh or a hierarchically layered network. In [45] the authors present a simple and energy efficient, fully decentralized procedure using just local rules to set up a clustered network from scratch. The resulting clustered network is called scale-free and obeys a Power law known from biology and sociology (small world phenomenon) [29, 1], yielding inherent favorable properties like robustness and energy efficiency. Self-adaptation allows WSN to perform even under dynamically varying environment conditions as well as in case of node and communication faults. A flock of birds typically rotates the most strenuous leading position under the strongest birds. In case of WSN, cluster heads receive sensor data from all cluster members and route aggregated data to the central gateway using neighboring cluster heads. Cluster heads thus dissipate their energy quite fast. Role changing between nodes depending on remaining energy level as proposed in X-Leach [15] and in [42] balances the energy burden among all cluster members and prolongs network lifetime considerably as depicted in Figure 2, which shows the results of simulation experiments.

Error resilience in biological systems is often based on mechanisms of self-healing or compensation. In case of severe faults most often not the whole organism fails, but just a functional degradation is experienced. The term graceful degradation has been coined for this behavior. In our technical context, a network must be able to identify and dismiss failing nodes. Neighboring or hibernating redundant nodes adopt their roles. Algorithms like XGAF [45] and MASCLE [44, 43] are able to determine the minimum number of active nodes to assure coverage and dynamically activate hibernating nodes or route around network holes. Thereby, the induced reduction of a network's functionality is kept as small as possible.

3. CHALLENGES AND APPROACHES

The above mentioned properties of regenerative systems provide ample challenges for all areas in modeling and simulation. In the following we will focus on modeling, simulation (i.e., the execution of models), and visualization methods. The work we are reporting has been driven by requirements of systems biology and constitute parts of the modeling and simulation framework JAMES II [17].

3.1 Modeling

One of the characteristics of regenerative systems is their variability when it comes to structure. Structure refers to components, interactions, behavior, and interface equally. A central question is how these dynamic structures can be modeled in an appropriate manner, e.g., in the context of molecular or cell biology [4].

Most modeling formalisms assume a rigid model structure, e.g., DEVS [57]. DEVS supports a modular hierarchical construction of models by distinguishing between atomic and coupled models. Atomic models are equipped with input and output ports, a state, transition functions, a time advance function that determines the next time-triggered event, and an output function. Coupled models

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907

comprise other models as components and define the coupling between those, they do not have a behavior of their own. As atomic models do, coupled models interact with the environment by receiving and sending events via their input and output ports. Due to the requirements of application areas like regenerative systems, extensions have been developed to support variable structures models. E.g. the variant dynDEVS supports variable composition, interaction, and behavior, ρ-DEVS extends dynDEVS by adding variable ports and with them dynamic interfaces, ml-DEVS builds on ρ-DEVS introducing different levels of abstraction to support multi-level modeling [53]. Thus, successively part of the rigidness of the original formalism DEVS has been overcome.

In contrast to DEVS, other modeling formalisms support variable structures from the outset. To those belongs the π-calculus [35]. The π-calculus is a model of concurrent computation and is based on the notion of naming. Names represent both interconnection links between active entities, called processes, and the data that these entities exchange through communication. Two concurrent processes can interact using a name they share the knowledge of: one process acts as a sender, the other as receiver. The message being transmitted is again a name, which the receiver henceforth knows and may use in further interactions. Thereby, networks with evolving connectivity can be described.

During the last decade, the π-calculus has seen quite a number of variants emerging. Many of those have been proposed for biological applications since the seminal work of Regev and Shapiro [41].

Space-π is a spatial extension of π-calculus supporting the representation of continuous space and molecular motions [23]. Concurrent processes are equipped with global positions and velocity vectors. The ability to communicate depends on the closeness of processes. The advantage of the Space-π approach is that intracellular structures and spatial effects can be represented in a very detailed manner. It is possible to build membranes, to introduce compartments, and active transportation processes. Even the impact of molecule sizes and shapes can be modeled. This makes the approach applicable to scenarios that are difficult to model with implicit representations of space, only. The operational semantics of Space-π is hybrid, i.e., combining continuous movement and discrete communication [47].

The idea that the ability to communicate depends on the current position of the processes and processes are equipped with further attributes, is generalized in Attributed π. Attributed π extends the π-calculus by attributed processes and attribute dependent synchronization. To ensure extendability, the calculus is parametrized with the language $\mathcal{L}$ which defines possible values of attributes [25]. Attributes subsume possible reaction rates and constraints as in higher order logic. The introduction of the language $\mathcal{L}$ avoids inventing completely independent calculi for the many reasonable choices of attribute values and constraints. Thus, depending on the language $\mathcal{L}$ Attributed π can express diverse compartment organizations. A non-deterministic and a stochastic semantics have been defined for Attributed π, where rates may depend on attribute values. Unlike Space-π which is hybrid, Attributed π is a discrete formalism.

Imperative π-Calculus is based on Attributed π and introduces imperative assignment operations to a global store. This forms a step toward multi-level modeling and simulation as information can be maintained and accessed at different levels. The extension allows for wide range of kinetics, e.g., including Michaelis Menten [33] and dynamic compartments with mutable configurations and variable volumes [24].

Developments as those from dynDEVS, over ρ-DEVS to ml-DEVS, and from the stochastic π-calculus, over Space-π, Attributed π to Imperative π-Calculus illuminate the driving force of cell biological applications in the field of modeling formalisms. Although these extensions have been designed for cell biological applications, they are applicable to the modeling of technical systems as well. In particular regenerative systems that exhibit similar characteristics, will benefit from these extensions. E.g., Space-π is currently being adapted for self-organizing sensor networks. DEVS variants that support variable structures have been applied to simulate ad-hoc networks evaluating decentralized strategies that utilize context information to assess the cooperativeness of other nodes [28].

3.2 Simulation

Operating on different scales in space and time is one of the characteristics of regenerative systems in general and biological systems in particular. Many intra-cellular dynamics, incl. the functioning of signaling pathways, depend on space, e.g., protein localization, cellular compartments, and molecular crowding. Thus, as already mentioned in the modeling section, spatial aspects, whether in a hybrid (Space-π) or a discrete manner (Attributed π or Imperative π-Calculus), are becoming of increasing relevance in systems biology. As the significance of stochasticity in intra and inter-cellular information processing is widely accepted, approaches that support both, i.e., stochasticity and space, are particularly promising, but also computationally expensive [49], which spurs on the development of new simulation algorithms and methods.

Algorithms commonly known under the term *stochastic simulation algorithms* (SSA) are based on sampling the Chemical Master Equation (CME): a partial differential equation describing the time evolution of the system's state probability distribution [14]. To address the problem of efficiency different strategies have been pursued: by introducing improved scheduling algorithms or data structures, [13], by trading accuracy for efficiency [6], by combining numerical integration and stochastic discrete event approaches [50], or by parallel and distributed simulation [5, 32].

One way of considering space within simulation is partitioning space into sub-volumes and extending the master equation with a diffusion term, which leads to the reaction-diffusion master equation (RDME). For sampling the RDME, several methods exist, to those belongs the Next Sub-volume Method (NSM) [8]. To increase their efficiency, parallel approaches have been developed as well [7].

The simulation costs increase further if one moves from the mesoscopic to the microscopic level. Microscopic algorithms operate with single particle detail, tracing the position of every particle and considering its individual features. While molecular and Brownian dynamics provide high accuracy, the effort required for position update and collision detection prevents simulating longer time periods or simulating models with many particles. Multi-resolution approaches offer a solution as they balance between accuracy and efficiency. E.g., by combining Brownian dynamics and the NSM [22] two different spatial levels are supported within a lattice. Macro elements are simulated at individual level. They affect the available space for reactions and diffusions at population level. Thus, phenomena like molecular crowding can be studied. According to our

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907

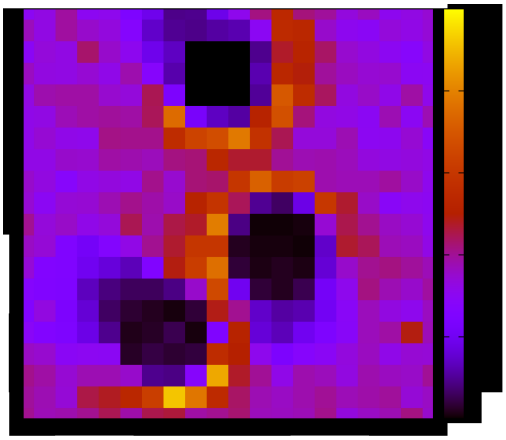


Figure 3: Impact of macro molecules on population propensities [21]

results [22], a decrease in the available volume can have a significant local effect on reaction propensities. The maximum propensity for single lattice cells was observed to be up to five times higher than the maximum in the dilute case (Figure 3).

Another impediment for a thorough evaluation of models lies in gaining access to sufficient computational resources. Whereas it is rather difficult to achieve a speedup in a grid-inspired environment with fine-grained parallel simulation [20], grid-inspired environments are very suitable for simulation replication and model parameter optimization, as those exhibit little or no data dependencies between portions of work [31].

The described methods give only a glimpse on recent activities in the area of simulation methods in systems biology. Similar problems imply similar solutions, thus solutions are quickly adapted in simulating biological and technical systems. The concept of multi-resolution simulation can also be found in network simulations, e.g., where network flows represented by fluids and packet-oriented flows are switched on demand [38]. As in systems biology, the goal is to cope with the calculation effort required and to balance efficiency and required detail in simulating those networks.

However, adopting concepts and cross fertilization between the application areas apply also in a broader manner. E.g., a new emphasis is being put on the necessity to equip formalisms with a clear operational semantics and relating this semantics to the simulation algorithm. How the simulator works is no longer guess work but subject to an unambiguous description related to the semantics of the formalism. Therefore, Phillips and Cardelli adopt the idea of Zeigler [57], to describe their SPIM simulator in terms of an abstract simulator [40]. One step further, the Attributed π simulator is directly rooted in the stochastic semantics and based on continuous time Markov chains, thereby forming a close bound between simulator and the formalism's operational semantics.

The insight that the performance of different simulation algorithms is influenced by a variety of factors, such as the model, sub-algorithms, and data structures, like event queues, is not new [11], however is newly fueled by the variety and quantity of recently developed simulation algorithms. In this context also the question of how performance evaluation of simulation algorithms can be done in an as little as possible biased manner, receives new attention.

Based on more than 40.000 simulation runs, the performance for a variety of SSA configurations for two models, i.e., the linear chain system and the cyclic chain system model, is tested [19]. According to the results, the execution speed of the direct reaction method (DRM) [14] and the next reaction method (NRM) [13] differs greatly between both models. The overhead of generating random numbers had less impact on SSA execution time than first assumed when the NRM was proposed [13]. However, not surprisingly the choice of the type of event queue influenced the performance of NRM significantly. This underlines that for efficiently simulating different models, different simulators, sub-algorithms, and data structures should be offered to be selected and configured on demand.

However, even if a simulation framework supports to plug in different simulators, sub-algorithms, and data structures, often simply too many possibilities exist for their combination, e.g., when evaluating exact and approximative SSA algorithms, 170 different sub-algorithm configurations were tested for one model instance [19]. In addition, performance might refer not only to speed, but to accuracy, memory consumption, or a combination thereof. Even for a simulation expert it is difficult to assess which of the many eligible *simulation configurations*, i.e., algorithmic set-ups, will deliver best performance in a concrete case. There are numerous ways to tackle this problem, e.g., by theoretically analyzing each simulation configuration, or by gathering performance data and applying data analysis and machine learning methods [9].

As already stated stochasticity plays an important role in biological systems and simulation replication is a necessity for all stochastic simulations. One way to improve replication efficiency is to ensure that the best configuration of the simulation system is used for execution. Even without any prior knowledge on simulator performance or problem instance, a highly efficient simulator can be configured online, e.g., by an adaptive replication mechanism that combines portfolio theory with simple policies from reinforcement learning. Preconditions are that the simulation system supports an on-the-fly configuration of simulation algorithms and that the number of required replications is sufficiently high. Figure 4 shows the consumed CPU time per policy and experiment. The experiments are executed by different SSA and stochastic π-calculus simulators that have been implemented in JAMES II. The stochastic π-calculus simulators rely again on SSA sub-algorithms to determine the next event. 25 and 24 parameter set-ups of SSA and stochastic π-calculus models have been tested, respectively. Each set-up was replicated 1000 times by each of the five policies, so that 125.000 (SSA) and 120.000 (stochastic π-calculus) simulation runs were executed for each experiment. Among the different reinforcement learning policies, epsilon-greedy performed best and achieved a good speed-up (up to 3.2) in comparison to the average case that has been produced by the random policy. However, please note that even before the learning algorithms are applied, one can restrict the space of possible configurations to the most promising ones by exploiting portfolio theory [10]. Without such supportive mechanisms for algorithm selection, the performance effect of choosing a suitable configuration can be dramatical, e.g., we observed a speed-up of over 1000 times when choosing the best instead of the worst configuration. By automatically configuring suitable simulators and reducing the time needed for experiments, experimentation is facilitated, be it for the purpose of model validation and analysis or the evaluation of newly developed simulation algorithms in

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907

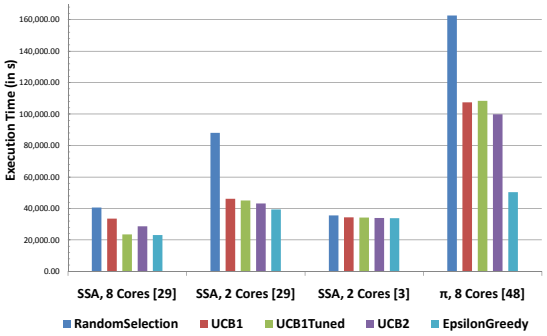


Figure 4: Adapting the simulator configuration online, using different learning policies and experimental settings, and its impact on the performance (for more detail see [10]). The numbers of available simulation configurations are given in square brackets for each experiment.

comparison to existing ones.

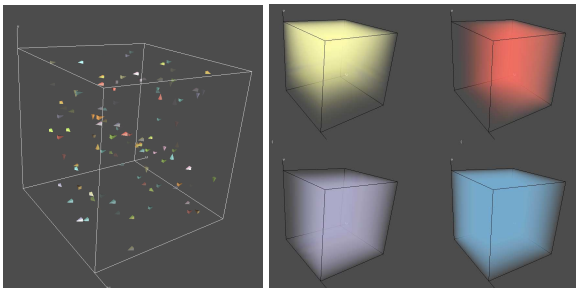
3.3 Visualization

Each of the different phases of modeling and simulation, i.e., model design, simulation experiment design, simulation, and interpretation of results, can be supported by visualization methods. Modeling and simulating regenerative systems provide some challenges for visualization. The main challenge visualization faces refers to the data as those have a spatial and temporal context, are heterogeneous and multi-variate. Classic visualization methods focus typically on only one of these dimensions or integrate multiple-dimensions only for a specific application context, e.g., for the field of terrestrial biogeochemistry in [48]. Yet, in the simulation of regenerative systems all of those play an equally important role, no matter if the simulation is conducted for the biology domain, the network sciences, or any other application area.

A tried and tested solution to this challenge is the use of so-called *linked views* – separate but tightly coupled, interactive views that each target a different aspect of the data. The benefits are obvious: no complicated all-round visualization techniques for heterogeneous, spatio-temporal, and multidimensional data need to be invented, but instead a number of simpler techniques are grouped and interlinked in a way that their combination fulfills the needed properties. This approach has been known for a while and the reader is referred to [2] for a concise overview of the issues involved. In combination with new visualization techniques, it is indeed a valuable approach to gain insight in models, experiments, and multi-run simulations alike.

Focusing on the visualization of large hierarchical models, the point-based hierarchy layout utilizes a very space-efficient node placement. It is shown in Figure 5 (left side), depicting a large dyn-DEVS model of the halobacterium in the context of the JAMES II framework which also provides additional views on the hierarchy. Other visualizations like the Coordinated Graph Visualization toolkit [52] focusing on networked instead of hierarchical model structures similarly allow a modeler to easily navigate through large models (Figure 5, right side).

For the visualization of spatio-temporal simulation data, Figure 6 shows multiple linked 3D-views, visualizing a small next sub-



(a) Event view (b) State view

Figure 6: Visualization of NSM [55].

volume simulation: reaction events and diffusion events are distinguished for multiple time points in the left view and the state, i.e., the concentration, is given for a single time point in the right view. Both are shown in their spatial context and support coordinated interactions in terms of zooming and turning [55].

The next step, the combination of the model visualization and the visualization of multiple simulation runs, is achieved by the integrated approach presented in [26]. Here, the model structure and its evolution in the course of different simulation runs is displayed in linked overview and detail views. While time-varying graphs occur frequently in many applications, e.g., social networks, adhoc and sensor networks, this visualization is targeted towards hypergraphs. These are a generalization of graphs, as they allow edges with more than two incident nodes. An example of such a hypergraph is a biochemical reaction network where an edge representing a reaction usually connects more than one reactant with more than one product. Figure 7 shows this visualization approach, which makes use of a novel table-based visualization for bipartite graphs, into which the hypergraphs are transformed. This enables the simulation expert to inspect and compare individual time steps, as well as gaining an overview of structural changes over time for a larger scale investigation. In Figure 7, the two detailed views of different time steps are shown in the upper part and the overview at the bottom. The overview shows the concentration of a few selected chemicals of our example, as well as the structural complexity as a green curve. Structural complexity can be measured, e.g., by calculating the amount of structural change between the hypergraph at time t and at time $t + 1$ using graph edit distances. Plotted against the time, these measures reveal the ups and downs of the overall structural changes in the reaction network, pointing at possibly interesting time points to explore in detail.

Yet, finally it is the goal to combine visualizations of all three stages: modeling, experiment setup, and simulation. This is done in the Mosan visualization framework [56] which covers the whole process in all its stages. It is shown in Figure 8. The upper left corner shows the network structure of this model, i.e., the biochemical reactions of the Wnt pathway as part of the Experiment View. This view is coordinated with the Multi-Run View (top right) for the selection of one run from all runs, the Node View (bottom left) for the detailed analysis of single-run data linked to nodes, and the Edge View (bottom right) that shows the single-run data connected to edges. Each of the nodes shows in a glyph-like fashion the overall gradient of a substance's concentration. Multiple runs and experiment setups can be compared and selected from numerous dia-

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907

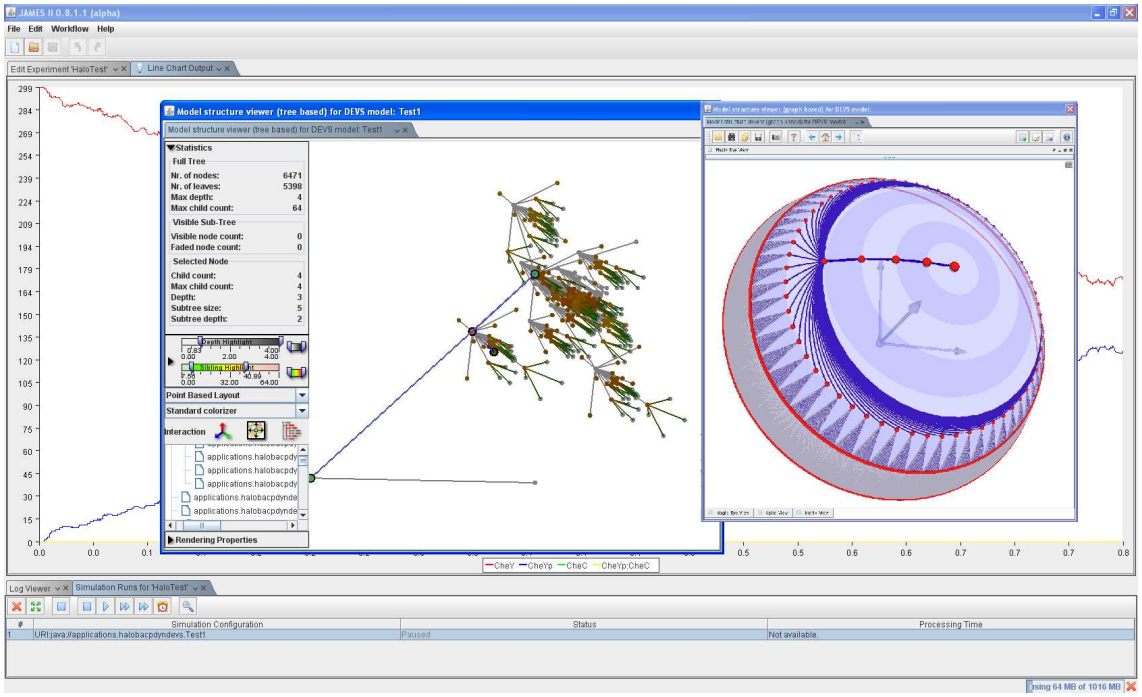


Figure 5: Visualization of model structures in JAMES II

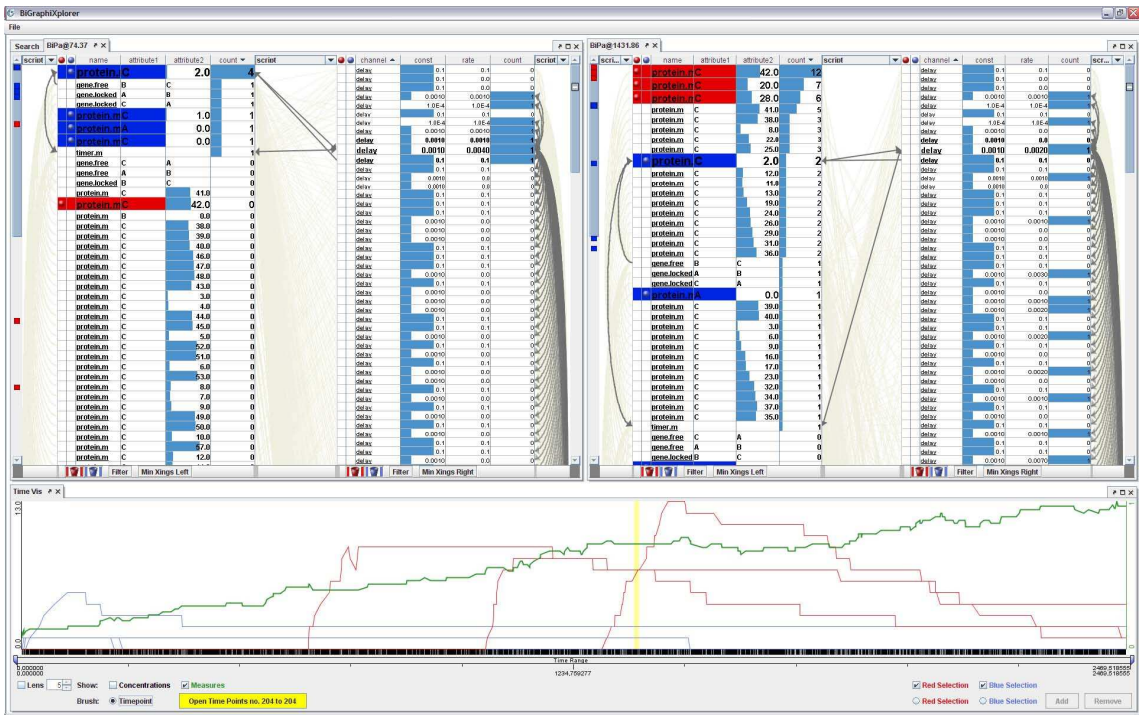


Figure 7: Screenshot of a visualization tool for time-varying hypergraphs [26].

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907

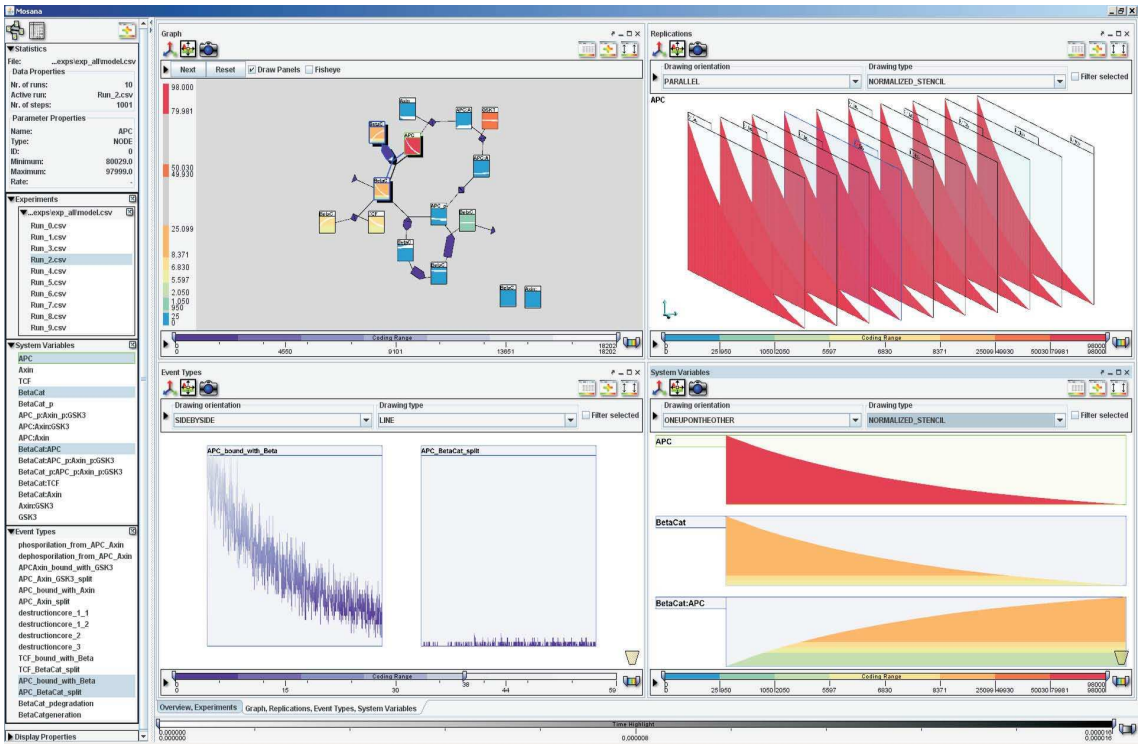


Figure 8: An integrated view on modeling and simulation applied to a part of the Wnt pathway [56]

grams displayed in stacked or Rolodex-style.

In summary, it can be said that the modeling and simulation of regenerative systems is a strong driving force behind many of the shown visualization techniques. E.g., the shown visualization of time-varying hypergraphs, which was specifically motivated by the task of depicting changing biochemical model structures, is the first of its kind. Yet at the same time, the approach of using multiple linked views has kept the inevitable complexity of these visualizations down to a minimum. Also, it has been shown by the selected visualization techniques presented here, that the evolution of them is quite similar to the ones described for the modeling formalisms: step after step the available toolkits have grown, now being up to the task of visualizing heterogeneous, multi-variate data in spatial and temporal contexts.

4. SUMMARY

Regenerative systems provide many challenges and thereby, opportunities for modeling, simulation, and visualization. The focus of our exploration has been on methodological approaches developed for biological applications addressing two important properties of regenerative systems, i.e., dynamic structures and operating at multiple spatial and temporal scales. To exemplify approaches and open challenges, we concentrated on research done in the context of the modeling and simulation framework JAMES II. Although the presented methods have been developed for biological systems, central concepts and entire formalisms are applicable for regenerative systems in general.

Whereas variable structures like dynamic interactions come easily to some modeling formalisms like π-calculus other formalisms like DEVS require extensions and this at a price of additionally burdening the formalism. Also the development of numerous π-calculus variants over the last years document the need for further improvements to ease the modeling of biological systems. Thus, although first steps are done to get a feeling what might be needed to model biological systems, it seems the modeling formalisms are still a far cry away from offering these features combined with an ease of use.

As models become more complex, the possibility of exploring and navigating hierarchical and network model structures is important. E.g., the adaptation of state of the art space-efficient visualization methods and their integration into simulation tools allow an overview and in-detail inspection of large, complex models on demand. However, as the model's structure is itself subject to changes, entirely new challenges arise as now structures are combined with a temporal dimension. For visualization purposes time is either mapped to time (animation) or to screen space (multiple time steps side by side). Visualizing dynamic hypergraphs in JAMES II is based on the latter and represents a very first step to supporting visually the analysis of large scale models with dynamic structures. The representation of meta information, like the amount of structural changes between events will help identifying interesting regions of simulation traces for further exploration.

Space has a crucial impact on the dynamics of regenerative biological systems, driving the development of new modeling formalisms, e.g., like Space-π, Attributed π, and Imperative π-Calculus. Whereas Space-π provides a continuous view on space, Attributed π and Imperative π-Calculus are discrete. These different perceptions of space are also reflected and sometimes even

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907

combined at the simulation level, e.g., when simulating certain molecules by Brownian dynamics and others at population level with NSM. Although the representation of space appears at first glance specific for biological systems, it also gains importance for technical systems. Sensor networks are only one obvious example for this. The more computing exploits decentralized and ad-hoc structures (e.g., when using cars as sensor and computing nodes to predict traffic congestion in an online manner [12]), the more important the location and movement of computing nodes becomes in realizing those systems and in evaluating them by simulation.

Due to operating on different temporal and spatial scales, simulating regenerative systems is computationally expensive. The desire to balance between efficiency of simulation and required detail drives the development of new simulation algorithms and innovative approaches that combine techniques from diverse fields, like machine learning, economics, and experimental algorithmics [34] to configure efficient simulators on demand.

The key to this type of solutions lies in modeling and simulation frameworks like JAMES II which is based on a "Plug'n simulate" concept. It facilitates integrating and evaluating modeling and simulation methods and supports different application areas. Thereby, re-use and alternatives do not end at the level of model, modeling formalisms, simulation algorithms, and tools, but refer to algorithms, sub-algorithms, data structures, and experimental setups as well. Not unlike the shown linked views in visualization that help designing and analyzing regenerative systems, its concept avoids all-around solutions that are difficult to maintain and relies on linking diverse plug-ins so that their combination fulfills the requirements.

The above efforts will help to improve the quality of "in-silico" experimentations referring to biological and technical regenerative systems alike. Modeling, simulation, and visualization forms a common ground for a better understanding and analysis of biological and technical regenerative systems, thereby also promoting the migration of concepts from regenerative biological to computer science systems and vice versa.

5. ACKNOWLEDGMENTS

We would like to thank Roland Ewald, Fiete Haack, Jan Himmelspach, Matthias Jeschke, Mathias John, Stefan Leye, Carsten Maus, Orianne Mazemondet, and Andrea Unger for their comments. This research is supported by the DFG (German Research Foundation) as part of the research training school "dIEM oSiRiS".

6. REFERENCES

[1] R.Z. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 47, 2002.

[2] M.Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *AVI '00: Proc. of the working conference on Advanced visual interfaces*, pages 110–119, New York, NY, USA, 2000. ACM.

[3] G.-Q. Bi and M.-M. Poo. Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci.*, 18(24):10464–10472, 1998.

[4] G. Broderick and E. Rubin. The realistic modeling of biological systems: A workshop synopsis. *ComPlexUs Modeling in Systems Biology, Social Cognitive and Information Science*, 3(4):217–230, 2006.

[5] K. Burrage, P.M. Burrage, N. Hamilton, and T. Tian. Compute intensive simulations for cellular models. In A. Y. Zomaya, editor, *Parallel Computation for Bioinformatics and Computational Biology*. Wiley, 2005.

[6] Y. Cao, D. T. Gillespie, and L. R. Petzold. Efficient step size selection for the tau-leaping simulation method. *J Chem Phys*, 124(4), January 2006.

[7] L. Dematté and T. Mazza. On parallel stochastic simulation of diffusive systems. In *CMSB*, pages 191–210, 2008.

[8] J. Elf and M. Ehrenberg. Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases. *Systems Biology, IEE Proc.*, 1(2):230–236, 2004.

[9] R. Ewald, J. Himmelspach, and A.M. Uhrmacher. An algorithm selection approach for simulation systems. In *Proc. of Principles of Advanced and Distributed Simulation*, pages 91–98. IEEE Computer Society, 2008.

[10] R. Ewald, S. Leye, and A.M. Uhrmacher. An efficient and adaptive mechanism for parallel simulation replication. In *Proc. of Principles of Advanced and Distributed Simulation*, 2009.

[11] A. Ferscha, J. Johnson, and S. J. Turner. Distributed simulation performance data mining. *Future Generation Computer Systems*, pages 157–174, September 2001.

[12] R.M. Fujimoto, M. Hunter, J. Sirichoke, M. Palekar, H. Kyoung Kim, and W. Suh. Ad hoc distributed simulations. In *Proc. of Principles of Advanced and Distributed Simulation*, pages 15–24. IEEE Computer Society, 2007.

[13] M.A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem.*, 104:1876–1889, 2000.

[14] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977.

[15] M. Handy, M. Haase, and D. Timmermann. Low energy adaptive clustering hierarchy with deterministic cluster-head selection. In *in IEEE MWCN*, pages 368–372, 2002.

[16] D.O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. Wiley, 1949.

[17] J. Himmelspach and A.M. Uhrmacher. Plug'n simulate. In *Proc. of the Annual Simulation Symposium*, pages 137–143. IEEE Computer Society, 2007.

[18] M.C. Huebscher and J.A. McCann. A survey of autonomic computing - degrees, models, and applications. *ACM Comput. Surv.*, 40(3), 2008.

[19] M. Jeschke and R. Ewald. Large-scale design space exploration of SSA. In *Proc. of Computational Methods in Systems Biology*, volume 5307 of *LNCS*, pages 211–230, Berlin / Heidelberg, 2008. Springer.

[20] M. Jeschke, A. Park, R. Ewald, R. Fujimoto, and A.M. Uhrmacher. Parallel and distributed spatial simulation of chemical reactions. In *Proc. of Principles of Advanced and Distributed Simulation*, pages 51–59, 2008.

[21] M. Jeschke and A.M. Uhrmacher. Multi-resolution spatial simulation for molecular crowding. In *Proc. of the Winter Simulation Conference*, pages 1384–1392. WSC, 2008.

[22] M. Jeschke and A.M. Uhrmacher. Multi-resolution spatial simulation for molecular crowding. In *Proc. of the Winter Simulation Conference*, pages 1384–1392. WSC, 2008.

[23] M. John, R. Ewald, and A.M. Uhrmacher. A spatial extension to the pi calculus. In *Proc. of From Biology To*

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907

Concurrency and back, volume 194 of *Electronic Notes in Theoretical Computer Science*, pages 133–148, 2008.

[24] M. John, C. Lhoussaine, and J. Niehren. Dynamic compartments in the imperative pi-calculus. In *Proc. Computational Methods in Systems Biology*, LNCS. Springer Verlag, 2009 (to appear).

[25] M. John, C. Lhoussaine, J. Niehren, and A.M. Uhrmacher. The attributed pi calculus. In *Proc. Computational Methods in Systems Biology*, volume 5307 of *LNBI*, pages 83–102. Springer Verlag, 2008.

[26] M. John, H.-J. Schulz, H. Schumann, A.M. Uhrmacher, and A. Unger. Exploring time-varying hypergraphs. In *Poster Compendium of the IEEE InfoVis*, 2009.

[27] H. Kitano, editor. *Foundations of Systems Biology*. MIT Press, 2001.

[28] F. Klan, M. Röhl, B. König-Ries, and A.M. Uhrmacher. Using context information to evaluate cooperativeness. In *Proc. of ACM symposium on QoS and security for wireless and mobile networks*, pages 98–105, New York, NY, USA, 2008. ACM.

[29] J. Kleinberg. The small-world phenomenon: An algorithmic perspective. In *Proc. of the 32nd ACM Symposium on Theory of Computing*, pages 163–170, 2000.

[30] E. Lee, A. Salic, R. Krüger, R. Heinrich, and M.W. Kirschner. The roles of APC and Axin derived from experimental and theoretical analysis of the Wnt pathway. *PLoS Biol*, 1(1):116–132, 2003.

[31] S. Leye, J. Himmelspach, M. Jeschke, R. Ewald, and A.M. Uhrmacher. A grid-inspired mechanism for coarse-grained experiment execution. In *Proc. of Distributed Simulation and Real-Time Applications*, pages 7–16. IEEE Computer Society, IEEE Press, 2008.

[32] S. Leye, C. Priami, and A.M. Uhrmacher. A bounded-optimistic, parallel beta-binders simulator. In *Proc. of Distributed Simulation and Real-Time Applications*, pages 139–148. IEEE Computer Society, IEEE Press, 2008.

[33] O. Mazemondet, M. John, C. Maus, A.M. Uhrmacher, and A. Rolfs. Integrating diverse reaction types into stochastic models - a signaling pathway case study in the imperative pi-calculus. In *Proc. of the Winter Simulation Conference*. WSC, 2009 (to appear).

[34] C. McGeoch. Experimental algorithmics. *Communications of the ACM*, 50(11):27–31, November 2007.

[35] R. Milner. *Communicating and Mobile Systems: the π-calculus*. Cambridge University Press, 1999.

[36] A. Morrison, M. Diesmann, and W. Gerstner. Phenomenological models of synaptic plasticity based on spike timing. *Biological Cybernetics*, 98(6):459–478, 2008.

[37] D.M. Nicol, O. Balci, R.M. Fujimoto, P.A. Fishwick, P. L'Ecuyer, and R. Smith. Strategic directions in simulation research (panel). In *Proc. of Winter Simulation Conference*, pages 1509–1520. WSC, 1999. Chairman-Page, Ernest H.

[38] D.M. Nicol and G. Yan. Discrete event fluid modeling of background TCP traffic. *ACM Trans. Model. Comput. Simul.*, 14(3):211–250, 2004.

[39] K. Pawlikowski, H.-D.J. Jeong, and J.-S.R. Lee. On credibility of simulation studies of telecommunication networks. *Communications Magazine, IEEE*, 40(1):132–139, Jan 2002.

[40] A. Phillips and L. Cardelli. Efficient, correct simulation of biological processes in the stochastic pi-calculus. In *Proc. Computational Methods in Systems Biology*, pages 184–199. Springer, 2007.

[41] A. Regev and E. Shapiro. Cells as Computation. *Nature*, 419:343, 2002.

[42] F. Reichenbach, A. Bobek, P. Hagen, and D. Timmermann. Increasing lifetime of wireless sensor networks with energy-aware role-changing. In *Self-Managed Networks, Systems, and Services, 2nd IEEE Workshop, Proc.*, volume 3996 of *LNCS*, pages 157–170. Springer, 2006.

[43] J. Salzmann, R. Behnke, M. Gag, and D. Timmermann. 4-MASCLE - improved coverage aware clustering with self healing abilities. In *Proc. of Multidisciplinary Autonomous Networks and Systems*, 2009.

[44] J. Salzmann, R. Behnke, D. Lieckfeldt, and D. Timmermann. 2-MASCLE - a coverage aware clustering algorithm with self healing abilities. In *Proc. of Intelligent Sensors, Sensor Networks and Information Processing*, pages 739–744, 2007.

[45] J. Salzmann, S. Kubisch, F. Reichenbach, and D. Timmermann. Energy and coverage aware routing algorithm in self organized sensor networks. In *Proc. of Networked Sensing Systems*, pages 77–80, 2007.

[46] H.M. Sauro, D. Harel, M. Kwiatkowska, C.A. Shaffer, A.M. Uhrmacher, M. Hucka, P. Mendes, L. Strömback, and J.J. Tyson. Challenges for modeling and simulation methods in systems biology. In *Proc. of the Winter Simulation Conference*, pages 1720–1730. WSC, 2006.

[47] A. Schäfer and M. John. Conceptional modeling and analysis of spatio-temporal processes in biomolecular systems. In *Proc. Asia-Pacific Conference on Conceptual Modelling*, pages 39–48, 2009.

[48] R. Sisneros, M. Glatter, B. Langley, J. Huang, F. Hoffman, and D.J. Erickson III. Time-varying multivariate visualization for understanding terrestrial biogeochemistry. *Journal of Physics: Conference Series*, 125(1):012093, 2008.

[49] K. Takahashi, S.N.V. Arjunan, and M. Tomita. Space in systems biology of signaling pathways - towards intracellular molecular crowding in silico. *FEBS Letters*, 579(8):1783–1788, March 2005.

[50] K. Takahashi, K. Kaizu, B. Hu, and M. Tomita. A multi-algorithm, multi-timescale method for cell simulation. *Bioinformatics*, 20(4):538–546, March 2004.

[51] W.F. Tichy. Should computer scientists experiment more? *Computer*, 31(5):32–40, 1998.

[52] C. Tominski, J. Abello, and H. Schumann. CGV – an interactive graph visualization systemy. *Computers & Graphics*, 2009. to appear.

[53] A.M. Uhrmacher, J. Himmelspach, and R. Ewald. Effective and efficient modeling and simulation with devs variants. In *Discrete-Event Modeling and Simulation: Theory and Applications*. CRC Press. Taylor and Francis., to appear.

[54] A.M. Uhrmacher and D. Weyns, editors. *Multi-Agent Systems: Simulation and Applications*. Taylor and Francis, 2009.

[55] A. Unger, E. Gutzeit, M. Jeschke, and H. Schumann. Viones – visual support for the analysis of the next-sub-volume method. In *Proc. of the IEEE IV*, pages 10–17, 2009.

[56] A. Unger and H. Schumann. Visual support for the understanding of simulation processes. In *IEEE PacificVis*, pages 57–64, 2009.

[57] B.P. Zeigler, H. Praehofer, and T.G. Kim. *Theory of Modeling and Simulation*. Academic Press, London, 2000.

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7907
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7907