SECTION 3- ELECTRONIC NOTES ON SIMULATION
SIMULATION OF MOLECULAR COMMUNICATION SYSTEMS

Simulation Tools for Molecular Communications

G. Reali [a], M. Femminella[a], L. Felicetti[a], A. Davy[b], Michael Barros[b] **, R. G. Cid-Fuentes[c], A. Cabellos-Aparicio[c], J.S. Pareta[c], E. Alarcon[c], P. Liò[d], P. Gresele[e], M. Malvestiti[e], W. Tavernier[f], Y. Koucheryavy[g], V. Petrov[g], S. Balasubramaniam[g], Ozgur B. Akan[h]**

[a] Consorzio Nazionale Interunivesitario sulle Telecomunicazioni (CNIT)-University of Perugia Research Unit. Via G. Duranti 93, Perugia, Italy.
[b] TSSG, Waterford Institute of Technology, Waterford, Ireland
[c] Universitat Politècnica de Catalunya, Departament d'Arquitectura de Computadors, Despatx 110. Jordi Girona, 1-3.
[d] Computer Laboratory, University of Cambridge, 15 JJ Thomson Avenue, Cambridge CB3 0FD, UK.
[e] Department of Medicine, University of Perugia, Piazzale Gambuli, 1 - 06132 Perugia, Italy.
[f] Iminds, Technologiepark-Zwijnaarde 19, 9052 Gent, Belgium.
[g] Department of Electronic and Communication Engineering, Tampere University of Technology, Tampere, Finland.
[h] Department of Electrical & Electronics Engineering, Koc University, Sariyer, Istanbul, Turkey

Abstract— *The field of molecular communications aims to utilize biological systems to create communication systems. This new paradigm shift, that has been developed and embraced by the computing and communication engineering researchers is highly multi-disciplinary, embodied with numerous challenges. One of these challenges is in developing a suitable simulation environment that can be used as a platform for researchers to embark into this new area of research. This paper is a survey of the most recent and versatile simulators for molecular communication systems. We highlight the peculiarities of these simulators in relation to the specific biological environment to be investigated.*

I. INTRODUCTION

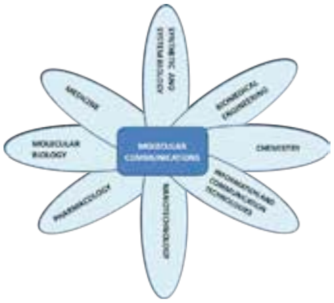
As the field of nanotechnology continues to unravel and solve numerous challenges that are found in the world today, new paradigms are continually being developed and support this area of science. One particular functionality that has been recently proposed is the "communication" at the nanoscale, which has led to the birth of molecular communication.

This paper illustrates, evaluates, and compares a set of existing simulators of Molecular Communications (MolCom).

MolCom is an emerging area which promises significant breakthroughs in some strategic socio-economic fields [1][2]. As shown in Fig. 1, it encompasses multi-disciplinary research that brings together information and communication technologies, molecular biology, physics, chemistry, biotechnology, as well as technologies that will realize the vision of transferring information within biological environments at extremely small scales, down to a size comparable to that of molecules.

The physical mechanisms that allow transferring information have been inspired by the biological mechanisms that exist in living bodies to exchange many types of signaling and particles, such as proteins, pheromones, and immune system activation signals. This multidisciplinary nature characterizes also the tools that are used to design and analyze such systems.

Figure 1. The multi-disciplinary nature of Molecular Communications.



Besides the analytical tools that have been developed [1], the complexity of MolCom usually requires the use of simulators tailored to the specific biological environment. In fact, for a simulator to be useful for investigating the detailed interactions that occur at extremely small scales, it must implement the elementary interaction mechanisms that characterize the analyzed environment, in terms of particle size, shape, interaction with the surrounding environment, and timescales. For this reason, different research groups have been developing various simulation tools tailored to their research needs. Developing a realistic simulation toolset and framework can be beneficial not only for communication engineering and networking researchers to further progress this new field of research, but also to provide opportunities for other disciplines (biotechnologists, medical doctors) to solve specific health care problems using communication functionalities. For example, virtual human organs can now predict drug effects that occur in the tissues and cells, while the simulator can play a supporting role in designing the process of delivering the molecules to the intended target, as well as investigate how the molecular structure of the drug will influence the motion within the environment. Simulators can also be used as partial substitute for experiments conducted in animals too, which will reduce cost as well as minimize any need for ethical approvals. In addition, simulators can exploit their best features in the field of personalized medicine. In fact, the ability to analyze immense data sets and extract all possible useful information is fundamental for the development of personal medicine.

II. REVIEW OF THE EXISTING SIMULATORS

In this section, some of the main simulators that have been developed to characterize the broad set of MolCom systems are revised. Table 1 reports the main features of some of the existing simulators. This set has been selected according to their adaptability to different types of molecular communication systems.

A. BiNS - Biological and Nano-Scale communication simulator

BiNS is a simulation package for MolCom systems developed at the University of Perugia [3]. Its customizable design provides a set of tools that allow creating objects modeling the behavior of biological entities, such as nodes (transmitters, receivers), carriers, or surrounding objects (e.g., vessel walls). In addition, BiNS permits to shape the simulated communication channel (e.g., the blood stream or in vitro experiment) with the desired accuracy. The simulator has been implemented in Java, and includes a generic type of software object, named Nano Object. Nodes and carriers are specific implementations of the Nano Object, and, although they share its general features, they can exhibit very different functions. The simulation is organized in discrete time steps. Each step consists of a number of phases, in which software objects are triggered in order to execute the operations associated with their specific behavior. The main phases are: transmission phase, reception phase, information processing phase, motion phase, object destruction phase (during which objects are removed due to lifetime expiration or because they exited from the area of interest), and collision management phase, which implements the elementary interaction between particles. BiNS has been validated through wet-lab experiments related to cardiovascular medicine. The list of simulated nano-objects are split into smaller lists, which can be handled in parallel by a thread pool. The simulated environment can be either unbounded or bounded by a surface of custom shape, referred to as simulation domain. Within a domain, the octree paradigm allows distributing the workload associate to the management of the objects in the domain volume to different threads.

B. CalComSim - Calcium Signaling-based Molecular Communication System Simulator

CalComSim is a calcium signaling-based molecular communication simulator, designed for both synthetic and natural communications found inside a 3D human tissues [4].

An integration of the biological models for the signaling process governed by stochastic solvers is used to handle multiple and parallel reactions in each cell of the tissue. Also, it implements three different types of tissues. They can be classified as: non-excitable (tissues which cannot propagate electric current), excitable (tissues which can propagate electric current) and hybrid (tissues capable of communicating with both excitable and non-excitable tissues). Each type of cell simulated are as follows: epithelial (non-excitable), smooth muscle cells (excitable) and astrocytes (hybrid). The biological models that are incorporated into the simulator are based on real experimental data. The simulator also includes the stochastic closing and opening behavior of the gap junction for each type of cell. The simulator can be used not only by telecommunication engineers and biological scientists, but also by pharmaceutical researcher that can design new drugs and treatments for diseases that emerge from impaired calcium signaling.

TABLE 1. Comparison of the most popular simulators for molecular communications systems

Simulator Name	BiNS	CalComSim	N3Sim	COMSOL Multiphysics	NS-3 based Simulators	NanoNS	BNSim	NCSim
Customization	Customizable simulation domains	Through Python programming	Via configuration file.	Very flexible model adaptation. Script-based module integration.	Through C++/Python programming	Through C++/OTcl programming	Through Java programming; selection of the simulation method	Through C++/Python programming
Licence	Open	Open	Open	Commercial	Open	Open	Open	Open
Purpose	Diffusion based MolCom	Calcium-signalling based molecular communications inside cellular tissues	Diffusion based MolCom	multipurpose platform designed for simulating physics-based problems.	Designed as a network simulator	Diffusion based MolCom	Bacteria-Based Nanonetworks	Flagellated bacteria-based MolCom
User Interface	Command line based with possible visualization of propagating particles	Command line based only	Command line based	Graphical user interface	Command line preferable. Animation tools available	Command line preferable. Animation tools available	Command line based	Command line based
Implementation techniques	Java, multi thread.	Python	Java	N/A	C++/Python	C++	Java	C++ multithread + Python
Deployment	Both single machine and cluster-based	Single machine	Single machine	Both single machine and cluster-based	Single machine	Single machine	Single machine	Any, point-to-point, by default

CalComSim has been implemented in Python, and requires the following dependencies: Numpy, Scypy, Pypy, Cpython. Numpy and scypy. They are used to access mathematical and scientific functions, while the Pypy is an alternative interpreter for the simulator. The Cpython is used to run C libraries that can be used in python codes. The Gillespie stochastic algorithm [5], was used to solve partial differential equation that model intracellular and intercellular calcium signaling. The simulator is controlled by a command line interface.

C. N3Sim

N3Sim [6] is a simulation framework for diffusion-based molecular communications. Transmitters encode the information by releasing particles into the medium, thus varying the concentration rate in their vicinity. The diffusion of particles through the medium is modeled as Brownian motion, taking into account particle inertia and collisions among particles. Receivers decode the information by sensing the local concentration in their neighborhood. The benefits of such a simulator are multiple: the validation of existing channel models for molecular communications and the evaluation of novel modulation schemes are just two examples. It implements a three-layer architecture. The user interface layer interacts with the user to read the input data for the simulation, while the data layer writes the simulation results to files. The domain layer contains the "intelligence" of the system, the molecular communication model. N3Sim is a Java-based simulator. The simulation parameters are determined by means of a text configuration file editable by the user.

The scalability of the simulator can be improved by selecting a higher simulation time step (at the expense of accuracy) or by deactivating the collisions among molecules in scenarios with a low molecular concentration. The time granularity of a simulation is defined by the user by selecting the simulation time step (typically a few ms).

D. COMSOL Multiphysics

COMSOL Multiphysics® is a commercial multipurpose platform designed for simulating physics-based problems through a unified workflow for electrical, mechanical, fluid, and chemical applications. It implements finite element analysis, for different physics and engineering applications. It is available for different operating systems. An example of the use of COMSOL Multiphysics for simulating a MolCom drug delivery system is presented in [7]. Nevertheless, this platform is tailored to macroscopic phenomena, such as flows. Modeling the interactions of biologic particles at molecular level requires the implementation of specific models. For example, the ligand-reception formation is not supported by specific libraries and has to be implemented by users. MATLAB® scripts can be used to implement new modules. The learning time for a fruitful use of the simulator is in the order of several months. The COMSOL package supports cluster computing. Thus, any simulation job can be deployed to any number of clustered computers.

E. NS-3 Based Simulators

NS-3 is a discrete-event network simulator for Internet systems, which was not developed for MolCom simulators. Nevertheless, its flexible structure has allowed implementing some basic elements of MolCom, in particular a simulation tool has been developed in the framework of the IEEE P1906.1 working group [8]. NS-3 is organized in different software libraries that can work together. User programs, written in C++ or Python programming languages, can adapt these libraries or be linked with them. External animators and data analysis and visualization tools are available. Nevertheless, in order to exploit the full potentials of the NS-3 the command line interface is preferable. The simulator is designed to be executed on a single machine. The minimal requirements to execute basic simulations are a gcc or clang compiler and Python interpreter. The simulation granularity is defined by users.

F. NanoNS

NanoNS is a molecular communication tool developed on top of NS-2 [9], which is an object-oriented simulator written in C++ with an object-Tcl (OTcl) interpreter. The main objective of the framework is to provide a simulation tool to create a better understanding of nanonetworks and facilitate the development of new communication techniques as well as provide a platform for validation of theoretical results. It incorporates the simulation modules for various nanoscale communication paradigms using a diffusive molecular communication channel. This is based on Brownian motion in aqueous medium, and is a consequence of the constant thermal motion of atoms, molecules, and their chemical reactions. An OTcl library, called ns-mol.tcl, is developed for a new node structure called NanoNode, which is based on the OTcl standalone class of Node. The library along with new nanonetwork components, parameters and methods for molecular communication provides the scheduling of the simulation, settings of the network topology and configuration of nanonetwork parameters.

F. BNSim

BNSim [10] is a multithread java simulator for bacteria networks. These networks interconnect engineered bacteria that communicate at nanoscales. BNSim integrates three simulation methods: (i) the Gillespie's stochastic simulation algorithm (SSA), which solves the chemical master equation in order to determine the time evolution of a chemical system; the extremely small granularity of events makes this approach unpractical for population-level problems; (ii) stochastic differential equations, used to model large-scale chemical system with a controlled level of approximation; (iii) a hybrid algorithm which integrates both the SSA and the methods based on differential equations. The approximated methods based on stochastic differential equations are used to model the behavior of species that are present in large quantities and generate scalability issues. It was estimated to have a speed increase of 5-7 orders of magnitude over the SSA approach [10].

G. NCSim

NCSim is a simulation framework for molecular communications, utilizing flagellated bacteria for information delivery [11]. The major focus of the framework is on different message encoding techniques. NCSim supports typical deployment policies, such as grid and random, as well as custom deployment policies. It can simulate several simultaneous links between the nanomachines. NCSim incorporates the stochastic model for bacteria mobility by Wang et al. [12], and the plasmid/chromosome transfer between bacteria through the conjugation process. Currently, only the 2-dimensional configuration is supported. The accuracy of the produced metrics in major scenarios has been confirmed via comparison with analytical results in simplified scenarios. NCSim consists of three modules: (i) physical (PHY) layer of bacterial nanonetworks, including deployment, bacteria mobility and conjugation, plus messages encoding/decoding; (ii) scenarios generator and simulation monitor; and (iii) plotting tool, intended to post-processing of raw simulation data and plots generation. The PHY module, as the most computational intensive, is implemented in C++. The two latter modules are written in Python for maintenance and extension simplicity. The user interacts with NCSim by writing small scripts on Python to define scenarios.

H. Other Related Simulators.

In [10] and in the references therein it is possible to find details of the following simulators: (i) Smoldyn, which simulates molecular-level biological interactions and particle diffusion; (ii) NFSim, which is a single-cell level simulators; (iii) Stochsim, developed for studying bacterial chemotaxis and the stochastic features of this signalling pathway; (iv) Agentcell, which is a population-level simulators working in conjunction with Stochsim; (v) RapidCell , which models the chemotaxis pathway through stochastic differential equations in a 2D environments, and (vi) BSim, which models the physicochemical interactions among bacteria in a 3D environment. In [13], the authors introduce a simulator design focusing on scalability, and adopting the high level architecture model, which is standardized under IEEE 1516. It is used to design a distributed simulation tool for molecular communication, so that different scalability options can be used to include additional processing power to shorten the execution time. Additionally, in the Project MINERVA, a Nervous NaNoNetwork Simulator, N4Sim, covering the electrical and molecular communications in the synaptic channels and neurons, is being developed. The simulator will be used to capture communication theoretical insights of neural diseases related to communication failures, e.g., multiple sclerosis, paralysis, Alzheimer's disease.

Diffusion based molecular communication have been also simulated by using general purpose scripting and programming languages, such as MATLAB . In this regard, it is worth to mention a MATLAB script shown in [14] that emulates droplet circulation in microfluidic circuits by leveraging the formal equivalence between droplet propagation and current propagation in electronic circuits. Finally, it is important to consider BioModels, which is a reference repository of pathway models.

III. Lesson Learned

The simulators shown above are characterized by both significant limitations and strengths. The main limitations consists of the lack of interworking functions between simulators. The main strength consists of the coverage of a large set of case studies and functions on which it is strategic to leverage for taking advantage of the capabilities of each other. In fact, the research challenges in the area are emerging frequently, and the cross-fertilization of the ongoing simulation activities will allow achieving research results much faster. For example, the ability of bacteria to communicate is significantly influenced by the physical features and flow of fluid in the environments they are invading. This phenomenon can be investigated by combining the capabilities of the mentioned simulators, handling both bacteria and fluid flow components in blood vessels. As a further example, combining the different facets of MolCom with microfluidics can open a wide range of research perspectives. For these reasons, the project CIRCLE pursues the realization of a combined simulation toolbox, to be used as a design tool for MolCom systems. The aim is to combine the behavior in complex MolCom systems, which are affected by a number of interdependent factors that make the fluid flow at any location depending on the properties of the entire system.

Acknowledgments
This work was supported by the EU project H2020 FET Open CIRCLE (Coordinating European Research on Molecular Communications) No. 665564.

REFERENCES

[1] M. Pierobon, I.F. Akyildiz, "A Physical End-to-End Model for Molecular Communication in Nanonetworks," IEEE Journal on Selected Areas in Communications (JSAC), 2010, 28(4), pp. 602–611.

[2] L. Felicetti, M. Femminella, G. Reali, P. Liò, " Applications of molecular communications to medicine: A survey", Nano Communcation Networks, 2015, DOI: 10.1016/j.nancom.2015.08.004.

[3] L. Felicetti, M. Femminella, G. Reali, "A simulation tool for nanoscale biological networks", Nano Communcation Networks, 2012, 3(1), pp. 2-18.

[6] M. T. Barros, S. Balasubramaniam, B. Jennings, " Comparative End-to-end Analysis of Ca2+ Signaling-based Molecular Communication in Biological Tissues," IEEE Transactions on Communications, 2015.

[5] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," J. Phys. Chem., vol. 81, no. 25, pp. 2340–2361, 1977.

[6] I. Llatser, I. Pascual, N. Garralda, A. Cabellos-Aparicio, M. Pierobon, E. Alarcón and J. Solé-Pareta, "Exploring the Physical Channel of Diffusion-based Molecular Communication by Simulation", in Proc. of IEEE GLOBECOM, Houston (USA), 2011.

[7] Y. Chahibi, M. Pierobon, S. O. Song, "Molecular communication Model of Nanoparticle-Body Interactions In Particulate Drug Delivery Systems" Asilomar Conference on Signals, Systems and Computers, 2013. doi: 10.1109/ACSSC.2013.6810452.

[8] S. F. Bush et al., "Defining Communication at the Bottom", IEEE Transactions on Molecular, Biological, and Multi-Scale Communications, 1(1), March 2015.

[9] E. Gul, B. Atakan, O. B. Akan, "NanoNS: A nanoscale network simulator framework for molecular communications" Nano Communication Networks, Elsevier, Volume 1, Issue 2, June 2010, Pages 138–156.

[10] G. Wei, P. Bogdan, R. Marculescu, "Efficient Modeling and Simulation of Bacteria-Based Nanonetworks with BNSim", IEEE Journal on Selected Areas in Communications, vol. 31, no. 12, 2013.

[11] S. Balasubramaniam, P. Lio' "Multi-hop Conjugation based Bacteria Nanonetworks", IEEE Transactions on NanoBioscience, vol. 12, no. 1, pp.47-59, March 2013.

[12] Z. Wang, M. Kim, and G. Rosen, "Validating models of bacterial chemotaxis by simulating the random motility coefficient," in In Proc. IEEE BIBE, pp. 1-5, Oct. 2008.

[13] A. Akkaya, G. Genc, T. Tugcu, "HLA based architecture for molecular communication simulation", Simulation Modelling Practice and Theory, 2014, 42, pp. 163-177.

[14] A. Biral, D. Zordan, A. Zanella, "Modeling, simulation and experimentation of droplet-based microfluidic networks" IEEE Transactions on Molecular, Biological, and Multi-Scale Communications, 2015.