SIMULATION OF BIOLOGICAL MODELS USING CAD TOOLS: NEEDS AND FUTURE

Víctor de la Hoz, César Pumar, Nicolás Quintero, Paula Cordero

Abstract: *In this paper, the main characteristics that are expected from this kind of software and the major tools that can be found at the present time are explained, as well as a proposal of a possible program that solves and fixes all the current lacks, for instance, the nonexistance of a multiplatform software, concluded after an exhaustive study of this proof of concept. The computer-aided design (CAD) tools are essential to achieve right and realiable results in Synthetic Biology researches. Furthermore, a comparison between two CAD's programs is made simulating a synthetic biological circuit, showing the results to study how new software improvements can be developed to get better conclusions in this kind of experiments, provided by new tools. The design of a tool that could be developed in the future is proposed, based on non conventional programming paradigms (DNA computing for instance), that allow researchers to solve multiple kind of real problems.*

Keywords: *Synthetic Biology, CAD tools, computational, free software, in silico, simulation*

ACM Classification Keywords: *J.6 COMPUTER-AIDED ENGINEERING, J.3 LIFE AND MEDICAL SCIENCES, I.6 SIMULATION AND MODELING*

Introduction

Natural Computing is a field in Artificial Intelligence, which initially observed biological behaviours and designed new algorithms based on them, implementing those algorithms in conventional and unconventional hardware (DNA or Quantum Computers). As a result they can be named artificial neural networks, evolutionary computation or swarm intelligence, which has been very useful in some computer science fields, such as distributed systems, complex combinatorial problems, logistics or allocation of resources and tasks. In the opposite direction, researchers try to find and implement computational models in biological substrates. This is known as Biological Computing, which can be considered a subfield of Synthetic Biology.

Synthetic Biology [Purnick and Weiss, 2009] is a new field which tries to build new biological systems or modify them following an engineering approach, so one of the main topics of this discipline is building biological components which can be modular and reusable. Once thoose components have been obtained, complex systems can be built based on the assembly of them. The Registry of Standard Biological Parts is a database founded in 2003 at MIT, which contains a lot of genetic components called Biobricks. These Biobricks are DNA strands that encode proteins and can be assembled to have a specific behaviour into the cell. There are a lot of computational behaviours which could be implemented into these strands, such as boolean logic [Weiss et al., 2003], communication protocols [Goni-Moreno, A., 2010], computer architecture concepts [Goni-Moreno, A. et al., 2012] or give alternative ways to solve classic algorithms in mathematics [Baumgardner, J. and Acker et al., 2009]. Its main goal, from a computational point of view, is to implement autonomous computation in biological substrates, something that actual natural computing substrates (like DNA) can not perform.

This article is motivated by the need of new software that uses bio inspired algorithms and other resources from natural computing and artifical intelligence to simulate different processes, biologicals or not, but that use biological models to run them, and can be used in others fields, for instance, economical or enviromental problems. Plus, new IT developments allow to use tools powerful enough, quality and reliable to investigate and support new projects, and motivate other researchers from different areas, who can make use of that tools and share knowledge. In this paper it is discussed the use of two differents tools (TinkerCell [Chandran, 2012] and Synbioss Designer [Hill et al., 2008]) for researchers in the field of Synthetic Biology, and how reliable results that can motivate future studies of these experiments in a laboratory can be obtained. Furthermore, it is discussed the joint use of the two tools to maximize

the features that they offer and give an idea of sharing the results with other researchers.

The *in silico* experiments give the advantage of being able to perform them in powerfuls machines and obtain a simulation in a fairly reliable way, corresponding to reality, and get support to obtain very reliable results. The computer-aided design (CAD) tools are essentials to achieve that goal, and software like TinkerCell or SynBioSS, make possible that many biologist build biological parts and networks. In addition, the free software characteristic of both tools is really important in the developement of new models, because, for instance, existing biological parts can be reused. For all of this, these CAD tools are proposed as an alternative way for traditional approaches of Synthetic Biology.

In recent years, many progress have been achieved in this field, and they led to improvements of the IT resources in some areas such as medicine [Karlsson and Weber, 2012], as well as new lines of research based on basic biological processes [Karr, J.R. et al., 2012] that give the possibility of new ways to understand more complex biological behaviours.

Free software and needs

A typical problem in computer research is that the tools which can be used are expensive or even unreacheable because of researchers' experience, background, or limited skills (typically computer skills). Free software can be used for researches, but generally there are some limitations: the software maintenance in such applications is almost nonexistent, unless an academic tool is being used by other research groups who can develope extensions or fix bugs to make software more reliable and fault-tolerant. In case of having to work with a payment application, the problem of the software mantenience is "solved", but it can not be developed freely to get improvements needed to success the research.

Beside all of this, the main advantage is that a tool developed by a certain group can be made to fit for its needs. For instance, if they are working with people who do not do well with computers (a typical case in interdisciplinary research groups), a tool with a smooth and intuitive Graphical User Interface (GUI) that let people work better and visualizing the results without any problems can be developed. The main disadvantage is that people with an specific background in software developement and knowledge in a specific field is needed to ensure the reusability, maintainability, usability and reliability of the software application.

In the field of Synthetic Biology different tools (some of them with a few years of development like Athena, JDesigner [Bergmann and Sauro, 2006], CellDesigner [Funahashi et al., 2003]) can be found, and others that are an evolution of the previous: Tinkercell [Chandran, 2012] and SynBioSS [Hill et al., 2008], engineering platforms for building and testing cellular circuits, but all of them have deficiencies that can not be resolved without a specific commitment to software development, that requires a level of effort and time that can not be addressed by a research group. The main goal to achieve with this kind of tools is to make them very reliable to ensure the success and sharing of the experiments carried out by them, and, secondly, providing the tool with specific features like a smooth GUI and powerful mathematical methods. The two studied CAD tools are discussed in the next sections.

TinkerCell

TinkerCell is a very useful software tool for Synthetic Biology that allow researchers to design DNA circuits and run complete simulations. One of the most attractive aspects is that it provides nice graphics thanks to its programming Application Programming Interface (API), that lets the use of differents programming languages, such as Python, Octave, C, and Ruby, which allow to make different scripts and import files from Matlab, for instance, and run mathematical models previously created. Furthermore, imported scripts in Systems Biology Markup Language (SBML) [Hucka et al., 2003] format can be run (there is an option to convert SBML files from Antimony files and compile them to run). This last point is important because SBML is used in differents enviroments, giving some advantages such as the possibility of create models for differents platforms and get that models can be used time after their creation, whithout the need of using a specific programming language to construct and develop biological networks in TinkerCell or another tool for Synthetic Biology that allows importing SBML files.
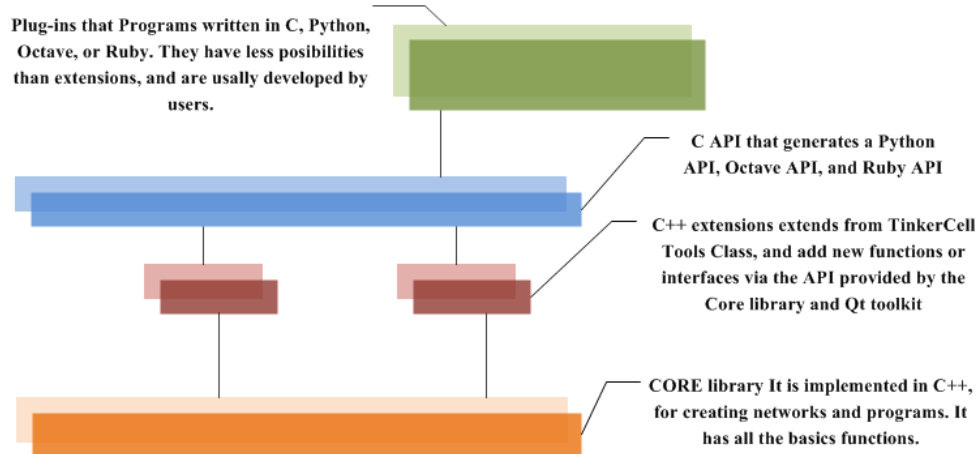
However, SBML is oriented to use models encoded with Extensible Markup Language (XML), which gives the ontology of parts and connectios of the models. XML files are used in many IA applications in the field of biology, and they can be imported in many tools, allowing to give a file to anyone, a biologist or mathematician for instance, who can understand all the ontology and develope the model with his own knowledge.
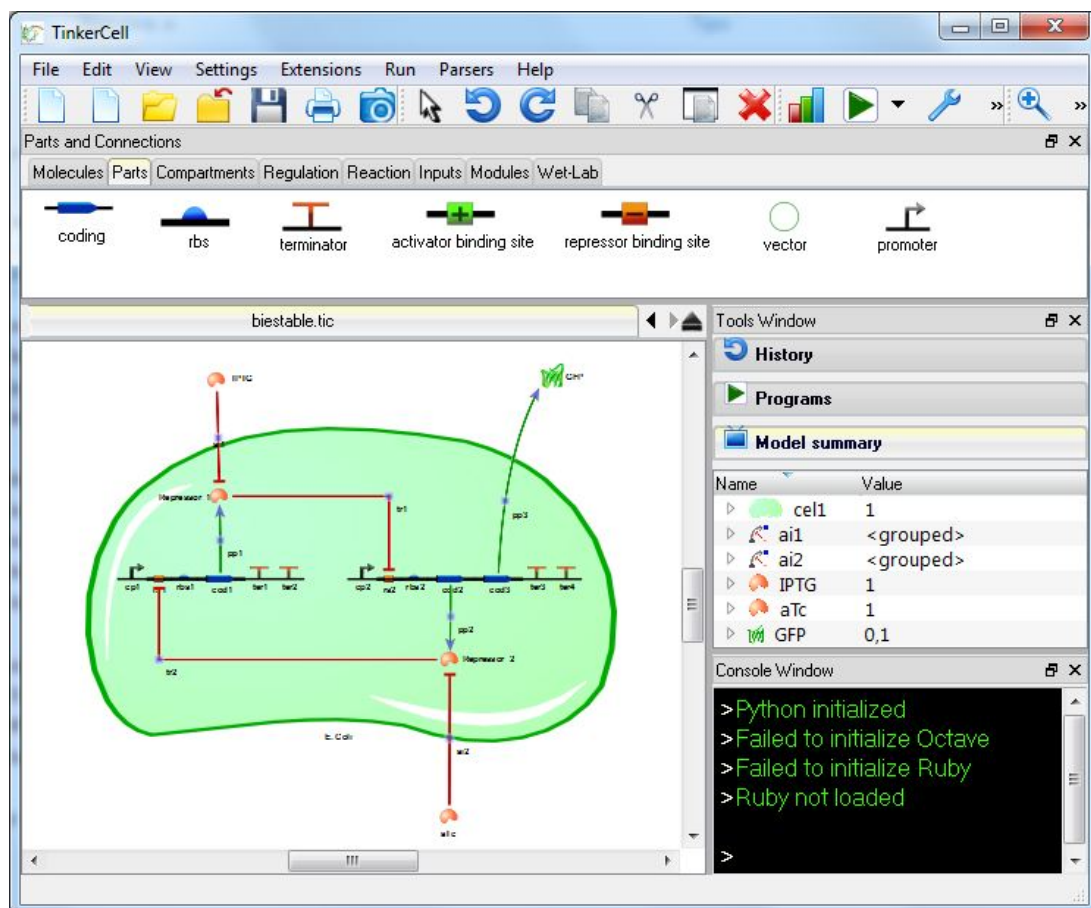

One of the useful characteristics of TinkerCell is the possibility of working with different modules, integrated in the tool. PySCeS [Olivier et al., 2005] can be run in the console, allowing the use of this powerful tool to simulate Celullar Systems. It has been developed in Python, and gives a lot of libraries with useful functions to use. In adition, mathematical models can be loaded to work with Octave. This functionality has been an advantage to simulate the Toggle Switch model. Also, after simulation, the results can be viewed and the parameters of the model can be modified to get a simulation that gives complete information of the experiment. Also, certain parameters of the

simulation can be modified for different results and study how they affect the simulation, obtaining multiple results for a comparison study.

TinkerCell also gives the chance to use extensions. Extensions are programs that can be integrated in TinkerCell. Its purpose is to allow the use of optional features, which give possibilities like, for instance, change the way of the dynamics of a reaction. This characteristics fit good with the open-source project of TinkerCell, because many people can contribute to it, developing useful extensions to make easier working with TinkerCell, adapting the tool to new researches in the field of Synthetic Biology.

SynBioSS

Sometimes there is a problematic designing genetic circuits, because of not having the necessary information to simulate the desired one, due to the nonexistence of an adequate database to work with. To solve it, SynBioSS can be used. This software suite lets the generation, storing, retrieval and quantitative simulation of synthetic biological networks. It consists of three components: SynBioSS Wiki, Designer [Weeding et al., 2010] and Desktop Simulator. SynBioSS Designer is a web service from SynBioSS that allows to create a kinetic model based on Biobricks. Biobricks are standard biological parts of DNA sequences, with defined structure and function. These Biobricks can be found in the SynBioSS database, connected directly with the iGem database, wich gives a lot of components and parts to use in the designs. The web service has a simple interface, where the Biobricks wanted to be used can be selected, previously searched, and create the circuit that, when finished, will be exported to an XML (SBML file) to use in SynBioSS designer or TinkerCell. The main advantage of this is that the Biobricks used in the iGem database are free: an own design can be created and shared with people, making this database better. Obviously, in the design of the circuit Operators, Proteins, and other components required to create a biological circuit are used, and they can be found in the iGem database too.
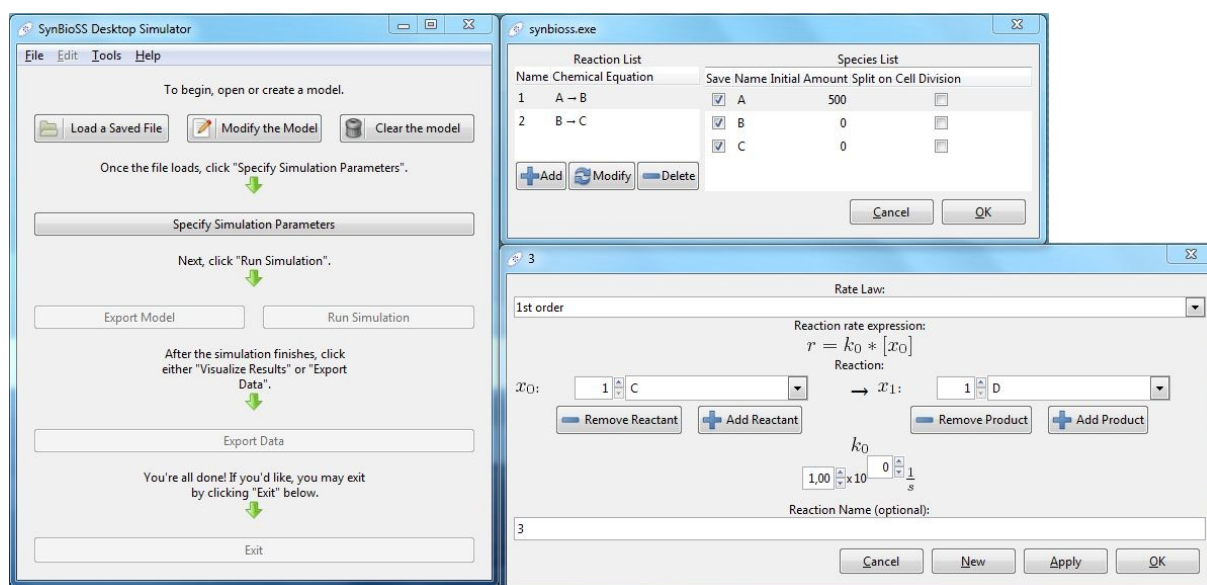

SynBioSS Desktop Simulator is an application that can be downloaded to simulate models generated with SynBioSS Designer. With it, the model previously created is loaded, and before simulating it, the initial parameters can be modified, and after, all the results can be exported to use them with other tools and make a more exhaustive study. The interface that SynBioss has is worse than TinkerCell, but it provides powerful mathematical methods and a good simulation report, showing a good review of the results obtained.

As a proof of concept, the parameterized study of these tools has been carried out by simulating the Toggle Switch, a synthetic biological circuit that is explained in the next section.

The toggle switch circuit

This synthetic biological system [Gardner, T.S. and Cantor, C.R. and Collins, J.J., 2000] consists of two genes, lacI and tetR, that are repressed between them, and a reporter called green fluorescent protein, that indicates which gene is activated. There are two inductors (aTc and IPTG), one for each gene. An inductor binds to the protein produced by its corresponding gene, preventing it to repress the other gene. This allows to change the active gene manually by inserting one inducer into the system.
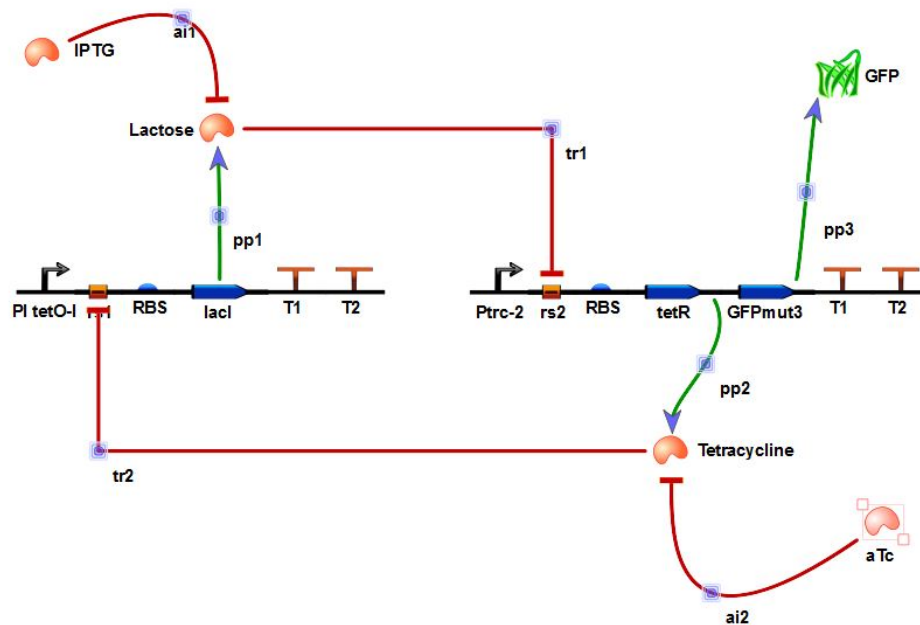
For example, the presence of aTc prevents Tetracycline to repress lacI, which freely produces Lactose and repress the production of tetracycline and green fluorescent protein. Moreover, the presence of IPTG causes the same efect with Lactose. In reality there are two kind of circuits, contained in two plasmids: pIKE and pTAK. pIKE was chosen
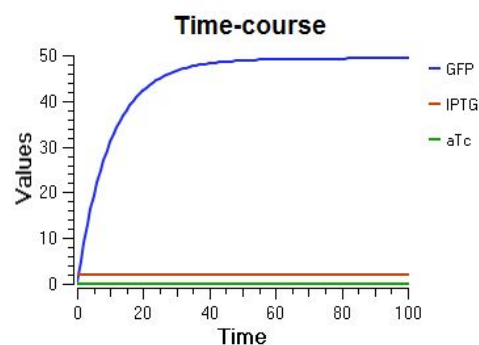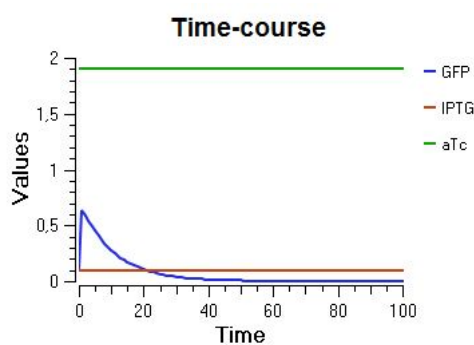
for the study, because the second uses a thermal pulse in place of aTc, which can not be modelled in the tools used for the simulation.

Comparison of the tools

TinkerCell provides a very suitable GUI for the use by most users. The modular framework is such an excellent way to work with diferent modules, study their behaviour and obtain reliable results with different levels of detail. The tool supports a lot of standards of Synthetic Biology (like the recent improvement of the use of SBML and Antimony scripts like imported modules), but in the future it has to be developed to support the major of the representations of Synthetic Biology Open Language (SBOL) and XML files in general.

Another great advantage of TinkerCell is its availability in the major paltforms: it can be run in Mac, Linux, and Windows (in 32 or 64 bits versions), making it more attractive and usefull, because, in adition with the free software and open project characteristics, it makes that many people can design and share biological parts for their designs. They are conected via remote databases too, making easier their own researches. SynBioSS has the same availability, and the extra feature that designing genetic circuits has to be done via Internet (the SynBioSS Designer can not be run in computers without Internet connection), which means that any specific machine is not needed to work. Plus, SynBioSS has a great database, the iGem's, which allows to share many biological parts with other researchers and get the suitable one.

Unless TinkerCell supports a lot of standards, a great improvement will be the possibility of supporting SBML files, making the tool more suitable for the different fields of Synthetic Biology, and simplifiying the spreading of the designs to other users that use other tools to develope their researches.

Comparative of the tools	
TinkerCell	✔ Suitable and smooth GUI that provides an easy way to create designs. ✔ Multiplatform. ✔ Several API's (C, Python). ✔ Flexible plug-in framework. ✖ Do not support SBML files to create the model, only to run simulations as scripts. ✖ WikiDust is not developed enough. ✖ It is not connected to MIT Biological Parts Database. ✖ Multiple stochastics simulation results are not allowed.
SynBioSS	✔ Web-Service tool that allows to run in any machine with internet conection. ✔ BioBricks are taken from Registry of Standard Biological Parts database. ✔ It allows to design own biological components. ✔ SBML files can be imported. ✖ Only genetic networks can be designed. ✖ SynBioSS-Wiki is not developed enough. ✖ External modules or scripts can not be used. ✖ Improved GUI.

Proposal of a new tool

Good simulations can be done using TinkerCell and SynbioSS simultaneously, so the method to do it is proposed here. First of all, a designer could construct an abstract model of the circuit, just thinking about the genes involved and their interactions. TinkerCell is siutable for this purpose. When the designer wants to take in account more details about the system he is simulating, SynbioSS designer is a good option, because he can choose real biobricks from the database it offers. Once the model is constructed and exported in SBML, TinkerCell can import it, but not directly. It has to use a python script included in the software that allows users to convert a SBML file into an antimony model, and simulate it. TinkerCell was chosen for this step because of its easiness to simulate and plot the results. But there is an issue in this step. TinkerCell can not show a graphical representation of antimony models, although it will be able to do it when the appropriate plug-in be developed. This joint use of the tools suggests about

a new simulation and modeling tool, but knowing that TinkerCell allows programmers to add new functionalities in a easy manner, it could be better to develop new modules for this tool that take the advantages of SynbioSS.

As it has been seen in the study, using both tools is needed to get complete results and extract conclusions. There is a limitation by the goal of the tools, which are made only for an specific purpose (for instance, genetic circuits in SynBioSS), plus, they are focused in the field of biology, and it is needed to abstract away: it consists of searching for a tool that, besides allowing to simulate biological networks and run simulations, it provides of bio inspired models to solve computationals and real problems. The goals to achieve in this simulation tool will be:

- Repository of bio inspired models, components and biological parts where people can freely access to check and extend their work, and can be used for future researches.

- Free database for researchers and experts to share knowledge.

- Open project to develop new features and create modules and extensions.

- Multiplatform and free software, which provides a GUI that allows a researcher of any field (biology, computer science, etc.) to work with the tool without the need of a complex background in CAD tools.

- Supporting new programming paradigms like DNA or bacterial computing.

- API from major language in the fields of computing and biology.

Broadly speaking, these are the characteristics that are expected. Also is a good idea that development of this tool is carried out by an interdisciplinary group, to contribute with ideas to suit the tool for all kind of researchers. The aim is to create a software that could be taken to simulate both the basic biological process to carry it out in a laboratory, and apply bio inspired models to solve problems, either real, computationals or mathematicals.

Conclusion

The main characteristics of the two CAD tools have been shown and compared simulating the toggle switch circuit, and based on the result a new tool is proposed, so it can be developed from scratch or improving an existing one, which allows not only designing and simulating new biological circuits implementing computationals models, but also lets researchers to create and check new bio inspired algorithms using resources they provide from a reliable biology knowledge, incorporating an important, additional feature: all the knowledge gained can be shared with the rest of the scientific community. This new algorithms will be useful for multiple areas (not only in the field of biology) like economics, mathematics, computer science, engineering or logistics. An alternative to the developement of new software could be using TinkerCell API: new modules implementing these features mentioned before could be developed and they would be useful to achieve this goal, using for it the typical standard formats in biology, such as SBML or SBOL.

As it has been seen in the results, CAD's tools generally provide a great GUI and multiple API's aim to simulate many biological process, that use good and realiable mathematical methods, but also lack a good repertory of useful biological and mathematical models that could be used in problems regardless of the nature of the problem or the field of study. The creation of a new databse with all kind of knowledge obtained (stored under the appropriate standards) will be a useful way to share freely all the technological advances.

Analyzing the characteristics of the tools explained in the paper, they suggest that a good approach is the developement of a tool that provides of the features explained before. For example, a typical mathematical model like predator-prey could be applied to the field of economics, or simulating a bacteria community (using swarm intelligence) that mimic some communication model, for instance. Anyway, the final goal is to achieve a tool that allows to use diferent models as modules, which could be loaded and simulate a process, using different parameters, variables and results of the simulation to interpret them as a study of the topic of interest.

The main goal is to generate a tool that provides of multiple bio inspired algorithms instead of focus in a traditional design program, because of the typical software architecture for this kind of programs is already implemented in lots

of CAD's, obtaining great results focusing only on a specific field of study, but not providing of designed models to help in general problems of real life, and forcing to spend too much time on process design. In a computational way is more interesting to have a multiplatform software that allows to simulate many kind of process and obtain reliable results that will be analyzed later, to solve, in a way reliable enough differents problems, and share the obtained results.

Acknowledgement

Bibliography

[Purnick and Weiss, 2009] Purnick, P. andWeiss, R. (2009). The second wave of synthetic biology: from modules to systems. Nature Reviews Molecular Cell Biology, 10(6):410-422.

[Karlsson and Weber, 2012] Karlsson, M. and Weber, W. (2012). Therapeutic synthetic gene networks. Current Opinion in Biotechnology.

[Lee et al., 2004] Lee, J., Shin, S., Park, T., and Zhang, B. (2004). Solving traveling salesman problems with dna molecules encoding numerical values. BioSystems, 78(1):39-47.

[Hug and Schuler, 2002] Hug, H. and Schuler, R. (2002). Dna-based parallel computation of simple arithmetic. DNA Computing, pages 321-328.

[Bach et al., 1996] Bach, E., Glaser, E., Condon, A., and Tanguay, C. (1996). Dna models and algorithms for npcomplete problems. In Computational Complexity, 1996. Proceedings., Eleventh Annual IEEE Conference on, pages 290-300. IEEE.

[Chandran, 2012] Chandran, D. (2012). TinkerCell: Computer-Aided Design for Synthetic Biology. University of Washington.

[Hill et al., 2008] Hill, A., Tomshine, J., Weeding, E., Sotiropoulos, V., and Kaznessis, Y. (2008). Synbioss: the synthetic biology modeling suite. Bioinformatics, 24(21):2551?2553.

[Bergmann and Sauro, 2006] Bergmann, F. and Sauro, H. (2006). Sbw-a modular framework for systems biology. In Proceedings of the 38th conference on Winter simulation, pages 1637-1645. Winter Simulation Conference.

[Funahashi et al., 2003] Funahashi, A., Morohashi, M., Kitano, H., and Tanimura, N. (2003). Celldesigner: a process diagram editor for gene-regulatory and biochemical networks. Biosilico, 1(5):159-162.

[Hucka et al., 2003] Hucka, M., Finney, A., Sauro, H., Bolouri, H., Doyle, J., Kitano, H., Arkin, A., Bornstein, B., Bray, D., Cornish-Bowden, A., et al. (2003). The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models. Bioinformatics, 19(4):524-531.

[Olivier et al., 2005] Olivier, B. G., Rohwer, J. M., and Hofmeyr, J.-H. S. (2005). Modelling cellular systems with PySCeS. Bioinformatics, 21(4):560-561.

[Weeding et al., 2010] Weeding, E., Houle, J., and Kaznessis, Y. (2010). Synbioss designer: a web-based tool for the automated generation of kinetic models for synthetic biological constructs. Briefings in bioinformatics, 11(4):394-402.

[Weiss et al., 2003] Weiss, R. and Basu, S. and Hooshangi, S. and Kalmbach, A. and Karig, D. and Mehreja, R. and Netravali, I. Y. (2003). Genetic circuit building blocks for cellular computation, communications, and signal processing. Natural Computing, 2(1):47-84.

[Goni-Moreno, A., 2010] Goni-Moreno, A. Y. (2010). Communication architectures for algorithmic computing in multi-strain bacterial communities. PhD thesis, Polytechnic University of Madrid

[Goni-Moreno, A. et al., 2012] Goni-Moreno, A. and Amos, M., Y. (2012). Continuous computation in engineered gene circuits. Biosystems.

[Baumgardner, J. and Acker et al., 2009] Baumgardner, J. and Acker, K. and Adefuye, O. and Crowley, S.T. and DeLoache, W. and Dickson, J.O. and Heard, L. and Martens, A.T. and Morton, N. and Ritter, M. and others, Y. (2003). Solving a Hamiltonian Path Problem with a bacterial computer. J Biol Eng, 3(11).

[Gardner, T.S. and Cantor, C.R. and Collins, J.J., 2000] Gardner, T.S. and Cantor, C.R. and Collins, J.J., Y. (2000). Construction of a genetic toggle switch in Escherichia coli. Nature, 403:339-342.

[Karr, J.R. et al., 2012] Karr, J.R. and Sanghvi, J.C. and Macklin, D.N. and Gutschow, M.V. and Jacobs, J.M. and Bolival, B. and Assad-Garcia, N. and Glass, J.I. and Covert, M.W. Y. (2012). A Whole-Cell Computational Model Predicts Phenotype from Genotype. Cell, 150(2):389-401.

Authors' Information

Victor de la Hoz - *Grupo de Computacion Natural, Facultad de Informatica, Universidad Politecnica de Madrid, Campus de Montegancedo s.n., 28660 Boadilla del Monte, Madrid, Spain;*
e-mail: v.delahoz@alumnos.upm.es

Cesar Pumar - *Grupo de Computacion Natural, Facultad de Informatica, Universidad Politecnica de Madrid, Campus de Montegancedo s.n., 28660 Boadilla del Monte, Madrid, Spain;*
e-mail: c.pumar@alumnos.upm.es

Paula Cordero - *Grupo de Computacion Natural, Facultad de Informatica, Universidad Politecnica de Madrid, Campus de Montegancedo s.n., 28660 Boadilla del Monte, Madrid, Spain;*
e-mail: paula.cordero@knowdle.com

Nicolas Quintero - *Grupo de Computacion Natural, Facultad de Ciencias Matematicas, Universidad Complutense de Madrid, Plaza de Ciencias 3, 28040 Madrid, Madrid, Spain;*
e-mail: nequintero@estumail.ucm.es