

Algebraic Modeling of Molecular Interactions

Oleksandr Letychevskiy¹, Vladislav Volkov², Yuliia Tarasich¹, Hanna Sokolova³ and Volodymyr Peschanenko¹

¹ Private Enterprise LitSoft, ap. 27, 83 Mykhailivska str., Kherson, 73000, Ukraine

² V.M.Glushkov Institute of Cybernetics of the NAS of Ukraine, 54 Volodymyrska str., Kyiv, 01030, Ukraine

³ Kherson State University, 27 University str., Kherson, 73003, Ukraine

Abstract

The approach of algebraic modeling of molecular interactions in some environment to determine the triggering of the studied properties is considered in the present article. The main idea of this study is to represent the actions of elementary particles in different molecular structures, in particular the motion of electrons in orbitals as algebraic equations for further processing. Behavior algebra specifications are used as the modeling language. The article also describes the formalization of the examples of atoms interaction (creating of chemical bonds) on the example of Ionic Bond. The formalization and properties analysis is considered with the usage of the insertion modeling platform. The study is at an early stage of development and the approach is demonstrated by some examples.

Keywords

formal verification, symbolic modeling, algebraic behavior, molecular models.

1. Introduction

Modeling of biochemical processes, in particular processes in the cell, is gaining importance in connection with the emergence of new viruses, cancer treatment, and other problems in the human community. Presentation of knowledge in the chemistry of organic substances has already acquired world standards, the existing data banks of proteins, enzymes, and many other substances are actively used in the research.

Today modeling is one of the main research methods. As an example, we note the award of the 2013 Nobel Prize in Chemistry for the development of multi-scale models of complex chemical systems.

Due to the high complexity a large number of biology and chemistry tasks cannot be effectively solved using traditional modeling methods due to the infinite number of possible scenarios of object behavior.

Algebraic modeling, in particular, the algebraic modeling of virus and cell interaction, which is the main goal of our research, has a number of significant advantages and, in particular, is characterized by high efficiency, speed, safety, and scalability of the experiment, which makes it possible to effectively search for means of counteracting the penetration, replication, and release of the virus, and the conditions under which such counteraction is possible at the molecular and cellular levels.

Formal verification anticipates the checking such properties of the system as the absence of deadlocks and non-determinisms, safety or liveness conditions. We can prove the reachability of demanded properties by using a symbolic modeling method developed in the scope of IMS [18] that involves the forward and backward symbolic modeling, the proving of the satisfiability of the property violation formula, invariant computation.

ICTERI-2021, Vol II: Workshops, September 28 – October 2, 2021, Kherson, Ukraine

EMAIL: oleksandr.letychevskiy@litsoft.com.ua (A. 1); vlad@issukraine.com (A. 2); yutarasich@mail.com (A. 3); annetsokolova96@mail.com (A. 4); vpeschanenko@mail.com (A. 5);

ORCID: 0000-0003-0856-9771 (A. 1); 0000-0002-4766-9284 (A. 2); 0000-0002-6201-4569 (A. 3); 0000-0001-9622-310X (A. 4); 0000-0003-1013-9877 (A. 5)



© 2021 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

This research considers the application of algebraic modeling of molecular interactions in some environment to determine the triggering of the studied properties. The idea is to represent the actions of elementary particles in different molecular structures, in particular the motion of electrons in orbitals, as algebraic equations for further processing.

We consider the possibility of studying arbitrary substances from the point of view of the possibility of their interaction based on symbolic modeling. Considering the possible interactions in the system it is possible to obtain the results of various reactions at the level of interaction of atoms, ions, and molecules as the sought or analyzed properties.

In this article we present our approach and model of atoms and substances interaction. It's the first step to the modeling of interaction of agents of high organization level such as agent-virus or a similar multilevel agent-cell.

2. Related works

The application of modeling methods for both continuous and discrete models takes place using different approaches: statistical, probabilistic, simulation, visual. Formal mathematical specifications for the description of knowledge of the behavior of atoms, molecules, and ions are used in software systems. There are language standards of formalized chemical and biological objects, such as the SBML model format for sharing and storing of biological models, which has extensive system support.

Systems such as CellDesigner, JDesigner, PathwayLab, PathwayAnalyser, Bio-Chem and others are used to visualize the kinetic modeling of biological systems.

Tools such as Cobra, CellNetAnalyser are compatible with such mathematical systems as MathLab, Gepasi, Copasi, GENESIS / Kinetiki, Jarnac are the software for modeling of biochemical processes. There is a number of software systems which are used for analysis and modeling of proteins: CAVER, SABER, POCKETOPTIMIZER, Protein WISDOM, EVODESIGN. As a tool for formal modeling and analysis of biological systems that exhibit random or probabilistic behavior also using a PRISM (a probabilistic model checker).

Software for structural analysis of biochemical networks is considered in [1]. The authors focus on the review of software for simulation and modeling of biochemical networks, in particular on tools which include topological analysis of network structure.

In the article [2] the analysis of modeling tools intended for modeling of biochemical systems is given. Tools such as GENESIS / Kinetiki, Jarnac / JDesign, Gepasi, SimTool were selected for analysis. Using each tool, time series simulations were performed using a specific pathway describing the concentration of the active form of the protein kinase.

In [3, 4] the functional features of the BioNetGen tool are considered.

An overview of the set of modeling tools is also presented in [5]. The main issue under discussion is the relationship between complex mathematical models and known arithmetic concepts in classical enzymology, as well as the relationship between modeling and experimental data. The authors note the serious underestimation and initial stage of developing the tools needed for effective modeling.

The use of pi-calculus as an alternative model to using differential equations or Monte-Carlo simulations was proposed by authors of BioSPI project [6]. Authors employ the stochastic pi-calculus and the ambient calculus and present results of the use of process algebras for modeling biological systems. The abstract machine for a variant of the stochastic pi-calculus for model the stochastic simulation of biological processes was presented in [7].

One of the most used tools is the BioChem system [8] for biochemical research. BioChem uses temporal logic to formalize biochemical systems and various modeling methods.

The development of algebraic systems - machines of solution, automatic proof of theorems began a new study using symbolic modeling, which allowed to derive the necessary knowledge from many formalized laws. The usage of algebraic techniques makes it possible to conduct research at different levels of abstraction and to operate with infinite entities. Along with deductive tools a modern computer science is actively using machine learning that has led to emergence of new research methods in chemistry and biology.

The present study is a continuation of the previous authors works where the used methods of algebraic programming and insertion modeling for verification and simulation of crypto-economics [9-11], legal, economics models [12], hardware systems [13] were presented.

3. Modeling of molecular interactions. Algebraic approach

We consider the theory of agents and environments [14] where the agent is presented as a transition system with the states defined as algebraic equations. Agents are interacting in some multilevel environment, which can be considered as an agent for the other environment. In our formalization, we have an environment with a set of different substances. Every substance contains some number of molecules, ionic, atom or metallic compounds. Substances are the agents in some environment where the substance is interacting being the environment for other agents which are atoms. In addition to agents that were specified in the model, other types of agents can be identified too. It depends on the formalized model, the level of abstraction, etc.

In a model, we define a type of an agent by the set of attributes of different types. We consider the integer, real, boolean, enumerated attributes. Some attributes are parametrized and considered as functional symbols. The types ATOM and SUBSTANCE are defined as the following agents:

```
ATOM:obj(  
  name: Periodic_Elements,  
  Orbital (int, int) -> int,  
  Levels: int,  
  valence:int,  
  spin: real,  
  electronegativity: real,  
  metal: bool,  
  elNum: int  
)
```

The attribute *name* is defined by enumeration type **Periodic_Elements** which contains all names of elements of the periodic table { H, He, Li, Be, B, C, N, O, F, Ne, Na, Mg,... } etc.

Orbital(I,J) is a functional symbol which defines the orbitals of an atom (*I* is the level number, and *J* is the cell number in sublevels, where it can be located up to two electrons).

Levels are the atom energetic levels.

Attribute **valence** is necessary for bonds establishment because it depends on the number of free electrons (electrons at the last level) and has different values for some atoms. It's calculated according to the electronic formulae of elements.

Spin is electron spin.

Electronegativity is atom's electronegativity.

Metal is an auxiliary attribute for determining the type of a bond.

elNum: is an auxiliary attribute to determine the number of level cells.

The agent SUBSTANCE is the following.

```
SUBSTANCE:obj(  
  atoms: int -> ATOM,  
  SubOrbital: (ATOM,ATOM) -> int,  
  AtNum: int  
)
```

Attribute **atoms** is a functional symbol which defines the set of atoms in a molecule.

SubOrbital (I,J) is the number of orbitals which are common to the *I* and *J* atoms.

AtNum is an auxiliary attribute to determine the number of atoms in the molecule.

The first step of our work is the modeling of the chemical bonds. So, the possible actions that illustrate the chemicals bonds are the following: **Covalent bond** (*Nonpolarized covalent bond* and *Polarized covalent bond*); **Ionic bond**; **Metallic bond** [15, 16].

To formalize chemical bonds we use the algebra of behaviors.

The notion of behavior algebra was introduced in 1997 by Gilbert and Letichevsky [14]. It was realized in the scope of the IMS [18].

Behavior algebra is a two-sorted universal algebra. The main sort is a set of behaviors and the second sort is a set of actions. This algebra has two operations (prefixing $a.u$ (where a is an action and u is a behavior) and nondeterministic choice of behaviors $u + v$), three terminal constants (successful termination Δ , deadlock 0 , and divergent behavior \perp), and an approximation relation.

The approximation relation \sqsubseteq is a partial order on the set of behaviors with the minimal element \perp . The behavior algebra is also enriched by two operations: parallel (\parallel) and sequential ($;$) compositions of behaviors.

The action language is built over some attribute environment in which all agents interact with one another. Every agent's actions are represented by the triple $B = \langle P, A, S \rangle$.

P is a precondition of the action and is presented as a formula in some basic logic language. The basic logical language is the set of formulae of first-order logic over polynomial arithmetic or other theories like enumerated types.

S - a postcondition which is also a formula.

A is a process or any text string that illustrates the agent transition. Thus, the agent can change its state if the precondition is true and the state will be changed correspondingly to the postcondition.

The action contains the semantics of electron motion that entails connection of atoms and creation or decay of substances. It consists of the pre-and postcondition being the formulae over the attributes of some type.

Let's consider the main actions of the atoms' interaction in the example of the **Ionic bond** actions.

1. CreateIMol1

```
CreateIMol1 = Forall(i:ATOM, j:ATOM) (  
(i != j && CONNECT(i,j) && i.valence != 0 &&  
j.valence != 0 && i.ready > 0 && j.ready > 0 &&  
i.metal != j.metal) ->  
    "ATOM#A: action 'IonicBond';"  
    (S1.atoms(S1.AtNum-1) = i;  
    S1.atoms(S1.AtNum) = j)  
) ,
```

The predicate `CONNECT (i, j)` - determines that the agents **i**, **j** have approached the communication distance and are ready to react.

If the agents are at a bonding distance, have a non-zero valence, one of them is a metal and both agents are ready to react - we get the substance with ionic bond. Speaking about the readiness of agents-atoms to create a bond, we set that between the many agents-atoms, atoms with the highest and lowest electronegativity are combining at first. To identify such agents and to know who will donate free electrons and who will receive, we introduce an additional attribute **ready**. Thus, for the agent with the highest electronegativity the value of **ready** is 1, with the lowest - 2.

If the precondition of the reaction is satisfiable, then the reaction is allowed and there are changes in the environment according to the postcondition. In this action we have established the set of atoms of substance that were bonded.

2. RewriteAtom1Orbitals1

```
RewriteAtom1Orbitals1 = Forall(i:ATOM, j:ATOM, k:int, m:int) ( 1<=k<=  
i.elNum && 1<= m <= j.valence && i != j && i.ready == 1 &&  
((S1.atoms(S1.AtNum-1) == i &&  
S1.atoms(S1.AtNum) == j) || (S1.atoms(S1.AtNum) == i && S1.atoms(S1.AtNum-  
1) == j)) &&  
i.Orbital(i.levels, k) == 1) ->  
    "ATOM#A: action 'IonicBond';"  
    ( i.Orbital(i.levels, k) = 2)
```

)

If atoms are connected by an ionic bond, one of them gives away free electrons and the other receive. Accordingly, we must change the number of electrons in their orbitals. In this action we check the membership of the atom to the created substance, by checking that this atom receives electrons from another atom (ready = 1) and change the value of electrons number for cells of level that contained one electron - i.Orbital (i.levels, k) = 2.

Then we have to recalculate the valence of atoms and to check that if all the atoms that make up the substance have a valence "0", then a new substance is created, if not, then it's a radical and we can add more atoms that will complement the required orbital.

As was mentioned above, the first experiment is to perform interactions between atoms. It implements the types of chemical bonds. We use the behavior algebra that defines the behavior of the agents in some environment. Every behavior consists of the actions and other behaviors and is presented as the system of behavioral equations. For example the behavior of the atoms in some environment is the following:

```
CREATE_BOND = (ChooseAtoms; ( (CovalentPolarBond + CovalentNonPolarBond)
+ MetalBond + IonicBond) )
...
IonicBond = (CreateIMol ; ChekIMol)
CreateIMol = CreateIMol1 . RewriteAtom1Orbitals1 . RewriteAtom2Orbitals1
...
```

So, we have some set of different atoms in some environment at different distance from each other. Firstly we choose the atoms that are in the communication distance and have maximal and minimal electronegativity (action **ChooseAtoms**). Next, according to their type (is metal or not) we can choose and model the type of chemical bond.

We provide symbolic modeling for arbitrary number of atoms and for working with the set of states of environment. This means that we can conduct experiments for different sets of atoms, the number and values of the attributes of which are set for each experiment. We can abstract from complicated formulae of calculating the distance or from the speed of particles. During symbolic modeling as result we get a set of different substances of molecular and non-molecular structure and we can check some property, for example, the possibility of the appearance of some substance or the amount of some substance.

For molecular substances we also use modeling of intermolecular interaction (electrostatic, polarization (induction) and dispersion; as well as exchange interaction and interactions due to the transfer of electronic charge) [19 - 21]. Those, we can modeling and found the values of the energy of highest and lowest occupied molecular orbitals, energy difference, dipole moment, electronegativity and etc.

The next experiments are to consider the set of different substances. How the interactions will be performed with the corresponding reactions? It will be defined as the possible resolution of behavioral equations.

The next stage is to consider the quantity of substance with different structure (molecular, ionic, atom or metallic). The reactions between them will take time and the speed of reaction depends of the environment (temperature, pressure) and is defined by the following equations:

— The rate of a chemical reaction (V_R):

$$V_R = -\frac{dC_R}{dt},$$

where C_R is the molar concentration of the reagent, and t is the time.

— Influence of temperature - Vant-Goff's rule:

$$V_{t_2} = V_{t_1} * \gamma^{\frac{t_2 - t_1}{10}},$$

where V_{t_2} is the reaction rate at a temperature t_2 , V_{t_1} - reaction rate at temperature t_1 , γ - temperature coefficient of reaction.

— Influence of the catalyst - Michaelis-Menten equation:

$$V = \frac{d[P]}{dt} = \frac{V_{max}[S]}{K_m + [S]},$$

where V_{max} is the maximum reaction rate, which is observed when the enzyme is completely saturated with the substrate, K_m - Michaelis constant - the concentration of the substrate in which the reaction rate is equal to half of the maximum, t is the time, P - product, S - substrate.

For modeling we consider the hybrid system that defines continuous and discrete processes. When some portion of substance starts to react with the other then the continuous process of reaction is triggered. Then we can obtain the other substances depending from the environment and the quantity of new substances will be changed. By modeling of such reactions we can resolve the problem of triggering of some conditions or properties.

For example we need to detect whether some substance reaches some critical quantity. It will be defined by the formula:

$$Q > \text{critical}.$$

4. Conclusion

Providing hybrid algebraic modeling we can detect whether the conjunction of environment that is defined by the values of agent attributes and the sought-for formula is true.

At this stage of our study we can make the experiments with different numbers of atoms and substances (such as molecules, radicals, metals, ions). We can get them from the world databases in the MOL format and translate to the language of behavior algebra. After that we can do modeling of their interaction (according to the chemical laws) and to study their properties. For the modeling substances interaction we use symbolic modeling combined with resolution of differential equation that models reaction.

After finishing of this part of study we plan to create the agent of high organization level. As example it will be the agent-virus, being an environment for agent of Substance type, or a similar multilevel agent-cell.

5. Acknowledgements

We would like to thank the company Private Enterprise LitSoft [22] for the opportunity to work with the platform for modeling of algebraic behavior for our research and experiments in the modeling area. We are also grateful to the Glushkov Institute of Cybernetics of the NAS of Ukraine for the theoretical and practical results in the field of verification that were used as a basis for our studies of formalization and algebraic modeling in the tokenomics projects area and to Kherson State University for the active supporting of Insertion Modeling System.

6. References

- [1] T. Rubina, E. Stalidzans, Software tools for structure analysis of biochemical networks, in: Proceedings of Applied Information and Communication Technologies 2010, Jelgava, Latvia, 2010, pp. 33–49.
- [2] Antti Pettinen, Tommi Aho, Olli-Pekka Smolander, Tiina Manninen, Antti Saarinen, Kaisa-Leena Taattola, Olli Yli-Harja, Marja-Leena Linne, Simulation tools for biochemical networks: evaluation of performance and usability, *Bioinformatics* 21(3) (2005) 357–363. doi:10.1093/bioinformatics/bti018.
- [3] J.R. Faeder, M.L. Blinov, W.S. Hlavacek, Rule-Based Modeling of Biochemical Systems with BioNetGen. In: I. Maly (Eds.) *Systems Biology*, volume 500 of *Methods in Molecular Biology (Methods and Protocols)*, Humana Press, 2009, pp. 113–167. doi:10.1007/978-1-59745-525-1_5.
- [4] M. L. Blinov, J. R. Faeder, B. Goldstein, W. S. Hlavacek, BioNetGen: software for rule-based modeling of signal transduction based on the interactions of molecular domains, *Bioinformatics* 20(17) (2004) 3289–3291. doi:10.1093/bioinformatics/bth378.

- [5] W. W. Chen, M. Niepel, P. K. Sorger, Classic and contemporary approaches to modeling biochemical reactions, *Genes & development* 24(17) (2010) 1861-1875. doi:10.1101/gad.1945410
- [6] THE BioSPI PROJECT, URL: http://www.wisdom.weizmann.ac.il/~biospi/index_main.html.
- [7] A. Phillips, L. Cardelli, A correct abstract machine for the stochastic pi-calculus, *Bioconcur'04, ENTCS*, 2004.
- [8] System Modeler Library Store, URL: <https://www.wolfram.com/system-modeler/libraries/biochem>.
- [9] O. Letychevsky, V. Peschanenko, M. Poltoratzkyi, Y. Tarasich, Our Approach to Formal Verification of Token Economy Models, in: *International Conference on Information and Communication Technologies in Education, Research, and Industrial Applications*, volume 1175 of *Communications in Computer and Information Science*, Springer, Cham, 2019, pp. 348-363. doi:10.1007/978-3-030-39459-2_16.
- [10] O. Letychevsky, V. Peschanenko, V. Radchenko, M. Poltoratzkyi, P. Kovalenko, S. Mogytko, Formal Verification of Token Economy Models, in: *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, IEEE, Seoul, Korea (South), 2019, pp. 201-204. doi:10.1109/BLOC.2019.8751318.
- [11] O. Letychevskyi, V. Peschanenko, V. Radchenko, M. Poltoratskiy, Yu. Tarasich, Formalization and Algebraic Modeling of Tokenomics Projects, in: *ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer*, Kherson, Ukraine, 2019, pp. 577-584.
- [12] A. Letychevsky, O. Letychevskyi, V. Peschanenko, M. Poltorackij, An Algebraic Approach for Analyzing of Legal Requirements, in: *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*, IEEE, Lisbon, Portugal, 2017, pp. 209-212. doi:10.1109/REW.2017.51.
- [13] V. Kharchenko, O. Letychevskyi, O. Odarushchenko, V. Peschanenko, V. Volkov, Modeling method for development of digital system algorithms based on programmable logic devices, *Cybernetics and System Analysis* 56 (2020) 710-717. doi:10.1007/s10559-020-00289-8.
- [14] D. Gilbert, A. Letychevsky, A model for interaction of agents and environments, in: D. Bert, C. Choppy (Eds.), *Recent Trends in Algebraic Development Techniques*, volume 1827 of *LNCS*, Springer-Verlag, Cham, Switzerland, 1999, pp. 311-328. doi:10.1007/978-3-540-44616-3_18
- [15] Richard F. W. Bader, *Atoms in Molecules. A Quantum Theory*, Oxford University Press, Oxford, 1990
- [16] P. S. V. Kumar, V. Raghavendra, V. Subramanian, Bader's theory of atoms in molecules (AIM) and its applications to chemical bonding, *Journal of Chemical Sciences* 128(10) (2016) 1527-1536. doi:10.1007/s12039-016-1172-3.
- [17] A. Letychevsky, D. Gilbert, A general theory of action languages. *Cybernetics and Systems Analysis*, 34 (1998) 12-30. doi:10.1007/BF02911258.
- [18] A. Letychevsky, O. Letychevskyi, V. Peschanenko, Insertion modeling and its applications, *Computer Science Journal of Moldova* 72(3) (2016) 357-370.
- [19] R. Rein, M. Pollak, *Molecular-Orbital Studies of Intermolecular Interaction Energies. I. On the Role of π Electrons in Intermolecular Interactions, with Some Application to DNA*, *The Journal of chemical physics* 47(6) (1967) 2039-2044.
- [20] I.G. Kaplan, *Introduction to the theory of intermolecular interactions*, Nauka Press, Moscow, 1982.
- [21] H. Ratajczak, W. J. Orville-Thomas (Eds.) *Molecular interactions. Vol. 1*. John Wiley & Sons Ltd., Chichester, New York, Brisbane, Toronto, 1980.
- [22] LitSoft Enterprise R&D, URL: <http://litsoft.com.ua/>.