Formalization and Algebraic Modeling of Tokenomics Projects

Oleksandr Letychevskyi[1[0000-0003-0856-9771]], Volodymyr Peschanenko[1[0000-0003-1013-9877]], Viktor Radchenko[1[0000-0003-4218-5376]], Maksym Poltoratskyi[2[0000-0001-9861-4438]] and Yuliia Tarasich[2[0000-0002-6201-4569]]

[1]Garuda AI V.B., Het Poortgebouw, Beech Avenue 54-62, Schiphol, 1119 PW, Netherlands
[2]Kherson State University, Universytets'ka St. 27, 73000, Kherson, Ukraine

{oleksandr.letychevskyi,volodymyr.peschanenko,viktor.radchenko}@garuda.ai, {mpoltorackiy,YuTarasich}@gmail.com

Abstract. This article provides a brief description of the technology and the methods and tools developed by the authors for token economy modeling and for the analysis and study of its properties. The article also describes the formalization of the tokenomics model on the example of the SKILLONOMY project and presents the specific and symbolic SKILLONOMY models and its simulation results.

Keywords: tokenomic modeling, blockchain, tokenomics, crypto-economics, symbolic modeling, insertion modeling.

Introduction

The development of blockchain technologies has led to the development of a new direction in the economy, referred to as crypto-economics or "tokenomics". Developing an effective tokenomics model is key to the long-term life of the blockchain project. Thus, it is important to understand and to analyze the effectiveness of a crypto-economy model.

There are many approaches to check the crypto-economy models, including games theory, probability and modeling approaches.

A review of the current state of information about the approaches, methods and tools for analyzing and simulating crypto-economic shows that the question of the right approach is still a real open question, with concerns about open access to a wide audience. The largest number of authors use the agent modeling as a tool for analysis. In [1], approaches for analyzing various crypto-economics are described. The author solves the problem associated with the modeling of microtokenomics using agent-based modeling. To optimize the analysis using agent-based modeling, the author also uses machine learning methods and genetic algorithms. In [2], the economic system as a discrete finite automaton is considered. The principle of simulation is based on the following methods: deterministic stochastic processes, stochastic processes and the Monte Carlo Simulation. Some examples of visualization and simulation modeling were also described in [3] and evaluated in [4]. Thus, we can conclude that there is still a relevant need for tokenomics models that can verify and simulate and that new solutions are required. The authors of this article proposes using methods of algebraic programming and insertion modeling [5] for verification and simulation of crypto-economics models using the example of the SKILLONOMY project [6].

In the first section of this article, a brief description of the proposed methods and tools is given. The second section of this article describes the formalization of the tokenomics model in the SKILLONOMY project. In the third and fourth sections, the specific and symbolic SKILLONOMY models and their simulation results are described.

1 Our contribution

We propose an algebraic approach to tokenomic modeling that is implemented in the scope of the insertion modeling system (IMS) [5].Insertion modeling focuses on building models and studying the interaction of agents and environments in complex distributed multi-agent systems [5,7].

With regard to mathematical refinements, we chose a transition system for the agent that is the most abstract mathematical concept, modeling a system that evolves over time. This approach is used for the formalization stage and the verification stage of the properties, which are the main stages of tokenomics analysis.

We used the behavior algebra specifications for the formalization for the insertion modeling method [7] and the deductive or symbolic method in IMS based on the such external provers and solvers, as Presburger – omega [8], Fourier-Motzkin - reallib (our tool), cvc3 [9], z3 [10] and MathSAT [11].

We considered the composition of the language of the basic protocols along with the language of UCM diagrams [12, 13] as the language for the formal presentation of system specifications. We considered the set of formulas of first-order logic over polynomial arithmetic as a basic logical language.

All the basic conceptions, such as environment, agents, basic protocols, behavior-algebra, etc., are considered in the SKILLONOMY project formalization example described in the next sections of the article.

2 Formalization of the tokenomics model

SKILLONOMY project is an educational online platform that tokenizes productive activities in the learning process and is focused on gaining monetized online knowledge and skills. The SKILLONOMY ecosystem is built around an IT platform that allows participants to effectively build and administer the relationships that are related to training, investing and sharing experience.

Developing the SKILLONOMY project required a set of essential functions of the blockchain for ensuring a stable and efficient system that works[6].

The main purposes of the tokenomics model formalization of the SKILLONOMY project are:

- the search for modeling errors, such as finding failings or possible contradictions;
- the search for effective scenarios of the system in the model, etc.;
- the possibilities for analyzing and predicting the model; and
- the possibilities for analyzing the feasibility of project financing.

The process of the formalization of the tokenomic project consists of the following steps: selection of the agents and definition of their attributes corresponding to the level of abstraction demanded, definition of agents' actions and the design of agents' behavior.

According to the project requirements, we determined the next set of agents for this model: coaches, managers, node owners, the platform owner, holders of tokens and the stock exchange, and students. Six categories of students are described: the first agent includes students whose average grade is equal 1, the second agent includes students whose average grade is equal 2, the third agent includes students whose average grade is equal 3, the fourth agent includes students whose average grade is equal 4, the fifth agent includes students whose average grade is between 3 and 4 and the sixth agent includes students whose average grade is equal 5.

The gent description will be present in the IMS language in the following form:

```
PLATFORM_OWNER
{
SMInvestToken: (int) -> real,
tokenICOStageEmission: (ICO_STAGE) -> real,
commonEmission: real,
 RESERVE: real.
...}
```

The interaction between agents is performed by the Basic Protocols (BP).

For this model, we have more than forty protocols. Examples of two of them are provided below:

- **studentONE_BuyTokens** — Starting from the seventh month (the end of the closed ICO), students (students whose average grade is equal 1) acquire tokens to replenish their skillmining account. Node receives cash income.

```
studen-
tONE_BuyTokens=Operator(((month>6)&(std_o.tokenAvailable<basicNe
eds*percStdON))->
    <("Purchase of tokens by students one")>
(std_o.tokenAvailable:=std_o.tokenAvailable+percStdON* basic-
      Needs - std_o.tokenAvailable;
listPurchases(ONE):=listPurchases(ONE) + (percStdON * basicNeeds
      - std_o.tokenAvailable) * tokenPrice;
```

```
stock.tokens:=stock.tokens - (percStdON *basicNeeds -
       std_o.tokenAvailable);
stock.bought:=stock.bought + (percStdON *basicNeeds -
       std_o.tokenAvailable))
```

- **stageEmission** — emits tokens for sale in periods (SEED, PRE_TGE, TGE_OPEN_SALE) in the first month of the period in which the emission is planned. In the precondition of these basic protocols we used the existential quantifier "Exist."

```
stageEmission=Operator(Exist
(i:ICO_STAGE)(((month=monthStartStage(i)))->
    <"Emission by period">
(platform.tokenICOStageForSale(i):=
             platform.tokenICOStageEmission(i);
platform.commonEmission:=platform.commonEmission -
             platform.tokenICOStageEmission(i))))
```

At the highest level, the SKILLONOMY model can be represented as a sequential and parallel composition of behaviors in the following form (this is the expression of behaviors algebra):

```
B0 = (UnlockBeh; EmissionBeh; ReductionBeh; {SaleBeh || Skill-
      MiningBeh || StockExchange }; PriceBeh;
      AddNewStdBeh; B2),
```

where: UnlockBeh — behavior of token unlocking, EmissionBeh — behavior of token emissions, SaleBeh — behavior of token sale, ReductionBeh — behavior of recounting the number of students when they reach their limit of purchasing ability; StockExchange — behavior of buying and selling tokens for the agent holders, SkillMiningBeh — Skillmining behavior, PriceBeh — behavior of token price change, B2 — behavior for the month counter and AddNewStdBeh — behavior of recounting the number of students after the arrival of new users on the platform. Each of these behaviors, in turn, represents a finite sequence of protocols. As example, lets describe the EmissionBeh behavior:

```
EmissionBeh = stageEmission.EB1 + not_stageEmission.EB1,
EB1 =  tokenHolders.EB2 + !tokenHolders.EB2,
EB2 =  skillMiningInvest + not_skillMiningInvest
```

where stageEmission, not_stageEmission, tokenHolders, skillMiningInvest and not_skillMiningInvest are the basic protocols.

Initial values for the initialization of the environment are given in a special logic formula. The construction of a specific model involves the determination of specific values of the agents and the environment attributes. Some values were defined initially in the documentation, but other parts were determined by conducting relevant surveys for target groups. The description, analysis and results from using the data obtained from the surveys is presented in the next section of the article.

3 Specific model. Simulation results

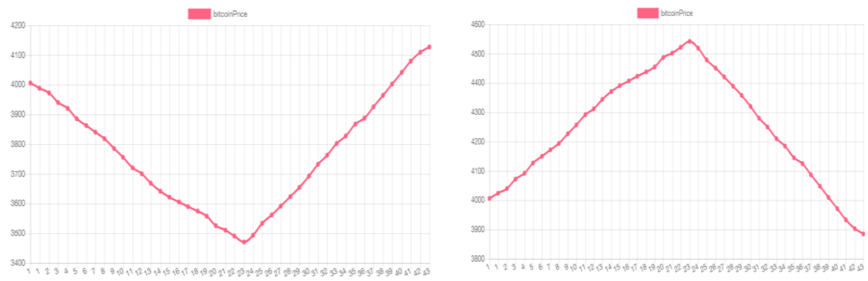
As described above, the construction of a specific model involves the determination of specific values of the agents and environment attributes. So, for example, we must determine the required ranges of the transfer of tokens between agents. Thus, it became necessary to conduct surveys, allowing us to determine the initial values of a number of attributes, such as the limit of students' ability to buy, the desire to sell to make a profit with an attachment to the time interval and the ratio of cashable profit to the income received, etc.

As a result of the simulation of the model and with the selection of different values of attributes, we were able to analyze the behavior of the SKILLONOMY model according to different bitcoin trends.

When we model the specifications, we simulated the activity of all agents that take part in such processes as sending and receiving tokens and selling and purchasing tokens on the stock exchange. In the process of modeling, we were able to monitor the dynamics of attributes and main tokenomic indicators.

We determined four different bitcoin trends: 1.The trajectory of bitcoin price is evenly falls up to the twenty-third month, and after this it evenly increases until to forty-third month; 2. The trajectory of bitcoin price evenly grows up to twenty-third month, and after this it evenly falls until to forty-third month; 3. There is uniform growth in bitcoin price; 4. There is a uniform fall in bitcoin price.

To better visualize the results we obtained, let us consider the behavior of the model according to first and second trends we noted. The trajectory of bitcoin price for first and second trend presented in Fig.1.

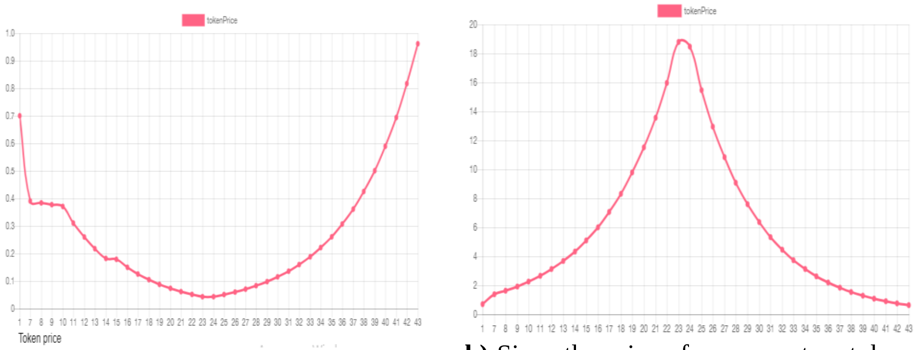


a) The initial price of bitcoin is 4006.1. From the first to the twenty-third month, the price of bitcoin falls to 3470.4. Between the twenty-third and the forty-third months, the price of bitcoin increases to 4127.4

b) The initial price of bitcoin is 4006.1. From the first to the twenty-third month, the price of bitcoin increases to 4541.8. Between the twenty-third and the forty-third months, the price of bitcoin falls to 3884.8

Fig.1. The trajectory of bitcoin price

The trajectory of the token price (based on the price of bitcoin and on the demand for a token inside the ecosystem) is represented in Fig.2.

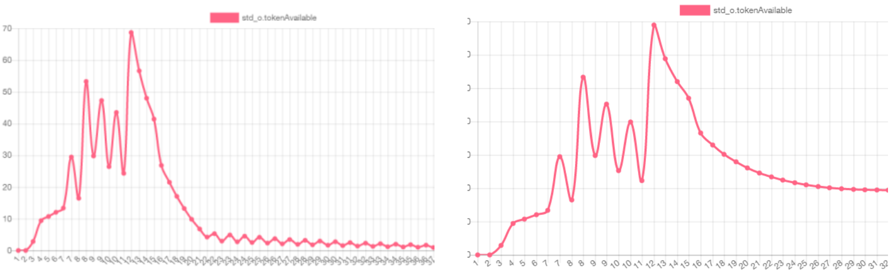


a) Since the price of an ecosystem token reacts to the fall of bitcoin price, the price of the token falls below the starting price up to twenty-third month. After this, the price evenly increases.

b) Since the price of an ecosystem token reacts to increases in bitcoin price, the price of the token increases up to the twenty-third month and then starts to fall almost to the initial price.

Fig.2. The trajectory of the token price

In our minds, the most interesting trajectory of the available tokens for students is the trajectory for students whose marks are equal (1), because the trajectory depends more on the bitcoins and the token prices than for other categories of students.



a) b)

Fig.3. The trajectory of the token price

As we can see in Fig.3 (part a), the students in this category do not approach the limit of purchased ability immediately because the token price is not large. Therefore, from the twenty-third month we see a small amount volatility. The students buy the missing tokens for mining. As seen in the part b of Fig.3, when the price of tokens has increased and is quite high, students in this category begin to buy the missing tokens for mining. When they buy tokens, they exceed their purchasing ability and begin to leave the system. The number of students becomes so small that they have enough of what they need for mining. Therefore, we do not see volatility on this chart.

Users can come to these and other conclusions using this process of studying the different attributes of agents and changing the initial values used in this project. This model allows changing hypotheses to evaluate the risks when selecting the worst conditions in the process of tokenomic modeling. One of most important advantages of tokenomic modeling is the opportunities to debug the system and to change the algorithm or boundary values of attributes to reach the demanded results.

4 Symbolic model

Unlike in concrete modeling, symbolic modeling provides us with opportunities to check the reliability and stability of our model.

The main feature of the system soundness is retaining the tokenomics indicators. Symbolic modeling cannot give us the opportunities to create charts as concrete modeling can, but symbolic modeling allows us to create formulae that characterize the monitored attributes. Thus, for this project all unknown parameters where considered in the supposed and desirable boundaries of the main indicators were checked.

As an example, let us consider such parameters as:

1) the price of a token, **2**) the portion of the sold tokens that are open sale and **3**) the purchasing ability of students.

In the given project they can be presented by the following formulas:

1) 0.5 < tokenPrice < 6; **2**) (pSale >= 0.01) && (pSale <= 0.5); **3**) (criticalLimit >= 10) && (criticalLimit <= 100).

After we denoted the initial unknown parameters as variables, we can evaluate their boundaries, if possible. In the process of tokenomic modeling for these examples, we can obtain the following formulas:

1) tokenPrice= F1(p1,p2,…), **2**) pSale= F2(s1,s2,...), **3**) criticalLimit= F3(l1,l2,...),

where p1,p2,…; s1,s2,...; l1,l2,... are the unknown parameters of tokenomics. Proving that these parameters enter into the described intervals can confirm or refute the property. This kind of property is called a safety property.

Another strength of the algebraic method is the ability to consider the security of the project. We must analyze the malicious or undesirable actions and create an additional formalization in the UCM behavior and basic protocols and combine it with a positive scenario. As example, in this model such actions could include the massive purchase of tokens or the lowering of the price by some subjects in market, etc.

The demanded boundaries of critical attributes shall be retained for stability.

Discussion and Conclusions

Using the proposed approach provides us with opportunities to study the reliability, stability, safety and properties of the models. Symbolic modeling allows us to evaluate the possible risks and to avoid losing money at critical moments, such as when the price of bitcoin falls, etc. It is possible to take into account the different scenarios of behavior to determine some predictions.

In the process of formalizing such models, there may arise some problems at the interdisciplinary level. Also, due to the complexity of the model, the problem of taking into account of possible scenarios may also arise. It is possible, that multiple attributes with undefined values may appear, the exact definition of which will be required for adequate model building. Accordingly, there is a need for additional research. In this case, as mentioned above, questioning and analysis methods were used to determine the initial values of some attributes for SKILLONOMY.

Acknowledgements

We would like to thank the company garuda.ai [14] for the opportunity to work with the platform for modeling, formal verification and testing of Blockchain / DLT systems behavior and cybersecurity research for our research and experiments in the modeling area. We are also grateful to the Glushkov Institute of Cybernetics of 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 the Kherson State University for the active supporting of Insertion Modeling System.

References

1. Kampakis, S.: Three Case Studies in Tokenomics. The JBBA, 1(2), 79-82 (2018).
2. Token Engineering Fundamentals, https://www.slideshare.net/MichaelZargham/token-engineering-presentation-5-1318.
3. ABM, https://github.com/ShrutiAppiah/Simulating-an-Economy-ABM.
4. Appiah, S.: Decentralized Organizations as Multi-Agent Systems - A Complex Systems Perspective. Technical Report (2017).
5. Letichevsky, A., Letychevskyi, O., Peschanenko, V.: Insertion Modeling and Its Applications. Computer Science Journal of Moldova 24 (3), 357-370 (2016).
6. SKILLONOMY, https:// SKILLONOMY.org/ru.
7. Letichevsky, A., Gilbert, D.: A Model for Interaction of Agents and Environments. In: Bert D., Choppy C., Mosses P.D. (eds) Recent Trends in Algebraic Development Techniques. WADT 1999, LNCS, vol. 1827, pp.311-328. Springer, Berlin, Heidelberg (1999).
8. Omega: a solver for quantifier-free problems in Presburger Arithmetic, https://coq.inria.fr/refman/addendum/omega.html.
9. CVC3, https://cs.nyu.edu/acsys/cvc3/.
10. Z3Prover, https://github.com/Z3Prover/z3.
11. Mathsat, http://mathsat.fbk.eu/.
12. ITU-T Recommendation, Z.151, User Requirements Notation (URN) – Language definition, https://www.itu.int/rec/T-REC-Z.151/en.
13. Letichevsky, A., Kapitonova, J., Letichevsky Jr, A., Volkov, V., Baranov, S., & Weigert, T.: Basic protocols, message sequence charts, and the verification of requirements specifications. Computer Networks, 49(5), 661-675. (2005).
14. Gruda AI BV, www.garuda.ai.