Security management of the functioning of a multi-node mobile cyber-physical system with a distributed registry based on an automatic model

O Ja Kravets¹, O A R Almusawi², Ju N Doroshenko¹, S N Mamedov³ and Yu V Redkin⁴

¹ Voronezh State Technical University, 14 Moscow ave., Voronezh, 394026, Russian Federation
² University of Wasit, El-Cut, Muhfaza Wasit, Iraq
³ Vladimir State University named after Alexander and Nikolay Stoletovs, 87, Gorky str., Vladimir, 600000 Russian Federation
⁴ Admiral Ushakov Maritime State University, 93, Lenaia ave., Novorossiysk, 353918, Russian Federation

E-mail: csit@bk.ru

Abstract. The problems and features of security management of the functioning of a multi-node mobile cyber-physical system with a distributed registry based on an automatic model are considered. The algorithm of the functioning of the system node allows for the possibility of increasing or decreasing its resources using various approaches. Models of stochastic automata with a variable structure are used to model such systems. The process of functioning of a node of a system with a distributed registry based on a chain of blocks in the form of a finite automaton with a variable structure and linear tactics is formalized, which ensures that the sequence of changing the variants of the node's behaviour strategy depends on the conditions of the environment with which it interacts by constructing state matrices.

1. Introduction

A multi-node mobile cyberphysical system with a distributed registry (DLT system) is conceptually a digital data storage technology based on creating a set of equivalent copies of this data distributed on network nodes [1, 2]. A special feature of DLT systems is the decentralized nature of copy synchronization, implemented on the basis of a class of algorithms for achieving VIC (in terms of DLT systems-consensus achievement) [3].

It is obvious that in such a broad sense, the solution to the problem of achieving consensus can be applied in various fields. Currently, a number of models have been proposed and are being improved that provide practical application of consensus-building algorithms. From the point of view of DLT structural organization approaches, they can be divided into models using the BC data structure (blockchain) [1] and models using a directed acyclic graph (DAG), the so-called blockchainless models [4]. Next, approaches based on the BC data structure are considered.

Generally, the DLT BC system is represented by a peer-to-peer network, whose nodes create, store and confirm transactions that are combined into a group described in a special data structure called a block. The distributed registry supported by such a system is represented by a BC containing records of
all transactions performed within the system, which is replicated to all nodes of the system. For each new block, the transactions included in it are confirmed, and additional information about transaction confirmation is stored in all previous BC blocks [5, 6].

2. A model of the functioning of a node of a system with a distributed registry based on a block chain

In real DLT BC systems, a reward mechanism for the work done is most often used to motivate nodes to perform the VIC algorithm. Each node strives to maximize revenue, using various behavioural strategies and their combinations, including the implementation of non-standard functions. At the same time, if each node of the system acts only in its own interests and does not strive to maintain the operability of the entire system – consistency, then its income will become zero, since the other nodes will not accept its BC. In these conditions, nodes tend to increase their income, while ensuring the efficiency of the entire system by constantly choosing a variant of the behaviour strategy – the formation of appropriate behaviour.

During the achievement of the VIC in DLT BC systems, the nodes of the system must be motivated to perform operations of verification, calculation and compliance with the rules for achieving the VIC. For example, in the DLT BC class of peer-to-peer payment systems, a mechanism for encouraging nodes, measured in cryptocurrency units, is used to motivate nodes to comply with the VIC algorithm and improve the performance of the entire system. Each node strives to maximize its reward, using various variants of the behaviour strategy, their combinations and the implementation of undeclared functions, which ultimately leads to the state of the system determined by the Nash equilibrium.

In real DLT BC systems, during the achievement of the VIC, the system nodes must constantly prove their participation by executing the Proof-of-X algorithm. The criterion for the success of a node (an increase in its profitability D when executing the Proof-of-X algorithm) is the share of node resources relative to the resources of the entire system. In addition, the algorithm of the functioning of the system node allows for the possibility of increasing or decreasing its resources, for example, the node of the Bitcoin peer-to-peer payment system can increase or decrease the amount of resources by changing its hash power, and the Ethereum node can change the ownership share (Proof-of-Stake) [1]. Let's generalize the share of resources of the m-th node when executing the Proof-of-X - hashpower algorithm and denote $h_m$, then the resources of the entire system will be equal $H = \sum_{m=1}^{v} h_m$, where $|A|$ is the power of the set A of the nodes of the system.

As a result, the ratio $h_m / H$ during the operation of the m-th node of the system will constantly change, which will lead to changes in the operating conditions. To model such features, it is advisable to use models of stochastic automata with a variable structure. In [7], a proof of the expedient functioning of such automaton structures in any stationary random environment is given.

3. Formal description

For a formal description of the process of functioning of the m-th node of the DLT BC system, a model of a finite automaton with a variable structure and linear tactics operating in the E environment is chosen. The environment E is a set of nodes $A = \{a_1, a_2, \ldots, a_n\}$, $a_n \notin A$ of the DLT BC system, involved in solving the problem of achieving mutual information coordination. In general, the structure of the DLT BC node of the system is shown in figure 1.

The stable states $\mu_i$ in which the automaton $a_n$ is located are combined into a set $B = \{b_1, b_2, b_3, \ldots b_j\}$ called a "tree", which consists of subsets $b_i$, called a "branch". Let's denote the indices of stable states of the automaton as: $i$ is the number of the "branch" of the tree B, and $j$ is the number of the state inside the "branch". We set the power of the set B equal to v. The change of states $\mu_i$ occurs taking into account the reactions of the medium $X(t)$. 


Figure 1. Automatic model of the DLT BC node of the system.

Then we divide the set of reactions of the medium \( X^n(t) \) into subsets: \( X^n(t)[+1] \) - the class of favourable reactions - adding a block to BC (the change of states \( \mu^1_j \) of the automaton in figure 1 is indicated by a solid line) and \( X^n(t)[-1] \) - the class of unfavourable reactions otherwise (the change of states \( \mu^1_j \) in figure 1 is indicated by a dotted line).

We assume that the elements \( x_k \) of the set of reactions of the medium within each of these classes \( X^n(t)[+1] \) or \( X^n(t)[-1] \) for an automaton \( a_m \) are indistinguishable. In the process of functioning, the nodes of the DLT BC system, depending on the input data and their internal state, perform a finite number of actions: checking the received transactions and blocks; updating the BC; transmitting newly created blocks to the network; relaying transactions over the network, etc. We will consider the actions of the node and the reactions of the system aimed directly at achieving consensus - the adoption of a single BC [8]. Under this assumption, the differences between the nodes of the system will consist only in the order of blocks and the agreement to add one of the possible blocks generated by different nodes in the time interval \( t \) to BC. BC, copies of which were accepted by the majority of the nodes of the DLT BC system, we will call "single".

Let's call the branches \( b_i \) of the tree \( B = \{ b_1, b_2, b_3, \ldots b_l \} \) variants of the node behavior strategy when the VIC is reached in the DLT BC system. At the same time, we will determine the number of options for the behavior strategy of the system node to four. We denote: \( b_1 \) the branch is the execution of the VIC algorithm with the probability \( p \) of an unintentional failure, \( b_2 \) the branch is an attempt to form a branch of the processed data, \( b_3 \) the branch is the publication of the found block, \( b_4 \) an attempt to implement a temporary blocking attack.

Since the environment \( E \) is dynamic, it is not possible to determine a priori analytically the necessary number of stationary states of the automaton – the optimal memory depth \( q_{\text{opt}} \) in each branch \( b_i \) of the tree \( B \). However, in [9] it is shown that during the experiments, an automaton with a variable structure itself determines \( q_{\text{opt}} \) for a specific implementation of the environment \( E \). Thus, for the generalized structure of the DLT node, the BC of the system is assumed \( q_{\text{opt}} \) to be equal to 1, while the exact value \( q_{\text{opt}} \) will be determined during an experimental study for specific algorithms for achieving VIC.

Then the branches of the tree \( B \) will have one stable state each, which determines the variant of the automaton's behaviour strategy, except because the implementation of the temporary blocking attack \( b_4 \) involves the fulfillment of the inequality:

\[
N_{\text{otr}} + 1 \leq N_{\text{block}}
\]  
(1)
where \( N_{otr} \) is the number of adverse reactions of the environment \( E \), \( N_{block} \) is the number of blocks found when executing the variant of the temporary blocking strategy. To perform (1), it is necessary to get ahead of a single BC by at least one block, which will require an additional stable state in the branch \( b_4 \).

4. Changing the states of the automaton

We define the change of states of the automaton \( \mu^t_j \) at a time \( t_n \) by the expression (2):

\[
\mu^t_j(t_{n+1}) = \Phi[\mu^t_j(t_n), X^n(t_{n+1})],
\]

where \( X^n(t_{n+1}) \) is the reaction of the medium at a time \( t_{n+1} \). \( \Phi[\mu^t_j(t_n), X^n(t_{n+1})] \) is the transition function that determines the change of states of the automaton under the influence of the input signal – the reaction of the medium.

We define the transition function \( \Phi[\mu^t_j(t_n), X^n(t_{n+1})] \) by a system of two state matrices \( a_i(X^n(t_n)) \) (\( i, j = 1, 2, ..., v \)) according to the number of classes of reactions of the medium \( X^n(t_n) [+1] \) and \( X^n(t_n) [-1] \). Each element of the matrix determines the probability of the automaton’s transition from one state \( \mu^t_j \) to another. At the same time,

\[
0 \leq a_i(X^n(t_n)) \leq 1,
\]

\[
\sum_j a_i(X^n(t_n)) = 1.
\]

The result of the automaton’s operation \( a_{m} \) at time points \( t = \{t_1, t_2, ..., t_n\} \) is a set of output signals \( Y^n(t_n) = \{y_1, y_2, ..., y_l\} \): new or alternative (an attempt to fork) blocks, updating its own BC.

Let us determine the dependence of the output signal \( Y^n(t_n) \) on the stable state \( \mu^t_j \) in which the automaton is located at the moment of time \( t_n \):

\[
Y^n(t_n) = F[\mu^t_j(t_n)]
\]

The sets \( Y^n(t_n) \) and \( B \) are equally powerful, that is, the bijection relation \( f : Y(t_n) \rightarrow B \) is defined for them. At the moment of time \( t_n \), being in any of the stable states of the branch \( b_1 \), the automaton \( a_{m} \) sends the corresponding signal \( y_1 \) to the medium \( E \). As a result of the reaction of the medium \( E \) to the signal \( y_1 \), an assessment of the actions of the automaton \( a_{m} \) is formed – a signal \( x_k \).

Based on this, the automaton model of the m-th node of the DLT BC system will take the following form, shown in figure 2.

**Figure 2.** The structure of the DLT BC node of the system in the form of a finite automaton with a variable structure and linear tactics.
5. Conclusion
The process of functioning of a node of a system with a distributed registry based on a chain of blocks in the form of a finite automaton with a variable structure and linear tactics is formalized, which ensures that the sequence of changing the variants of the node's behaviour strategy depends on the conditions of the environment with which it interacts by constructing state matrices.

References
[1] Raval S 2017 Decentralized Applications Blockchain Technology in action (St. Petersburg: Peter Publishing House) 240
[2] Ben-Or M 1983 Another advantage of free choice: completely asynchronous agreement protocols Proceedings of 2nd ACM Symposium Principles of Distributed Computing (Canada) 27-30
[3] Kumari S and Rohil H 2014 Taxonomy of Distributed Consensus Algorithms International Journal of Computer Science and Information Technologies 5(3) 3921-3
[4] Li Y, Bin C, Peng M, Zhang L, Zhang L, Feng D and Yu J 2020 Direct Acyclic Graph based Blockchain for Internet of Things: Performance and Security Analysis IEEE/ACM Transactions on Networking 99 DOI: 10.1109/TNET.2020.2991994
[5] Filimonov P A, Izotova T Y, Gorelik S S and Eskov S S 2019 The question simulation of internet data exchange, according the features of functioning the hybrid content delivery networks Journal of Physics: Conference Series 1399(3) 033017
[6] Bano S and Sonnino A 2019 SoK: Consensus in the Age of Blockchains Proceedings of the 1st ACM Conference on Advances in Financial Technologies (Zurich, Switzerland) 183-98
[7] Varshavsky V And 1973 Collective behavior of automata (Moscow: Nauka) 408
[8] Shirinkina E V and Romansky R 2020 Assessment of the synergetic efficiency of industrial companies reengineering processes Journal of Physics: Conference Series 1679(3) 032014
[9] Varshavsky V and and Pospelov D A 1984 The orchestra plays without a conductor: reflections on the evolution of some technical systems and their management (Moscow: Nauka) 208