MIPRO 2020, September 28 - October 02, 2020, Opatija, Croatia

Microservice approach to the qualitative study of attractors of binary dynamic systems based on the Boolean constraint method

G.A. Oparin, V.G. Bogdanova, and A.A. Pashinin
Matrosov Institute for System Dynamics and Control Theory of SB RAS, Irkutsk, Russia
prn51@mail.ru, bvg@mail.ru, apcrol@gmail.com

***Abstract* - In recent years, due to the active development of distributed computing technologies, the automation of computations based on these technologies became very important for solving the problems of qualitative research of binary dynamic systems, which are widely used as mathematical models in cryptography, bioinformatics, the study of fault tolerance of computer networks, and some other domains. In this regard, the issues are actualized of developing tools and providing infrastructures for the microservice-based implementation of scientific applications, which are characterized by the dynamism of resource requirements and oriented to use in a hybrid cloud environment. We offer microservice-oriented tools for the automation of solving computationally complex problems of a qualitative study of attractors of binary dynamic systems in a hybrid computing environment. This study is based on the Boolean constraint method and allows the finding of evolutionary parameters of system states (such as, for example, radius, period, and branching) without modeling system dynamics. These tools provide the agents implemented as web-services for the integration of resources of on-premises computers with cloud resources using Dew computing. An example of the application of the developed tools is given.**

Keywords - microservice; microservice-oriented tools; qualitative analysis; hybrid computations

I. INTRODUCTION

Binary dynamic systems (BDS) are widely used as basic mathematical models in many subject areas, in particular in cryptography, bioinformatics, economics, and sociology. We apply the declarative approach to a qualitative analysis of BDS based on the Boolean Constraint Method (BCM) [1]. The fundamental point of BCM is the use of formal definitions of dynamic properties, presented in the language of applied logic of predicates with limited quantifiers of existence and universality, understandable to a specialist in dynamic analysis of systems. Such specification of property (instead of its verbal description) disambiguates the interpretation of the meaning of the property. Based on this approach, a Boolean model of the dynamical property for BDS, whose functioning is considered at a finite time interval, is constructed, and its feasibility is checked. The uniqueness of the constructed Boolean model lies in the fact that it contains restrictions describing both the dynamics of the BDS and the specification of its required property.

A BCM-based qualitative study of BDS requires the software to build Boolean models and verify their feasibility. The computational complexity of some problems of this study leads to the necessity of the use of a high-performance computing environment to solve them. We developed the above software as Applied Microservices Package (AMP) for BCM-based Qualitative Analysis of BDS (BCM-QABDS). The advantages of this AMP are the reduction of computational complexity by using small, loosely coupled, reusable, autonomous microservices [3], and simplicity of its scaling, deploying, and updating. Applied software is implemented as microservices interacting by using lightweight communication mechanisms. This implementation is based on the authors' technology HPCATAMP (High-Performance Computing Automated Technology for implementation of Applied Microservices Package) [2]. The AMP, based on this technology, is an integrated intelligent system for managing distributed computations in a specific subject domain. This system automates these computations using a semantic peer-to-peer (P2P) network of agents deployed on nodes of the computational field (CF). Agents installed on these nodes control the launch of computational microservices.

Further development of the HPCATAMP uses advantages of the integration of P2P network and agents technology for providing additional cloud resources. This possibility is provided due to the openness of this resulted computational network, mobility of AMP agents, and new cloud version HPCSOMAS-MSC of the previous basis software platform [2] of this technology. An agent-manager for processing the request of cloud resources was developed in the new version. The technological level of knowledge used by agents for organizing hybrid computing, allocating additional cloud resources, and ensuring synchronization of on-premises and user cloud data was added to the conceptual model of the subject domain.

A conceptual model of knowledge representation using this technology in the subject domain of a BCM-based qualitative study of BDS is presented. An illustrative example of the usage of the BCM-QABDS for the study of attractors of BDS is considered.

II. Related Work

Currently, one of the software development trends is an integration of distributed and cloud computing technologies in solving resource-intensive complex scientific and applied problems. Some publications (for example, [4-5]) discuss the development of tools and infrastructures for implementation of resource-require scientific applications based on microservice technology, including applications oriented to use in a hybrid cloud environment [6, 7]. Hybrid cloud infrastructure combines the reliability and availability of software running on on-premises computers with the ability to scale computing to the cloud when peak loads occur [4]. Cloud integration models in a hybrid environment are considered in [8, 9]. In [10], a taxonomy of cloud deployment patterns is presented. The template selection is determined by the aims of forming hybrid infrastructure and the requirements for developing service-oriented applications. In our research, a hybrid infrastructure based on a template similar to the "on-premises pattern" [9] is used to solve the problems of studying dynamics and structural-parametric synthesis of BDS based on BCM [1]. The active development of new and improvement of existing methods for the qualitative study of BDS (mainly models of genetic networks) observed in the last decade. However, only a small number of publications are devoted to the direction of the application of web and distributed technologies in creating tools for the qualitative study of BDS. Examples of it are the web-based tool Analysis of Dynamic Algebraic Models (ADAM) [11] and CoLoMoTo [12]. ADAM automatically converts several types of discrete models into polynomial dynamical systems and analyses their dynamics using computer algebra tools. The CoLoMoTo provides access to the range of command-line tools for the qualitative study of the Boolean network (BN). The BoolSi [13] is intended for distributed simulation of synchronous BN. This tool is oriented for use in a high-performance environment. However, it has a command-line interface. We develop an AMP BCM-QABDS intended to automate distributed and parallel solving the problems of a qualitative study of BN in a high-performance hybrid environment on dedicated CF resources. This tool is cross-platform, has web and API interfaces. CF can include both cloud and on-premises resources, in particular clusters, dedicated servers, and personal computers (PC). Unlike the listed above approaches, a BCM-based declarative approach for the qualitative study of BDS allows data parallelism by splitting the Boolean model of the verifying BDS property. This approach provides significantly increasing the dimension of the BDS and the time interval of its functioning.

III. Conceptual Model of BCM-Based Qualitative Analysis Using HPCATAMP

A. Problem Formulation

We consider a synchronous autonomous BDS, which has the form

$$x^t = F(x^{t-1}), \quad (1)$$

where x is the state vector of the dimension n ($x \in B^n$, $B = \{0,1\}$), $t \in T = \{1,2,\ldots,k\}$ is the discrete time, and $F(x)$ is the vector function of logic algebra called the transition function ($F : B^n \to B^n$). Let us define the trajectory $x(t, x^0)$ of (1) for each initial state $x^0 \in B^n$ as the finite sequence of states $x^0, x^1, \ldots, x^k$ from the set B^n. The state x^i is a successor of the state x^{i-1}, and the state x^{i-1} is a predecessor of the state x^i. In an autonomous BDS, each state has only one successor, and the number of predecessors of this state can vary from zero to 2^n-1. A cycle of length k is a closed trajectory ($x^0 = x^k$) in which other states are pairwise distinct. An equilibrium state is the cycle of the length k=1. A cycle without predecessors is an isolated one. The non-isolated cycle is called an attractor. Let X^* is the attractor. The region of attraction (basin) of an attractor of the radius k is the set of all states, from which the set X^* is reachable in $i \le k$ time steps.

A qualitative analysis of BDS, based on the study of the structure of the state space, requires solving the following problems:

- Searching equilibrium states and cycles including cycles of the given length k, and verifying their isolation property
- Description of attractors (their number, type, length)
- Determining the attraction region (basin) of the attractor and its radius, or constructing the attraction region of the given radius k
- Checking the property of belonging the state to the attractor and determining the distance (number of steps) to this attractor
- Determining immediate predecessors for given states
- Checking the reachability property of the target set X^* from the initial values set X^0.

B. BCM-based Qualitative Study of BDS

Solving the above problems based on BCM includes three stages:

- Building a property model in the form of Boolean constraints based on the description of the dynamics of the system (1) and property specifications [1]
- Verification of satisfiability of this model
- Post-processing the results.

We present the equations from [1], which are required for solving these problems.

The system (1) with the initial state $x^0 \in X$ for $t = \{1,2,\ldots,k\}$ is equivalent the one Boolean equation ([1])

$$\Phi(x^0, x^1, \ldots, x^k) = \vee_{t=1}^{k} \vee_{i=1}^{n} (x_i^t \oplus F_i(x^{t-1})) = 0. \quad (2)$$

Equation (2) for k=1 (one-step transitions is considered only) has the following form:

$$L(x^0, x^1) = \vee_{i=1}^{n}(x_i^1 \oplus F_i(x^0)) = 0 . \quad (3)$$

The state x^1 is the successor of x^0, and the state x^0 is the predecessor of x^1 in (3). All immediate predecessors x^0 of the state $s \in X$ are solutions of the next Boolean equation:

$$L(x^0, x^1)\Big|_{x^1=s} = 0 . \quad (4)$$

Predecessors of the state s are absent if there are no solutions to the equation (4). A cycle of length k (if it exists) is a solution to the Boolean equation

$$\Phi(x^0, x^1, \ldots, x^k)\Big|_{x^0=x^k} \vee R(x^0, x^1, \ldots, x^{k-1}) = 0, \quad (5)$$

where the condition for pairwise differences of all states of the set C of a cycle of length k is given by the expression

$$R(x^0, x^1, \ldots, x^{k-1}) = \vee_{\substack{1 \le q \le k-1, \\ k \bmod q = 0}} \wedge_{i=1}^{n} y_i^q = 0,$$

$$y_i^q = (x_i^0 \cdot x_i^q \vee \overline{x_i^0} \cdot \overline{x_i^q}) .$$

Let the elements s of the set C be determined by the solution of the Boolean equation $G^c(s) = 0$. Then the condition of isolation of the cycle is equivalent to the absence of solutions to the following Boolean equation [1]:

$$G^c(s) \vee L(x^0, x^1)\Big|_{x^1=s} \vee \overline{G^c}(x^0) = 0 . \quad (6)$$

The attraction region of the attractor X^* of the radius k is determined by the set of solutions of the Boolean equation

$$\Phi(x^0, x^1, \ldots, x^k) \vee G^*(x^k) = 0 \quad (7)$$

concerning variables $x^{k-1}, x^{k-2}, \ldots, x^0$.

The set of parameters AMP BCM-QABDS consists of Boolean models and variables used for their building, sets of states, descriptions of functions, and description formats. The set of modules includes programs for building Boolean models, converting formats from one to another, verifying the truth of models, and post-processing the results. Both of these sets represent algorithmic knowledge for a qualitative study of BDS based on BCM.

C. Knowledge Representation

Algorithmic knowledge about mathematical models, methods, and techniques for their research have a complex hierarchical structure in which there are three conceptually isolated layers: computational, schematic, and production knowledge [14]. In this study, the development of HPCATAMP technology in the direction of hybrid computing automation with Dew-computing support [15] required the inclusion of a new layer of knowledge - a technology layer. This layer provides information support for decision-making by managing agents, in particular, about the need for synchronization, about the possibility of connecting cloud resources. The conceptual model of HPCATAMP-based AMP usage has the form:

CM=<C, A, P, G, T, P^F, M, N, U^P, U^T, R>,

where C is the set of *complexes*, which are isolated (structurally) sets of knowledge. These complexes combine objects of the subject domain, corresponding to a family of research methods that intersect in sets of parameters and operations. In a hybrid infrastructure, set C includes three subsets, $C^P \subset C$, $C^D \subset C$, and $C^C \subset C$. The first subset contains agents PSA (Problem Statement Agent), the second – web services installed on the on-premises resources (Dew PSA), and the third – newly developed agents CRA (Cloud Resource Agent), designed to allocate resources in the cloud at the request of the PSA. The set A is divided into four subsets of agents: A^C, A^P, A^I, and A^G. Elements of the subset A^C are DSA (Distributed Solver Agent) that control the launch of the Computational Microservice (CM). Subsets A^P, A^I and A^G consist of, respectively DSAProd agents (DSA Production) using the productional knowledge for the CM launch; the initial (DSAInit), and target (DSAGoal) agents, which are created by the PSA and DPSA when the user formulates the problem [16]. The set of parameters P includes input and output parameters stored in the local knowledge base (KB) of agents from the set A. The set of parameters $P = \bigcup_{i=1}^{n_A} P_i^L$, where n_A is a number of all agents of AMP, P_i^L are the set of input and output parameters from the local (distributed on CF nodes) knowledge base (KB) of the i-th agent. Elements of the set G are active groups of agents formed for the non-procedural (declarative) statement (NPS). Each new NPS is added in the set T, initially empty. The structure T_i ($T_i \in T$) includes an input (T_i^I) and output (T_i^O) parameters, names of which are selected from the parameter dictionary P^F, and parameter values (T_i^D). The set P^A includes values of input parameters by which the output ones are calculated.

The sets M and N contain computational microservices and CF nodes, and the sets U^P and U^T include references to user profiles and their data updating tables. The set R = {In, Out, Pr, Inc, Imc, Isc, Icv} (Fig. 1) consists of relationships of the following types: many-to-many (pairs of the Cartesian product of corresponding sets), one-to-one, and one-to-many (functional relationships, the semantic of which are presented in [17]). For example, In has many-to-many type:

$$In(P^L, A^C) \subset P^L \times A^C : (P_i^L, A_i^C)\Big| P_i^L \in P^L, A_i^C \in A^C .$$

D. Architecture and Functioning of AMP

The scheme of interaction of agents (previously deployed to the nodes of the CF) is presented in Fig. 2. The active group G_i is formed for the NPS $T_i = (T^I; T^O; T^D)$ or briefly, $T_i = (T^I; T^O)$, by the logical inference on the distributed KB of DSA. Fragments of relationships In (P^L,

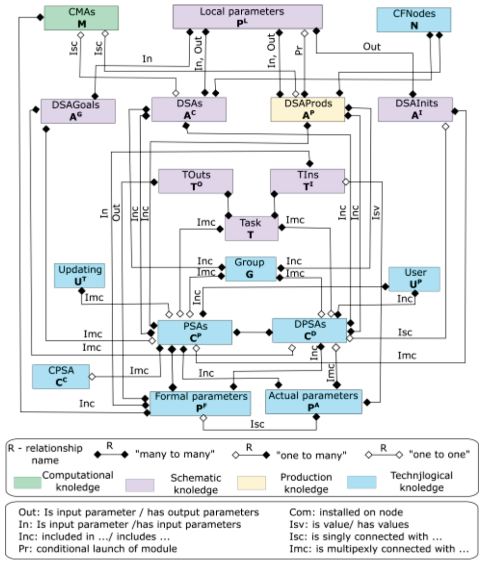

Figure 1. Conceptual model of HPCATAMP-based AMP

A^C) and *Out* (A^C, P^L) ($In \in R$, $Out \in R$) of this KB determine pair interactions of DSA in the process of group formation and solving a problem. The KB contains relationships $Com \subset A^C \times N$ also. The PSA sets the attribute IP of $N_i \in N$ in KB of DSA during its initialization after a preliminary availability check. PSA needs technological knowledge when executing a user request to organize backup processes, synchronize local and cloud knowledge bases, handle CF node failure situations, conduct testing when updating a microservice by a developer, and attract additional cloud resources. Backup and data synchronization is required to save the results and the ability to continue to work when the Internet is turned off. The user installs Dew PSA on his computer and configures it to synchronize the results of computations and input files from available PSA agents upon request, schedule, or following the options specified in the profile.

Scaling cloud computing at the top level of abstraction is as follows. DSAs manage the message queue, and CMAs control the task queue. In the case of queue congestion, the CMA informs DSA about resource exhaustion. DSA sends a request for an additional resource to the PSA, where it is registered. PSA contacts

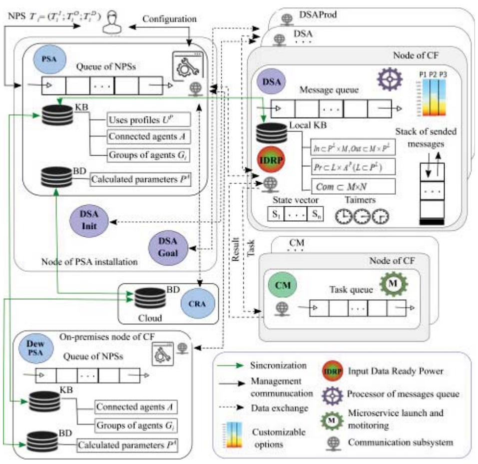

Figure 2. The interaction of the AMP agents

TABLE I. HYBRID INFRASTRUCTURE

Resource	Resource type	Installed software: *Agents*	Installed software: *Additional software*
Nodes of the cluster «Akademik V.M. Matrosov» [18]	On-premises	PSA, Ag_6, Ag_7, Ag_{11}, Ag_{13}	Tomcat Server
VDS [19]	Private Cloud	PSA Ag_1-Ag_5 Ag_8-Ag_{10}, Ag_{12}, Ag_{13}	SageMath VirtualBox, Docker Server, Docker Container
	Public Cloud	CRA Ag_{13}	Kernel Virtual Machine Tomcat Server
PC	On-premises	DPSA, Ag_1-Ag_3, Ag_{12}, Ag_{14}-Ag_{16}	Tomcat Server

with the special CPSA agent if the option to use a cloud resource is set in the user profile. In this case, the CPSA launches the Docker container with the required CMA, returning a message about the allocation of the resource and the network address for communication with it. Otherwise, a request-rejection message is sent, and tasks are executed in the order of the task queue.

IV. ILLUSTRATIVE EXAMPLE

We used the AMP BCM-QABDS for the study of attractors of BDS. This problem has a practical application in bioinformatics. Attractors correspond to a combination of gene expressions, which specifies a particular cell type or cell fate of an organism [21, 22]. The sizes of the basins of attraction may indicate which attractor is more stable [21]. The computing resources on which the AMP BCM-QABDS microservices are deployed are shown in Table 1. We study the attractors using the following scheme for verifying BDS properties. A search for equilibrium states is carried out according to (4). The isolation property of the found equilibrium states (*ES*) is checked using (6). The equilibrium state is an attractor if the isolation property is not satisfied (there are no immediate predecessors in the array *A_IP*). The basin of attraction (*AB*) is founded as solutions of equation (7). The fragment of the computational model for this scheme is presented in Fig. 3. Problems of verifying these properties are solved both separately and jointly (as task flow).

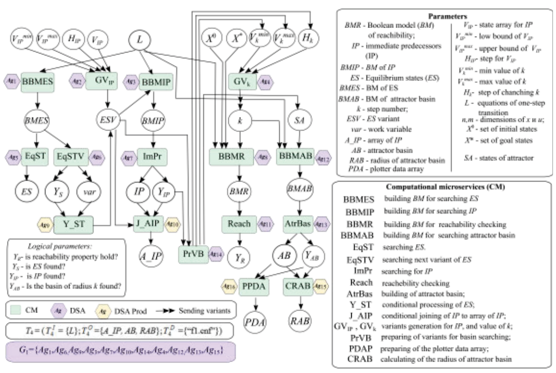

Figure 3. The fragment of the computational model

For example, in the first case, the following NPSs for these problems are used: $T_1 = (T_1^I = \{L\};\ T_1^O = \{ES\}; T_1^D = \{\text{“f1.cnf”}\})$; $T_2 = (T_2^I = \{L,\ ESV\}; T_2^O = \{A\_IP\})$; $T_3 = (T_3^I = \{L,\ SA\};\ T_3^O = \{AB,\ RAB\})$. In the second case, the following NPS is formulated:

$$T_4 = (T_4^I = \{L\}; T_4^O = \{A\_IP,\ AB,\ RAB\}; T_4^D = \{\text{“f1.cnf”}\}).$$

In the above NPSs, the text “f1.cnf” from the set T_i^D is the file name of the description of function L (4), which is built preliminary [16]. All NPSs are formulated using the web-interface of PSA (Dew PSA). Sets T_i^I and T_i^O are selected from the P^F. For the NPS T4, the active group G4 = {Ag1, Ag6, Ag9, Ag3, Ag7, Ag10, Ag14, Ag4, Ag12, Ag13, Ag15} is formed (Fig.3). The group of DSAs performs decentralized control for solving the problem based on direct pair interactions. Agents behavior is coordinated by the input data readiness (the event control is used) [20]. In the case of task flow (NPS T4), computations are carried out by parallel-pipeline way [14]. The next equilibrium state is processed according to the above scheme. For testing developed AMP, the BDS was used from [21] and [23]. The NPS T_4 was used to study a phase portrait for BDS with state vector dimensions n=15 [21] and n=11 [23], dynamic of which is described correspondently by the following equations:

$$\begin{aligned}
&x_1^t = x_{10}^{t-1} \cdot x_{14}^{t-1} \cdot x_{15}^{t-1} \cdot x_1^{t-1} \vee x_5^{t-1} \cdot x_{14}^{t-1} \cdot x_{15}^{t-1} \cdot x_1^{t-1} \vee \\
&\vee x_7^{t-1} \cdot x_2^{t-1}, x_2^t = x_2^{t-1}, x_3^t = \overline{x_5^{t-1}} \cdot \overline{x_{13}^{t-1}}, x_4^t = \overline{x_6^{t-1}}, \\
&x_5^t = \overline{x_{10}^{t-1}} \cdot \overline{x_{13}^{t-1}} \vee x_7^{t-1} \cdot \overline{x_{10}^{t-1}} \vee x_4^{t-1} \cdot \overline{x_{10}^{t-1}}, \\
&x_6^t = \overline{x_7^{t-1}}, x_7^t = \overline{x_{13}^{t-1}} \vee \overline{x_6^{t-1}}, x_8^t = \overline{x_{13}^{t-1}}, \\
&x_9^t = x_9^{t-1} \cdot \overline{x_{15}^{t-1}} \vee x_9^{t-1} \cdot \overline{x_{10}^{t-1}}, \\
&x_{10}^t = x_7^{t-1} \cdot x_{10}^{t-1} \cdot x_{15}^{t-1} \vee \overline{x_8^{t-1}} \cdot \overline{x_{13}^{t-1}} \vee x_7^{t-1} \cdot \overline{x_{12}^{t-1}} \vee \\
&\vee x_7^{t-1} \cdot \overline{x_{11}^{t-1}} \vee x_7^{t-1} \cdot x_9^{t-1} \vee \overline{x_5^{t-1}} \cdot x_7^{t-1} \vee x_7^{t-1} \cdot \overline{x_8^{t-1}}, \\
&x_{11}^t = 1, x_{12}^t = 1, x_{13}^t = \overline{x_5^{t-1}} \cdot x_6^{t-1} \cdot \overline{x_7^{t-1}}, \\
&x_{14}^t = x_5^{t-1} \cdot x_{14}^{t-1} \cdot x_{15}^{t-1} \cdot x_1^{t-1} \vee x_{10}^{t-1} \cdot x_{14}^{t-1} \cdot x_{15}^{t-1} \cdot x_1^{t-1} \vee \\
&\vee x_7^{t-1} \cdot x_{10}^{t-1} \vee x_7^{t-1} \cdot x_1^{t-1}, x_{15}^t = x_7^{t-1}.
\end{aligned} \tag{8}$$

$$\begin{aligned}
&x_1^t = x_2^{t-1}, x_2^t = x_3^{t-1} \cdot x_{10}^{t-1}, x_3^t = x_4^{t-1} \cdot x_7^{t-1}, \\
&x_4^t = x_5^{t-1} \cdot x_7^{t-1}, x_5^t = x_8^{t-1} \cdot x_9^{t-1}, x_6^t = x_3^{t-1}, \\
&x_7^t = x_6^{t-1} \cdot x_8^{t-1}, x_8^t = x_5^{t-1} \cdot x_9^{t-1}, x_9^t = x_8^{t-1}, \\
&x_{10}^t = x_8^{t-1} \cdot x_{11}^{t-1}, x_{11}^t = 0.
\end{aligned} \tag{9}$$

Table II shows the results of solving NPS T_4 for these systems. The speedup of distributed parallel-pipeline computations was achieved 7.5 and 2.8 times in comparison with the BCM-based sequential execution. The AB solution result (the attractor basin), is visualized using the NPS $T_5 = (T_5^I = \{AB,\ Y\_AB\};\ T_5^O = \{PDA\})$ (Fig. 4). The decimal state codes of the attractor are calculated by the formula $\sum_{i=1}^{n} a_i * 2^{i-1}$. The Dew PSA

TABLE II. RESULTS OF STUDY

DD	No	*Attractor*	Attractor Type		
			Length	*Basin size*	*Max radius*
(8)	1	100110110011011	1	8	2
(8)	2	000110110011011	1	440	3
(8)	3	110110110011011	1	448	3
(8)	4	000001000011100	1	512	2
(8)	5	010001000011100	1	512	2
(8)	6	010001001011100	1	256	2
(8)	7	000001001011100	1	256	2
(8)	8	111100110111011	1	15168	9
(8)	9	101100110111011	1	568	4
(8)	10	001100110111011	1	14600	9
(9)	1	00000000000	1	1792	4
(9)	2	00001001100	1	240	10
(9)	3	00111111100	1	16	4

with microservice agents less demanding on resources (for example, the builder of the Boolean model, sequential SAT and 2QBF solvers, and post-processor of results) can be installed on on-premises computers. Parallel solvers SAT and 2QBF are installed on on-premises high-performance clusters. BCM-based sequential execution has the advantage in comparison with the sequential BNS-solver [21, 24] for BDS of both the small and the large dimensions n of the state vector (Fig. 5). The BNS achieved the time limit in twelve hours in the searching of cycles of the given length on the value k=30 of BDS, which is the model of stream cipher based on shift registers (Fig. 6). The BCM-based approach allows increasing the interval k of functioning this BDS by splitting constructed Boolean model for searching cycles and subsequent parallel solving subtasks (Fig. 6). The previously available number of resources on NPS startup is allocated, and the dynamical request, according to the need for additional cloud resources, is sent for providing more effective the execution of these subtasks

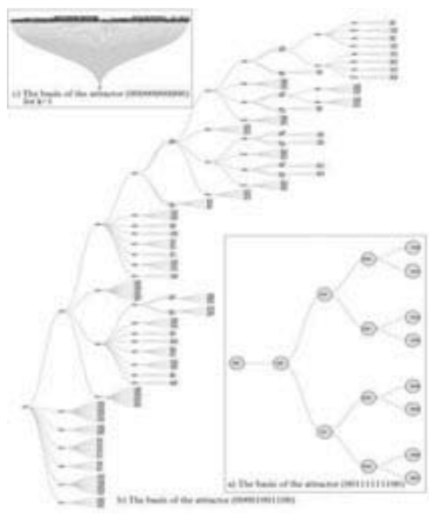

Figure 4. Basins of attractors of BDS (9)

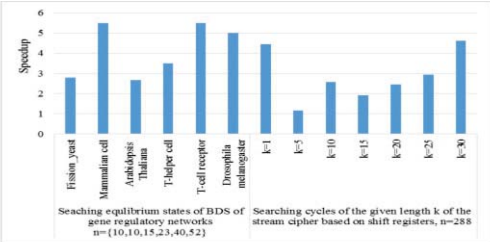

Figure 5. The speedup (S=$T_{\rm BCM}$ / $T_{\rm BNS}$) of BCM- approach in comparison with the BNS solver, based on algorithmic approach, in sequential solving qualitative analysis problems

in the case of queuing delay of CMA (Fig. 3). The resource is freed after completing the tasks sent on it.

V. Conclusion

We proposed further development of HPCATAMP technology for automation of hybrid computing, allocating additional cloud resources, and ensuring synchronization of on-premises and user cloud data. We developed a new version of the program platform based on which this technology is implemented. Using this technology, we developed a package of applied microservices BCM-QABDS for the qualitative study of BDS based on the Boolean constraint method, oriented on an application in the different subject domains, where BDS are used as models of research objects. The conducted experiments confirmed the effectiveness of the offered approach and developed software for solving practically significant scientific problems of the qualitative study of BDS.

Acknowledgment

The present investigation was supported by Russian Foundation of Basic Research, projects no. 18-07-00596.

References

[1] G. Oparin, V. Bogdanova, and A.Pashinin, “Qualitative analysis of autonomous synchronous binary dynamic systems,” in Mathematics in engineering, science and aerospace (MESA), vol. 10, no. 3, 2019 pp. 407-419.

[2] G.A. Oparin, V.G. Bogdanova, A.A. Pashinin, and S.A. Gorsky, “Microservice-oriented Approach to Automation of Distributed Scientific Computations,” in Proceedings of the 42nd International Convention on Information and Communication Technology, Elec-tronics and Microelectronics (MIPRO), IEEE, 2019, P. 236-241.

[3] S. Newman, “Building Microservices”. O’Reilly, 2015.

[4] V. Navale and P.E. Bourne, “Cloud computing applications for biomedical science: A perspective,” PLOS Comp Biol, vol. 14(6):e1006144, 2018.

[5] M.A. Netto, R.N. Calheiros, E.R. Rodrigues, R.L. Cunha, and R. Buyya, “HPC Cloud for Scientific and Business Applications: Taxonomy, Vision, and Research Challenges,” ACM CSUR, vol. 51(1), 2018, pp. 8:1-8:29.

[6] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” 2011, https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf [online, accessed: 24-Jan-2020].

[7] K. Bakshi, “Secure hybrid cloud computing: Approaches and use cases,” IEEE Aerospace Conference, Big Sky, MT, 2014, pp.1-8.

[8] D. Merkel, F. Santas, A. Heberle, and T. Ploom, “Cloud Integration Patterns,” 4th European Conference on Service-Oriented and Cloud Computing (ESOCC), Taormina, Italy, 2015, pp.199-213.

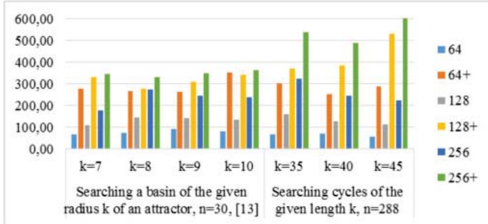

Figure 6. The speedup S=T / T^{+}, where T and T^{+} are runtimes correspondently on the time-independent number of nodes (e.g., 64) and number nodes increasing by cloud node temporarily (e.g., 64+)

[9] “CSCC: Practical Guide to Cloud Computing,” https://www.omg.org/cloud/deliverables/CSCC-Practical-Guide-to-Hybrid-Cloud-Computing.pdf [online, accessed: 19.10.2019].

[10] L.C. Ochei, A. Petrovski, and J.M. Bass, “A Novel Taxonomy of Deployment Patterns for Cloud-hosted Applications: A Case Study of Global Software Development (GSD) Tools and Processes,” in International Journal on Advances in Software, vol. 8(3 & 4), 2015, pp. 420-434.

[11] F. Hinkelmann, M. Brandon, B. Guang, R. McNeill, G. Blekherman, A. Veliz-Cuba, and R. Laubenbacher, “ADAM: Analysis of Discrete Models of Biological Systems Using Computer Algebra,” in BMC Bioinformatics, vol. 12:295, 2011.

[12] A. Naldi, C. Hernandez, N. Levy, G. Stoll, P.T. Monteiro, C. Chaouiya, T. Helikar, A. Zinovyev, L. Calzone, S. Cohen-Boulakia, D. Thieffry, and L. Paulevé, “The CoLoMoTo Interactive Notebook: Accessible and Reproducible Computational Analyses for Qualitative Biological Networks,” Frontiers in Physiology, vol. 9, 2018, article 680.

[13] V. Oles and A. Kukushkin, “BoolSi: a tool for distributed simulations and analysis of Boolean networks,”, arXiv:1910.03736v2.

[14] I.V. Bychkov, G.A. Oparin, V.G. Bogdanova, and A.A. Pashinin, “Intellectual Technology for Computation Control in the Package of Applied Microservices.” in Proceedings of the 1st International Workshop on Information, Computation, and Control Systems for Distributed Environments, ICCS-DE 2019, CEUR-WS Proceedings, vol. 2430, 2019, pp. 15-28.

[15] Y. Wang, “Definition and Categorization of Dew Computing,” OJCC, vol. 3(1), 2016.

[16] G.A. Oparin, V.G. Bogdanova, and A.A. Pashinin, “Automation of Microservices Creation for Qualitative Analysis of Binary Dynamic Systems,” in Proceedings of the 1st International Workshop on Information, Computation, and Control Systems for Distributed Environments, ICCS-DE 2019, CEUR-WS Proceedings, vol. 2430, 2019, pp. 88-98.

[17] D.C. Tsichritzis and F.H. Lochovsky, Data Models. Prentice Hall Professional Technical Reference, 1982.

[18] “Irkutsk Supercomputer Center of SB RAS,” http://hpc.icc.ru/ (accessed: 24-Jan-2020).

[19] https://firstvds.ru/ (accessed: 24-Jan-2020).

[20] I.V. Bychkov, G.A. Oparin, V.G. Bogdanova, and A.A. Pashinin, “Service-oriented technology for development and application of decentralized multiagent solvers for applied problems,” Herald of computer and information technologies, vol. 12. pp. 36-44, 2018. [In Russian].

[21] E. Dubrova, “Self-Organization for Fault-Tolerance,” In Self-Organizing Systems, K.A. Hummel and J.P.G. Sterbenz, Eds. IWSOS 2008, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, vol. 5343, 2008, pp. 145-157.

[22] S.A. Kauffman, “The Origins of Order: Self-Organization and Selection of Evolution”. Oxford University Press, Oxford, 1993.

[23] O. Reyes, R. C. Laubenbacher, and B. Pareigis, “Boolean Monomial Dynamical Systems,” in Annals of Combinatorics, vol. 8, 2004, pp.425-439.

[24] E. Dubrova and M. Teslenko, “A SAT-Based Algorithm for Finding Short Cycles in Shift Register Based Stream Ciphers,” in IACR Cryptology ePrint Archive, 2016. https://eprint.iacr.org/2016/1068 [online, accessed: 24-Jan-2020].