Agent-based approach to hybrid modeling queuing systems

A G Feoktistov[1], R O Kostromin[1], O Yu Basharina[1,2], A Tchernykh[3,4,5] and V I Dmitriev[2]

[1]Matrosov Institute for System Dynamics and Control Theory of SB RAS, Lermontov St. 134, Irkutsk, Russia, 664033
[2]Irkutsk State University, Karl Marx St. 1, Irkutsk, Russia, 664003
[3]CICESE Research Center, Ensenada-Tijuana Highway, Ensenada, Mexico, 22860, 3918
[4]Ivannikov Institute for System Programming of RAS, Alexander Solzhenitsyn st. 25, Moscow, Russia, 109004
[5]South Ural State University, Lenin prospect 76, Chelyabinsk, Russia, 454080

agf@icc.ru

Abstract. Nowadays, the study of the behavior of social, economic, and technical queuing systems at different stages of their design and operation is a challenge for simulation modeling. The complexity of the dynamic structure of such systems is owing to a number of factors. Among them are a large number of system characteristics and relations between them, the existence of a large spectrum of distribution laws of random events, the need to collect and analyze current data on the system under study, and the presence of various constraints. Moreover, relations can be represented by functional, statistical, ambiguous, or other mappings. Unfortunately, simulation modeling tools are not widespread in designing and applying business processes, unlike other products of information and computation technologies (system of the office administration, warehouse management, etc.). In this regard, we present an approach to developing web-service for simulation modeling of queuing systems. Within the approach, we automate some stages of web-service development and use. To this end, we apply tools that support distributed computing with parameter variations in a heterogeneous environment, which includes virtualized resources. We provide a multi-agent management of computations. The practical use of the tools is shown on the example of developing a web-service for simulation modeling of a typical health and care institution.

1. Introduction

Queuing Systems (QSs) are used to model the processes of customer service, production, material handling, transportation, communication, health and care, etc. Applying modeling based on the Discrete Event Simulation (DES) allows us to significantly facilitate the selection of rational schemes for the QS work, determination of the needed control actions, and prediction of the results of their impact [1]. Within the framework of such modeling, a simulation program to run the QS model is developed. The researcher adjusts the model parameters by varying their values in permissible domains and obtains the target values of the model outputs, which are then optimized.

Distributed Computing with Parameter Variations (DCPV) causes high combinatorial complexity of the simulation. Moreover, they lead to large overhead costs of RAM and disk space. In this regard, a qualitative study of QSs is impossible without applying the High-Performance Computing (HPC) systems. Nowadays, such systems are represented by the resources of the public access centers, grid systems, and cloud or fog platforms. Without the use of HPC, a strong reducing in details of the QS model is required. Unfortunately, such reduction can lead to significant distortion of the simulation results. A distortion arises owing to the fact that some combinations of parameters affecting the simulation results cannot be considered.

Integration of simulation modeling with parallel and distributed computing provides the opportunity of carrying out large-scale experiments, ability to generate a larger number of initial data variants, and extension of the spectrum of problems to be solved [2]. At the same time, it should be noted the current trend of integrating methods of simulation and optimization [3].

Developing and applying a simulation software that effectively and correctly reproduces the QS under study is a challenge [4, 5]. Thus, there is a need for high-level tools that may assist the QSs developers to automate these processes.

The paper addresses developing and applying simulation models based on the integrated use of conceptual programming, knowledge engineering, multi-criteria optimization, distributed computing, multi-agent and service-oriented technologies.

The paper is structured as follows. Section 2 presents a brief overview of related work. Aspects of the proposed approach are given in Section 3. Section 4 provides an analysis of the practical experiment results. Finally, Section 5 concludes the paper and shows future works.

2. Related work

Among all-purpose DES systems, there is a large spectrum of the well-known tools for QSs modeling [6-8]. However, many tools do not fully use available HPC capabilities. Some tools do not support the service-oriented technologies that provide the required components of modern modeling tools. Many tools are provided as part of expensive software.

Eldabi T. et al. [9] note that a modern trend in intellectualization in relation to the QSs study is the use of hybrid models based on the conformed application of analytical and simulation models in combination with multi-agent technologies. The agent-based approach provides the necessary properties of self-organization of the system under study, inherent in the real environment of its functioning [10, 11].

Zhang X. [12] presents the current advances of DES in its applying to decision-making for the operation improvement of health and care institutions. He highlights a constantly increasing relevance of DES modeling in health and care service and management. This is evident for identifying service bottlenecks and decision-making support based on the integration of current and retrospective data.

In particular, Luo L. et al. [13] point out the importance of efficient queuing. They study the impact an emergency reservation policy applied to stochastic patient arrivals to reduction wait times and other service metrics.

Onggo B.S.S. et al. [14] show that many DES tools can represent a simple healthcare servicing and create a simulation model automatically. At the same time, they have a number of limitations related to the capabilities for data-driven servicing and decision-making.

Capan M. et al. [15] address a relevant problem of the improvement in decision-making through the efficient modeling techniques taking into account the existing restrictions. In particular, they conclude that simulation modeling as one of the operation research methods can be used to support in decision-making related to the development of health policies and service delivery by optimizing the performance of system components and analyzing their interaction under the current limitations.

Moreover, Zhang C. et al. [16] indicate the usefulness of modeling for non-technical skill training in health and care servicing.

Thus, research in the subject area under consideration is being actively pursued. Figure 1 highlights the perspective directions in developing DES in studying QSs. However, many problems still remain unresolved.

To this end, we propose a new approach to the study of QSs. In comparison with the aforementioned approaches, it is based on the integration of conceptual programming, knowledge engineering, multi-criteria optimization, distributed computing, multi-agent and service-oriented technologies. Within the approach, we provide virtualization of computing resources.

Results from previous studies related to uncertainty mitigation, data storage and transfer, clouds integration, and computation management for cloud computing are presented in [17-21].

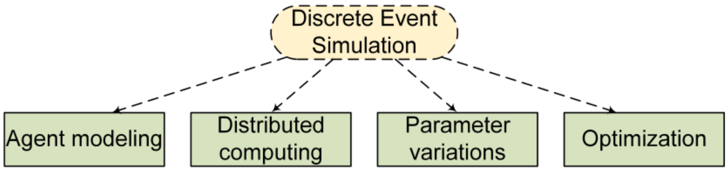

Figure 1. DES in studying QSs.

3. Approach aspects

We use specialized methods and tools that provide the automation of constructing and applying simulation models in the GPSS language to study QSs within DES [22] (Figure 2). Instances of a GPSS model are run in a distributed computing environment with different sets of model inputs. The search for optimal solutions is based on the use of a set of methods for multi-criteria selection of modeling results. Among them are Pareto optimal, lexicographic, and majority methods [23]. The selection of the specific method from the aforementioned ones depends on the information completeness provided by experts in a subject domain.

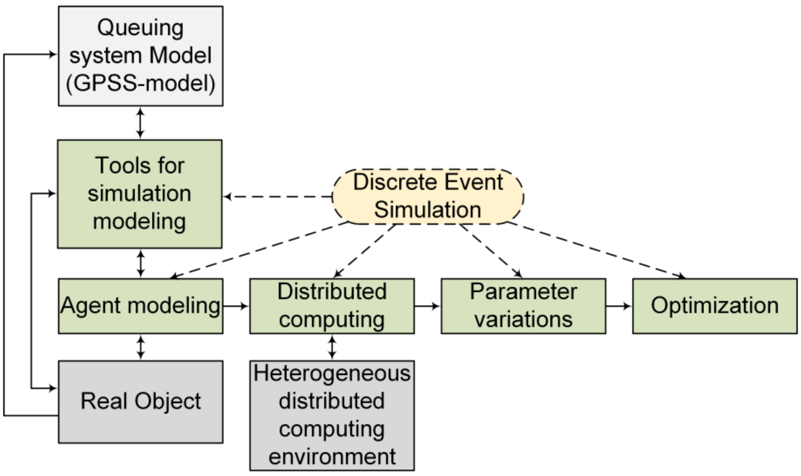

Figure 2. Proposed scheme of constructing and applying simulation models.

The structure $(m/m/k)$:(FCFS/n/$Infinite$) is used as a base model for QS [24]. It assumes a Poisson distribution for the transaction arrival time into QS and an exponential distribution for their service time intervals. The model includes the finite number k of parallel services. A type of the queue service discipline is First-Come-First-Served (FCFS) with the finite number n of requests in the queue accepted for service. The capacity of the source generating transactions is unlimited.

The model hybridity is ensured by the use of agents that represent information and management systems of real objects and transfer current and retrospective data to the model. Unlike the well-known systems that implement agent-based modeling [25], for example, MASDK [26] or AnyLogic [27], agents are developed using JADE [28]. The model of their behavior is designed using a special add-on to this tool. The add-on provides a description of both the subject domain for the object represented by the agent and its actions. Interaction with the agent is implemented through calls to external functions using the built-in language PLUS [22]. Features of the design of agents, models of their behavior, and aspects of the Multi-Agent System (MAS) operation are presented in [29, 30].

The GPSS model is implemented as an application that supports DCPV and generates a set of independent jobs. Such job executes the model with one of the variants of the initial data for model inputs. Jobs are launched in parallel in the computing environment. The computations are well scalable due to the lack of interactions between jobs. The application can use libraries of ready-made typical QSs simulation models in the GPSS language. Analysis of the computation results and decision-making is carried out according to the following scheme:

- Solving the direct problem in order to determine the set of optimal variants of the observed variables (criteria for the functioning quality of the QS under study),
- Solving the inverse problem for searching the optimal variants for the values of the input variables (parameters that determine the conditions for the QS functioning),
- Decision-making and formation of control actions in the system based on the determined optimal variants for the values of the input variables.

In searching optimal solutions, we use the aforementioned set of multi-criteria selection methods. The use of one method or another depends on the degree of completeness of the information provided by experts in the subject domain.

The simulation web-service is created automatically based on the REST approach according to the model specification in the JSON data transfer format (Figure 3).

```
{
   "name": "QS simulation model",
   "model": "qs_model.gps",
   "title": "Simulation model of a typical health and
             care institution",
   "description": "Simulation model in the GPSS language
                   of a typical health and care
                   institution. Version 1.2.",
   "parameters": {
      "inputFile": "Variant.txt",
      "dataPath": "~/models/QS simulation model/",
      "input": [
         {
            "type": "number",
            "name": "Int1",
            "gpssName": "Int1",
            "title": "Patient receipt interval",
            "value": "1.5",
            "variable": "true",
            "step": "0,5",
            "minValue": "0,5",
            "maxValue": "3"
         },
         ...
      ],
      "output": [
         {
            "type": "string",
            "name": "Result",
            "title": "GPSS report",
            "resultPath": "~/models/QS simulation
model/out"
         }
      ]
   },
   "commands": {
      "start": "./start.sh ${input_num} ${input_text}"
   }
}
```

Figure 3. Fragment of the model specification.

The server hosts an application (node.js) that accepts requests over the HTTP protocol. In accordance with the requests, scripts in the BASH language are executed using node.js on behalf of an unprivileged user. These scripts launch Virtual Machines (VMs) in computing resource queues and start the simulation process. We apply OpenStack for virtualizing resources of the heterogeneous distributed computing environment that are used for executing the GPSS models [31].

Figure 4 shows a scheme of multi-agent management of DCPV.

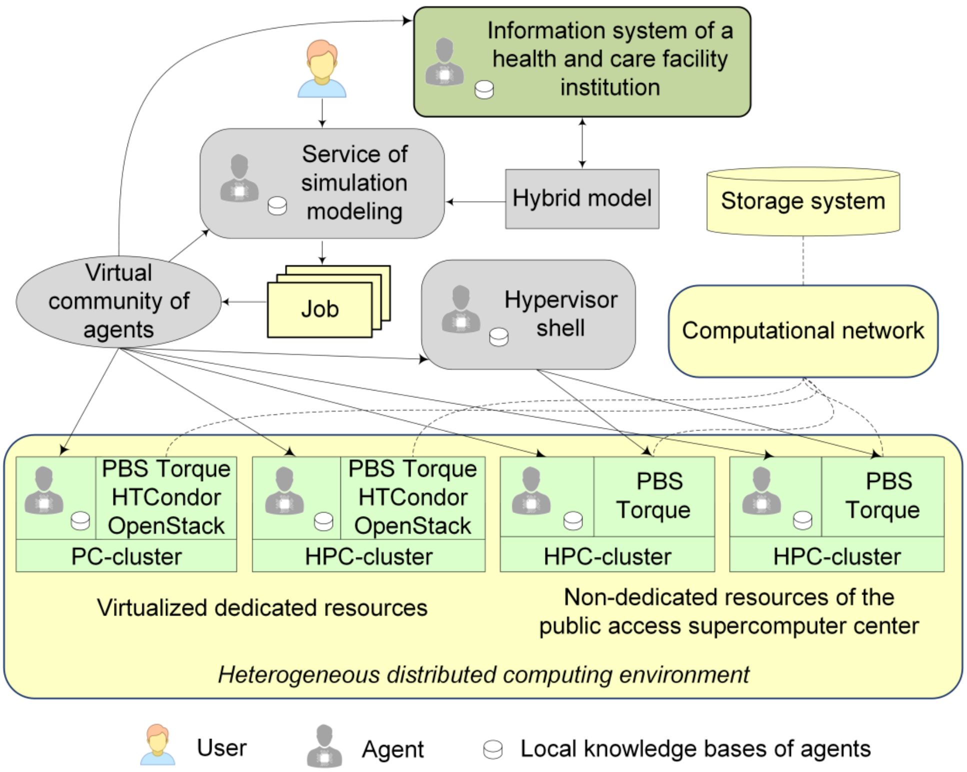

Figure 4. Multi-agent management of DCPV.

In addition, we can redistribute the computational load of virtualized resources to the free non-dedicated nodes in the environment. To this end, we apply a hypervisor shell for scheduling and running VMs from the queues of local resource managers of non-dedicated nodes [32]. All computing resources of the environment are represented by agents.

4. Practical experiment

The developed tools were applied for creating the web-service of simulating modeling the functioning of a typical health and care institution. Such an institution consists of several main subdivisions (polyclinic, diagnostic center, hospital, and other structures). Each subdivision provides a set of single-channel and multi-channel services. The structural and parametric characteristics of the institution were formed on the basis of generalized year report data from one of the medical institutions of the Irkutsk region.

Figure 5 shows a scheme of patient service. The registry is the first serving multi-channel device. At the registry, patients receive destines depending on their categories and requests of health and care services.

Specialists receive patients. Some services are paid at the cashbox. After receiving the patient by specialists, the patient's service in the system ends. In addition, the patient can be destined to the laboratory or hospital.

The simulation results provide information about a load of individual areas of the institution, queue waiting, the average duration of patients in the system, and other important data. This knowledge helps in decision-making for optimizing the institution work by reducing queues, redistributing patient flows and services, and improving the working regime of specialists. This is especially important in times of epidemics when it is necessary to restructure usual schemes of work and adapt existing resources to new challenges [33].

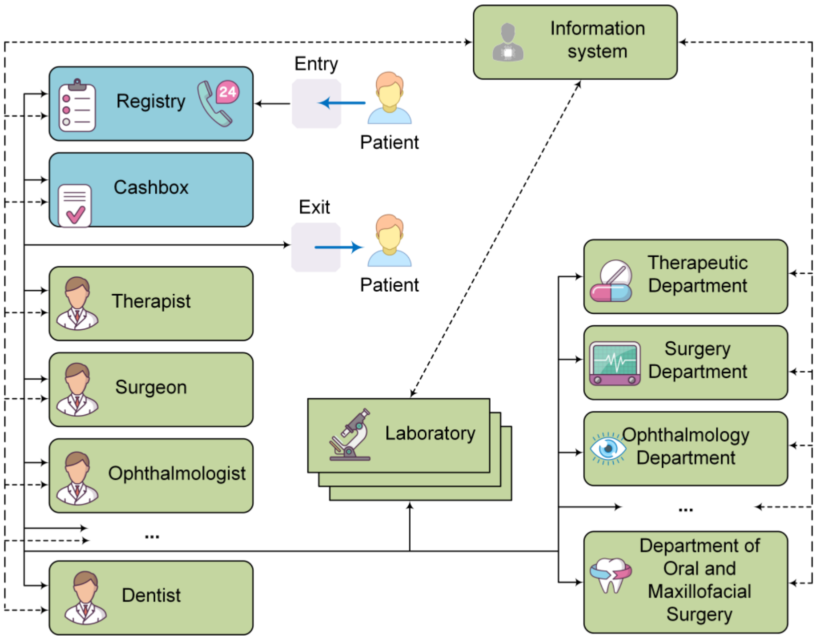

Figure 5. Typical service scheme for patients.

Table 1. Model segment description.

Segment parameter	Parameter specification
Transactions	Patients of various categories that are needed different health and care services
Services	Registrars
Input variables of the segment	Number of registrars, intensity of transaction flows, intensity of service, intensity of service faults, etc.
Output variables of the segment	Queue size, queue waiting time, number of customers of different categories per day, service load, etc.
Characteristics of flows	Number of flows, number of isolate flows, number of services per-flow

```
; REGISTRY
          ; Multichannel device: Registry
          ; Functions: Func_After_Reg, Func_pay
          ; Input variables: Intl, Otkl1, TimeReg, OtklReg
          ; Input labels: Met1, Met2
          ; Output labels: Met3, Met4, Met8, Met9, Met10

          GENERATE  X$Int1,X$Otkl1          ; Receipt of a transaction at the registry
          QUEUE     Common_queue            ; Receipt of a transaction in the queue for collecting statistics
                                             of the patient's stay in the clinic
Met1      QUEUE     Registry                ; Transaction queued up at the registry
          ENTER     Registry                ; Start of service at the registry
          DEPART    Registry                ; Leaving a transaction from the queue
          ADVANCE   X$TimeReg,X$OtklReg     ; Transaction service
          LEAVE     Registry                ; Completion of transaction service at the registry
          TRANSFER  FN,Func_pay             ; Transaction transfer to further service at the cashier
Met2      TRANSFER  FN,Func_After_Reg       ; Transition of a transaction to further servicing by specialists
```

Figure 6. Fragment of the program code in the GPSS language.

Institution departments are represented in the simulation model by the following segments: Registry, Cashbox, segments for modeling input and output flows of patients, segments for modeling specialist operations, Laboratories, Hospital departments. Each of them has its own characteristics and a set of input and output variables. As an example, Table 1 demonstrates the description of the Registry segment. Fragment of the program code in the GPSS language is shown in Figure 6.

Figure 7(a) and Figure 7(b) demonstrate the part of results in solving one of the problems in the study of a typical health and care institution. The problem is to determine the required additional resources (in our case, human ones) for meeting the given constraints of servicing while increasing the patient flow. Figure 7(a) provides a graph showing the patient's waiting time in the registry queue. With an increase in the number of patients with a constant number of registrars, this time increases nonlinearly. When critical values are reached, it becomes necessary to attract additional resources by adding new registrars to the system. The number of registrars varies from 2 to 4. The number of patients reaches 744. The OY axis reflects the average time that patients are in the Registry queue.

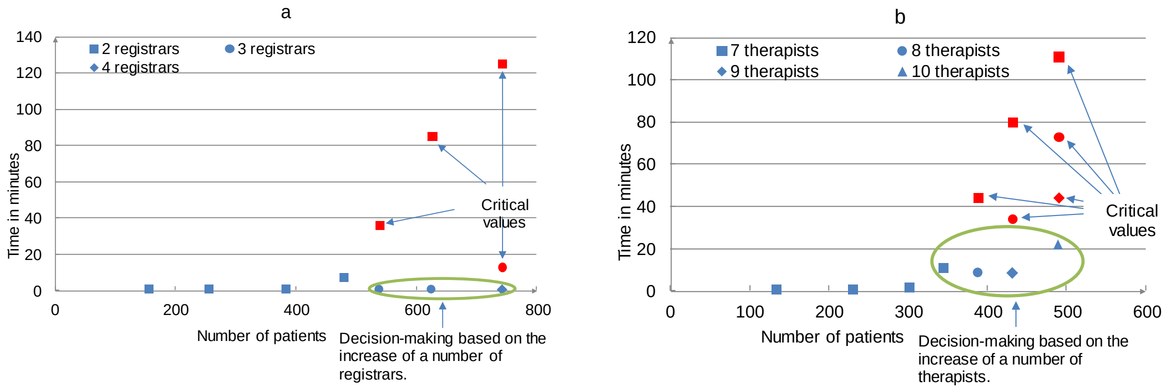

Figure 7. Patient waiting time in the queue at the Registry (a) and the therapist (b).

Thus, in modeling, the critical values of the number of patients are determined. At such critical values, it is necessary to apply control actions to satisfy the given constraints. Different variants of control actions can be investigated. The choice of actions is based on a multi-criteria analysis of the simulation results.

Figure 7(b) reflects the results of modeling the waiting time in the queue to the therapist. The number of therapists varies from 7 to 10. The number of patients reaches 506. The OY axis reflects the average time that patients are in the therapist queue.

When solving this problem through running the model with different variants of the initial data, results similar to those shown in Figure 7 are also determined for other institute specialists, Cashbox, Laboratories, and Hospital departments simultaneously.

Computational experiments were performed based on typical industry standards for patient care times. The model allows you to vary the values of its variables. We can also introduce external disturbances into the model that affect the patient care process. For example, we can change the number of queues, create isolated service pipelines, form new patient flows, and determine the optimal service structure. To study the considered model, a modular application has been developed. It supports DCPV. The modular approach makes it possible to easily modify the model by adding new segments implemented by separate modules.

The experiment based on industry-standard patient service times [34]. However, we can introduce external disturbances into the model that affect these times in practice.

In the case of applying parameter sweep computations, the number n of variants for the initial data can grow exponentially with an increase in the number of input parameters of the model and the number of their possible values. It is determined by the following formula:

$$n = k_1 \times k_2 \times \dots \times k_m,$$

where k_i is a number of possible values for the ith parameter, $i = \overline{1, m}$, m is a number of input parameters. For example, for 10 parameters, each of which has 4 different values, we generate 1,048,576 variants of the initial data. In practice, simulation models are more complex. They have more parameters and their values. Moreover, we need the determined number of times to run the model with each variant to provide the required veracity of the stochastic results of simulation modeling.

Therefore, it is impossible to perform large-scale computations with such models on a personal computer in a reasonable time. Thus, the need for HPC is obvious.

Figure 8(a) and Figure 8(b) show speedup of computations and efficiency of the resource use in virtualized dedicated resources of the environment. These parameters were achieved in executing the application for a different number of VMs with different characteristics under the MAS and Condor DAGMan management correspondingly. The linear speedup in Figure 8(a) and efficiency equal 1 in Figure 8(b) are given for clarity of the results.

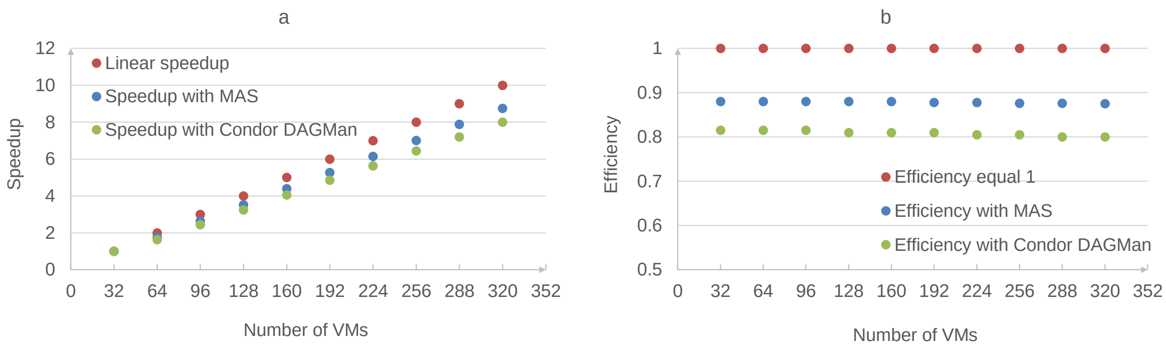

Figure 8. Computing speedup (a) and resource use efficiency (b) for DCPV.

In comparison with the Condor DAGMan meta-scheduler, MAS demonstrates the advantages in both the speedup of computations and efficiency of the resource use. These advantages are achieved due to predicting the job execution time and characteristics of the VMs in distributing the computational load.

Moreover, the distribution of part of the jobs into free slots of non-dedicated resources of the public access supercomputer center allows us to reduce the problem-solving makespan by about 10-15%. The possibility of such distribution is provided by our hypervisor shell.

In this experiment, both of these parameters remain high enough with an increase in the number of VMs. Thus, the considered results demonstrate the scalability of the developed application.

Figure 9 and Figure 10 illustrate the web-service interface. Figure 9 demonstrates the screenshot of fragments of the form for preparing and running a simulation model. Figure 10 shows the screenshot with the form for analyzing of the obtained experiment results through the use of the aforementioned multi-criteria methods.

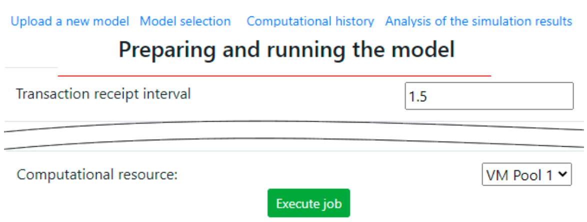

Figure 9. Screenshot of the form for preparing and running a model.

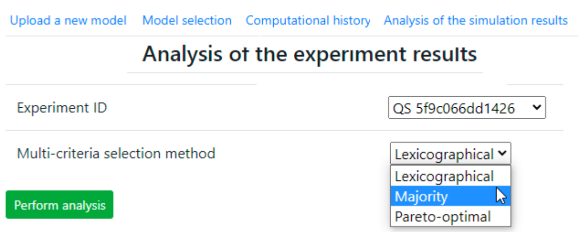

Figure 10. Screenshot of the form for analyzing of the experiment results.

5. Conclusions

The paper is devoted to a new approach to the subject-oriented study of QSs based on simulation modeling in the heterogeneous distributed computing environment. This research direction is extremely relevant. It is owing to the well-known simulation tools often do not take full advantage of distributed computing and service-oriented programming possibilities. Moreover, unlike such tools, we additionally provide the integrated use of the opportunities of knowledge engineering, multi-criteria optimization, and multi-agent technologies.

We presented a software and hardware platform based on virtualized resources. Applying this platform, we developed tools for the QSs simulation. The developed tools have been used to create a service for simulation modeling. The high-level user interface of the tools provides flexible and convenient access for subject domain specialists (managers, IT-officer, administrators, and other participants in the decision-making process) to the service components during the QSs simulation.

The created service has been used to solve a model problem of analyzing the effectiveness of patient servicing in a typical health and care institution. The experiments have shown high scalability of computations performed using the service.

A further direction of our work is the implementation of the developed software and hardware platform for the study of structural and parametric features of health and care facility institutions in Irkutsk region. In collaboration with specialists from these institutions, we plan to develop behavioral models for agents representing patients and medical staff.

Acknowledgments

The study is supported by the Russian Foundation of Basic Research, project no. 19-07-00097.

References

[1] Pidd M 2004 *Computer simulation in management science* (New York: Wiley) p 332

[2] Sulistio A, Yeo C S and Buyya R 2004 Taxonomy of computer-based simulations and its mapping to parallel and distributed systems simulation tools *Softw. Pract. Exper.* **34(7)** 653–673

[3] Panadero J, Juan A A, Mozos J M, Corlu C G and Onggo B S 2018 Agent-based simheuristics: extending simulation-optimization algorithms via distributed and parallel computing *Proc. of the Winter Simulation Conference (WSC 2018)* (IEEE Press) pp 869–880

[4] Cordasco G, Spagnuolo C and Scarano V 2016 Toward the new version of D-MASON: efficiency, effectiveness and correctness in parallel and distributed agent-based simulations *Proc. of the International Parallel and Distributed Processing Symposium Workshops (IPDPSW 2016)* (IEEE Press) pp 1803–1812

[5] Karolius S and Preisig H 2018 Developing simulation tools for interdisciplinary modelling *Proc. of The 59th Conference on Simulation and Modelling (SIMS 2018)* (Oslo Metropolitan University, Norway, Linköping University Electronic Press) **153** pp 210–215

[6] Dias L M S, Vieira A A C, Pereira G A B and Oliveira J A 2016. Discrete simulation software ranking – A top list of the worldwide most popular and used tools *Proc. of the 2016 Winter Simulation Conference (WSC 2016)* (IEEE Press) pp 1060–1071

[7] Rashidi H 2017 Discrete simulation software: a survey on taxonomies *J. Simul.* **11(2)** 174–184

[8] Suh Y K and Lee K Y 2018 A survey of simulation provenance systems: modeling, capturing, querying, visualization, and advanced utilizatio. *Hum.-centric Comput. Inf. Sci* **8(1)** 27

[9] Eldabi T, Balaban M, Brailsford S, Mustafee N, Nance R E, Onggo B S and Sargent R G 2016 Hybrid simulation: historical lessons, present challenges and futures *Proc. of the 2016 Winter Simulation Conference (WSC 2016)* (IEEE Press) pp 1388–1403

[10] Serugendo G D M, Gleizes M P and Karageorgos A 2005 Self-organization in multi-agent systems *Knowl. Eng. Rev.* **20(2)** 165–189

[11] Kovtunenko A, Bilyalov A and Valeev S 2018 Distributed streaming data processing in IoT systems using multi-agent software architecture *Lect. Notes Comput. Sc.* **11118** 572–583

[12] Zhang X 2018 Application of discrete event simulation in health care: a systematic review. *BMC Health Serv. Res.* **18(1)** 687
[13] Luo L, Zhang Y, Qing F, Ding H, Shi, Y and Guo H 2018 A discrete event simulation approach for reserving capacity for emergency patients in the radiology department. *BMC Health Serv. Res.* **18(1)** 452
[14] Onggo B S S, Proudlove N C, D'Ambrogio S A, Calabrese A, Bisogno S and Levialdi Ghiron N 2018 A BPMN extension to support discrete-event simulation for healthcare applications: an explicit representation of queues, attributes and data-driven decision points. *J. Oper. Res. Soc.* **69(5)** 788–802
[15] Capan M, Khojandi A, Denton B T, Williams K D, Ayer T, Chhatwal J, Kurt M, Lobo J M, Roberts M S, Zaric G, Zhang S and Schwartz J S 2017 From data to improved decisions: Operations Research in healthcare delivery. *Med. Decis. Making* **37(8)** 849–859
[16] Zhang C, Grandits T, Härenstam K P, Hauge J B and Meijer S 2018 A systematic literature review of simulation models for non-technical skill training in healthcare logistics. Advances in Simulation **3(1)** 15
[17] Tchernykh A, Schwiegelsohn U, Talbi E G and Babenko M. 2019 Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability. *J. Comput. Sci.-Neth.* **36** 100581
[18] Tchernykh A, Babenko M, Chervyakov N, Miranda-López V, Kuchukov V, Cortés-Mendoza J M, Deryabin M, Kucherov N, Radchenko G and Avetisyan A 2018 AC-RRNS: Anti-collusion secured data sharing scheme for cloud storage. *Int. J. Approx. Reason.* **102** 60–73.
[19] Tchernykh A, Cortés-Mendoza J M, Bychkov I, Feoktistov A, Didelot L, Bouvry P, Radchenko G and Borodulin K 2019 Configurable cost-quality optimization of cloud-based VoIP. *J. Parallel Distr. Com.* **133** 319-336
[20] Tchernykh A, Tchernykh, A., Miranda-López V, Babenko M, Armenta-Cano F, Radchenko G, Drozdov A Y and Avetisyan A 2019 Performance evaluation of secret sharing schemes with data recovery in secured and reliable heterogeneous multi-cloud storage. *Cluster Comput.* 22(4) 1173–1185
[21] Feoktistov A, Sidorov I, Tchernykh A, Edelev A, Zorkalzev V, Gorsky S, Kostromin R, Bychkov I and Avetisyan A 2018 Multi-Agent Approach for Dynamic Elasticity of Virtual Machines Provisioning in Heterogeneous Distributed Computing Environment. *Proc. of the International Conference on High Performance Computing and Simulation* (IEEE Press) pp 909–916
[22] Feoktistov A G, Kostromin R O, Fereferov E S, Tchernykh A, Basharina O Yu, Dmitriev V I and Kurzybova Y V 2019 Toolkit for simulation modeling of queue systems in Grid *Proc. of the 1st International Workshop on Information, Computation, and Control Systems for Distributed Environments (ICCS-DE'19)* (CEUR-WS Proceedings) **2430** pp 51–59
[23] Ho W, Xu X and Dey P K 2010 Multi-criteria decision making approaches for supplier evaluation and selection: A literature review *Eur. J. Oper. Res.* **202(1)** 16–24
[24] Taha H A 2016. *Operations research: an introduction* (Cambridge: Pearson) p 862
[25] Abar S, Theodoropoulos G K, Lemarinier P and O'Hare G 2017 Agent based modelling and simulation tools: A review of the state-of-art software. *Comput. Sci. Rev.* **24** 13–33
[26] Multi-agent System Development Kit. Available at: http://meta-guide.com/bots-agents-assistants/multi-agent-system-development-kit-masdk (accessed: 13.10.2020)
[27] AnyLogic: Simulation modeling software tools and solutions for business. Available at: https://www.anylogic.com/ (accessed: 13.10.2020)
[28] Bellifemine F, Bergenti F, Caire G and Poggi A 2005 Jade: A Java Agent Development Framework *Mu. S. Art. Soc. Sim. Org.* **15** 125–147
[29] Feoktistov A, Tchernykh A, Kostromin R and Gorsky S 2017 Knowledge elicitation in multi-agent system for distributed computing management *Proc. of the 40th International Convention on information and communication technology, electronics and microelectronics*

(MIPRO’17) (Riejka: IEEE) pp 1350–1355
[30] Feoktistov A, Kostromin R and Tchernykh A 2018 Agent Behavior Model for Distributed Computing Management in the Environment with Virtualized Resources *Proc. of the 41st International Convention on information and communication technology, electronics and microelectronics (MIPRO 2018)* (IEEE Press) pp 1153–1158
[31] Open Source Cloud Computing Infrastructure. Available at: https://www.openstack.org/ (accessed: 13.10.2020)
[32] Feoktistov A G, Sidorov I A, Sergeev V V, Kostromin R O and Bogdanova V G Virtualization of heterogeneous HPC-clusters based on OpenStack platform *Bull. South Ural State Univ. Ser. Computational Math. Softw. Eng.* **6(2)** 37–48
[33] Currie C S M, Fowler J W, Kotiadis K, Monks T, Onggo B S, Robertson D A and Tako A A 2020 How simulation modelling can help reduce the impact of COVID-19. *J. Simul.* **14** 1–15
[34] Order of the Ministry of Health of the Russian Federation, June 2, 2015, no. 290n. Available at: https://minzdrav.gov.ru/documents/9082-prikaz-ministerstva-zdravoohraneniya-rossiyskoy-federatsii-ot-2-iyunya-2015-g-290n-ob-utverzhdenii-tipovyh-otraslevyh-norm-vremeni-na-vypolnenie-rabot-svyazannyh-s-posescheniem-odnim-patsientom-vracha-pediatra-uchastkovogo-vracha-terapevta-uchastkovogo-vracha-obschey-praktiki-semeynogo-vracha-vracha-nevrologa-vracha-otorinolaringologa-vracha-oftalmologa-i-vracha-akushera-ginekologa (accessed: 13.10.2020)