IOP Conf. Series: Materials Science and Engineering **177** (2017) 012017 doi:10.1088/1757-899X/177/1/012017

Simulation model of load balancing in distributed computing systems

I A Botygin, V N Popov and S G Frolov

Tomsk Polytechnic University, 30, Lenina ave., Tomsk, 634050, Russia

E-mail: bia@tpu.ru

Abstract. The availability of high-performance computing, high speed data transfer over the network and widespread of software for the design and pre-production in mechanical engineering have led to the fact that at the present time the large industrial enterprises and small engineering companies implement complex computer systems for efficient solutions of production and management tasks. Such computer systems are generally built on the basis of distributed heterogeneous computer systems. The analytical problems solved by such systems are the key models of research, but the system-wide problems of efficient distribution (balancing) of the computational load and accommodation input, intermediate and output databases are no less important. The main tasks of this balancing system are load and condition monitoring of compute nodes, and the selection of a node for transition of the user's request in accordance with a predetermined algorithm. The load balancing is one of the most used methods of increasing productivity of distributed computing systems through the optimal allocation of tasks between the computer system nodes. Therefore, the development of methods and algorithms for computing optimal scheduling in a distributed system, dynamically changing its infrastructure, is an important task.

1. Introduction

Currently, the design and production preparation in mechanical engineering implemented a wide range of engineering calculations (strength, hydrodynamic, thermal, etc.). These tasks require a much more machine resources than the systems of design or production preparation. Modern systems of automation engineering calculations, depending on the scale of the task, are carried out parallel calculations using different methods and structures of distributed computing. One of the major problems in distributed computing is scheduling of loading of nodes in the system.

Today, there are many methods of the load balancing: Round Robin, Weighted Round Robin, Least connections, Weighted least connections, and others. The load balancing is also divided into static, semi-dynamic and dynamic. Static balancing is performed prior to the start of the task, semi-dynamic load balancing requires that the distribution of tasks carried out at the stage of initialization and dynamic balancing is performed in the course of computation. If we talk about how to interact the load balancing in a computer system, the balancing is divided into central, where the special compute node (manager) distributes the modules of a task between calculators, and distributed, in which status data is exchanged between all nodes in a distributed computing system by using a special algorithm [1–14].

The article describes the method of computational load balancing in distributed automation systems which are focused on multi-agent and multi-threaded data processing. A scheme for the control of request processing from the terminal devices is proposed. This scheme provides an effective dynamic horizontal scaling of computing power at high loads. A GPSS-model and the results of model experiments for developed algorithm load dispatching are represented. This model and results show the effectiveness of the algorithm even with a significant increase in the number of connected nodes and with an increase in architecture of distributed computing system.

2. The architecture of a distributed computing system

Modeling of functioning of load balancing algorithm was investigated in the functional structure of a distributed computing system shown in Figure 1.

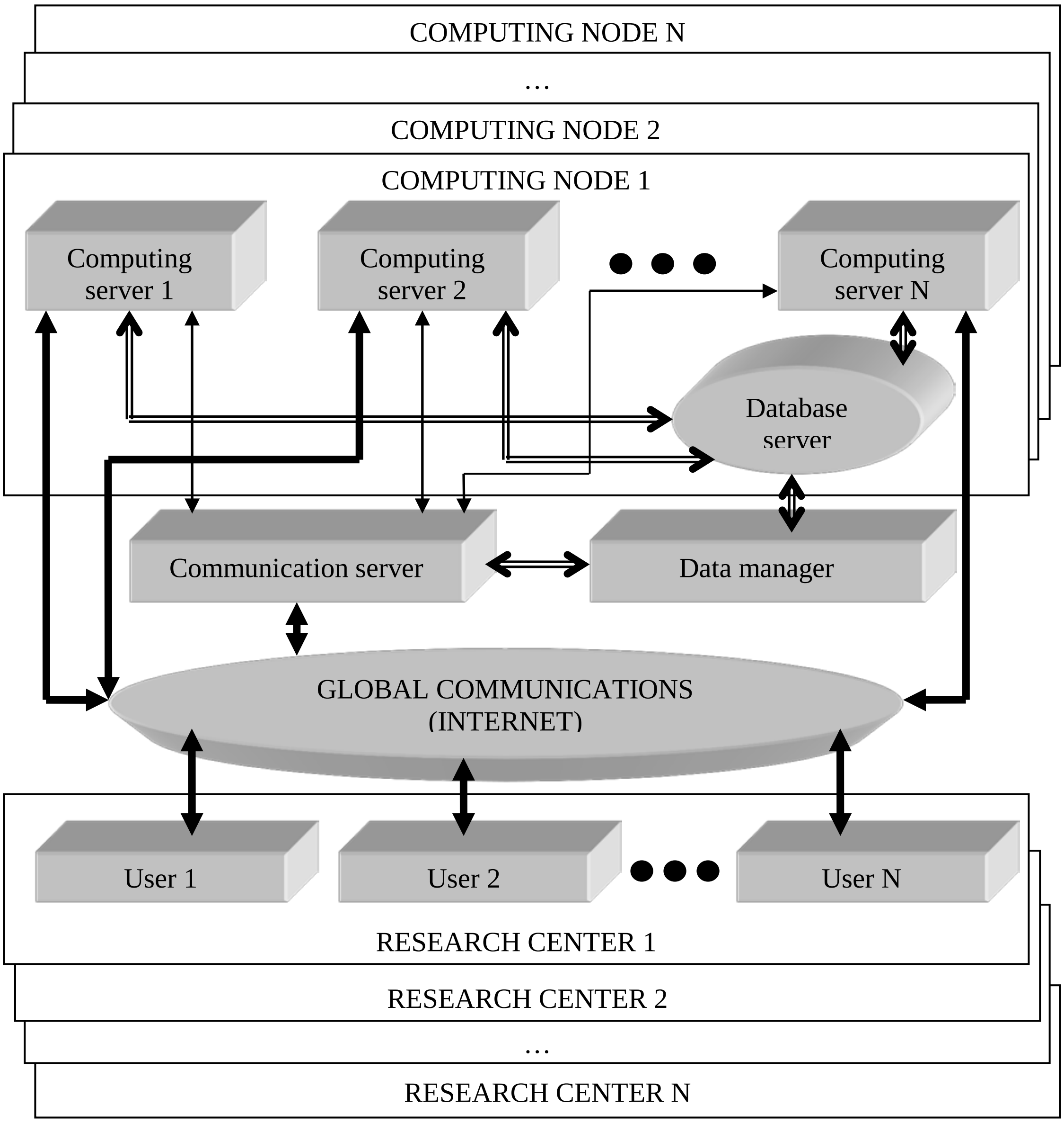


Figure 1. Architecture of distributed computing data processing system.

A feature of this system is the ability to increase performance using horizontal scaling computing nodes and resources. The system architecture is based on the components used for the construction of modern tools of distributed computing (grid-systems). But the internal structure of the complex, the contents of functional components (middleware) protocols and their interaction are original. Architecture of this complex virtualizes three basic technical resources on which the high-performance data center (computer systems, data storage systems and global communication) is constructed, and

IOP Conf. Series: Materials Science and Engineering **177** (2017) 012017 doi:10.1088/1757-899X/177/1/012017

then collects them into a single virtual computer to provide users of this center with its resources as services.

Computing servers, a communication server and a database server are allocated in the structure of distributed computing data processing system. Planning and dispatching of data processing is entrusted to the special communication server. The interaction of all users with this complex is carried out only through the communication server. The main task of the communication server is to provide optimal loading of the available computing servers and to provide users with real-time mode (online). In this situation, the user does not know what particular network node performs his task. The user just consumes a certain number of virtual processing power capacity available in the network. Mathematical data processing is performed using of the computing servers.

A scheme of the developed algorithm for selecting of the computing server by means of the communication server is shown in Figure 2.

The algorithm uses information that is stored in the profiles (meta descriptions) of computing servers (a table of processors, a table of compliance of tasks and a table of tasks).

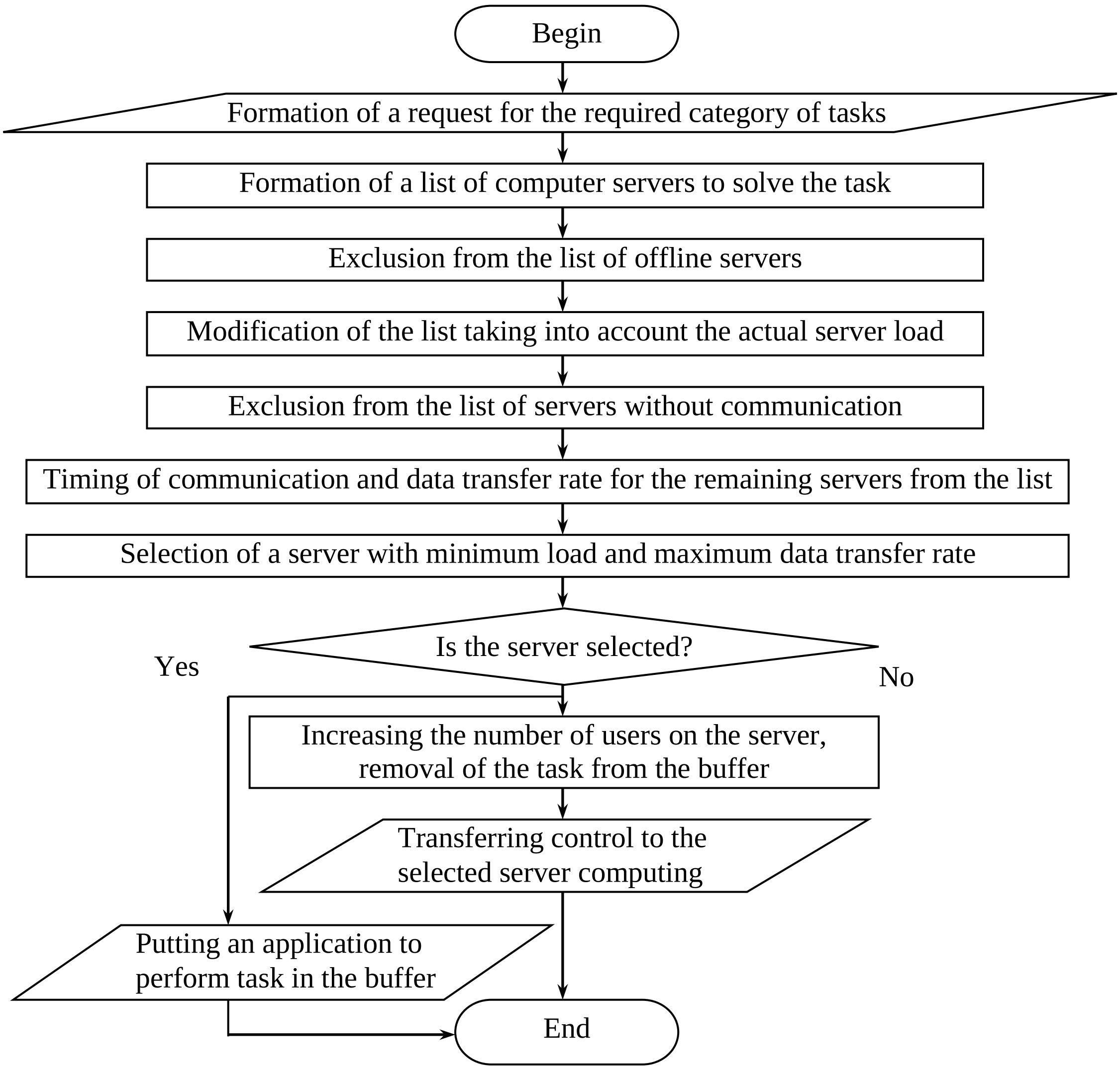


Figure 2. Algorithm of functioning of communication server

The main steps of the algorithm:

1. Creating a list of computer servers to solve a task.
2. Servers which are offline are deleted from the list.

3. Servers which are busy are deleted from the list.
4. Servers which have no connection by technical reasons are deleted from the list.
5. Determination of the connection time and data transfer speed of the servers.
6. Determination of the server with the most free computing power.
7. If there are several servers with the same processing power, then the server that has a greater data transfer rate is selected.
8. The value of the field ‘Number of users’ is increased by one.
9. In the absence of servers with free computing capacity, an application for the task execution is placed in the buffer of the communication server.
10. Control is transferred to the selected computing server.

The selection criterion is to find a server which has a minimum load and a maximum speed of data exchange.

3. GPSS-model for assessment of the effectiveness of the load balancing algorithm

A simulation model in the GPSS World language was developed for assessment of the effectiveness of a distributed computing system.

Figure 3 shows an enlarged model of the load balancing algorithm.

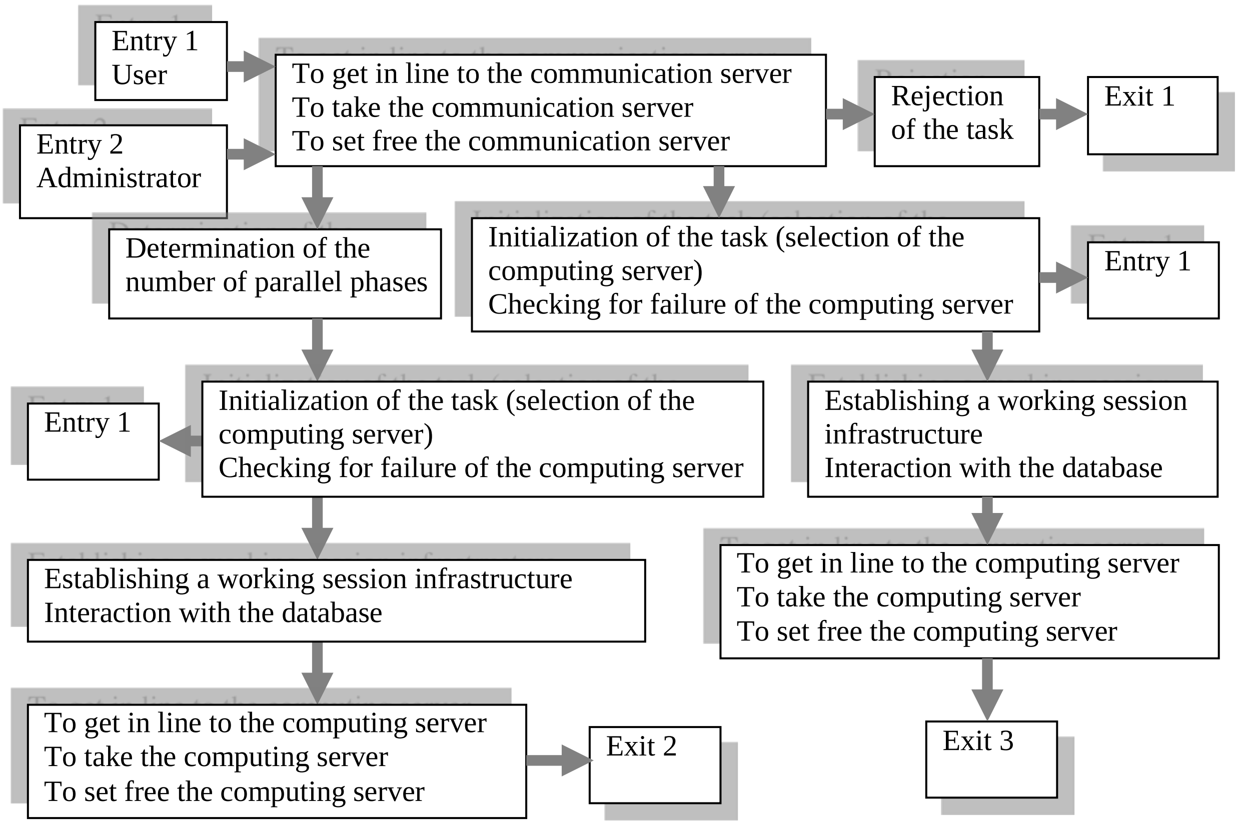
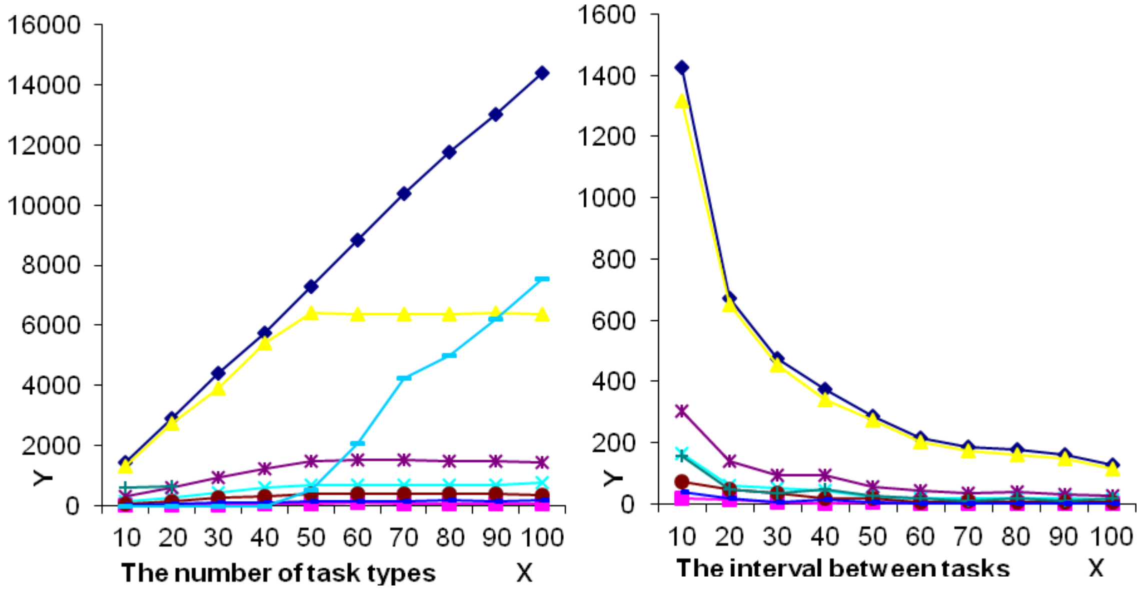


Figure 3. Gpss model for assessment of effectiveness of load balancing algorithm.

The main objects of the modeled system are flows of input tasks, computing segments, computing servers of segments, the communication server, database servers, the database manager. Calculation of the moments of task occurrences (a random value associated with the time interval between occurrences of two neighbor tasks) is performed according to the normal distribution law. Service intervals are also a random value which is distributed exponentially.

Variable parameters of the model are: the number of computing servers, maximum time of task processing, the number of task types, average interval between tasks, the proportion of parallel tasks, the proportion of rejected tasks, the average time of task processing by the communication server, deviation from the average task processing time of the communication server, failure probability of a computing server, recovery probability of a computing server, the recovery average time of a computing server, deviation from the recovery average time of a computing server, the number of communication servers, the maximum time of administrator task processing. Typical tasks in this model are the tasks of collection, storage and processing of meteorological data for the problems of analysis and forecasting of climatic processes [15, 16].

The authors conducted a study of dependency of some criteria (the total number of tasks, the number of rejected tasks, the number of tasks without parallelism and with parallelism, the number of emergency situations with the recovery and without the recovery (for tasks without parallelism and with parallelism), the number of tasks in the queue) from the following parameters: the number of computing servers, maximum time of task processing, the number of task types, the interval between tasks, the number of communication servers (Figure 4).



a b

The total number of tasks; The number of rejected tasks; The number of tasks without parallelism; The number of tasks with parallelism; The number of emergency situations with the recovery; The number of emergency situations without the recovery; The number of emergency situations with the recovery (for the tasks with parallelism); The number of emergency situations without the recovery (for the tasks with parallelism); The number of tasks in the queue

Figure 4. Simulation results of distributed computing system: a – dependences of criteria on quantity of task types; b – dependences of criteria on interval between tasks.

The Y-axis shows the obtained values of the test criteria (the absolute numerical value of the amount), and the X-axis shows the absolute numerical values of the corresponding variable parameters. In all experiments, simulation was performed within 24 hours (simulation time).

The following performance parameters of computing servers were chosen in the model: the number (over 100), the maximum time of task processing (up to 100 units), the number of communication servers (up to 10). This combination of parameters has essentially no effect on the studied criteria. The number of task types significantly affect the number of tasks performed in the system and the

IOP Conf. Series: Materials Science and Engineering **177** (2017) 012017 doi:10.1088/1757-899X/177/1/012017

communication server queue.

Changing the range of task income greatly affects the studied criteria especially in the early stages of its increase.

Simulation experiments of the functional structure of a distributed computing system have shown its efficiency even with a significant increase in the number of connected nodes and scale of architecture. The proposed model parameters and static criteria for evaluating its operation allowed comprehensively and independently evaluate the quality of the developed model.

4. Conclusion

The question of the load planning in a distributed computing system should be addressed at an early stage of development of any service. Initially, the problem of inadequate performance of a distributed computing system in connection with increasing loads can be solved by building up the capacity of its servers but the effective method is the optimization of the algorithms and program codes.

References

[1] Kshemkalyani A D and Singhal M 2008 *Distributed Computing: Principles, Algorithms, and Systems* (Cambridge University Press)

[2] Md. Firoj Ali and Rafiqul Zaman Khan 2012 The study on load balancingstrategies in distributed computing system *Int. J. of Comp. Science and Engineering Survey* **3** 19-30

[3] Jain P and Gupta D 2009 An Algorithm for Dynamic Load Balancing in Distributed Systems with Multiple Supporting Nodes (Academy Publisher) pp 232-236

[4] Chhabra A, Singh G, Waraich S S, Sidhu B and Kumar G 2006 *Word Academy of Sci., Eng. and Techn.* 39-42

[5] Mesheryakov R, Moiseev A, Demin A, Dorofeev V and Sorokin V 2016 *Key Eng. Mat.* **685** 943-947

[6] Sanchez-Rodriguez D, Macías E M and Suarez A 2003 *Int. Conf. on Parallel and Distributed Processing Techniques and Applications* (Las Vegas, Nevada, USA) vol 1 (Providence: CSREA Press) pp 473-479

[7] Semchedine F, Bouallouche-Medjkoune L and Aissani D 2011 *J. of Network and Comp. Appl.* **34** 1123-30

[8] Sherstnev V S 2005 *The 9th Russian-Korean Int. Symp. on Sci. and Techn.* (Novosibirsk) vol 1 (Novosibirsk: NSTU Press) pp 696-700

[9] Shubhinder Kaur and Gurpreet Kaur 2015 *Int. J. of Comp. Applications* **125** vol 9 pp 25-28

[10] Imas O, Kaminskaya V and Sherstneva A 2015 *Proc. of 2015 Int. Conf. on Interactive Collaborative Learning* (Florence, Italy) vol 1 (IEEE Xplore) pp 511-514

[11] Shilpa Gambhir and Er Sonia Goyal 2014 *Int. J. of Adv. Res. in Electronics and Communication Eng.* **3** 659-663

[12] Fouzi Semchedinel, Louiza Bouallouche-Medjkoune, Leila Bennacer, Naim Aber and Djamil Aissani 2012 *Wireless Personal Communications* **67** 105-112

[13] Shubhinder Kaur and Gurpreet Kaur 2015 *Int. J. of Comp. Applications* **121** 45-47

[14] Rafiqul Z Khan and Md F Ali 2014 *Int. J. of Information Technol. and Comp. Sci.* **6** 65-71

[15] Popov V N, Botygin I A and Koshelev N V 2016 *Key Eng. Mat.* **685** 925-929

[16] Botygin I A, Popov V N, Tartakovsky V A and Sherstnev V S 2015 *Proc. SPIE 21st Int. Symp. Atmospheric and Ocean Optics: Atmospheric Physics* (Tomsk) vol 9680 (SPIE Digital Library) pp 1-4