The problem of choosing the optimal performance in the multiprocessor computer system design

S N Efimov¹,², V A Terskov¹,², I YU Sakash³, V V Molokov⁴ and D L Nikiforov⁴

¹ Siberian State University of Science and Technology named after M. F. Reshetnev, 31, Krasnoyarsky Rabochy Av., Krasnoyarsk, 660014, Russia
² Siberian Federal University, 79, Svobodny Av., Krasnoyarsk, 660041, Russia
³ Krasnoyarsk State Agrarian University, 90, Mira Av., Krasnoyarsk, 660049, Russia
⁴ Siberian Law Institute of the Ministry of Internal Affairs of Russia, 20, Rokossovskogo Str., Krasnoyarsk, 660130, Russia
⁵ E-mail: efimov@bk.ru

Abstract. The article presents the formulation of the problem of optimizing the structure of multiprocessor computing systems designed to solve control problems in real time. The features of this problem, which influence the choice of optimization methods, have been studied. It is concluded that this problem can be effectively solved using evolutionary methods of optimization. The considered models for finding performance can be used to optimize the architecture of multiprocessor computing systems. Besides, it should be taken into account that the resources allocated for the development and operation of computing systems are always limited. Therefore, it is advisable to consider the problem of optimizing the structure of a computing system as multi-criterial one: one criterion is the performance, and the other one is the cost of development and operation of the system. Acquired results can be used in development of multiprocessor computing systems for real-time systems, which is going to reduce the cost of development and operation of these control systems.

1. Introduction

Many modern control systems are real-time systems, where performance is crucial: a control action must be provided in the required time, otherwise it becomes useless. This class of control systems includes those which used in, e.g., rocket and space industry, air traffic control or combat operation control, etc. [1].

Performance requirements for computing systems used in real-time control systems are constantly increasing due to the complexity increase in plants.

Computers’ speed increase has traditionally been going in two ways: processors clock frequency increase and the development of multiprocessor systems. It can be stated that today the possibilities of increasing the clock frequency have been exhausted due to physical limitations [2]. This means that real-time control systems are going to be inevitably based on multiprocessor computing systems (MCS).

MCS performance is usually assessed by solving test problems. To solve a problem of choosing the optimal MCS performance during design it is proposed to assess this performance based on analytical dependencies which consider impact of main features of both the problems and the MCS [3,4].
2. Performance model and formulation of the optimization problem

A computing system is considered as a closed queuing system (QS) with a limited queue. The studied MCS consists of N types of processors containing \( m_i \) \((i=1,2,...,N)\) processors of each time with the average execution time of a single instruction \( T_{0i} \). The processors are connected with RAM via a single bus. The service time of a request from a processor of \( i^{th} \) type is \( \tau_i \). It is assumed that the time interval between two adjacent requests obeys Poisson distribution law with the parameter \( \nu_i \). The time interval between two adjacent services obeys exponential distribution law with the parameter \( \mu_i \). Bus load intensity \( \rho_i = \nu_i / \mu_i \). At any time, the system is in one of the states \( j_1, j_2, ..., j_N \). At the same time, the system contains \( j_i \) requests from processors of \( i^{th} \) type, where \( j_i \leq m_i, i=1,N \), the bus is busy with servicing, \( \sum_{i=1}^{N} j_i - 1 \) requests are in queues for servicing.

Composing the system of Kolmogorov-Chapman equations [5,6] according to the general rules for queuing systems, we obtain a system of linear differential equations for the probabilities of states in which the system can be.

If the bus load intensity \( \rho_i < 0 \), then there is a stationary mode. Equating the derivatives in this system to zero, we obtain a system of linear algebraic equations for the states probabilities in the stationary mode.

Solving the system of equations with one of the numerical methods of linear algebra [7], we obtain an analytical expression for the states probabilities of the studied QS, which can be used to determine the performance of the MCS:

\[
P_{h_1,h_2,...,h_N} = \frac{\prod_{i=1}^{N} \rho_i^{h_i}}{\sum_{h_1=0}^{m_1} \sum_{h_2=0}^{m_2} \cdots \sum_{h_N=0}^{m_N} \prod_{i=1}^{N} \rho_i^{h_i}}.
\]

The MCS performance depends on the total number of requests in the queue for servicing, which is characterized by the value of the average queue length \( L_{avg} \), and on the coefficients of relative losses of performance for processors of each type \( \theta_i \) [8].

The total number of requests in the queue can be determined using the concept of a set of stationary probabilities \( P_L \):

\[
P_L = \sum_{j_1=0}^{m_1} \sum_{j_2=0}^{m_2} \cdots \sum_{j_N=0}^{m_N} P_{h_1,h_2,...,h_N}, \text{ где } \sum_{i=1}^{N} j_i = L;
\]

\[
L_{qp} = \sum_{L=1}^{M} P_L \cdot L, \text{ где } M = \sum_{i=1}^{N} m_i;
\]

\[
\theta_i = 1 + \frac{L_{qp} \tau_i}{T_{0i} + \tau_i}.
\]

Then the MCS performance is calculated using:

\[
\Pi = \sum_{i=1}^{N} \frac{m_i}{\theta_i}.
\]

Obviously, by increasing the number of redundant hardware components, the performance of a multiprocessor computing system can be brought to any given level [9]. However, such systems can be too expensive in development and/or operation. That’s why the use of performance models should also
take into account the cost of the resulting systems. The cost of hardware production can be viewed as the sum of the costs of the components.

Built models make it possible to proceed to the formalization of the problem of choosing the optimal architecture options for multiprocessor computing systems. At the same time, two groups of criteria are obvious:

- performance criteria that must be maximized (the probability of being in a state when performance is sufficient to produce a control action, etc.);
- cost criteria that must be minimized (cost of a system, operation cost, repair cost, etc.).

At the same time, constraints will be imposed on the variables of the problem, e.g., power consumption, weight, etc. To simplify the problem, the cost criteria can be translated into constraints, since as a rule there are upper bounds for all cost characteristics of the system specified by the customer. Selecting the leader among the performance criteria, we get a single-objective conditional optimization problem with a set of constraints into which the rest of criteria will go. Besides, there is going to be a set of natural constraints (e.g., the number of hardware components is an integer and a positive).

Let’s examine the type of variables in our optimization problem. We are going to assume that the maximum number of processor types $N$, maximum and minimum possible number of processors of each type (for processors $m_i^+$ and $m_i^-$ respectively, $i = 1, \ldots, N$) are given.

Here is a formal notation of the optimization problem for the structure of multiprocessor computing system:

$$R_0(m_1, \ldots, m_N) \rightarrow \text{max},$$

under conditions

$$R_i(m_1, \ldots, m_N) \geq R_i^0, \quad i = 1, \ldots, L_R,$$
$$C_l(m_1, \ldots, m_N) \leq C_l^0, \quad l = 1, \ldots, L_c,$$
$$m_i \leq m_i \leq m_i^*, \quad i = 1, \ldots, N.$$

In this problem, the following notation is used:

$R_0$ – leading performance evaluation criterion;
$R_i$, $i = 1, \ldots, L_R$ – secondary performance evaluation criteria;
$C_l$, $l = 1, \ldots, L_c$ – cost evaluation criteria;
$R_i^0, C_l^0$ – maximum acceptable criteria levels translated into constraints.

When designing an optimal MCS structure, one cannot focus on the maximum performance of special processors, but it is necessary to choose the structure in such a way as to ensure the maximum performance of the entire MCS as a whole. For the formal problem statement, it means that the values of the average execution time of a single instruction $T_{0i}$ by processors of the $i^{th}$ type cannot be constant, but have to be included in the number of optimization variables. Moreover, system parameters $v_i$ and $\mu_i$ become functions of $T_{0i}$, that is $v_i = v(T_{0i})$, $\mu_i = \mu(T_{0i})$. This leads to a significant complication of the optimization problem, turning it into a two-tier hierarchical problem:

$$(R_0^*(T_{01}, \ldots, T_{0i}, T_{0N}), R_i^*(T_{01}, \ldots, T_{0i}, T_{0N}), C_l^*(T_{01}, \ldots, T_{0i}, T_{0N}) \rightarrow \text{extr},$$

where $R_0^*$, $R_i^*$ and $C_l^*$ – optimization problem solution.

First of all, it should be noted that the space of possible solutions is a discrete, since the MCS configuration is defined by the number of processors of various types, which can only be integers. At the same time, the power of the search space grows rapidly with the increase in the number of processor types.

If we roughly estimate the power of the optimization space, then we get the total number of possible configurations more than $1.6 \cdot 10^{20}$. At the same time, problem constraints are not going to significantly reduce the number of search points.
A significant problem for solving this optimization problem is the method of calculating the object functions (criteria), which are mostly set algorithmically.

There are all signs of a complex optimization problem: algorithmically set functions, different type of problem variables, a variable number of sought variables, a large search area for an optimal solution.

When solving such optimization problems, evolutionary optimization algorithms have proven themselves well [10-16]. Therefore, the study of evolutionary algorithms effectiveness in optimizing the structure of multiprocessor computing systems can be indicated as a possible direction for further research.

3. Conclusion
The considered models for finding performance can be used to optimize the architecture of multiprocessor computing systems. At the same time, one must take into account that the resources allocated for development and operation of MCS are always limited. Therefore, it is advisable to consider the problem of optimizing the structure of a computing system as multi-criterial one: one criterion is the performance, and the other one is the cost of development and operation of the system.

Thus, this article presents the formulation of the problem of optimizing the structure of multiprocessor computing systems. In the future it is proposed to study the effectiveness of using evolutionary optimization methods to solve this problem.

Acquired results can be used in development of multiprocessor computing systems for real-time systems, which is going to reduce the cost of development and operation of these control systems.

References
[1] Vasiliev V A, Legkov K E and Levko I V 2016 Real-time systems and their application areas Information and Space 3 68-70
[2] Sutter H 2005 The free lunch is over: A fundamental turn toward concurrency in software Dr. Dobb’s Journal 30(3) http://www.gotw.ca/publications/concurrency-ddj.htm
[3] Buttazzo G 2011 Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications (New York: Springer)
[4] Efimov S N, Terskov V A and Serikova O YU 2021 Algorithm for finding the probability of functioning with a targeted performance of the hardware and software complex of real time systems Bulletin of the Voronezh Institute of the Ministry of Internal Affairs of Russia 1 72-81
[5] Wentzel A D 1996 Course of the theory of random processes (Moscow: Nauka)
[6] Ross S M 2014 Introduction to Probability Models (Los Angeles, CA: University of Southern California) 187
[7] Bakhvalov N S, Zhidkov N P and Kobelkov G M 2004 Numerical methods (Moscow: BINOM)
[8] Terskov V A and Efimov S N 2003 Analytical method for evaluating performance indicators for parallel computing systems Information technologies 7 13-20
[9] Efimov S N, Tyapkin V N, Dmitriev D D and Terskov V A 2016 Methods of Assessing the Characteristics of the Multiprocessor Computer System Adaptation Unit Journal of Siberian Federal University. Mathematics & Physics 9(3) 288-95
[10] Tarhov D A and Radchenko D S 2015 Distributed optimization algorithms Modern information technology and IT education 11 404-8
[11] Coello C A C 2017 Constraint-handling techniques used with evolutionary algorithms Proc. of the Genetic and Evolutionary Computation Conference Companion 675-701
[12] Semenkina O E, Popov E A and Semenkin E S 2021 Cooperative self-configuring nature-inspired algorithm for a scheduling problem IOP Conf. Ser.: Mater. Sci. Eng. 12080
[13] Goldberg D E 1989 Genetic algorithms in search, optimization, and machine learning (Reading, MA: Addison-Wesley Professional)
[14] Polyakova A S, Lipinskij L V and Semenkin E S 2020 An automated system for forming the composition of a team using a multicriteria genetic algorithm Pat. of the Russian Federation No 2020663770
[15] Lytkina E M, Efimov S N, Terskov V A and Yarkov K V 2021 Method of taking into account constraints when optimizing the architecture of the hardware-software complex of the real-time control system Transport Science and Technology 1 63-7

[16] Semenkin E S and Sopov E A 2005 Probabilities-based evolutionary algorithms of complex systems optimization Proc. Int. Conf. Intelligent Systems and Intelligent CAD 1 (Moscow: Fizmatlit) 77-8