Software complex for realization of mathematical models, methods and algorithms of estimation of time of execution of complex problems in multiprocessor computer systems

Yuriy Klushyn

Computer Engineering Department, Lviv Polytechnic National University, UKRAINE, Lviv, S. Bandery street 28a, E-mail: vladina_t@ukr.net

Abstract – To solve the problem of forecasting, a complete software package was developed based on mathematical models, methods and algorithms of direct stochastic modeling and multilevel stochastic modeling, which are used to estimate the execution time of complex software systems in multiprocessor computer systems.

Keywords: parallel computing systems, complex of interconnected works, direct stochastic modeling, Markov process, function of distribution of random variable.

I. Introduction

At present, the efficiency of using computers and, in particular, parallel computing systems (CS), is estimated in some cases not so much by traditional performance parameters (speed of various operations, their mixtures, typical computing procedures), as the execution time of specific tasks or their sets.

Such an approach is important for evaluating computing systems that operate in control loops, where the main criterion for the quality of a CS is its ability to solve a problem in a certain time. The study of the effectiveness of parallel computing systems for responsible applications of this kind is based on an analysis of the structure of the interrelationships of parallel-sequential tasks (fragments) of a given set [1 - 3] and parameters of its parallelism [4].

In connection with the use of parallel computing systems in real-time control systems, the problem of a priori assessment of the "suitability" of such aircraft for solving a specific set of tasks, which is set by the user, for the required time becomes obvious.

With regard to parallel computing systems , this problem is called the prediction of the execution time of complex software systems; the latter are usually defined by graph models and are considered as complexes of interrelated works (CIW) - tasks and / or their parallel-sequential fragments (subtasks, processes, program modules).

The development of accurate mathematical models and algorithms for analyzing the functioning of parallel computing systems on CIW, which the user sets, with a random execution time of each work (process) would solve the actual problem of reliable analytical evaluation of the execution time of each specific CIW on the computing system.

II. Block - diagram of the software package

The program complex consists of five modules:

- module for setting the structure of the graph;
- module for determining the states of the breaking Markov process (BMP);
- module of transformation of the matrix Q to a block triangular form;
- module for calculating the average time;
- module for calculating the distribution function.

In Fig. 1 shows a block diagram of the links of all the modules listed.

III. The module sets the structure of the graph CWR

This module is used to enter the initial data into the program and consists of two submodules - A and B.

The submodule A is used to enter the initial data using the algorithm described in [5,7]. As the initial data are used:

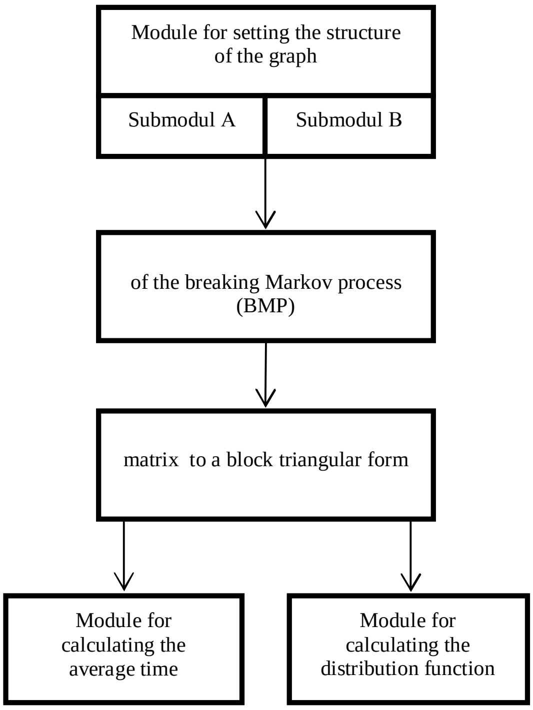

Fig. 1. Block - diagram of the software package.

- connection table of the vertices of the graph, which describes the graph of a given CIW)
- parameters characterizing the given CIW (see below);
- parallel parity parameter - the number *k* of servicing devices (processors KS);
- parameter cr, which specifies the dispatching criterion.

Parameters characterizing a given CIW are:

- *N*- the total number of all jobs specified by the CIW;
- μ_j - intensity of service for each job;
- b_j is the degree of connectivity of each work of this CIW, which determines the number of successors
- r_j - the rank of work in the given CIW.

The parameter *cr*, which specifies the scheduling criterion, can take the following values:

- 0 - dispatching according to the rank criterion: "r_j";
- 1 - dispatching according to the criterion of the connection level of the vertices "r_j / b_j";
- 2 - dispatching according to the criterion "the choice of work with the smallest number";
- 3 - dispatching according to the criterion "choice of work with the largest number".

Submodule B serves to set the source data using algorithms for buoyant stochastic modeling [5]. In this submodule, the method of uniform distribution of works of a given CIW in terms of tiers [7] is also used. The source data is the connection table [7] and the above parameters (N, μ_j, b_j, r_j).

IV. Module of determining the states of the breaking Markov process (BMP) when performing a specific CIW

The module implements the algorithms described in Section 2, as well as in [5, 7].
Sets the initial state of the system.
State of the system:
Pool work = N-1
Work buffer = 0
Operating device operation = 1

Number of States = 1
Current state = 1

The module uses three working arrays:

- array *Q (s, s)*, which determines the matrix *Q* of the transition intensity of the considered BMP;
- arrays *X1 () X2 ()*, elements of which is the state of the system xp (m, iw, jn)

It should be noted that in direct stochastic modeling, the dimension of the array *Q (s, s)* can, in principle, be estimated using Eg. (1).

$$S^* = 2 + \sum_{l=1}^{k} C_{N-2}^{l}\,, \qquad (1)$$

where *k* is the number of servicing devices, $k = \overline{1, N-2}$; $C_{N-2}^{l} = \dfrac{(N-2)!}{(N-2-l)!\,l!}$.

This is advisable for CIW with a small amount of work, when the number of states of weapons of mass destruction is deliberately small. For large dimensions of the model, there is a sharp increase in the amount of computation required for processing the matrix *Q*, and hence an increase in the cost of computer memory. Therefore, to reduce the amount of computer memory used, it is desirable to determine the actual dimension of the array *Q (s, s)*, which is one of the results of the implementation of the algorithms for determining the states of the BMP in this software module. An additional opportunity to save computer memory when implementing the proposed methods is a special numbering and state order in the process of constructing the matrix *Q (s, s)* [5,7].

V. Module of transformation of the matrix to a block triangular form

This program module not only transforms the matrix *Q* of the intensities of transitions to a block triangular form, but also forms this matrix in a compressed form, which simplifies and speeds up the implementation of algorithms for obtaining numerical characteristics of the system under study.

To bring the matrix to a triangular shape, it is necessary and sufficient to order the elements of the state vector *X*.

To reduce the amount of used memory and accelerate the calculation, a special representation of the lower triangular matrix *Q* of dimension *n * n* is used in the form of two arrays:

- array *wq (m, p)* values of zero elements in the row of the matrix *Q*,
- array *iq (m, n)* of numbers of nonzero elements in the row of the matrix *Q*.

Since in the matrix *Q* all diagonal elements are nonzero, in the array *iq (m, n)* the first column is used to indicate the type of row in which the nonzero elements described in it are located. The module also uses an array *num (p)* of the number of nonzero elements in rows of various types and an array *vec (n)* of the probability vector of the initial states of a system of dimension *s*.

VI. Module for calculating the average value of CIW execution time

With the help of this module, the average value of CIW *T* execution time is calculated on homogeneous resources parallel computing systems. In this case, the module uses one working array *T(s, 4)*, in which the current value *T* is remembered.

VII. Module for calculating the distribution function

With the help of this module, the distribution function of a random variable *T* is calculated - the execution time of the complex of interrelated works of a parallel KS. The algorithm for finding the distribution function of a random variable is described in [6].

VIII. Conclusions

Based on the analysis of mathematical models, methods and algorithms of direct stochastic modeling and belt-based stochastic modeling, as well as studies, a software package designed to solve the problem of predicting the execution time of complex branched tasks on parallel computing systems in full. The given software package calculates the average value and the distribution function of the execution time of a set of interrelated tasks on homogeneous resources of a parallel computing system, and consists of five software modules, which are interconnected. The interconnection of the modules is shown in Fig.1.

Its characteristics on the test example:

- the total amount of computer memory used for processing the transition intensity matrix, corresponds to the implementation of a set of interrelated works on N <100 jobs - about 260 kB;
- the construction of a matrix of transition intensities on a computer of average power is about 7 minutes;
- when calculating the model as a whole - about 16 minutes.

References

[1] W.W. Chu, K.K. Leung, “Module replication and assignment for real-time distributedprocessing system”. // "Proc IEEE". 1987. 75. N5. pp. 547-562.

[2] A.S. Khritankov, “Mathematical model of performance characteristics of distributed computing systems”. Computer science, management, economics. WORKS OF MIPT. - 2010. - Volume 2, No. 1 (5), p. 110-115.

[3] A.N. Ivutin ., E.V. Larkin, “Prediction of the execution time of the algorithm”. Magazine. News of TSU. Technical science. Issue number 3/2013 C 301-315.

[4] P.L. Bocharov, V.V. Ignatushchenko, “Mathematical models and methods for evaluating the effectiveness of parallel computing systems on complexes of interrelated works” // Tez. report international conf, “High-Performance Computing Systems in Management and Scientific Research,” Alma-Ata, 1991, p. 6.

[5] Y.S. Klushin, “ Improving the accuracy of estimating the execution time of folding software systems in multiprocessor computer systems for belt stochastic modeling”. Bulletin of NU "Lviv Polytechnic" №881. Computer systems and networks. - Lviv: NU “LP”, 2017.

[6] Y.V. Preidunov, “Development of mathematical models and methods for predicting the implementation of complex software systems on parallel computing systems”. Cand. course work. M .: Inst. Of Problems of Management RAS, 1992.

[7] Y.S Klushin, “Reducing the number of states of the Markov process when executing complex software systems on parallel computers”. Scientific Bulletin of Chernivtsi University. Computer systems and components. 2016. T. 7. Vol. 2, pp. 53-62.