ISSN 1512-1232

On Multicriteria Algorithm for Specific Problem of Scheduling Theory

Liana Lortkipanidze[1], Nana Odishelidze[1], Ketevan Kuthashvili[2], Liana Karalashvili[2]

[1]Computer Science Department, Faculty of Exact and Natural Sciences, Iv. Javakhishvili Tbilisi State University, 0143, Georgia
[2] School of Information Technologies, Engineering and Mathematics, The University of Georgia, 0171, Georgia

Abstract

One of the areas of discrete optimization problem - the scheduling theory is considered. As it is known, the problems of scheduling theory are of NP difficulty and only in the certain cases it has been managed to construct the algorithm of polynomial difficulty. In the paper it is considered the problem for which the set of additional resources and partially ordered set are empty. Under such conditions the effective algorithm is constructed to order the sequence of tasks. The schedule length and maximal price of tasks' implementation are considered as the measure of the algorithm effectiveness. The constructed algorithm takes into account the construction of tasks implementation schedule. It is possible to construct such schedule, which gives Pareto-optimal solution for both criteria.

Keywords: *scheduling theory, multicriteria optimization, Pareto-optimal solution.*

1. Introduction

Many practical problems, for instance, transport or management and running of industry process, under conditions of fixed resources require scheduling of tasks at a time. The given system of tasks must be implemented by certain set of resources or by means /devices of services.

In terms of tasks system and the given properties of resources with certain restrictions to them we have to construct an efficient algorithm of the task implementation sequence, which gives possibility to attain efficiency by certain measure of optimum. Under measure of optimum there may be considered scheduling length in terms of time, average time of being in the tasks system or maximum cost of the system.

As it is known, schedule such problems is of NP difficulty [5,37] and requires great deal of applications of the modern applied mathematics.

Basically difficulty is caused by great volume of tasks. In such situations, to receive best decisions, new methods are creating and practical recommendations of planning and control are producing. [13,30,39], [5].

Because of above-mentioned, for the certain problem it is actual to construct comparatively accurate mathematical model and create such algorithms, which totally will use the specific character of the problem and give possibility of the optimal decision in polynomial time.

In this paper, on the basis of schedule theory common methods, the mathematical model and algorithm are constructed, for which task implementation is possible by single-step multiprocessor system, where processors are mutually half-interchangeable, but set of additional resources and partially ordered set are empty.

ISSN 1512-1232

2. Setting of a problem

Let there is given a set X={ξ_1, ξ_2, . . . , ξ_n} of ***n*** tasks, which must be fulfilled during [0, T] time. These tasks must be implemented by means of ***m*** processors P={p_1,p_2, . . . , p_m}. Each ***j***-th task enters the system at $t^0_{\xi_i} \in [0, T]$ moment and its implementation is possible by any processor from the subset of processors $Q_i \subset P$. L_i is an ***i***-th row of *L* matrix. Matrix ***L*** is composed in the following way: if it is possible to implement any ***i***-th task by means of p_j processor, then its q_{ij} (i=1,2, . . . n, j=1,2, . . .,m) element equals 1, otherwise it equals 0.

Furthermore, price matrix Ω is given. Its elements $\mathbf{w}_{ij}$ (i=1,2, . . . n, j=1,2, . . .,m) express the implementation price of ***i***-th task on ***j***-th processor and, besides, there are given quantities d_i (i=1,2, . . . n), which show the directive time of implementation of each task (the ***i***-th task implementation must start not later than d_i moment). If it is needed to construct continuous scheduling, then directive terms may not be taken into account, and in this case these terms will be equal to zero.

τ matrix elements τ_{ij} express the needed term (duration) to implement ***i***-th task on the ***j***-th processor. Besides, there are given $t^0_{p_j}$ quantities (j=1,2, . . . ,m), in which by the initial moment of [0, T] period ***j***-th term from the previous started tasks shows a release time moment of the ***j***-th processor.

It is necessary to compose a schedule ***S*** of the task implementation sequence, i.e. to find such t_{ξ_i} quantities and sort out l_{ij} processors, which *i-th* element (*i* = 1,2, . . . n) shows ξ_i –th $q_{ij} \in Q_i -$ on ***j***-th processor initial moment of implementation $(j = 1,2,\dots,m)$ and the following condition will be satisfied.

$\rho_1(S^*) = \min_S \rho_1(S)$

or

$\rho_2(S^*) = \min_S \rho_2(S)$

3. Construction of algorithm

Procedure consists of ***n*** iterations. For each ***k***-th iteration selection of t_{ξ_i}quantities, ξ^*_k task and its corresponding processor, on which it is possible to start implementation of the task at t_{ξ_i} moment, takes place. Selection of these quantities originates the following way:

$$\tilde{t}^{(k)}_{\xi_i} = \min_{j \in Q_i} t^0_{p_j}, \quad i=1,2,\dots,n,$$

$$t^{(k)}_{\xi_i} = \max(d_i, \min_{j \in Q_i} t^0_{p_j}). \quad i=1,2,\dots,n. \qquad (1)$$

Under condition (1) for all tasks, entered to the system during [0, T] time interval, possible start moments will be evaluated and also those processors will be selected, which are ready to implement the mentioned tasks. To select the task, which must be implemented first, let us evaluate $(t^{(k)}_{\xi_i} + \tau_i)$ quantities (*i* = 1,2, . . . n) and select the minimal one among them. Let this minimal meaning is $\eta = \min_{\xi_i \in X^k_1} (t^{(k)}_{\xi_i} + \tau_i)$. Let for each ***k***-th iteration selected ξ_i be $\xi^*_{(k)}$, and the number of its corresponding processor, which is selected by formula (1) and for which $t^{(k)}_{\xi_i}$ is maximal, be $j^*_{(k)}$.

For each ***k***-th iteration let us compose a set $X^k_1 \subset X$, which elements are selected by the following way: if any $\xi_i \in X^k_1$, then the following conditions must be satisfied

$$t^{(k)}_{\xi_i} < \eta,$$

$$j^{*(k)} \in Q_i \qquad (2)$$

ISSN 1512-1232

$$\tilde{t}^{(k)}_{\xi_i} - t^{(k)}_{\xi_i} \geq 0 .$$

This is a set of those tasks, which are ready to start implementation by that moment before $\xi^*_{(k)}$ task is finished. We shall call X_1^k set as the set of alternative tasks. Let us evaluate meanings of functions ρ_1 and ρ_2 for the task $\xi^*_{(k)}$. Furthermore, it is possible that ω_i and τ_i $(i=1,2,...,n)$ quantities by themselves are dependent functions of any parameter. In this algorithm they are piece-wise constant functions.

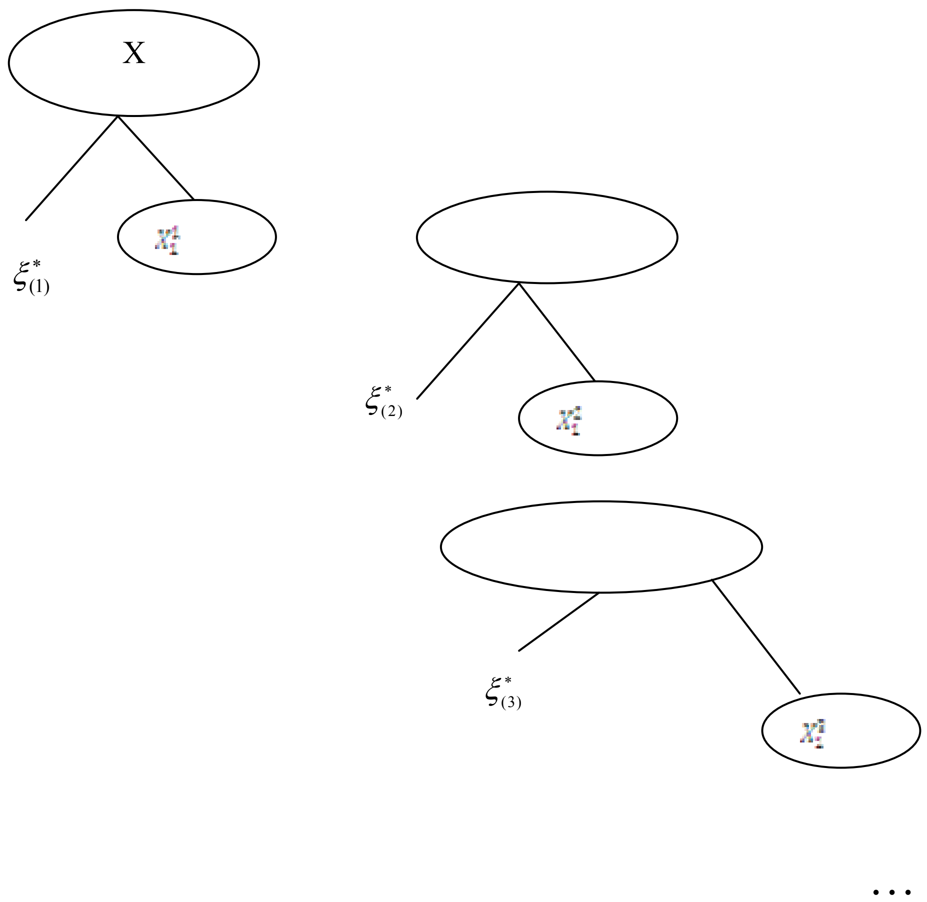
After first iteration let us exclude $X \setminus \{\xi^*_{(1)}\}$ element, change $t^{(2)}_{P_{(j_1^*)}} = t^{(1)}_{\xi^*_{(1)}} + \tau_{i_1}$ quantity and construct $\tilde{t}$ vector according formulas (2). For the second iteration we will select two tasks $\xi^*_{(1)}$ and $\xi^*_{(2)}$ and will evaluate

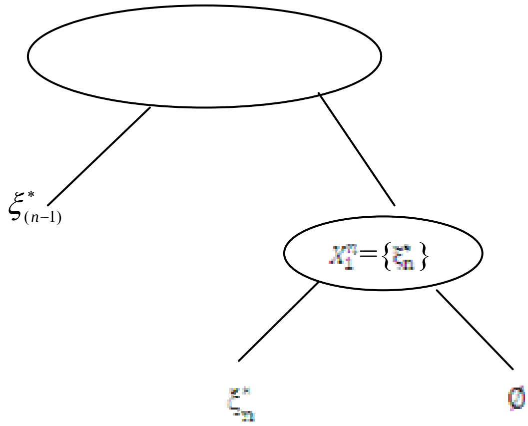
$$\rho_1 = \max(\rho_1 \ , \ t^{(2)}_{\xi(2)} + \tau_{i_2}),$$
$$\rho_2 = \rho_2 + (t^{(2)}_{\xi(2)} + \tau_{i_2} - d_{i_2}) * \omega_{i_2} .$$

For the last iteration all tasks will be distributed and furthermore for each step of iteration X_1^k sets are constructed.

Then the backward phase starts.

This process schematically may be expressed in the following way:





When all tasks are distributed, meanings of the functions (1) will be evaluated and the first version of scheduling will be constructed. After that sorting and, respectively, correction of scheduling will take place.

During backward phase we pass to the set X_1^{n-1} and, if it is not empty, we will consider all possible permutations. For each permutation function meanings (1) will be evaluated and, if in the certain permutation case the received result is better than previous one, then function meanings (1) will be changed and corresponding version will be saved. After that we pass to the set X_1^{n-2} and implement the same procedure. In the case, when the set happens empty, we pass directly to the set X_1^{n-2}. By this way we go on the backward phase until attain X_1^1 set. When all elements sorting of the set X_1^1 happens, the backward phase is over, the optimal scheduling is constructed and meanings of functions (1) are evaluated.

Assume that in terms of (1') criterion the optimal scheduling is S_1^* and optimal meaning of its corresponding functional is $\rho_1(S_1^*)$, but in terms of (1'') criterion the optimal scheduling is S_2^* and optimal meaning of its corresponding functional is $\rho_2(S_2^*)$.

After the corresponding scheduling of both criteria is constructed we will pass to search Pareto optimal solution.

Let us construct two functional:

$$\rho_3(S^*) = \min_S \sqrt{\sum_{i=1}^{2}\left(\frac{\rho_i(S)-\rho_i(S^*)}{\rho_i(S^*)}\right)^2}$$

$$\rho_4(S^*) = \min_i \max_S \left|\frac{\rho_i(S)-\rho_i(S^*)}{\rho_i(S^*)}\right|,\ i = 1, 2.$$

For ρ_3 functional we may use the same procedure by means of which we were looking for the optimal solution for ρ_1 functional, but in the case of ρ_4 functional we may use the same procedure by means of which we were looking for the optimal solution for ρ_2 functional.

Dislike the direct sorting method this algorithm implements sorting in $X_1^k \subset X \setminus \{\xi_k^*\}$ sets, which (as it is shown from the given example) contains much less terms than $X \setminus \{\xi_k^*\}$ set and very often is empty. Therefore number of steps decreases essentially.

Besides, in the case of one criterion $X_1^k \subset X \setminus \{\xi_k^*\}$ constructed sets during direct phase will be used in the case of other criterion, since they actually determine admissible range and, furthermore, we use just these sets, when we implement sorting in the case of searching Pareto optimal solution.

Though, since we have to save these sets on each step on the backward phase, the volume of memory is increasing. But saving of these sets essentially reduces the calculation time. That is why it is necessary in terms of speed.

ISSN 1512-1232

4. Conclusions

For the certain problem of schedule theory two algorithms with different criteria are constructed. Then there is considered a problem which implies to find Pareto optimal solution. The importance of this algorithm means that to find Pareto optimal solution does not require to construct a new algorithm, (by which the optimal solution is found in the case of one criterium) it is possible to solve the problem with two criteria.

References

1. V. Tanaev, V. Gordon, I. Shafranskiĭ Scheduling Theory. Single-Stage Systems, Dordrecht Boston : Kluwer Academic Publishers, 1994.
2. Michael L. Pinedo. Scheduling: Theory, Algorithms, and Systems. Springer. New York ,USA 2012.
3. By Vincent T'Kindt, Jean-Charles Billaut Multicriteria Scheduling: Theory, Models and Algorithms. Springer-Verlag Berlin Heidelberg 2006.
4. K.Kutkhashvili Algorithm for Multicriterion Problem of the scheduling theory. Georgian Academy of Sciences A. Eliashvili Institute of Control Systems Proceedings, Tbilisi, № 9, 2005.
5. Kutkhashvili K., Gabisonia V. The algorithm for a problem of the scheduling theory considering two criterion. Georgian Academy of Sciences A. Eliashvili Institute of Control Systems Proceedings, Tbilisi, №13. 2009.

Article received: 2014-09-09