Scheduling Theory Problem with an Unfixed Start Time

Ketevan Kutkhashvili The
University of Georgia
Tbilisi, Georgia
e-mail: Kkutkhashvili@yahoo.com

Neli Kilasonia
GTU Institute of Control Systems
Tbilisi, Georgia
e-mail: Kilasonianeli1@gmail.com

https://doi.org/10.62343/csit.2024.10

***Abstract* — The paper discusses the problem when the achievement of the optimal goal depends on the sequence of tasks, i.e., when it is necessary to arrange them in time. Such problems are known to belong to calendar scheduling problems and are in general of NP complexity. It is possible to build a polynomial algorithm only in the case of a certain configuration of the initial parameters. In the paper, one such configuration is considered and financial cost minimization is taken as a criterion for achieving the goal.**

***Keywords* — discret optimization, scheduling problem, algorithm.**

I. INTRODUCTION

Solving various technical, economic, organizational and social problems is often impossible without building a mathematical model. Especially in the case, when a very large amount of data needs to be processed to achieve some goal, and this data is of a discrete nature, it becomes necessary to use discrete optimization methods.

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.

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 problems are of NP difficulty [1]-[5] and requires great deal of applications of the modern applied mathematics. Basically difficulty is caused by great volume of tasks. In such situations, new methods are created to receive best decisions and practical recommendations of planning and control [2]-[5].

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

For this problem, an algorithm of complexity P is built, which is based on a combined method consisting of the branch and bound and statistical methods.

II. PROBLEM FORMULATION

Given a set of tasks, X={ξ_1,...,ξ_n}, which must be executed in [0, T] period by means of P={P_1,...,P_m}, j=1,..., m processors. Obviously, the problem is relevant when n is a much larger number than m. Processors are partially interchangeable. They can differ both in terms of speed and functionality. Therefore, the matrix [τij]i=1,...,m, j=1,...,n , whose τij element shows the duration of execution of the task ξj on the Pi processor is given in advance.

Also is known {ω ij } i=1,...,m, j=1,...,n matrix, whose ωij element shows the price of execution of task ξj on Pi processor. In order to simplify the problem, let's assume that the tasks are mutually independent and additional resources are not taken into account.

The arrival time of each task in the system is not known exactly, but the estimated time is given in the form of an interval $t_i^0 \in [a_i; b_i]$. where a_i is the estimated minimum time for the i-th task to enter the system, and b_i is the estimated maximum time for the i-th task to enter the system.

The schedule S must be constructed in such a way that one task can be executed on one processor without interrupts, and one processor cannot execute several tasks at the same time. Our goal is to assign to each task a number t0ξj (Pi), which at the time of the start of processing of the j-th task indicates the Pi processor on which the j-th task should be executed. Among all such mappings S that satisfy the above conditions, one needs to find a mapping S* for which the following conditions are satisfied

$$\rho(S^*) = \min_S \rho(S) = \min_S \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{m} \omega_{ij} f_{ij}(s),$$

where $f_{ij}(s)$ is the processing time of the j-th task according to schedule S shows on the Pi processor, and ρ expresses the cost of the task system.

III. ALGORITHM DESCRIPTION

The algorithm includes the following main blocks:

- Determination of Xk system and formation of τ_i^k , $t_i^k \in [a_i; b_i]$ and $\omega_{ij}(Z)$ for the components included in it;
- Determination of the exact time of entry into the system for each task $t_i^0 \in [a_i; b_i]$ or $t_i^0 = a_i$;
- Branching rule - it defines the branching strategy and process of the search tree. Its objective is to divide the solution space into disjoint subsets, each of which is either cut or formed for consideration at a later stage, $X_1^{(k)} \subset X$;

- Selection rule - selects the tree vertex from which the next branch should start. For this, we choose $\eta = \min_{\xi_i \in X_1^k}(t_{\xi_i}^{(k)} + \tau_i)$;
- Calculation of the value of the characteristic function and expert evaluation - based on this calculated value, the branch of the tree is evaluated and those branches that do not contain the desired solution are cut. $\rho_k = \rho_k + (t_{\xi_{(k)}}^{(k)} + \tau_{i_l} - d_{i_l}) * \omega_{i_l}$;
- Forming a new step. mint $\min t_{P_{(j_1^*)}}^{(2)} = mint_{\xi_{(1)}^*}^{(1)} + \tau_{i_1}$, $X_1^{(k+1)} \subset X$. A transition to the second step is made if the endpoint of the interval [0, T] is not reached. Otherwise, we go to the next stage;
- Construct and compute a lower bound function - which matches each private solution to its lower bound value. $\rho_l(S_k) \leq \rho_l(S_{k+1})$;
- Checking and correction of the upper limit of the value - the upper limit of the value is initially equal to any of the full prices, which are known in advance by approximation or reasonable judgment. And if there is no such opinion in advance, then it is equal to mechanical infinity;
- Establishing a schedule for the case when $t_i^0 = b_i$;
- Selecting a common distribution from both trees and determining the best starting time for tasks whose arrival time in the system is uncertain;
- Checking the end of the algorithm operation - the operation of the algorithm is completed if all subsets of the set X_k turn out to be empty.

Due to the fact that the influence of the abundance of additional resources is neglected during scheduling, and also that the tasks are mutually independent, it is possible to construct an algorithm of O(n3) complexity. In addition, it is possible to include an expert in the process of executing the algorithm to specify the time of entry into the system of tasks.

For this problem, an algorithm of complexity P is built, which is based on a combined method consisting of the branch and bound and statistical methods.

REFERENCES

1. V. Tanaev, V. Gordon, I. Shafranski Scheduling Theory. Single-Stage Systems, Dordrecht ; Boston : Kluwer Academic Publishers,1994.
2. P. Chrეtienne, E.G. Coffman, J.Karel Lenstra Scheduling Theory and Its Applications. Wiley; 1st edition, 1995.
3. K. Kutkhashvili., 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.
4. P.Brucker, B.Jurisch, B.Sievers A branch and bound algorithm for the job-shop scheduling problem. Discrete Applied Mathematics. Volume 49, Issues 1–3, 30 March 1994, Pages 107-127
5. F. Werner, L. Burtseva, Y. Sotskov, Algorithms for Scheduling Problems. MDPI, Basel, Switzerland. 2018.