Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Decision Support Systems 38 (2004) 171–182
Decision Support Systems
www.elsevier.com/locate/dsw

Decision-making algorithms in two-level complex operation system

Jerzy Józefczyk*

Institute of Control and Systems Engineering, Wroclaw University of Technology, Janiszewski St. 11/17, 50-370 Wroclaw, Poland

Available online 15 August 2003

Abstract

A two-level manufacturing operation system composed of two interconnected subproblems, i.e. scheduling of independent, non-preemptive tasks on unrelated moving executors as well as motion control of a group of moving executors performing the tasks is investigated. As the performance index of the two-level system, the makespan is assumed. Three heuristic solution algorithms for the two-level system are presented. The first algorithm ensures the current modification of solutions for the scheduling subproblem during the decision procedure of the two-level system. In the second one, an iterative approach is applied, which consists in successive implementation of the solution algorithms for both subproblems. The third algorithm uses on-line procedure, which enables determination of the best solution in the current step of the decision procedure. Comparisons of the algorithms as well as a numerical example are also presented.

Keywords: Scheduling; Simulation; Decision-making systems; Flexible manufacturing systems; Vehicle scheduling

1. Introduction

For complex operation systems composed of operations (tasks, jobs), which are mainly characterized by execution times, different decision-making problems are solved including scheduling and allocation. It is assumed that the execution times for tasks are not given a priori but they are direct or indirect results of solving the decision-making problems, which consist in controlling the execution of tasks under consideration. Taking into account the control of task execution in operation system makes it more complex on the one hand and closer to real-life applications on the other hand. Complex manufacturing systems like flexible manufacturing systems are the main application area for systems under consideration. In such systems, the movement of different elements should be taken into account and, roughly speaking, the movement is the subject of control. The movement of plants to be produced as well as the movement of executors being the performers of technological tasks can be distinguished as the most important cases. Further investigation will be confined to the second case. Then, each task consists of two parts: driving-up of the executor towards the plant to perform the job (transportation task) and the execution of the job to forward the performance of the plant (technological task). Both parts should be controlled. One can easily notice that two decision-making subproblems,

* Tel.: +48-713202979; fax: +48-713203884.
E-mail address: Jerzy.Jozefczyk@pwr.wroc.pl (J. Józefczyk).

0167-9236/03/$ - see front matter
doi:10.1016/S0167-9236(03)00098-8

i.e. solving the selected subproblem for operation system and control of the execution of tasks are interconnected. It is convenient to describe them in the form of a two-level decision-making system. The first subproblem being the more strategic one is placed at the upper level. For such a general and complex problem, particular decision-making subproblems can be generated and investigated as well as different solution algorithms can be derived and applied. The idea of the two-level decision-making system for complex manufacturing operation system has been first introduced in Ref. [2]. It corresponds to the development of the area of complex operation systems, e.g. Refs. [1,11–13].

Now, it is assumed that task scheduling is considered as the upper level decision-making subproblem. Moreover, at the lower level, only the motion of executors is investigated and the control of job performance is not taken into account (Fig. 1). The subproblems from both levels taken separately have been intensively studied and described in detail in numerous papers, e.g. Refs. [2–7,10]. The presentation as well as evaluation of solution algorithms both exact and approximate can be found therein.

The main objective of the paper is to present and compare via computer simulation three approximate decision-making algorithms for a two-level system. The description of algorithms for subproblems is omitted. The selected results of the computer simulation, which verifies the solution algorithms for the two-level systems, are also given. In the next section the problem is stated. Then, in Section 3, heuristic control algorithms are presented. Section 4 comprises results of the computer simulation. Final remarks complete the paper.

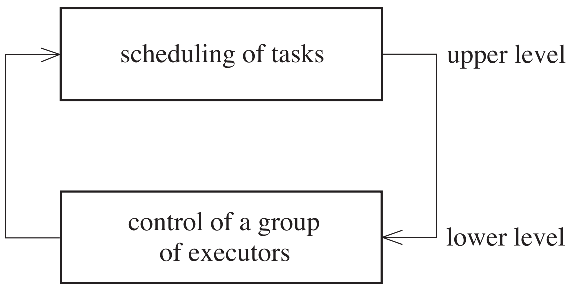

Fig. 1. Two-level complex operation system with scheduling of tasks and motion control of a group of executors.

2. Problem formulation

The decision-making problem for the two-level system is stated as the optimization problem. This formulation is preceded by a short presentation of the subproblems from both levels.

2.1. Upper level subproblem

The movement of executors, which is taken into account, makes the scheduling problems more difficult. It leads to problems of task scheduling with moving executors, which can be treated as a generalization of classical scheduling problems. In order to explain more precisely the idea of task scheduling with moving executors the main notions will now be introduced. The basic notion is the *task*, which can be understood like in the scheduling theory but having its own meaning by reason of the fact that the movement of executors is taken into account. The *executor* is usually a technological device, which performs tasks at a place called workstation located on the plane or in space. From among all workstations, a depot is distinguished, where each executor starts and ends its work, and where no task is performed. The main idea of the generalized scheduling problem under consideration is the following. To perform the task the executor should drive-up to the workstation. Therefore, each task is twofold and consists in driving-up to the workstation and performing of a *job* at this workstation. Generalizing the term 'task' leads to generalization of execution times, which are the main data for every scheduling problem. Then the *execution time* is the sum of the *driving-up time* and the *time the job is performed*. Such a generalization defines a new scheduling problem in which not only a subset of tasks for each executor should be derived but also the routes of executors are to be determined. The necessity of determining the routes results from the fact that the order of performing tasks by an executor has an influence on the quality of scheduling for the whole set of tasks. Versions with moving executors can be formulated and solved for all known classical scheduling problems. In the paper, a simple scheduling problem with independent, non-preemptive tasks and unrelated executors as well as the same ready times is considered. However, it deals with the situation when each executor to perform the job should drive-up to a