Computers & Operations Research 67 (2016) 184–192

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

An efficient new heuristic for the hoist scheduling problem

Adnen El Amraoui [a], Mohsen Elhafsi [b,*]

[a] Université d'Orléans, Laboratoire PRISME, 63 av. de Lattre deTassigny,18020 Bourges, France
[b] School of Business Administration, University of California Riverside, 900 University Ave., Riverside, CA 92521-0203

A R T I C L E I N F O

Available online 19 October 2015

Keywords:
Transportation resources
Dynamic scheduling
Mixed Integer Linear Programming (MILP)
Heuristic
Bounded processing times
Initial products sequence

A B S T R A C T

In this paper, we study the hoist scheduling problem. The latter is often encountered in electroplating processes where a variety of jobs have to be processed in small quantities and in a very short amount of time. Basically, the problem consists in scheduling the hoist's movements in order to achieve two main objectives: Higher productivity and better product quality. In order to achieve these two goals, we first formulate the problem as a Mixed Integer Linear Programming Model. Then, due to the problem complexity, we develop an efficient heuristic procedure to obtain the hoist's job processing sequence. Extensive numerical experiments show that the heuristic performs extremely well compared to a lower bound obtained through the mixed linear programming model and gives the optimal makespan for a large number of problem instances. Furthermore, comparison with the best available heuristic in the literature, shows that ODEST always outperforms the heuristic and achieves an improvement (i.e., reduction) of the makespan (hence the throughput of the line) of up to 43%.

1. Introduction

Scheduling systems with transportation resources, in particular material-handling hoists (or robots), have been the focus of extensive research in the last two decades. The hoist scheduling problem (HSP) is usually encountered in the surface treatment industry, particularly in electroplating processes which are used in the production of printed circuit boards. These are in turn used as components in electronic products such as laptops and printers. Such systems usually involve processing several types of parts or products and are characterized by frequent processing order changes.

A surface treatment process [27] is a multiple-stage production system where the first and the last stages are the loading and unloading stations, respectively, and each of the other stages consists of a soaking bath, or tank, containing a chemical solution. Based on its processing requirement, a part, or batch of parts, visits a sequence of tanks where it is soaked for a specific duration. In general, the soaking duration depends on the type of part being processed and can be either fixed or vary within an acceptable time interval. However, in a production line, all parts go through the same sequence and have the same soaking duration at each visited tank. Such systems are usually mass production systems that are characterized by high throughput rates. Parts are transferred from one stage to the next via a transport system consisting of a hoist or hoists that are mounted on a rail system above the tanks. The latter are arranged in a straight line which sets the axis of the hoist movement. The hoist is a controllable material handling automaton that is designed to pick up a part from one workstation, transport it, and put it down in another workstation. Hence, to ensure a high throughput rate and high quality parts, efficient part-dispatching and hoist movement scheduling are necessary.

The above described systems can be considered as a generalization of flow shop systems with additional features: no-intermediate storage, no-wait conditions, flexible processing times, non-negligible transfer time, among others. As a result, scheduling parts (or jobs) on such systems is challenging. The difficulty is due to several operating constraints: Jobs have to be removed from the soaking tank before or as soon as they reach the maximum soaking duration otherwise they would become defective (melted in acid, or become too hot for instance); Lack of intermediate storage points between soaking tanks. Thus, semi-finished parts cannot be stored for later use since otherwise they would lose their chemical properties; and hoist transport durations are of the same order of magnitude as the soaking durations, hence cannot be ignored. For instance, at the time a job, at a particular tank, reaches its maximum soaking duration, the transport hoist must be positioned above that tank (otherwise the part becomes defective) and at the same time the next tank

* Corresponding author.
E-mail addresses: adnen.el-amraoui@univ-orleans.fr, elamraoui.adnen@gmail.com (A.E. Amraoui), Mohsen.elhafsi@ucr.edu (M. Elhafsi).

http://dx.doi.org/10.1016/j.cor.2015.10.006

receiving this job must be free (because of the lack of intermediate stocking option).

The objective of the schedule in such systems can take different forms: maximize throughput (by minimizing the completion time of the last task), maximize service level (by minimizing tardiness), or minimize production costs (by minimizing the average task flow time).

In the above class of scheduling problems, the resources are not only limited to the processing units but also include the material handling resources such as the hoist or hoists. As such, scheduling the hoist movements in order to process a set of parts becomes the most important task since a workstation schedule can be easily obtained once the hoist movement schedule has been generated. Such schedule can be cyclic or dynamic. In a cyclic hoist scheduling problem (CHSP), a static schedule is continuously repeated to produce one or several types of parts. In this case, the mix of parts is invariant over time and it is assumed that new items are available at the loading station at fixed time intervals and are introduced to the line in a cyclical fashion. Cyclic scheduling is the most common approach in surface treatment facilities [3] due to several advantages. Production operations can be easily managed due to the fixed number of steps in the cyclic schedule and the system performance is known in advance. Despite these advantages, cyclic schedules are limited. In particular, it is difficult to obtain cyclic schedules for systems operating multiple part-types. In such a case, even a small change in the parts order results in a redesign of the line which usually entails a costly expenditure.

In contrast, since surface treatment facilities generally deal with a variety of part types, flexibility of the scheduling method as well as the availability of workstations become a priority. Given the variability in processing steps of the different part types, some parts may only use a subset of the workstations. In such a case, dynamic scheduling proved to be more efficient than cyclic.

In a dynamic environment, the production manager updates the schedule dynamically whenever new parts, with various types, arrive to the loading station. In this case, the schedule is updated while processing of the current parts is ongoing. The updated sequence must take into account the parts that are in process as well as the newly received parts while maintaining feasibility. Such a scheduling problem is known as the dynamic hoist scheduling problem (DHSP). However, in practice since processing cannot be preempted (otherwise parts are rendered defective), processing cannot be halted and hence the current schedule has to be maintained while generation of a new feasible schedule takes place. Furthermore, even if schedules can be generated as often as new parts are added to the ones under the current schedule, changing schedules too often results in a loss of capacity. This may be a big factor for production lines that are designed to operate with high throughput rates.

In this paper, we consider a facility with a single handling resource (a hoist). The hoist has to perform a sequence of moves in order to accomplish a set of jobs, with varying processing requirements, while satisfying processing and transport resource constraints. Our objective is to determine a feasible schedule (*i.e.*, a sequence) that minimizes the total processing time of a set of jobs (*i.e.*, the makespan), denoted by C_{max}, while, at the same time, satisfying surface treatment constraints, thus ensuring high quality products. The first goal can be achieved by introducing new jobs, to the line, as soon as possible. The second goal can be achieved by guaranteeing that a job's processing duration is confined within specified lower and upper processing time requirements. If a job spends less time than the lower required duration or more time than the upper required duration, the processed part is considered defective.

The paper is organized as follows. In Section 2, we present an extensive survey of the relevant literature. In Section 3, we describe the problem studied in this paper. In Section 4, we present a Mixed Integer Linear Programming Formulation of the problem. In Section 5, we describe a heuristic solution approach. In Section 6, we present computational experiments and discuss the results. Finally, in Section 7, we conclude the paper and suggest future research directions of the work.

2. Literature review

Since the first studied model by [23], automated electroplating systems have been widely studied [21]. Several mathematical models [1,10–12,14,18,2,19,20,26,28,31,32,5,7,4]) and heuristic algorithms [17,30,16,24,29,7] have been developed. However, only few papers focused on the DHSP.

In order to help with the exposition, we use the notation introduced by Manier and Bloch [21] which consists of a four-field scheme, $\alpha|\beta|\delta|\gamma$, where α indicates the type of the HSP. In particular, when the problem is static and the schedule is cyclical, α=CHSP [8,9], and when the scheduling problem is dynamic, α=DHSP. For the sake of brevity, the reader is referred to Manier and Bloch [21] for the definition of the parameter β, δ and γ.

The literature dealing with the DHSP has focused, for the most, on the development of heuristics. The first heuristic algorithm for the DHSP was proposed by Song et al. [25] who studied a CHSP|10//diss|/12|Other problem. This iterative procedure is known in the literature as the EST (Earliest Start Time) heuristic. The EST heuristic attempts to schedule the hoist loading operations for jobs one after the other; and if a transport constraint is violated, the entire job hoist moves are re-scheduled and the introduction time of the job, into the line, is delayed. The authors use a set of rules to minimize the time between the insertions of two consecutive parts into the line. However, they assume that processing times are not bounded but fixed to the minimum times required and also assume a single job per part. Yih [29] studied a DHSP|5//diss|100/100, 7|C_{max} problem. He proposes an iterative heuristic procedure by introducing jobs to the line as early as possible and uses the minimum operation time to define the starting hoist move times. Ge and Yih [13] studied a DHSP|mt//diss|100/nps, nop|C_{max} problem. The authors propose a tree search procedure to construct the schedule. The proposed tree is constructed progressively by adding a new hoist move at each level where priority is given to the introduction of a new carrier to the line whenever possible. Cheng and Smith [6] studied the same problem as Yih [29]. They use a graph model to represent conjunctive constraints to obtain a partial schedule and compute the longest path in the graph in order to adjust starting hoist move dates as well as to check the consistency of the proposed schedule. Chu and Proth [7] studied a system involving a single machine (or robot) and a set of jobs where the processing duration of each job is upper and lower bounded. The problem can be considered as the equivalent of a dynamic hoist scheduling problem with infinite tanks capacity and can be classified as DHSP|1//diss|400/4,4|C_{max}. In order to solve the problem, the authors develop three heuristics, based on priority rules, to solve large-size problems, and a branch and bound procedure to obtain optimal solutions for small size problems. Spacek et al. [26] use a P-temporal Petri network to model the DHSP|1,12,3//|/14|C_{max} problem. To reduce the complexity of the problem, the authors solve the problem by relaxing the hoists' unit capacity constraints. A repair procedure, inspired from Yih [29], is then applied to the schedule when an overlap of time windows is detected. Hindi and Fleszar [15] propose a non-standard constraint problem to solve a DHSP|5//diss|100/100, 6|C_{max} problem using a similar approach to that of Cheng and Smith [6]. Paul et al. [22] studied the DHSP|18//diss|40/4, 12|C_{max} problem. They extended the approach proposed by Song et al. [25] to the case of several part-type jobs and showed, through an illustrative example, that using lower bounds for soaking times can provide good schedules. Kujawski and Świątek [16] studied the real-time scheduling of a multi-hoist electroplating production line (DHSP|mh,13//ass|