Computers & Operations Research 44 (2014) 52–65

Contents lists available at ScienceDirect

Computers & Operations Research

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

Heuristics for an assembly flow-shop with non-identical assembly machines and sequence dependent setup times to minimize sum of holding and delay costs

Javad Navaei [a,b], Seyyed Mohammad Taghi Fatemi Ghomi [b,*], Fariborz Jolai [c], Ashkan Mozdgir [d]

[a] Intelligent Manufacturing Systems(IMS) Centre, University of Windsor, Windsor, ON, Canada
[b] Department of Industrial Engineering, Amirkabir University of Technology, Tehran 1591634311, Iran
[c] Department of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran
[d] Payame Noor University, 19395-4697 Tehran, Iran

ARTICLE INFO

Available online 31 October 2013

Keywords:
Assembly flow-shop
Non-identical assembly machines
Holding and delay costs
Simulated annealing
Imperialist competitive algorithm
Sequence dependent setup time

ABSTRACT

This paper addresses the two-stage assembly flow-shop scheduling problem with non-identical assembly machines at the second stage to minimize a sum of holding and delay costs. Although there are more than one assembly machine in many manufacturing systems, to the best of our knowledge, the two-stage assembly flow-shop scheduling problem (TSAFSP) has never been addressed with more than one assembly machine at stage two. Moreover, setup time is an inevitable factor in many cases and so in this paper, for more reality, sequence dependent setup times are considered for both stages. After extending mathematical modeling, to solve the addressed problem, four hybrid meta-heuristics are developed. A simulated annealing algorithm (SA) and an imperialist competitive algorithm (ICA) in order to find a sequence of jobs at the first stage and a heuristic (HEU) and again SA for assigning addressed jobs to assembly machines in stage two; therefore, these hybrid meta-heuristics are SA+HEU, ICA+HEU, SA+SA and ICA+SA. Computational results reveal that ICA+HEU outperforms all other algorithms; however, the run time of SA+HEU is the smallest among the algorithms.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Nowadays, because of diversity of demands and its competitiveness, manufacturing corporations have to decrease their batch production whereby they can amend their production flexibility. So, for these kinds of manufactures, applying an appropriate scheduling is vital. One of the problems similar to this situation is the two-stage assembly flow-shop scheduling problem (TSAFSP). This problem was initially addressed by Lee and Cheng [21] and [33] and consists of two stages. At the first stage there are m machines while the second one consists of one or more assembly machines. There are n available jobs at time zero. Each job requires $m+1$ operations. Each machine can process only one job at a time and there is no preemption. The first m operations of a job are processed by m different machines at stage one. Then, the $(m+1)$th operation of the job, assembly process, is done at the second stage. The last operation of the job can start if only all its first m operations have been finished at the first stage. There are many cases in real life which coincide with this kind of problem. One of these cases is the production of personal computers. Each customer according to his/her need orders a particular set of modules: a monitor, a mouse, a keyboard, a hard disc, a CPU, and so forth. It is obvious that different customers select different types/configurations from each module. Despite the fact that there may be few options for each module (e.g. only six different configurations for the CPU module) from which a customer can select, a large variety of finished products can be obtained by combining different types of each module at the packing stage (assembly stage). Different modules are first performed on different independent feeder lines (at the first stage) and then all the produced modules are assembled at a packing station (second stage) [33]. Moreover, the problem is applied in other areas such as in database distribution [1], batch production and supply chain management [23] and invoice printing system [43]. After Lee and Cheng [21] and [33] some researchers attempted to extend the problem by adding some limitation or considering practice relevant characteristics, like three-stage assembly flow-shop [18,26], set up times [3], blocking [16] and so forth.

To the best of our knowledge, no one has considered TSAFSP with more than one assembly machine at the second stage whereas in reality most of huge manufacturing lines use more than one assembly machine in order to satisfy the demand effectively. Therefore, in this paper, we use w non-identical assembly machines

* Corresponding author. Tel.: +98 21 66545381; fax: +98 21 66954569.
E-mail address: fatemi@aut.ac.ir (S.M.T. Fatemi Ghomi).

0305-0548/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2013.10.008

($w>1$) at the second stage and consider the problem as “two-stage hybrid assembly flow-shop scheduling problem” or TSHAFSP. In TSHAFSP, after processing all of m operations of a given job at the first stage, the job can be assembled just by one of w non-identical assembly machines at the second stage.

In addition, in most of real manufacturing systems, sequence dependent setup time is an ineluctable factor. Allahverdi and Ruiz and Allahverdi [36] and Allahverdi and Al-Anzi, [3] assumed the independent setup time in their works while Hendizadeh et al. [19], Yu et al. [18,42] used dependent setup time to make their problems more viable. Therefore, in this paper, to have a more realistic problem, sequence dependent setup times are considered in both stages.

In TSAFSP, researchers have used different objectives such as makespan, mean completion time, lateness, tardiness, number of tardy jobs and so on [2,3,24,29,39]. In real life many managers are curious about total cost in order to find the best sequence of jobs to minimize their expenditures. Two important types of costs derived by sequencing of jobs are holding and delay costs. It is obvious that if a job is completed before its due date, it will impose holding cost for the corporation. On the other hand, if a job is accomplished after its due date, it will impose delay cost. Thus, in this paper we aim to minimize the sum of holding and delay costs as the objective of the problem. To the best of our knowledge none of the research works in TSAFSP have used the sum of holding and delay costs as an objective function, whereas we can find these kinds of objectives in other similar areas [12,15].

Different optimal solution techniques, such as branch-and-bound procedures [17,39,40] have been developed for TSAFSP. Thanks to the combinatorial nature of the problem, accessibility to exact solution for large size of problems is nearly impossible. The TSAFSP with only two machines at the first stage and a single assembly machine at the second stage is considered as NP-hard problem [21,33]. Therefore, our problem which is the generalization of the TSAFSP is NP-hard too. Some researchers have used meta-heuristic algorithms to solve the TSAFSP. The most notable of this group of algorithms used in optimization problems are evolutionary computation methods such as evolutionary algorithms [2], simulated annealing (SA) [2,22,24,27,29], variable neighborhood search [24,29], self-adaptive differential evolution (SDE) [2], genetic algorithm (GA) [10,16,4], tabu-search [1,18], particle swarm optimization [1] and hybrid of them [38]. On account of the fact that this problem contains two phases which are finding a sequence of jobs at the first stage and assigning addressed jobs to w assembly machines at the second stage, more than one heuristic or meta-heuristic to solve the problem is required. This paper considers four combinations of meta-heuristics and a new heuristic to solve the problem: combination of SA and SA (SA+SA), ICA and SA (ICA+SA), and each of SA and ICA with the heuristic is used to assign jobs to the assembly machines at the second stage (SA+HEU and ICA+HEU). In other words, four hybrid meta-heuristics are proposed. SA is one of the most practical meta-heuristics used in many papers such as [2], simulated annealing (SA) [2,24,29]. Allahverdi and Al-Anzi [2] showed that SA outperforms ant colony optimization and SDE in TSAFSP. On the other hand, ICA is one of the recent meta-heuristics which was initially proposed by Atashpaz-Gargari and Lucas [5] and has been shown to have a good performance [37,31]. It is the reason why we use these metaheuristic algorithms to solve the problem.

In this paper we address the two stage hybrid assembly flow shop scheduling problem (TSHAFSP) with sequence dependent setup time in both stages to find the best sequence and assignment in order to minimize the sum of holding and delay costs according to the due date of each job.

Considering planning horizon, the length of the planning horizon for a scheduling decision is short [7] because of the fact that this type of decision is mostly made in an operational level. However, the length, depending on the type of the product, may vary from hours in one industry to months or even years in another industry. It is noteworthy that when a set of n jobs is completed, for the new set of jobs, a new planning is considered. In other words, for each production series there is a specific planning. In the present work, we have not used any specific time unit (minutes, days, week, etc.) in order to accentuate the fact that our problem can be applied for different industries with different length of the planning horizon. The paper is organized as follows. Section 2 gives the problem description and extends a mix integer non linear programming (MINLP) mathematical model. Section 3 introduces the proposed hybrid meta-heuristics while Section 4 gives their performances and comparison of results. Finally, Section 5 is devoted to summaries of results and some possible further researches.

2. Problem statement

The two stage hybrid assembly flow-shop scheduling problem can be stated as follows. n jobs are available in time zero. Each job has ($m+1$) operations where the first m operations should be done at the first stage by m parallel machines. Each of m machines can process one part of a job at a time and no preemption is allowed. No priorities are considered for accomplishing the jobs. All machines in this stage are distinctive and require sequence dependent setup time. The assembly process of a job can be started at the second stage only when all the parts of the job have been completed at the first stage. At the second stage, there are w non-identical assembly machines. Sequence dependent setup times are considered for the second stage too. When a job is accomplished at stage one, it can be assembled just by one of w assembly machines at stage two. One should note that although there may be some idle assembly machines at a time, a job may wait to assign to an assembly machine which is busy at that time; on account of the fact that assembly times can differ from one assembly machine to the other one for a given job, waiting for being assembled by the busy machine is reasonable. Moreover, the sequence of jobs at stage one can change when jobs enter to the second stage because of the sequence dependent setup times at stage two.

Regarding the objective function, there are two types of costs: holding and delay costs. Holding costs are associated to those costs imposed by storing unsold products. Warehouse personnel, space, utilities and insurance are some examples of holding costs. On the other hand, delay costs are imposed when a job is accomplished after its due date. Delay costs are based on a kind of agreement between the company and its customer thereby the company is committed to deliver its customer's order before a predefined due date. Each of holding and delay costs is further subdivided to fixed and variable costs. Fixed costs, in the present work, are those costs that are time-independent. In contrast, variable costs of a given job refer to those types of costs that vary depending on the completion time of the job. Note that in this paper just one unit of each job is produced and hence, the number of units of each job does not affect the costs.

So, a finished job may have only one of the following three different statuses: completed before its due date, completed after its due date and completed exactly in its due date (just in time). Regarding this fact and aforementioned definitions of holding and delay costs, a delayed job has only delay costs and we do not consider any holding costs for that job. Similarly, a job finished before its due date has only holding costs. In the case that the job is accomplished exactly at its due date (just in time), the job has

Download English Version:

https://daneshyari.com/en/article/475611

Download Persian Version:

https://daneshyari.com/article/475611

Daneshyari.com