Computers & Operations Research 39 (2012) 1713–1723

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

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

A genetic algorithm with tabu search procedure for flexible job shop scheduling with transportation constraints and bounded processing times

Q. Zhang, H. Manier, M.-A. Manier *

UTBM; Université de Technologie de Belfort-Montbéliard, Laboratory Systems and Transport, 90010 Belfort Cedex, France

ARTICLE INFO

Available online 17 October 2011

Keywords:
Flexible Job Shop Scheduling Problem with transportation
Bounded processing times
Genetic algorithm
Tabu search
Flexible manufacturing system
Robotic cell
Surface treatment facility
Disjunctive graph

ABSTRACT

In this paper, we propose a model for Flexible Job Shop Scheduling Problem (FJSSP) with transportation constraints and bounded processing times. This is a NP hard problem. Objectives are to minimize the makespan and the storage of solutions. A genetic algorithm with tabu search procedure is proposed to solve both assignment of resources and sequencing problems on each resource. In order to evaluate the proposed algorithm's efficiency, five types of instances are tested. Three of them consider sequencing problems with or without assignment of processing or/and transport resources. The fourth and fifth ones introduce bounded processing times which mainly characterize Surface Treatment Facilities (STFs). Computational results show that our model and method are efficient for solving both assignment and scheduling problems in various kinds of systems.

1. Introduction

This paper deals with a Flexible Job Shop Scheduling Problem (FJSSP) with transportation constraints involved by critical handling resources, in which the processing durations of jobs may be limited by lower and upper bounds. FJSSP is a generalization of the classical job shop scheduling problem which allows an operation to be processed by any machine from a given set. Here we study an extension of FJSSP which considers bounded processing times and transport constraints. Thus processing times are not fixed and transportation tasks cannot be neglected. Such problems may be mainly encountered in three kinds of production systems: Flexible Manufacturing Systems (FMS), Robotic Cells (RC) and Surface Treatment Facilities (STF). FMS are highly automated production systems producing a wide variety of job types. Many FMS consist of several machine tools and tool handling devices such as Automated Guided Vehicles (AGVs) or robots. During the last three decades, FMS scheduling problem has received a great deal of attention because reducing lead time is always a very important goal for industry. Two basic types of problems are studied in FMS: real-time scheduling and offline scheduling. Both aspects are studied by several researchers [1–5]. In some automated manufacturing systems, several machines are grouped into cells which are responsible for treating a subset of tasks of the entire production, and material handling is usually performed by one (or a few) robots or automatic guided vehicles (AGVs). This kind of cell can be called robotic cell. Robotic cells with no storage buffers between machines are widely used in practice. There are mainly two types of problems in robotic cells [6]. The first one is referred to unbounded processing time windows which means each part has a precisely defined processing time on each machine and can wait on the machine indefinitely long after it has been processed [7–9]. The second one is known as zero-width processing time windows or no-wait [10]. STF belongs to the last class and has been widely studied since 1970s. Such a system is composed of tanks containing chemical or electrolytic baths and hoists are responsible for transportation tasks. Processing times are limited by minimum and maximum durations. The generic name of problems encountered in STF is Hoist Scheduling Problems (HSPs) [11]. Ref. [12] classified the problems in STF into four types: CHSP (cyclic HSP), PHSP (predictive HSP) which are static problems, DHSP (dynamic HSP) and RHSP (reactive HSP). For the last two types of systems (RC and STF), most of the studied problems are cyclic ones in which the types of parts are limited to one or a few. Cyclic schedules are efficient when producing identical parts or only a few types of parts. However, few researchers considered scheduling of job-shop with transportation in these systems. The performance of the cell becomes highly dependent on the interaction between the processingresources and material handling system.

In this paper, we study job shop scheduling problems for manufacturing systems with various characteristics (FMS, RC, STF) but with critical transport resources. Job shop scheduling and vehicle scheduling are both NP-hard problems, and only very special

* Corresponding author.
E-mail addresses: qiao.zhang@utbm.fr (Q. Zhang), herve.manier@utbm.fr (H. Manier), marie-ange.manier@utbm.fr (M.-A. Manier).

0305-0548/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2011.10.007

cases may be polynomially solved [13]. Several researchers have devoted to study job shop scheduling problems with transportation constraints in various systems. However, the progress is limited as this kind of problem is difficult to solve even for simplified and small size cases. Ref. [14] integrated transport constraints in the scheduling problem with one robot. Refs. [15] and [16] considered a job shop problem with several robots, with fixed operation times and fixed assignment of machine for each job's operation. Ref. [17] studied a two machine flow shop scheduling problem with intermediate transportation with a single transporter. Ref. [18] studied coupled task problem and one-machine robotic cell problems. It reported new algorithmic procedure for this problem with or without tolerances on the distance. Ref. [19] considered a cyclic hoist scheduling problem with a single hoist, but without assignment problem. Ref. [20] studied the predictive and dynamic HSP by applying stochastic methods. Ref. [21] proposed an intelligent scenario selection method in dynamic hoist scheduling problem. In recent years, a few FMS researchers have studied the flexible job shop problems with transportation such as [22], but with fixed processing times. Overall, only some cases studied in FMS considered both assignment and transportation. Most of the problems studied in robotic cells and STF are not job shop and few of them consider assignment problems. So until now, few benchmarks could be found about the more general flexible job shop scheduling with both assignment problems and bounded processing times.

The scheduling problem of manufacturing systems has been addressed using exact and/or heuristic methods. Exact methods are usually suitable for simple or particular FMS. Several researchers have studied the job shop scheduling problem in FMS by considering transport resources. Ref. [23] proposed a dynamic programming approach to construct optimal machine and vehicle schedules. Ref. [24] developed a mixed integer programming (MIP) formulation raising this constraint on the vehicles. Refs. [25,26] elaborated a genetic algorithm. Refs. [27,28] proposed, respectively, neural networks and tabu search approaches. Ref. [29] described a hybrid method composed of a genetic algorithm for the scheduling of machines and a heuristic for the scheduling of vehicles. Ref. [30] applied a decomposition method where the master problem (scheduling) is modelled with constraint programming and the subproblem (conflict free routing) with mixed integer programming. Ref. [16] proposed an efficient neighbouring system and implemented into three different metaheuristics: iterated local search, simulated annealing and their hybridization. Approximated methods are often applied for the offline scheduling of FMS. Ref. [31] used a mixed integer linear program (MILP) to find optimal solutions for the FMS Scheduling Problem with one vehicle. Refs. [16,22] used a dynamic method to test FMS instances with and without assignment problems of processing resources. Ref. [7] proposed a polynomial algorithm for finding the optimal cycle in a robotic cell with production of identical parts. Ref. [32] used a kangaroo algorithm with a multi-agent model to solve real-time hoist control. Ref. [33] applied a branch and bound search process to determine optimal cyclic hoist schedule in a single hoist electroplating line. Ref. [34] used constraint logic programming to solve cyclic multi-hoist scheduling problem with collision constraints. Ref. [35] proposed an evolutionary approach by encoding empty moves of hoists in HSP to optimize the couple (minimal cycle time, number of hoists). Ref. [20] combined a tabu search with a modified procedure of shifting bottleneck heuristics to solve Dynamic HSP. A disjunctive graph is proposed to represent each solution. The proposed heuristic finds feasible schedules and then add them into the graph. Ref. [36] developed a mixed integer linear programming model to schedule multiple tanks.

In this paper, we propose a generic model which considers flexible processing resources and transport constraints with possible bounded processing times. Based on this model, scheduling problems in each of the three systems mentioned above can be treated as special cases in this model. A genetic algorithm with tabu search procedure is applied to solve the offline problems. In Section 2, the associated notation and model are presented. Section 3 describes the procedure of the hybrid algorithm. Various kinds of instances are tested for each class of manufacturing systems and the computational results are provided and discussed in Section 4. Finally, Section 5 is devoted to the conclusion and perspectives.

2. Flexible Job Shop Scheduling Problem with transportation and bounded processing times

The Flexible Job Shop Scheduling Problem (FJSSP) is extended from the classical job shop scheduling problem which allows an operation to be processed by one machine chosen from a given set. Because of this, the assignment of each operation to its processing machine and the sequencing on each machine are needed to be determined. For this kind of model, all loaded and empty transport times cannot be neglected. Furthermore, all the processing times are bounded.

2.1. Problem statement

The most basic version of job shop can be defined as follows: a set $\{J_1, J_2, \ldots, J_n\}$ of n independent jobs of varying sizes needs to be scheduled on m machines. Job J_i consists of O_i operations. Each operation is associated with a set of processing machines and a processing time. When a job has finished an operation and needs to move to another machine, a transport resource is required. There are r identical transport resources (vehicles) responsible for transportation tasks, and all jobs can be transported by any of them. All the loaded/empty transportation times are job-independent but machine-dependent. The objective function is to minimize both the makespan C_{max} and storage which is defined as the total waiting time before and after each machine during production.

For the proposed model, some assumptions are made including the following ones:

- machine failures and vehicle failures are ignored,
- input and output buffers are unlimited,
- processing and transport resources can handle at most one job at a time,
- each task can only be assigned to one and only one resource,
- all machines are available at time zero,
- all jobs are ready at time zero,
- each job consists of its fixed processing sequence,
- once an operation is treated on a machine, it cannot be interrupted until it is finished (no preemption is allowed).

2.2. Notation

The FJSSP with transportation and bounded processing times we consider here is a problem which includes n jobs operated on m machines and transported by r transport resources. Some symbols and notations have been defined as follows:

Input data

J_i	job i ($i \in [1,n]$).
O_i	number of operations for job J_i ($i \in [1,n]$).
OP_{ij}	jth operation of J_i ($j \in [1,O_i]$).
P^-_{ij}, P^+_{ij}	minimum and maximum processing times for OP_{ij}.
C_i	completion time of J_i.