Applied Soft Computing xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Applied Soft Computing

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

Two-objective stochastic flow-shop scheduling with deteriorating and learning effect in Industry 4.0-based manufacturing system

Yaping Fu [a], Jinliang Ding [b], Hongfeng Wang [b], Junwei Wang [c,*]

[a] Institute of Complexity Science, Qingdao University, Qingdao, 266071, China
[b] College of Information Science and Engineering, State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, 110819, China
[c] Department of Industrial and Manufacturing System Engineering, The University of Hong Kong, Hong Kong, China

ARTICLE INFO

Article history:
Received 30 September 2016
Received in revised form
17 November 2017
Accepted 3 December 2017
Available online xxx

Keywords:
Industry 4.0
Flow-shop scheduling
Multi-objective optimization
Uncertainty
Time-dependent processing time

ABSTRACT

Industry 4.0 is widely accepted in manufacturing industry since it guides a novel and promising production paradigm. A new characteristic in Industry 4.0-based manufacturing systems is that the applications of advanced intelligent machines which have communication, self-optimization and self-training behaviors. Based on this new change, this study investigates a flow-shop scheduling problem under the consideration of multiple objectives, time-dependent processing time and uncertainty. A mixed integer programming model is formulated for this problem, and a fireworks algorithm is developed where some special strategies are designed, e.g., explosion sparks procedure and selection solution procedure. Simulation experiments on a set of test problems are carried out, and the experimental results demonstrate that the model and proposed algorithm can achieve a satisfactory performance by comparing with three state-of-the-art multi-objective optimization algorithms.

1. Introduction

Recently, the competition among manufacturing enterprises has become more and more fierce, many countries and organizations have proposed some novel production paradigms to integrate the manufacturing process [1,2]. Industry 4.0 is one of the most popular concepts in advance manufacturing fields, and it has been regarded as a future direction of manufacturing industry [3,4]. Industry 4.0 is the fourth industrial revolution which applies the principles of cyber-physical systems, internet and future-oriented technologies, and smart systems with human-machine interaction paradigms. The Internet of Things and Services enables to network the entire factory to form a smart factory [5–7]. It significantly influences the production environment in the execution of operations. In contrast to conventional manufacturing systems, the introduction of information and communication systems into industrial network leads to a steep rise in the degree of automation, and advance intelligent machines can collect real-time information for dynamic self-optimization, self-training and self-maintaining behaviors, which can synchronize themselves and influence upon production processes.

Although the framework of Industry 4.0 embeds with the latest technologies and intelligent algorithms, a smart factory allows itself to be built on the foundations of classical manufacturing systems [8,9]. Therefore, some key issues and techniques also play an important role in an Industry 4.0-based manufacturing system [10–13]. Scheduling has been accepted as a useful tool by manufacturing systems to realize a higher utilization of resource, and many optimization models and methods have been proposed and applied in the past few years [14,15]. Nevertheless, scheduling in Industry 4.0-based manufacturing systems will become more complex and difficult since advanced intelligent machines are widespreadly applied. With more advanced analytics, the advent of cyber-physical systems and cloud computing framework, Industry 4.0-based manufacturing systems will be able to achieve huge amounts of data that helps advanced intelligent machines to be self-optimization, self-training and self-maintaining [16]. As the market environment becomes increasingly competitive, a decision-maker has to move toward frequent production change in order to provide the customers with more production variety [17]. In this situation, on the one hand, the advanced intelligent machines need to execute the self-optimization and self-training to make themselves more proficient for the new products, which makes the processing time for a product shorter with the time continues; on the

* Corresponding author.
E-mail address: jwwang@hku.hk (J. Wang).

https://doi.org/10.1016/j.asoc.2017.12.009
1568-4946/

other hand, they also wear away as the time continues, and the processing time for a product becomes longer if it is processed later. Thus, time-dependent processing time usually occurs. It is noticeable that the self-optimization and self-training of intelligent machines is difficult to know in advance since it depends on many factors, e.g., product structure and training algorithm. Moreover, customers' involvement should be established to integrate their requirements with production process in making decisions [18]. Therefore, scheduling problem in Industry 4.0-based manufacturing systems is challenged by these characteristics.

In this study, we consider these characteristic in an Industry 4.0-based manufacturing system, and propose a flow-shop scheduling problem with multiple objective functions, time-dependent processing time and uncertainty. In order to deal with it, this study designs a multi-objective discrete fireworks algorithm that incorporates some novel techniques to make it more powerful.

The reminder of this paper is organized as follows. Section 2 devotes to literature review. Section 3 presents the problem definition in which the formulation and notations are described, and a mixed integer programming model is presented. Section 4 proposes the solution method and Section 5 carries out simulation experiments and shows the analysis on the experimental results. The final section gives the conclusion of this paper and outlooks the future work.

2. Literature review

Flow-shop scheduling is one of the most well-known scheduling problems in manufacturing systems [19,20]. It also has widespread applications in Industry 4.0-based manufacturing systems [21]. The time-dependent processing in flow-shop can be classified into two categories, i.e., deteriorating and learning effect [22–24]. The deteriorating effect indicates that the delay in starting to process a job will increase its processing time, while the learning effect means that the actual processing time of job becomes shorter when it is processed later. There has been a growing interest in flow-shop deteriorating and/or learning scheduling in the past few years [25,26]. Lee et al. [27] considered a flow-shop deteriorating scheduling problem with minimizing the maximum tardiness, and used a branch and bound algorithm to solve it. Cheng et al. [28] investigated a two-machine flow-shop deteriorating scheduling problem, and designed a branch and bound algorithm to minimize the total flow time. Vahedi-Nouri et al. [29] studied the learning effect in a flow-shop scheduling problem with available constraints under the objective of minimizing the total flow time, and proposed a simulating annealing algorithm with heuristic rule. Wang et al. [30] investigated a two-machine flow-shop scheduling problem with the deteriorating and learning effect and utilized a branch and bound algorithm to minimize the makespan. Shiau et al. [31] studied a two-agent two-machine flow-shop scheduling problem with the learning effect in order to minimize the total completion time of an agent subject to the maximum tardiness of the other agent. Liu et al. [32] considered a no-wait two-machine flow-shop scheduling problem with the learning effect, where the objective function was to find an optimal sequence of jobs and optimal resource allocation. Note that most of the existing studies on deteriorating and/or learning scheduling in a two-machine flow-shop due to its complexity, and their application is rather limited in manufacturing systems.

Since product structures become more complex and varied, a real-world manufacturing system usually has multiple machines to execute a series of operations. Moreover, the applications of advanced intelligent machines and techniques which Industry 4.0 advocates make scheduling problems in manufacturing systems more complicated and difficult, e.g., their self-optimization, self-training and component abrasion lead to the time-dependent and uncertain processing time for jobs. In addition, a multi-objective optimization design also needs to be considered since a decision-maker usually wants to optimize cost-desired and service-oriented objectives simultaneously. Therefore, this study will investigate a two-objective stochastic flow-shop deteriorating and learning scheduling problem. In order to deal with it, a multi-objective discrete fireworks algorithm is developed, where the solution representation, explosion sparks procedure and selection solution procedure are designed based on its characteristics. Based on experiments on test problems, this study will validate the effectiveness and feasibility of the model and the proposed algorithm by comparing with three state-of-the-art multi-objective algorithms.

3. Problem description

A two-objective stochastic flow-shop deteriorating and learning scheduling problem in this study can be described as follows. There are n jobs that need to be processed on m machines following the same route, their normal processing time on machines are random variables that obey a known Normal distribution, and the actual processing time of jobs depends on their starting time and processed positions in a schedule. Considering the cost-desired and service-oriented factors are both important when making a scheduling decision, two objective functions, that is, to minimize the makespan and the total tardiness, will be considered in detail. Obviously, the former can increase the utilization of machines, and the latter can improve the satisfactory level of customers. The goal is to find a trade-off solution set for decision-makers to choose a preferred schedule. A feasible schedule must meet the following requirements: each machine can only process one job at a time, each job can be processed on one machine at a time, the preemption is not allowed and the setup time of jobs is included in their processing time. To build a mathematical model for this problem, the parameters and decision variables are defined as follows.

Indices$J$job index set, $J = \left\{1, 2, \cdots, n\right\}$, where n denotes the number of jobs.Mmachine index set, $M = \left\{1, 2, \cdots, m\right\}$, where m represents the number of machines.

Symbol variablesξp_{ij}normal processing time of job j on machine i.α_{ij}deteriorating rate of job j on machine i.β_{ij}learning coefficient of job j on machine i.ξS_{ij}starting time of job j on machine i.ξC_{ij}completion time of job j on machine i.ξC_jcompletion time of job j on the last machine.ξp_{ij}actual processing time of job j on machine i.$d_j$due date of job j.$[r]$job index in the r-thposition in a schedule.ξC_{max}maximal completion time of all jobs, i.e., makespan.Ga large number.ξa symbol represents that its corresponding variable is a random variable.

Note that ξp_{ij}, ξS_{ij}, ξC_{ij}, ξC_j, ξp_{ij}, $\xi C_{\max}$ are random variables. In the investigated problem, the actual processing time of jobs is defined as a function of their starting time and their processed positions in a schedule, that is, $\xi p_{ij} = \left(\xi p_{ij} + \alpha_{ij} \cdot \xi S_{ij}\right) \cdot r^{\beta_{ij}}$.

Decision variablesx_{jr}0-1 integer decision variable. If job j is processed in the r-th position, $x_{jr} = 1$. Otherwise, $x_{jr} = 0$.

Based on the notations and decision variables defined in the above, a mixed integer programming model can be built as follows.

$$\min E\left(\xi C_{max}\right) \tag{1}$$

$$\min \sum_{j=1}^{n} max\left(E\left(\xi C_j\right) - d_j, 0\right) \tag{2}$$

$$\sum_{j=1}^{n} x_{jr} = 1, r = 1, 2, \cdots, n. \tag{3}$$