European Journal of Operational Research 252 (2016) 737–749

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

Rescheduling on identical parallel machines with machine disruptions to minimize total completion time

Yunqiang Yin[a], T. C. E. Cheng[b], Du-Juan Wang[c,*]

[a] Faculty of Science, Kunming University of Science and Technology, Kunming 650093, China
[b] Department of Logistics and Maritime Studies, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
[c] School of Management Science and Engineering, Dalian University of Technology, Dalian 116023, China

ARTICLE INFO

Article history:
Received 29 March 2015
Accepted 24 January 2016
Available online 29 January 2016

Keywords:
Combinatorial optimization
Production
Rescheduling
Bicriterion analysis
Two-dimensional fully polynomial-time approximation scheme

ABSTRACT

We consider a scheduling problem where a set of jobs has already been assigned to identical parallel machines that are subject to disruptions with the objective of minimizing the total completion time. When machine disruptions occur, the affected jobs need to be rescheduled with a view to not causing excessive schedule disruption with respect to the original schedule. Schedule disruption is measured by the maximum time deviation or the total virtual tardiness, given that the completion time of any job in the original schedule can be regarded as an implied due date for the job concerned. We focus on the trade-off between the total completion time of the adjusted schedule and schedule disruption by finding the set of Pareto-optimal solutions. We show that both variants of the problem are $\mathcal{NP}$-hard in the strong sense when the number of machines is considered to be part of the input, and $\mathcal{NP}$-hard when the number of machines is fixed. In addition, we develop pseudo-polynomial-time solution algorithms for the two variants of the problem with a fixed number of machines, establishing that they are $\mathcal{NP}$-hard in the ordinary sense. For the variant where schedule disruption is modeled as the total virtual tardiness, we also show that the case where machine disruptions occur only on one of the machines admits a two-dimensional fully polynomial-time approximation scheme. We conduct extensive numerical studies to evaluate the performance of the proposed algorithms.

1. Introduction

Most modern production and service systems operate in a dynamic environment in which unexpected *disruptions* may occur, necessitating changes in the planned schedule, which may render the originally feasible schedule infeasible. Examples of such disruption events include the arrival of new orders, machine breakdowns, order cancellations, changes in order priority, processing delays, and unavailability of raw materials, personnel, tools, etc. Rescheduling, which involves adjusting the original schedule to account for a disruption, is necessary in order to minimize the effects of the disruption on the performance of the system. This involves a trade-off between finding a cost-effective new schedule and avoiding excessive changes to the original schedule. The degree of disruption to the original schedule is often modeled as a constraint or part of the original scheduling objective (Hall, Liu, & Potts, 2007; Hall & Potts, 2004, 2010; Hoogeveena, Lentéb, & T'kindtb, 2012; Jain & Foley, 2016; Liu & Ro, 2014; Qi, Bard, & Yu, 2006; Unal, Uzsoy, & Kiran, 1997; Wang, Liu, Wang & Wang, 2015; Yan, Che, Cai, & Tang, 2014; Yang, 2007; Yuan & Mu, 2007). Variants of the rescheduling problem can be found in many real-world applications such as automotive manufacturing (Bean, Birge, Mittenthal, & Noon, 1991), space shuttle missions (Zweben, Davis, Daun, & Deale, 1993), shipbuilding (Clausen, Hansen, Larsen, & Larsen, 2001), short-range airline planning (Yu, Argello, Song, McCowan, & White, 2003), deregulated power market (Dahal, Al-Arfaj, & Paudyal, 2015), etc.

The literature on rescheduling abounds. For recent reviews, the reader may refer to Aytug, Lawley, McKay, Mohan, and Uzsoy (2005), Billaut, Sanlaville, and Moukrim (2002), Ouelhadj and Petrovic (2009), and Vieira, Herrmann, and Lin (2003). In this paper we review only studies on machine rescheduling with unexpected disruptions arising from machine breakdowns that are directly related to our work. Leon, Wu, and Storer (1994) developed robustness measures and robust scheduling to deal with machine breakdowns and processing time variability when a right-shift repair strategy is used. Robustness is defined as the

* Corresponding author. Tel.: +86 13840846586.
E-mail address: wangdujuan@dlut.edu.cn (D.-J. Wang).

http://dx.doi.org/10.1016/j.ejor.2016.01.045
0377-2217/

minimization of the bicriterion objective function comprising the expected makespan and expected delay, where the expected delay is the deviation between the deterministic makespan in the original and adjusted schedules. Their experimental results showed that robust schedules significantly outperform schedules based on the makespan alone. Ozlen and Azizoğlu (2011) considered a rescheduling problem on unrelated parallel machines, where a disruption occurs on one of the machines. The scheduling measure is the total completion time and the deviation cost, which is the total disruption caused by the differences between the original and adjusted schedules. They developed polynomial-time algorithms to solve the following hierarchical optimization problems: minimizing the total disruption cost among the minimum total completion time schedules and minimizing the total completion time among the minimum total disruption cost schedules. Qi et al. (2006) considered a rescheduling problem in the presence of a machine breakdown in both the single-machine and two parallel-machine settings with the objective of minimizing the total completion time plus different measures of time disruption. They provided polynomial-time algorithms and pseudo-polynomial-time algorithms for the problems under consideration. Zhao and Tang (2010) extended some of their results to the case with linear deteriorating jobs. Liu and Ro (2014) considered a rescheduling problem with machine unavailability on a single machine, where disruption is measured as the maximum time deviation between the original and adjusted schedules. Studying a general model where the maximum time disruption appears both as a constraint and as part of the scheduling objective, they provided a pseudo-polynomial-time algorithm, a constant factor approximation algorithm, and a fully polynomial-time approximation scheme when the scheduling objective is to minimize the makespan or maximum lateness.

In this paper we address the issue of how to reschedule jobs in the presence of machine breakdowns. We assume that a set of jobs has been optimally scheduled according to the *shortest processing time (SPT) rule* to minimize the total completion time on m identical parallel machines (Baker, 1974). However, the processing of most of the jobs has not begun. This situation arises when schedules are planned in advance of their start dates, typically several weeks earlier in practice. Based on the SPT schedule, a lot of preparation work has been made, such as ordering raw materials, tooling the equipment, organizing the workforce, fixing customer delivery dates, etc. Due to unforeseen disruptions, different from the setting in Ozlen and Azizoğlu (2011), we consider the situation where machine breakdowns may occur on more than one machine, and the disruption start time and the duration of a machine disruption may differ on different machines. This necessitates rescheduling the remaining jobs in the original SPT schedule. However, doing so will disrupt the SPT schedule, causing havoc on the preparative work already undertaken. Thus, on rescheduling, it is important to adhere to the original scheduling objective, say, the total completion time, while minimizing the disruption cost with respect to the SPT schedule. In this paper we use the maximum time deviation or the total virtual tardiness, where the completion time of a job in the SPT schedule can be regarded as an implied due date for the job concerned here, to model the disruption cost with respect to the SPT schedule. Instead of modeling the degree of disruption over the original schedule as a constraint or part of the scheduling objective, we focus on the *trade-off* between the total completion time of the adjusted schedule and schedule disruption by finding the set of Pareto-optimal solutions for this bicriterion scheduling problem. The purpose of this paper is twofold. One is to study this innovative and more realistic scheduling model. The other is to ascertain the computational complexity status and provide solution procedures, if viable, for the problems under consideration.

To motivate our scheduling problem, consider a practical example related to the manufacturing of containers. In this context, the manufacturing process is labor intensive and several machines are deployed to manufacture a host of various container types. The preparation for the manufacturing of each type of containers incurs a high operational cost. The factory takes customer orders during the current scheduling cycle (usually a month) and generates an optimal production schedule for the next period that minimizes the scheduling cost measured as the total flowtime for manufacturing all the ordered containers. In other words, the factory uses the SPT dispatching rule to schedule the manufacturing of the containers. The delivery times and related transport arrangements of the finished containers are then determined accordingly. However, unexpected disruptions such as machine breakdowns may occur that will render the machines unavailable for certain periods of time, which will affect the utilization of the machines and order delivery to customers. This will result in the original SPT schedule no longer optimal, wreak havoc on the preparative work already undertaken, and make an impact on the subsequent transport arrangements. Hence, it is important to react quickly to such disruptions whereby the affected jobs need to be rescheduled with a view to reducing the scheduling cost, while not causing excessive schedule disruption with respect to the original SPT schedule (excessive schedule disruption will increase the operational cost or result in the loss of customer goodwill). From the viewpoint of the production planner, a trade-off between the scheduling cost and deviation from the original SPT schedule is desired. This situation can be modeled as our problem of rescheduling on identical parallel machines with machine disruptions to minimize the total completion time.

The rest of the paper is organized as follows: In Section 2 we formally formulate our problem into two variants. In Section 3 we analyze the computational complexity and derive structural properties that are useful for tackling the two variants of the problem under study. In Section 4 we develop a pseudo-polynomial-time dynamic programming solution algorithm for the variant with the maximum time deviation as the schedule disruption cost and a fixed number of machines, establishing that it is $\mathcal{NP}$-hard in the ordinary sense. In Section 5 we also develop a pseudo-polynomial-time dynamic programming solution algorithm for the variant with the total virtual tardiness as the schedule disruption cost and a fixed number of machines, establishing that it is $\mathcal{NP}$-hard in the ordinary sense, and convert the algorithm into a two-dimensional fully polynomial-time approximation scheme for the case where machine disruptions occur only on one of the machines. In the last section we conclude the paper and suggest topics for future research.

2. Problem statement and definitions

There are n jobs in the job set $J = \{1, 2, \ldots, n\}$ to be processed without interruption on m identical parallel machines $\{M_1, \ldots, M_m\}$, which can deal with only one job at a time. All the jobs are available for processing at time zero. Each job j has a processing requirement of length p_j. We assume that the jobs have been sequenced in an optimal schedule that minimizes the total completion time. It is well known that the jobs should be scheduled in the SPT order with no idle time between them for this purpose, whereby the jobs are sequenced successively on the m machines, i.e., jobs $i, m+i, \ldots, m\lfloor n/m \rfloor + i$ are successively scheduled on machine i without any idle time, $i = 1, \ldots, m$, where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x. Let π^* denote the sequence in which the jobs are scheduled in this SPT order. Hereafter, we assume that all the p_j are known non-negative integers and the jobs are sequenced in the SPT order such that