Omega ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Omega

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

Parallel-machine scheduling of deteriorating jobs with potential machine disruptions ☆

Yunqiang Yin [a,*], Yan Wang [a], T.C.E. Cheng [b], Wenqi Liu [a], Jinhai Li [a]

[a] Faculty of Science, Kunming University of Science and Technology, Kunming, China
[b] Department of Logistics and Maritime Studies, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

A R T I C L E I N F O

Article history:
Received 10 August 2015
Accepted 27 July 2016

Keywords:
Scheduling
Deteriorating jobs
Disruptive environment
Fully polynomial-time approximation scheme

A B S T R A C T

We consider parallel-machine scheduling of deteriorating jobs in a disruptive environment in which some of the machines will become unavailable due to potential disruptions. This means that a disruption to some of the machines may occur at a particular time, which will last for a period of time with a certain probability. If a job is disrupted during processing by a disrupted machine and it does not need (needs) to re-start after the machine becomes available again, it is called the resumable (non-resumable) case. By deteriorating jobs, we mean that the actual processing time of a job grows when it is scheduled for processing later because the machine efficiency deteriorates over time due to machine usage and aging. However, a repaired machine will return to its original state of efficiency. We consider two cases, namely performing maintenance immediately on the disrupted machine when a disruption occurs and not performing machine maintenance. In each case, the objective is to determine the optimal schedule to minimize the expected total completion time of the jobs in both non-resumable and resumable cases. We determine the computational complexity status of various cases of the problem, and provide pseudo-polynomial-time solution algorithms and fully polynomial-time approximation schemes for them, if viable.

1. Introduction

Contemporary production and service systems often operate in a dynamic and uncertain environment, in which unexpected events may occur from time to time. Should the expected events be *disruptions*, they may cause some resources (machines or facilities) to be unavailable for a certain period of time, which will directly affect the utilization of the resources and ultimately customer service. Examples of disruptive events occurring during production abound, e.g., machine breakdowns, power failures, and shortages of raw materials, personnel, tools, etc. Research on scheduling that takes disruptions into account is commonly known as scheduling with availability constraints, which has been extensively investigated in the literature. Lee et al. [13], Sanlaville and Schmidt [23], Schmidt [25], and Ma et al. [19] survey and summarize the major results and practices in this area.

Machine scheduling with availability constraints can be categorized into two major classes. One class is where the machine unavailability is deterministic due to some internal factors such as preventive maintenance. In this case, both the disruption starting time and duration are either fixed in advance [4,8–11,20,31,34,33] or are decision variables in the scheduling model [5,18,21,22,28–30,32]. The other class is where the machine unavailability is stochastic [1–3], which is caused by machine breakdowns or other internal and external factors. Lee and Yu [15] consider single-machine scheduling with potential disruptions due to external factors, e.g., bad weather (typhoons and snowstorms), labour strikes, power shortages, etc. In such a case, the disruption starting time is roughly known (should it happen); however, the disruption duration is unknown until the damage is made. They provide pseudo-polynomial-time algorithms to solve the problems of minimizing the expected total weighted completion time and the expected maximum tardiness. Subsequently, Lee and Yu [16] extend the results to the parallel-machine case to minimize the expected total weighted completion time.

☆This manuscript was processed by Associate Editor Strusevich.

* Corresponding author.
E-mail address: yinyunqiang@126.com (Y. Yin).

http://dx.doi.org/10.1016/j.omega.2016.07.006

Please cite this article as: Yin Y, et al. Parallel-machine scheduling of deteriorating jobs with potential machine disruptions. Omega (2016), http://dx.doi.org/10.1016/j.omega.2016.07.006

We pursue the stream of research initiated by Lee and Yu [15,16]. We consider scheduling of jobs on m identical parallel machines that are subject to potential disruptions in a deteriorating production environment, which means that the job processing times will deteriorate over time. Some of the machines may become unavailable for a period of time over the scheduling period due to potential disruptions arising from worker shortage, power shortage, etc. In such a case, we often know the disruption starting time (should it happen) in advance, yet the duration is unknown until it happens. That is, there is a possibility that a disruption will happen at a particular time and the disruption will last for a certain duration with a certain probability. So the machine unavailability will only be revealed at the time when the disruption occurs. Thus we assume that once a disruption occurs, we will know its duration. Specifically, we consider two cases. One is to perform maintenance immediately on each of the disrupted machines when a disruption occurs and the other is not to perform machine maintenance, where performing machine maintenance will improve the efficiency of the machine by returning it to its original state of efficiency at the expense of the cost incurred from maintenance. With known probabilities of all the unexpected events, the scheduling objective is to find an optimal schedule for the jobs to minimize the expected total completion time of the jobs. We extend the work of Lee and Yu [16] in three major ways as follows:

- We consider the scheduling problem in a deteriorating production environment, i.e., the actual processing time of a job grows when it is scheduled for processing later because the machine efficiency deteriorates over time due to machine usage and aging, which more accurately reflects real-life production.
- We assume that machine unavailability will only occur on some of the machines, which is the case where the factory has backup power to keep some of the machines working when the disruption occurs due to power shortage, whereas Lee and Yu [16] assume that machine unavailability will happen on all the machines.
- We include the case where the disruption may not happen (i.e., $\zeta_\gamma = 0$) in the non-resumable case, which Lee and Yu [16] do not consider.

The purpose of this paper is twofold. One is to study a more realistic and complex scheduling model that takes both potential machine disruptions and job deterioration into consideration. The other is to ascertain the computational complexity status and provide solution procedures, if viable, for the problems under consideration.

The rest of the paper is organized as follows: In Section 2 we introduce the notation and formally formulate our problems. In Section 3 we derive some structural properties of the optimal solutions that are useful for tackling our problems. In Sections 4 and 5 we analyze the computational complexity status and provide solution procedures, if viable, for the problems in the non-resumable and resumable cases, respectively. In the last section we conclude the paper and suggest topics for future research.

2. Problem formulation

We formally describe the general problem under consideration as follows: There are n independent jobs in the job set $N = \{J_1, J_2, \ldots, J_n\}$ to be processed on m identical parallel machines $M_1, M_2, \ldots, M_m$ over a scheduling period T. All the n jobs are available for processing at time zero. Each job needs to be processed on one machine only and each machine is capable of processing any job but at most one job at a time. The machines will experience deterioration in efficiency due to usage and aging [26]. As a result of deterioration in machine efficiency over time, the actual processing time of a job will become longer if it is scheduled for processing later. Specifically, if job J_j is processed on machine M_i and starts processing at time t, we define its actual processing time as $p_{jt} = p_j(1+at)$, where p_j is the normal processing time of job J_j and a $(a > 0)$ is the deteriorating rate common to all the jobs. Some of the m machines may become unavailable due to potential machine disruptions, each of which will last for a period of time with a certain probability. Without loss of generality, we assume that a machine disruption will happen at time r, which makes the first ι machines $M_1, \ldots, M_\iota$, $1 \le \iota \le m$, unavailable and the duration will take γ $(\gamma = 0, 1, \ldots, s)$ time units with a probability ζ_γ, which is the same for all the disrupted machines once the anticipated disruption occurs. Here, ζ_0 is the probability that the disruption will not happen and s is the maximum possible duration.

In order to reduce the effect of machine deterioration, an option strategy is to perform machine maintenance, which will improve the efficiency of the machine by making it return to its original state of efficiency. It follows that the actual processing time of job J_j will be the same as its normal processing time if it is the first job that starts processing on a repaired machine after maintenance. However, we focus on the case where the effect of machine deterioration on the objective value during the scheduling period is smaller compared with the cost incurred from the maintenance duration and the maintenance cost, which is reasonable in most production and service systems. As a consequence, we consider the following two cases:

Case 1: Perform maintenance immediately on each of the disrupted machines with a fixed duration, denoted by κ, when a disruption occurs by several maintenance workers (or teams). Thus, the disrupted machines will become unavailable during the time interval $[r, r + \max\{\gamma, \kappa\}]$ under the scenario γ $(\gamma \ge 1)$.

Case 2: Do not perform machine maintenance.

The reason for considering Case 1 is twofold. One is that it may reduce the effect on the objective value caused by machine disruptions because performing maintenance can improve the efficiency of the machines. The other is that it may reduce the effect on the objective value caused by performing maintenance because disrupted machines have unavailable time intervals due to machine disruptions, which can reduce the effect from maintenance duration. In each case, we consider both the *resumable* and *non-resumable* cases. If a job is disrupted during processing by a disrupted machine and it does not need (needs) to re-start after the machine becomes available again, it is called the resumable (non-resumable) case (see [12]). Assume that the last job started before but not completed at r is job J_j. In the non-resumable case, J_j needs to re-start at times $r + \max\{\gamma, \kappa\}$ and $r + \gamma$ for Case 1 and Case 2, respectively, under scenario $\gamma \ge 1$, while in the resumable case, processing of the remaining part of J_j will continue at times $r + \max\{\gamma, \kappa\}$ and $r + \gamma$ for Case 1 and Case 2, respectively, without any penalty.

We assume throughout the paper that p_j, r, s, and κ are known positive integers such that $\kappa \le s$ and a is chosen such that ap_j is a positive integer for all $j = 1, \ldots, n$. The objective is to determine an optimal schedule to minimize the expected total completion time of the jobs, i.e., $E(\sum_{j=1}^{n} C_j)$, where C_j denotes the completion time of job J_j in a given sequence.