Solving Scheduling Deteriorating Jobs with Rate Modifying Activity

by

Yücel Yılmaz Őztűrkoğlu

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
May 9, 2011

Keywords: scheduling, rate-modifying activity-deteriorating jobs,
single machine

Approved by

Robert L. Bulfin, Chair, Professor Emeritus of Industrial and System Engineering
Jorge Valenzuela, Associate Professor of Industrial and System Engineering
Emmett J. Lodree, Assistant Professor of Industrial and System Engineering

Abstract

Traditional scheduling problems assume that the processing time of a given job is always fixed. However, the processing times may change in real industrial applicants. To reflect the real world, we use two new phenomenons in this study. The first one is known as the deterioration job where the job processing times are defined by functions of their starting times and positions in the sequence. And, the other one is rate-modifying-activity is an activity which affects and changes the production rate of the machines. This study aims to determine the work sequence, the number of breaks and their positions within the work sequence while considering the deterioration jobs and rate-modifying-activities.

In this dissertation, three different mathematical models are introduced to solve the scheduling problems. The first model addresses a classical single machine problem with makespan and total flow time objectives to consider deteriorating jobs and rate-modifying-activity. The second model introduces a single worker scheduling problem which considers workers' physiological factors while determining the break time. The third model introduces a bi-criteria scheduling model with deteriorating jobs and rate-modifying-activities on a single machine. Also, several different heuristic algorithms are developed for large-sized problems. For each mathematical model and heuristic, independent experiments are performed to analyze the effectiveness of these algorithms.

Acknowledgments

First and foremost I would like to thank Dr. Robert L. Bulfin for his guidance, support endless patience and being my father in Auburn. Without him and his continuous guidance, I never would have seen the end of this study. Special thanks go to Dr. Jorge Valenzuela and Dr. Emmett J. Lodree for their valuable suggestions and opinions. Also, I would like to thank Dr. Alice E. Smith for her support throughout my doctoral study.

My sincere appreciation is also extended to all my friends who made my years in Auburn memorable. I would like to thank to my parents, Dr. Zeki Yilmaz and Dr. Canan Yilmaz for their unlimited love and continued support; to my brother Dr. Yetkin Z. Yilmaz and my husband Omer Ozturkoglu. Finally, my thanks go to my biggest motivator to complete this study, my little princess Duru.

I dedicate this dissertation to the memory of my uncle Dr. Cihangir Ciftci and my grandma, Hale A. Ciftci, who truly believed in me.

Table of Contents

List of Tables

Chapter 2

Chapter 3

Chapter 4

CHAPTER 1

INTRODUCTION

1. Background

The competition between companies in the same industry has become vital due to the recession in the national and the international economy. Nowadays, the cost of production and the response to the customers’ requirements are becoming more important on a daily basis for many of the industries to be successful in the market place. So, this leads them to redesign their processes and jobs. Hence, reducing the completion time of the product becomes one of the most important factors in job design. In this study, to be able to reduce the completion time of the products, we focus on the sequencing and scheduling of jobs, as well as some benefit activities.

In both manufacturing and service industries, jobs might have variable processing times. Delaying a job may result in additional time to complete it, such as cleaning dirty dishes the longer they wait, the harder they are to clean. These kinds of jobs are known as *deteriorating jobs* in the literature; if processed later they take more time than when processed earlier. In other words, deteriorating jobs are tasks which need more time and effort to complete the process than when they are done earlier. Gupta and Gupta (1988), and Browne and Yechiali (1990) initiated studies on scheduling deteriorating jobs. Some examples of deteriorating jobs are searching for an enemy under growing darkness, treating a patient under worsening health conditions or producing steel under decreasing oven temperature (Mosheiov 1996).

If a job deteriorates, there is an activity called a *rate-modifying activity* (rma) which recovers the lost time of processing for a job because of the deterioration. Lee and Leon (2001) first introduced the rma which changes the production rate of a processor such as machines and workers. In the scheduling literature, an rma is defined as a maintenance or repair activity of a machine. So, when an rma is taken, there is some improvement on processing of that job, i.e., its processing time decreases.

In this study we deal with two similar problems of scheduling a set of deteriorating jobs on a single machine. We propose three different models which reflect real life situations in which the processing time of a job increases or decreases depending on its initial processing time and other activities such as maintenance or repair of a machine, break for workers, etc. In our first model, we schedule deteriorating jobs on a machine and an rma as a maintenance activity. In our second model we schedule tasks processed by a human worker where the rma as a break given to the worker in order to let him/her recover. Since our processor is a worker, we consider some physiological factors which may affect processing of the job and cause deterioration of the job. In the last model, we have bi-criteria objectives to satisfy both customers and managers with real life assumptions. Therefore, managers should consider more than one measure when they try to find the best schedule for their production process.

The general problem we deal with in this study is to determine the best work sequence, the number of rmas and the position of each rma for our specified model assumptions and objectives.

2. Major Contributions

This study differs from existing studies in several ways.

Contribution 1. We develop the first mathematical model to schedule deteriorating jobs and rma to reflect real industrial situations. Another difference of the first model is our processing times. Specifically, we consider nonlinear deterioration which depends on the position. Up until now, all other research papers have considered time-dependent deterioration.

Contribution 2. Another important contribution of this research is that we consider physiological factors of workers when scheduling the jobs. There has been no research which simultaneously considered deteriorating jobs and rma under some physiological constraints. Except for Lodree and Geiger (2010), to find an optimal sequence position for an rma, ergonomic factors have been mostly ignored in scheduling problems. So, this research acts as a bridge which connects scheduling with ergonomic issues. Thus, this study is an important milestone.

Contribution 3. In our second model, we assume that jobs deteriorate because of worker fatigue. We define rate modifying activities as a resting period of workers. All other studies use rma as maintenance or repair activity of a machine. As workers tire, the job processing time increases. In the existing literature, jobs deteriorate because of the waiting time. Hence, as that kind of job waits in the queue to be processed, its processing time increases.

Contribution 4. The literature is rich for job scheduling problems when we consider either deteriorating jobs or rma .But these are not directly related to customer and manager satisfaction at the same time. We propose the first bi-criteria model to consider rma and deteriorating jobs simultaneously.

In our research the following three major questions are addressed to minimize the completion time of all given jobs:

1) How should jobs be scheduled?

2) How many rmas are needed?

3) Where are these rmas in the schedule?

These questions are addressed for the problem with and without physiological factors.

3. Organization of the Dissertation

The rest of the dissertation is organized as follows. In Chapter II, we present a mathematical model for scheduling deteriorating jobs with rma. An extension of the model, which considers physiological factors is discussed in Chapter III. In Chapter IV, we discuss a bi-criteria scheduling model. Conclusion and future work are provided in Chapter V.

References

1. Browne S. and Yechiali U. (1990). Scheduling deteriorating jobs on a single processor. *Operations Research*, 38, 495-498.

2. Gupta, J.N.D. and Gupta, S.K. (1988). Single facility scheduling with nonlinear processing times. *Computers and Industrial Engineering*, 14, 387-393.

3. Lee, C.Y. and Leon, V.J. (2001). Machine scheduling with a rate-modifying activity. *European Journal of Operational Research*, 128, 119-128.

4. Lodree, E. J. and Geiger, C.D. (2010). A note on the optimal sequence position for rate-modifying activity under simple linear deterioration. *European Journal of Operational Research*, 201 (2), 644-648.

5. Mosheiov, G. (1996). Λ-Shaped policies to schedule deteriorating jobs. *Journal of the Operational Research Society*, 47, 1184-1191.

CHAPTER 2

SINGLE MACHINE SCHEDULING WITH DETERIORATING JOBS AND RATE-MODIFYING-ACTIVITIES

Abstract

In this study, we examine the scheduling a set of deteriorating jobs on a single processor. We propose a model which reflects real life situations in which the processing time of jobs change depending on its initial processing time and activities, such as machine maintenance or a break for workers. A rate-modifying-activity (rma) is a maintenance activity given to a machine to restore it to its original state. The general problem we deal with in this study is to determine the job sequence, the number of rmas, and rma positions within the work sequence. We formulate a unique integer program to solve this model for makespan and total completion time objectives. We also propose efficient heuristic algorithms for solving large size problems for both makespan and total completion time objectives. Polynomial algorithms for several special cases are derived.

1. Introduction

In the last four decades, scheduling researchers have primarily concentrated on problems with a standard set of assumptions. One of these assumptions is that processing times of the jobs are constant. But in reality, processing times may change due to various factors such as deterioration and wear phenomena.

A deteriorating job can be defined as a job which takes more time when processed later than when processed earlier. In our study, an increase in processing times is caused by machine deterioration. After a while, the machine needs more effort to accomplish the task, this is described as the deterioration rate of jobs.

Constant speed of machines or fixed processing times can be changed by *rate-modifying activities* (rma). The rma was first introduced by Lee and Leon (2001). The processing times of the jobs vary depending on whether a job is scheduled before or after the rma because the rma lets the machine or worker recover. After machines are maintained, they tend to have different speeds than before. In our study we define an rma as a maintenance activity during which the machine stops for a given period of time. After an rma is completed, the capability of the machine is expected to return to normal.

In our study, the scheduling model we propose includes both deteriorating jobs and rma. More specifically, we develop a mathematical model to determine job sequences and placement of breaks under makespan and total completion time. We follow the three–field notation introduced by Graham et al. (1979) to describe scheduling problems. This notation is $\alpha \mid \beta \mid \gamma$, where α denotes the worker/machine condition, β indicates the characteristics of the problem and γ shows the performance measure. Hence, we study $1 \mid p_{ij} = (1+\alpha_j)^{i-1} p_j, rm \mid \sum_{i=1}^{n} C_i$ and $1 \mid p_{ij} = (1+\alpha_j)^{i-1} p_j, rm \mid C_{\max}$. In this notation p_{ij}, α, rm and C represent actual processing time, deterioration rate, rma and completion time of jobs respectively.

The remainder of this chapter is organized in eight sections. In Section 2, a literature review is given. The problem definition is in Section 3. A mathematical model is presented in Section 4. In Sections 5, an algorithm for makespan is presented. Heuristics for both objectives are given in Section 6. Computational results are given in the Section 7. Finally, the conclusions are presented in the last section.

2. Literature Review

Classical machine scheduling problems have been widely studied by many researchers. Recently, researchers have started to give more attention to scheduling problems with different characteristics including deteriorating jobs, learning effects or rate-modifying activities. Makespan, total completion time, total weighted completion time, maximum lateness, maximum tardiness and number of tardy jobs are the most commonly studied performance measures.

Scheduling deteriorating jobs was first introduced by Gupta and Gupta (1988), and Browne and Yechiali (1990). Gupta and Gupta (1988) introduced a scheduling model with variable processing time of a job which is a polynomial function of its initial processing time. Then Browne and Yechiali (1990) mentioned deteriorating jobs; their processing times increase while they await service. They considered that a processor loses its efficiency with a certain rate as soon as it finishes its operation. In their model, all jobs are available at the beginning with their initial processing times. If the processing of a job is delayed, then the required time to process that job increases linearly based on its initial processing time. They constructed a scheduling problem to minimize the makespan for n jobs on a single machine. Mosheiov (1991) considered the problem of minimizing total completion time of jobs with different deteriorating rates and found that the optimal sequence of this problem is V-shaped. V-shaped scheduling indicates that “jobs are arranged in descending order of growth rate if they are placed before the minimal growth rate job, and in ascending order if placed after it.” (Mosheiov (1991). Cai et al. (1998) developed a fully polynomial time approximation scheme to minimize makespan for deteriorating jobs. Also Kubiak and Vende (1998) investigated the computational complexity of makespan under deterioration. They developed a heuristic and branch-and-bound algorithm for the problem. Kovalyov and Kubiak (1998) presented a fully polynomial approximation scheme for a single machine scheduling problem to minimize makespan of deteriorating jobs. Cheng and Ding (2000) studied a single machine to minimize makespan with deadlines and increasing rates of processing times. They found that both problems are solvable by a dynamic programming algorithm. Bachman and Janiak (2000) considered a single machine scheduling problem minimizing maximum lateness under linear deterioration. They presented two heuristics and proved that the maximum lateness problem is $NP-hard$. Bachman et al. (2002) showed that total weighted completion time is $NP-hard$ for single machine scheduling in which the job processing times are decreasing linear functions dependent on their start times. These models all have the processing time of a job as a function of its start time. None of these works are valid for position based deterioration.

Rma is a phenomenon in scheduling problems appearing in the last decade. In the scheduling literature, rma is defined as a maintenance or repair activity which improves the condition of the machine. Qi et al (1997) considered a problem where multiple maintenance activities need to be scheduled with jobs on a single machine. Also Lee and Chen (2000) scheduled jobs and maintenance activities to minimize total completion time on a set of jobs on parallel machines. They proposed branch-and bound algorithms for solving medium sized problems. Lee and Leon (2001) introduced a different perspective of scheduling maintenance activities. They studied a scheduling problem with maintenance activities which is commonly found in electronic assembly lines. One of the main decisions their model addresses is whether to stop the machine and fix the problem or to let it work with a lower production rate. Hence, processing times of jobs may change after a maintenance activity. They also studied various performance measures including makespan, total completion time, total weighted completion time and maximum lateness. They developed polynomial algorithms for solving problems of minimizing both makespan and total completion time. In addition, they developed pseudo-polynomial algorithms to solve the total weighted completion time problem. They used the start time of the maintenance activity as a decision variable in their model. However, their model does not include the possibility of machine breakdowns. Lee and Lin (2001) studied single machine scheduling problems involving repair and maintenance activities which they also called rma. They focused on two types of processing cases which are resumable and nonresumable. Their objective functions sought to minimize the expected makespan, total expected completion time, maximum expected lateness, and expected maximum lateness respectively. If they decide not to do maintenance activities for the problem of minimizing the expected makespan and the total expected completion time, they obtained these interesting results: i) when the cumulative distribution function of x, F(x), which indicates that machine breaks down if there is no maintenance activity, is concave then the sequence of the jobs is in SPT (shortest processing time) order; ii) if F(x) is convex, then the sequence of the jobs is in LPT (largest processing time) order.

He et al. (2005) studied a single machine to minimize makespan and total completion time of jobs. They assumed that an rma is not always valid because an activity needs some additional resources such as operators and equipment. These resources may not be available all the time. Thus, they considered the problem with a restricted rma. If the rma must be performed, they called it mandatory (man.); otherwise, it is called optional (opt.). They analyzed the computational complexity of both makespan and total completion time. Table 1 presents their complexity results.

Table 1. Complexity Results of He et al. (2005)

$1/rm,opt/C_{\max}$	$NP-hard$ if all $\alpha_i < 1$	Pseudo-polynomially solvable in $O(nms_m)$ time
$1/rm,man/C_{\max}$	$NP-hard$ if all $\alpha_i < 1$	FPTAS running in $O(mn^2/\varepsilon)$
$1/rm,opt/\sum C_i$	$NP-hard$ if all $\alpha_i \le 1$, Open if $\alpha_i < 1$	Open in general and pseudo-polynomially solvable for the agreeable rate case in $O(nms_m)$ time
$1/rm,man/\sum C_i$	$NP-hard$ if all $\alpha_i \le 1$, Open if $\alpha_i < 1$	Open in general and pseudo-polynomially solvable for the agreeable rate case in $O(nms_m)$ time

To minimize makespan, they presented a pseudo-polynomial time algorithm and a fully polynomial time approximation scheme (FPTAS). To minimize the total completion time, they proposed a pseudo-polynomial algorithm as a special case. When they fixed the start times for the rma, availability constraints can be applied.

There has been little research that simultaneously considered time-dependent processing times and rma. Lodree and Geiger (2010) integrated time dependent processing times and rma for assigning a single rma to a position. They showed that a single rma should be inserted in the middle of the optimal job sequence to minimize makespan.

To the best of our knowledge, except for the recent study of Lodree and Geiger (2010), the scheduling problem with the effects of deterioration and rma has not been studied in the literature. We also differ in that our deterioration depends on job position rather than start time.

3. Problem Definition

The problem we study in this paper is to schedule a set of n independent jobs $J = \{J_1, J_2, ..., J_n\}$ and one or more rma's for a single machine. All jobs are available for processing at all times. The machine (worker) can do only one job at a time. Each job has a deterioration rate α which reflects a delay time (worker's fatigue) from processing jobs. We assume that the deterioration rate α has the same effect on processing times of different jobs and it changes the processing time of the job nonlinearly based on its position.

Let us define model parameters and variables as follows:

Model Parameters:

n is the number of jobs to be sequenced

i indicates the position number which is from 1 to *n*

k indicates the position number which is from 0 to *n* (k=0 is initial position)

j indicates the job number which is from 1 to *n*

α $0 < \alpha \leq 1$, is the constant deterioration rate of jobs when delayed by one position.

q is the fixed period of time to perform an rma.

p_j is the initial processing time of job j before deterioration.

p_{ji} is the processing time of job *j* if done *i* positions after an rma or the initial position, i.e.

$$p_{ji} = (1+\alpha)^{i-1} p_j \tag{3.1}$$

Model Variables:

$$x_{ijk} = \begin{cases} 1 & \text{if job j is in the ith position after an rma which is done just before position } k+1, \\ 0 & \text{otherwise} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if an rma is done before position i,} \\ 0 & \text{otherwise zero} \end{cases}$$

C_i Completion time of the job in position *i*.

$C_{\max}$ Completion time of the job in the last position.

In addition, our model assumptions are given as the following:

- There is only one machine (worker).
- The deterioration of a job depends on its position.
- Jobs are non-preemptive.
- After an rma, jobs revert to their initial/base processing time p_j. That means the machine (worker) recovers completely after an rma (100% recovery).
- Deterioration process is the same after an rma.

In our research the following three questions are addressed to minimize the completion time or makespan of all given jobs:

1) How should jobs be scheduled?

2) How many rmas are needed?

3) Where are rmas in the schedule?

4. The Mathematical Model

As mentioned before, we consider two performance measures: total completion time and makespan. Hence, independently our objective function is to minimize each of those performance measures in our models.

Minimize $\quad Z = \sum_{i=1}^{n} C_i \qquad i = 1,..,n \qquad$ (total completion time)

or

Minimize $\quad Z = C_{\max}, \quad C_{\max} \geq C_i \qquad i = 1,..,n \qquad$ (makespan)

The related constraints with our model are given as follows.

$$C_1 = \sum_{j=1}^{n} p_{j1} x_{1j0} \tag{4.1}$$

$$C_i = C_{i-1} + \sum_{k=1}^{i} \sum_{j=1}^{n} p_{jk} x_{i,j,i-k} + q\, y_i \qquad i = 2,.....,n \tag{4.2}$$

$$\sum_{i=1}^{n} \sum_{k=0}^{i-1} x_{ijk} = 1 \qquad j = 1,.....,n \tag{4.3}$$

$$\sum_{k=0}^{i-1} \sum_{j=1}^{n} x_{ijk} = 1 \qquad i = 1,.....,n \tag{4.4}$$

$$x_{kji} \leq y_{i+1} \qquad k = 2.....,n \qquad j = 1,.....,n \qquad i = 1,.....,k-1 \tag{4.5}$$

$$x_{ijk} \in \{0,1\} \qquad i = 1,.....,n \quad j = 1,.....,n \qquad k = 0,.....,n \tag{4.6}$$

$$y_k \in \{0,1\} \qquad k = 2,.....,n \tag{4.7}$$

$$C_i \geq 0 \qquad i = 1,.....,n \tag{4.8}$$

In constraint (4.1), the completion time of the job in position $i = 1$ is equal to the processing time of the job assigned to position $i = 1$. Before the first position, there is no rma ($y_1 = 0$). In constraint (4.2), the completion time of the job in position i is equal to the completion time of the job in position $i - 1$ plus the processing time of the job assigned to position i plus the rma time if assigned. In constraint (4.3), each job is assigned to exactly one position. In constraint (4.4), each position is scheduled for only one job. Constraint (4.5) requires an rma to be done in the related position if jobs are scheduled after rma and to control the sequence of the rma. Constraints (4.6), (4.7) and (4.8) show that the variables should be binary and non-negative.

5. Fundamental Properties and Special Cases

In this section, we develop some fundamental properties and develop polynomial algorithms for the unit processing time problem. First, we examine the problem without rmas in Theorem 1 and then develop the result with a rma in Theorem 2.

Theorem 1. For $1/\, p_{ij} = (1+\alpha)^{i-1} p_j \,/ C_{\max}$, LPT sequencing minimizes the makespan**.**

Proof: Suppose schedule S minimizes makespan and is not in LPT order. Then there must be a pair of jobs in S , say job i and job j, with job i immediately after job j in the k^{th} and $k+1^{st}$ positions, and $p_i < p_j$. Let B be the set of jobs before job i, and A the set of jobs after job. Let $C(A)$ and $C(B)$ be the sum, of processing times in sets A and B respectively. Now consider the schedule $S^{'}$, where $S^{'}$ the same as S except job i and j have been interchanged and the sets of jobs A and B are in the same position in both schedules. $S=(\pi,i,j,\pi^{'})$ and $S^{'}=(\pi,j,i,\pi^{'})$ where π and $\pi^{'}$ denote partial sequences.

The makespan for S is;

$$\begin{aligned} C(S) &= C(B)+p_{(n-1)i}+p_{nj}+C(A) \\ &= C(B)+(1+\alpha)^{n-2}p_i+(1+\alpha)^{n-1}p_j+C(A) \qquad (5.1) \end{aligned}$$

and the makespan for $S^{'}$ is;

$$\begin{aligned} C(S^{'}) &= C(B)+p_{(n-1)j}+p_{ni}+C(A) \\ &= C(B)+(1+\alpha)^{n-2}p_j+(1+\alpha)^{n-1}p_i+C(A) \qquad (5.2) \end{aligned}$$

Subtracting equation (5.1) from (5.2) we get

$$C(S)-C(S^{'})=\alpha(1+\alpha)^{n-2}(p_j-p_i)>0$$

This implies the makespan of $S^{'}$ is smaller than S, which contradicts the assumption that S was optimal. Therefore, an optimal solution must be in LPT.

Theorem 2. For $1/\ p_{ij}=(1+\alpha)^{i-1}p_j, rm\ /\ C_{\max}$ the sequence of the jobs in an optimal solution is always in LPT order between any given pair of rmas.

Proof: Let $C(B)$ be the total completion time of the jobs before given rma and $C(A)$ be the total completion time of the jobs after given second rma. Now consider the schedule $S = \{\pi, rma, i, j, k, rma, \pi^{'}\}$ (SPT order) and $S^{'} = (\pi, rma, k, j, i, rma, \pi^{'})$ (LPT order) where π and $\pi^{'}$ denote partial sequences. The sets of jobs A and B are in the same position in both schedules. We assume that schedule S is an optimal schedule. $S^{'} = (\pi, rma, i, j, k, rma, \pi^{'})$. Let us assume that, $p_i < p_j < p_k$ and after rma the first job assigned is in the $(n-1)^{th}$ position. The m is defined as a duration of maintenance activity (rma).

The makespan for S is;

$$\begin{aligned}
&C_{\max}(S_{n-1}) = C(B) + m + p_i \\
&C_{\max}(S_n) = C(B) + m + p_i + p_j(1+\alpha)^{n-1} \\
&C_{\max}(S_{n+1}) = C(B) + m + p_i + p_j(1+\alpha)^{n-1} + p_k(1+\alpha)^n \\
&C_{\max}(S_{n+2}) = C(B) + m + p_i + p_j(1+\alpha)^{n-1} + p_k(1+\alpha)^n + m + C(A) \qquad (5.3)
\end{aligned}$$

and the makespan for $S^{'}$ is;

$$\begin{aligned}
&C_{\max}(S_{n-1}^{'}) = C(B) + m + p_k \\
&C_{\max}(S_n^{'}) = C(B) + m + p_k + p_j(1+\alpha)^{n-1} \\
&C_{\max}(S_{n+1}^{'}) = C(B) + m + p_k + p_j(1+\alpha)^{n-1} + p_i(1+\alpha)^n \\
&C_{\max}(S_{n+2}^{'}) = C(B) + m + p_k + p_j(1+\alpha)^{n-1} + p_i(1+\alpha)^n + m + C(A) \qquad (5.4)
\end{aligned}$$

Subtracting, equation (5.3) to (5.4) we get;

$$C_{\max}(S) - C_{\max}(S^{'}) = \left|(1+\alpha)^n - 1\right|(p_k - p_i) > 0$$

The schedule $S^{'}$ is better than schedule S which contradicts the assumption that S was optimal. This means that between given two rma, the schedule is always LPT order, like schedule $S^{'} = \{\pi, k, j, i, \pi^{'}\}$.

5.1. A Polynomial Algorithms for Makespan with Identical Processing Times

In this section, we give a polynomial algorithm for $1/\, p_{ij} = (1+\alpha)^{i-1} p, rm \;/ C_{\max}$ which is a special case with $p_j = p$. Suppose we have a given number of rmas, say m. Then any schedule can be divided into $m+1$ groups of jobs scheduled between rmas and the beginning and of the schedule. Let the rmas be scheduled before positions $k_1, k_2, ..., k_m$.

Let $d = \left\lfloor \frac{n}{m+1} \right\rfloor + r$ and $k_1 = d+1$,

$$k_i = \begin{cases} k_{i-1} + d & i = 2, ..., m-r \\ k_{i-1} + d + 1 & i = m-r+1, ..., m \end{cases}$$

A schedule with the m rmas before positions k_i, $i = 1, ..., m$ is called a *balanced schedule.* This implies the number of jobs in each of the $m+1$ groups is as equal as possible, either having d or $d+1$ jobs. If $r = 0$, each group has exactly d jobs and we say the schedule is *perfectly balanced.*

Theorem 3. Balanced schedules are optimal for $1/\, p_{ij} = (1+\alpha)^{i-1} p, | rm | = m \;/ C_{\max}$.

Proof: We will assume an unbalanced schedule is optimal and show a contradiction. The makespan for a schedule with rmas at $k_1, k_2, ..., k_m$ is;

$$C_{\max} = \sum_{i=1}^{k_1 - 1} p(1+\alpha)^{i-1} + q + \sum_{i=1}^{k_2 - k_1 - 1} p(1+\alpha)^{i-1} + ... + q + \sum_{i=1}^{k_m - k_{m-1} - 1} p(1+\alpha)^{i-1}$$

A balanced schedule has $m+1-r$ groups with d jobs and r groups with d+ 1 jobs. An unbalanced schedule must either have more than r groups with d+ 1 jobs or at least one group with d+2 jobs. Assume an unbalanced schedule of the first type has minimal makespan. Without loss of generality; let the group before k_i consist of d-1 jobs and the group after have d+ 1 jobs. The contribution to makespan of these two groups will be;

$$\begin{aligned} M(S) &= \sum_{i=1}^{d-1} p(1+\alpha)^{i-1} + \sum_{i=1}^{d+1} p(1+\alpha)^{i-1} \\ &= p + p(1+\alpha) + ... + p(1+\alpha)^{d-2} + p + p(1+\alpha) + ... + p(1+\alpha)^{d} \end{aligned}$$

Now let $S^{'}$ be an identical schedule to S except one job between k_{i+1} and k_{i+2} is moved between k_{i-1} and k_i. Then

$$\begin{aligned} M(S^{'}) &= \sum_{i=1}^{d} p(1+\alpha)^{i-1} + \sum_{i=1}^{d} p(1+\alpha)^{i-1} \\ &= p + p(1+\alpha) + ... + p(1+\alpha)^{d-1} + p + p(1+\alpha) + ... + p(1+\alpha)^{d-1} \end{aligned}$$

and

$$M(S) - M(S^{'}) = p(1+\alpha)^{d} - (1+\alpha)^{d-1} > 0,$$

So S cannot be optimal. The other case can be shown non-optimal in the same way.

To solve $1/\, p_{ij} = (1+\alpha)^{i-1} p, |\, rm \,| = m \;/ C_{\max}$, we create a balanced schedule for each possible value of m. This is done in Algorithm 1.

Algorithm 1

```
Step 0. Set $m = 0$, $C_{\max} = \infty$.
Step 1. Create a balanced schedule with $m$ rmas and optimal makespan $C_{\max}(m)$
        If $C_{\max}(m) < C_{\max}$, $C_{\max} \leftarrow C_{\max}(m)$
        If $m = n$, stop
Step 2. Set $m = m + 1$, and go to Step1.
```

Step 1 requires at most $O(n)$ operations and it will be repeated at most n times, so the algorithm is $O(n^2)$.

6. The Heuristic Algorithms

Although our mathematical model, which we discussed in the previous section, can solve some problems in a reasonable run time, larger problems (e.g. n>50) are difficult to solve without considerable computational effort. Therefore, we propose heuristic algorithms to solve large problems.

6.1. Heuristic for Makespan

For problems with a large number of jobs, Heuristic 1 provides a solution very close to the optimal solutions.

Let us define;

b Job with largest processing time within the set of unassigned jobs.

s Job with smallest processing time within the set of unassigned jobs.

m Potential position number after rma or initial position.

$J_{[i]}$ Job in position i

$$y_i = \begin{cases} 1 & \text{if an rma is placed before position i} \\ 0 & \text{otherwise} \end{cases}$$

p_{ij} Processing time of job j in position i as $p_{ij} = (1+\alpha)^{k-1} p_j$

C_i Completion time of the job in position i

The proposed heuristic first applies the LPT rule to sort the jobs. The job which has the largest processing time is assigned to position 1. Then calculate the incremental amount of processing time of the first job which has smallest processing time. If this amount is greater than the rma time (q), then do an rma and assign the next largest job to the current position immediately after the rma. Otherwise do not do an rma and assign the smallest job to the current position without doing an rma.

The procedure for the proposed heuristic for makespan problem is stated as follows:

Heuristic 1

Step 1. Order the jobs in descending order (LPT) of processing times $p_{j.}$

set $b = \max\ p_j \quad j = 1,..,n$

Step 2. Assign the job with largest processing time to position $i = 1$ and no rma at the beginning

$y_1 = 0$, $b = n$, $s = 1$

Set $C_1 = p_{1n}$, $J_{[1]} = b$

$b = b - 1$, $s = 1$ and $k = 2$.

Step 3. Calculate;

$D = p_{s[k]} - p_{s[k-1]}$

If $D < q$ go to Step 4

$y_i = 1$, $J_{[k]} = b$

Set $C_i = C_{i-1} + p_{ib} + q$

$k = k + 1$, $b = b - 1$

Go to Step 5.

Step 4. $y_i = 0$, $J_{[k]} = s$

Set $C_i = C_{i-1} + p_{is}$,

$s = s+1$, $m = m+1$ and $k = k+1$

Step 5. If $k = n$, Stop the algorithm, Makespan $= C_{[n]}$,otherwise go to Step 3.

This algorithm is $O(n \log n)$. It is well known that Step 1, can be completed in $O(n \log n)$. In Step 2, there is only one iteration and this iteration can be completed in $O(1)$. There are at most n-1 iterations in step 3 and 4 and go through once for each job. Each iteration can be completed in $O(n-1)$. Lastly, Step 5 can be completed in $O(n)$.

6.2. Heuristic for Total Completion Time

The heuristic is developed for the problem of total completion time with given multiple rma's.

In the proposed heuristic, let J be a set of n jobs, $J = \{J_1, J_2, ...J_n\}$ and R be a set of m rmas', $R = \{R_1, R_2, ...R_m\}$ represents the position of the last job before the rma. Let S represent a schedule of n jobs and m rmas on a single machine, $S = \{(J_1, J_2, ...J_n)(R_1, R_2, ...R_m)\}$. CT_{m+1} is the total processing time of the assigned jobs in the part m and assume $\{CT_1, CT_2, ..., CT_{m+1}\}$ is the separate total processing of jobs in each part. The steps of the developed heuristic are given below:

Heuristic 2

Step 1: Split S into $m+1$parts so that each part includes a set of jobs before each given consecutive rma. $S=\{S_1,S_2,....S_{m+1}\}$. And assume $\min\{CT_1,CT_2,.....,CT_{m+1}\}$ is the separate total completion of jobs in each part.

Step 2: Order jobs based on LPT rule.

Step 3: Assign one job, which has the largest processing time in the unscheduled list of jobs, into each part.

$$S=\left[\{J_1\},\{J_2\},\{J_{m+1}\}\right]$$

Step 4: Calculate total completion time of the each part.

$$CT_1 = p_1,\ CT_2 = p_2,\ CT_{m+1} = p_{m+1}$$

Step 5: To assign the other jobs in each part, first select $\min\{CT_1,CT_2,.....,CT_m\}$ and then start to assign the next job to $\min\{CT\}$.

$$S=\left[\{J_1,J_{m+4}\},\{J_2,J_{m+3}\},\{J_{m+1},J_{m+2}\}\right]$$

And calculate new total completion time of the each part with adding deteriorated processing times.

$$CT_1 = p_1 + p_{i,m+4},\ CT_2 = p_2 + p_{i,m+3},\ CT_{m+1} = p_{m+1} + p_{i,m+2}$$

Step 6: Repeat Step 5 until all jobs are assigned. Hence, the positions of the rma are naturally determined based on the scheduled jobs in each part. The best schedule is the S^*, which gives the smallest flowtime.

7. Computational Results

In this section, we conduct three experiments to analyze the effectiveness of our model. Experiment 1 focuses on the computational time to solve the proposed mathematical model for minimizing makespan and total completion time. In Experiment 2, we compare the computational effectiveness of all of the heuristic algorithms with the proposed mathematical model. In Experiment 3, an experimental design is built to estimate the relationship between parameters of the problem.

To conduct our analyses on the proposed models, we identify four experimental factors: deterioration rate (α), RMA time (*q*), mean processing time (*M*) and variance of processing time (*V*). Also, we specify three levels for each factor. So this is a 3^4 experimental design with six two-factor, four three-factor and one four-factor interactions. The defined levels of those factors in the experiments are:

1) α : 0.02, 0.04 and 0.08
2) q : 5, 10 and 15
3) M : 20, 40 and 80
4) V : 0.20, 0.40 and 0.80

Using the variance and mean of processing time, we produced an interval for each combination; [18, 22], [10, 30], [1, 40], [36-44], [20-60], [1-80], [72-88], [40-120] and [1-160]. We tested our models for 50 jobs, and 810 instances (10 replications) for each experiment.

Experiment 1: Performance of the Mathematical Model

The proposed mathematical model is coded using AMPL and solved by CPLEX 9.1 on a computer with a Pentium IV 2.8 GHz processor and 1GB of RAM. Ten replications of each of the combination were run for each performance measure. The average computational time (in seconds) of each of the combination for each objective is given in Table 2 and Table 3, respectively.

Table 2. Average Run Time (sec.) for Total Completion Time

Jobs numb. (n)	Det. Rate (α)	Rma time (q)	Ave. Time (sec.)
25	0.02	5	9.2
	0.02	10	16.1
	0.02	15	29.7
	0.04	5	14.5
	0.04	10	20.1
	0.04	15	20.2
	0.08	5	9.9
	0.08	10	20.2
	0.08	15	21.2
50	0.02	5	58.3
	0.02	10	91.5
	0.02	15	119.5
	0.04	5	28.2
	0.04	10	56.6
	0.04	15	61.1
	0.08	5	11.6
	0.08	10	32.7
	0.08	15	45.1

Table 3. Average Run Time (sec.) for Makespan

Jobs numb. (n)	Det. Rate (α)	Rma time (q)	Ave. Time (sec.)
25	0.02	5	11.6
	0.02	10	12.9
	0.02	15	18.4
	0.04	5	15.6
	0.04	10	22.7
	0.04	15	23.1
	0.08	5	12.6
	0.08	10	19.8
	0.08	15	22.4
50	0.02	5	78
	0.02	10	109.4
	0.02	15	101.9
	0.04	5	51.2
	0.04	10	83.6
	0.04	15	93.8
	0.08	5	25.5
	0.08	10	51.8
	0.08	15	66.8

The efficacy of the model is based on the average run time in seconds. The average time for 50 jobs for total completion time is 56.06 seconds and for the makespan it is 73.55 seconds. As seen in the tables, as the rma time increases, run time for both models increase. We expect this result because increasing the rma time means fewer rmas and jobs deteriorate more. For example, if the rma time is very small compared to deterioration, we expect the model to have many rmas. When the deterioration rate increases with the fixed rma time, run time for both models decreases for the same reason. Also, the problem of minimizing total completion time requires less time than that of makespan.

Experiment 2: Performance of the Heuristic Algorithms

The input parameters are the same as in Experiment 1. The heuristic algorithms were coded in Java. The solution quality percentage error is calculated by using the following equation:

$$e = \left\lfloor \left(F_h - F_{opt} \right) / F_{opt} \right\rfloor * 100$$

In this equation, F_h is the measure of the heuristic model and F_{opt} is the measure of the optimal solution obtained by the mathematical model. When we tested our models for 10 replications for each specific set of conditions, we obtained the average percentage errors of the heuristic models, which are given in the Table 4.

Table 4. Comparison of Heuristic 1 and Mathematical Model for Makespan

			Ave. Error of Heuristic 1 (%)		
Mean	Ranges of Process Time	α	RMA time		
			5	10	15
20	[18,22]	0.02	2.06	3.70	4.90
		0.04	0.86	4.06	4.86
		0.08	0.54	0.59	0.84
	[10,30]	0.02	1.35	2.01	5.59
		0.04	1.96	4.36	4.72
		0.08	0.31	0.43	2.95
	[1,40]	0.02	2.60	4.28	4.91
		0.04	1.77	5.47	5.78
		0.08	0.19	0.45	2.14
40	[36,44]	0.02	1.40	2.93	3.20
		0.04	0.06	0.92	3.53
		0.08	1.20	1.11	2.98
	[20,60]	0.02	1.68	4.39	5.85
		0.04	0.01	1.30	3.51
		0.08	0.42	1.61	3.57
	[1,80]	0.02	1.71	4.07	5.28
		0.04	1.04	2.66	3.03
		0.08	0.39	2.36	4.58
80	[72,88]	0.02	0.42	1.28	2.60
		0.04	0.06	0.65	1.61
		0.08	0.01	0.01	0.81
	[40,120]	0.02	0.59	1.71	2.92
		0.04	0.05	0.97	1.90
		0.08	0.07	0.04	1.02
	[1,160]	0.02	0.7	1.84	3.01
		0.04	0.54	1.01	2.28
		0.08	0.43	0.64	1.55

The average percentage error of all 810 instances is calculated as 2.06 % with the worst instance 5.85% for makespan. We also ran Heuristic 1 for the total completion time objective.

Table 5. Comparison of Heuristic 1 and Mathematical Model for Total Completion Time

			Ave. Error of Heuristic 1 (%)		
Mean	Ranges of Process Time	α	RMA time		
			5	10	15
20	[18,22]	0.02	2.06	3.70	4.90
		0.04	0.86	4.06	4.86
		0.08	0.54	0.59	0.84
	[10,30]	0.02	1.35	2.01	5.59
		0.04	1.96	4.36	4.72
		0.08	0.31	0.43	2.95
	[1,40]	0.02	2.60	4.28	4.91
		0.04	1.77	5.47	5.78
		0.08	0.19	0.45	2.14
40	[36,44]	0.02	1.40	2.93	3.20
		0.04	0.06	0.92	3.53
		0.08	1.20	1.11	2.98
	[20,60]	0.02	1.68	4.39	5.85
		0.04	0.01	1.30	3.51
		0.08	0.42	1.61	3.57
	[1,80]	0.02	1.71	4.07	5.28
		0.04	1.04	2.66	3.03
		0.08	0.39	2.36	4.58
80	[72,88]	0.02	0.42	1.28	2.60
		0.04	0.06	0.65	1.61
		0.08	0.01	0.01	0.81
	[40,120]	0.02	0.59	1.71	2.92
		0.04	0.05	0.97	1.90
		0.08	0.07	0.04	1.02
	[1,160]	0.02	0.7	1.84	3.01
		0.04	0.54	1.01	2.28
		0.08	0.43	0.64	1.55

The average percentage error of all 810 instances is calculated as 1.85 % with the worst instance 4.39% for total completion time. If the deterioration rate is fixed and the rma duration increases, average percentage error increases. When the rma time increases, scheduling the jobs based on the order results in a larger error than scheduling the job by investigating each job separately. On the other hand, if the duration of the rma is fixed and the deterioration rate rises, the average percentage error decreases. In this case, jobs are deteriorating more because of the increasing deterioration rate with fixed rma time. The difference between the required additional time due to the deteriorated job and the rma time converges. Hence, the position of the rma becomes less critical. Additionally, when the variation of the mean of processing time increases, the average percentage error of all instances in that group increases.

Table 6 and Table 7 shows the comparison of the computational times for both the proposed models for makespan and total completion time objectives. By comparing the heuristic model with the mathematical model, it is clear that the heuristic model requires much less computation time than the proposed mathematical model.

Table 6. Comparison of Run Times of Heuristic 1 for Makespan with 50 Jobs

		rma time=10	
Ranges of Process Time	A	Mathematical Model (sec.)	Heuristic Model (sec.)
[18,22]	0.02	93	0.05
	0.04	82.13	0.03
	0.08	43.15	0.02
[36,44]	0.02	88.53	0.04
	0.04	61.43	0.03
	0.08	48.2	0.04
[72,88]	0.02	72.6	0.03
	0.04	41.8	0.02
	0.08	17.3	0.02
[10,30]	0.02	98	0.03
	0.04	113.13	0.03
	0.08	88.26	0.02
[20,60]	0.02	100.72	0.05
	0.04	95.4	0.04
	0.08	45.13	0.03
[40,120]	0.02	86.23	0.04
	0.04	43.03	0.03
	0.08	16.56	0.02
[1,40]	0.02	129.5	0.05
	0.04	121.7	0.04
	0.08	104.83	0.04
[1,80]	0.02	123.6	0.08
	0.04	80.96	0.03
	0.08	40.23	0.03
[1,160]	0.02	90.61	0.05
	0.04	34.51	0.05
	0.08	11.12	0.02

Table 7. Comparison of Run Times of Heuristic 1 for Total Comp. Time with 50 Jobs

		rma time=10	
Ranges of Process Time	A	Mathematical Model (sec.)	Heuristic Model (sec.)
[18,22]	0.02	42.11	0.03
	0.04	67.97	0.04
	0.08	40.08	0.03
[36,44]	0.02	59.35	0.04
	0.04	64.17	0.05
	0.08	41	0.03
[72,88]	0.02	58.34	0.03
	0.04	21.44	0.03
	0.08	19.10	0.01
[10,30]	0.02	52	0.04
	0.04	97.61	0.05
	0.08	72.69	0.04
[20,60]	0.02	59.38	0.03
	0.04	89	0.05
	0.08	51.64	0.04
[40,120]	0.02	80.74	0.03
	0.04	21.09	0.03
	0.08	16	0.01
[1,40]	0.02	92.50	0.04
	0.04	101.03	0.04
	0.08	77.12	0.04
[1,80]	0.02	119.45	0.06
	0.04	51.18	0.05
	0.08	46.39	0.03
[1,160]	0.02	71.40	0.04
	0.04	30.18	0.04
	0.08	18.73	0.02

Therefore, as the number of jobs increases, the computational time for the mathematical model increases dramatically in comparison to the heuristic model. For example, we tested both models with 100 jobs with 5 replications. We used a deterioration rate of 0.08, the rma time of 5, and the ranges of processing time is [1,160] with mean 80. Based on the results, the average percentage error is 0.369%. However, while the averages run time for the mathematical model is 2861.66 seconds, the heuristic model is only 46.5 seconds. This shows that the heuristic model has reasonable error with very short computational time for a large number of jobs compared to the mathematical model.

Table 8. Results of Heuristic 2 for Total Completion Time

n = 50			**Ave. Error of Heuristics (%)**
Rma Time	Det. Rate	Num. of Rma	Heuristic2
5	0.02	20	0.62
	0.04	23	0.53
	0.08	24	0.51
10	0.02	19	0.63
	0.04	19	0.78
	0.08	21	0.62
15	0.02	18	0.65
	0.04	20	0.71
	0.08	20	0.88

The results of the average percentage error for minimizing total completion time to Heuristic 2 are presented in Table 8. Heuristic 2 gives very close to optimal solutions while the average percentage error is 0.65%.

Experiment 3: Experimental Design

To estimate the effects of input factors of the model, an experimental design is built. The input parameters are the same as in previous experiments. Table 9 shows the result of the experimental design for the problem of minimizing completion time.

Table 9. Number of Rmas Versus Factors for Total Completion Time Objective

Source	D.F.	SS	MS	F	P	
Det. Rate (α)	2	15500	7750.02	16064.2	0.00<0.05	sig.
RMA Time (q)	2	9015.83	4507.92	9343.96	0.00<0.05	sig.
Mean (M)	2	13148.7	6574.35	13627.2	0.00<0.05	sig.
Variance (V)	2	159.9	79.95	165.72	0.00<0.05	sig.
α *q	4	508.9	127.23	263.71	0.00<0.05	sig.
α *M	4	878.39	219.6	455.18	0.00<0.05	sig.
α *V	4	433.77	108.44	224.78	0.00<0.05	sig.
q *M	4	441	110.25	228.52	0.00<0.05	sig.
q * V	4	311.49	77.87	161.41	0.00<0.05	sig.
M* V	4	368.91	92.23	191.17	0.00<0.05	sig.
α *q*M	8	67.8	8.48	17.57	0.00<0.05	sig.
α *q*V	8	731.8	91.48	189.61	0.00<0.05	sig.
α *M*V	8	918.42	114.8	237.96	0.00<0.05	sig.
q *M*V	8	772.51	96.56	200.16	0.00<0.05	sig.
α *q*M*V.	16	1490.8	93.18	193.13	0.00<0.05	sig.
Error	729	351.7	0.48			
Total	809	45100	R^2 = 99.22%			

Table 9 clearly indicates that 99.22% of the variation in the number of rmas, which is a main variable of the model, is explained by all factors. Table 10 shows that 54.46% of the variation in total run time of the model is explained by all factors. This is the evidenced by the fact that the interaction of some factors is not significantly different.

Table 10. Total Run Time Versus Factors for Total Completion Time Objective

Source	D.F.	SS	MS	F	P	
Det. Rate (α)	2	433002	216501	122.29	0.00<0.05	sig.
RMA Time (q)	2	258126	129063	72.90	0.00<0.05	sig.
Mean (M)	2	488619	244310	138.00	0.00<0.05	sig.
Variance (V)	2	89859	44930	25.38	0.00<0.05	sig.
α *q	4	7338	1834	1.04	0.38<0.05	sig.
α *M	4	5565	1391	0.79	**0.535**	not sig.
α *V	4	12511	3128	1.77	0.13<0.05	sig.
q *M	4	4678	1169	0.66	**0.620**	not sig.
q * V	4	16207	4052	2.29	0.058	sig.
M* V	4	71023	17756	10.03	0.00<0.05	sig.
α *q*M	8	27955	3494	1.97	0.04<0.05	sig.
α *q*V	8	10868	1358	0.77	**0.632**	not sig.
α *M*V	8	43040	5380	3.04	0.00<0.05	sig.
q *M*V	8	19171	2396	1.35	0.21<0.05	sig.
α *q*M*V.	16	55393	3462	1.96	0.01<0.05	sig.
Error	729	1290563	1770			
Total	809	2833916	R^2 = 54.46%			

According to the experimental design results, there are significant differences among deterioration rate, rma time, mean processing times, variance of mean processing times and their interactions at the level of significance of 0.05. If one of the factors is changed, the optimal number of rma and their sequence position changes also. Furthermore, the change in the interactions between the model factors affects the result to a lesser degree.

8. Summary

This paper investigates a scheduling problem with deteriorating jobs and rate-modifying-activity simultaneously. First, we present a mathematical model with the objective of minimizing the makespan and total completion time. Our model can decide the sequence in which jobs should be scheduled, how many rmas to use, if any, and where to insert them in the schedule. We show that, as the number of jobs increases the computational time to solve the problem increases dramatically for the mathematical model. We provide polynomial time algorithms for unit processing time for makespan to solve the problem optimally. Several theoretical proofs are proposed to solve the problem with different special cases.

We propose heuristics for both makespan and total completion time. Then, we present some computational experiments. According to the experimental results, the performance of the proposed model and the heuristics are quite satisfactory. Also, the heuristic models give a reasonable error for 50 jobs when compared to the mathematical model. These heuristic algorithms are able to construct near optimal solutions with much less computational time for large number of jobs $(50 \geq)$.

References

1. Bachman, A., Janiak, A. and Kovalyov, M.Y. (2002). Minimizing the total weighted completion time of deteriorating jobs. *Information Processing Letters*, 81(2), 81-84.

2. Bachman, A. and Janiak, A. (2000). Minimizing maximum lateness under linear deterioration. *European Journal of Operational Research*, 126(3), 557-566.

3. Browne S. and Yechiali U. (1990). Scheduling deteriorating jobs on a single processor. *Operations Research*, 38, 495-498.

4. Bureau of Labor Statistics, US Dept of Labor (2009). Generated report-Number of nonfatal occupational injuries and illnesses involving days away from work by selected industry. Retrieved from www.bls.gov.

5. Cai, J.Y. and Cai, P. (1998). On a scheduling problem of time deteriorating jobs. *Journal of Complexity*, 14, 109-209.

6. Cheng, T. C.E. and Ding, Q. (2000). Single machine scheduling with deadlines and increasing rates of processing times. *Acta Informatica*, 36, 673-692.

7. He Y., Ji, M. and Cheng, T.C.E. (2005). Scheduling with a restricted rate-modifying activity. *Naval Research Logistics*, 52, 361-369.

8. Graham, R.L., Lawler, E.L., Lenstra, J.K. and Rinnooy, K. A.H.G. (1979). Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annual Discrete Math*.**,** 5, 287–326.

9. Gupta, J.N.D. and Gupta, S.K. (1988). Single facility scheduling with nonlinear processing times. *Computers and Industrial Engineering*, 14, 387-393.

10. Kovalyov, Y. M. and Kubiak, W. (1998). A fully polynomial approximation scheme for minimizing makespan of deteriorating jobs. *Journal of Heuristics*, 3, 287-297.

11. Kubiak W. and Vende, S. (1998). Scheduling deteriorating jobs to minimize makespan. *Naval Research Logistics*, 45, 511-523.

12. Lee, C. Y. and Chen, Z. L. (2000). Scheduling of jobs and maintenance activities on parallel machines. *Naval Research Logistics,* 47, 61-67.

13. Lee, C.Y and Lin, C.S. (2001). Single-machine scheduling with maintenance and repair rate-modifying activities. *European Journal of Operational Research*, 135, 493-513.

14. Lee, C.Y. and Leon, V.J. (2001). Machine scheduling with a rate-modifying activity. *European Journal of Operational Research*, 128, 119-128.

15. Lodree, E. J. and Geiger, C.D. (2010). A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration. *European Journal of Operational Research*, 201 (2), 644-648.

16. Mosheiov, G. (1991). V-Shaped polices to scheduling deteriorating jobs. *Operation Research*, 39, 979-991.

17. Mosheiov, G. (1996). Λ-Shaped policies to schedule deteriorating jobs. *Journal of the Operational Research Society*, 47, 1184-1191.

18. Qi, X., Chen, T. and Tu, F. (1997). Scheduling with the maintenance on a single machine. Working Paper, Department of Computer System Sciences, Nankai University, China.

CHAPTER 3

SCHEDULING JOBS TO CONSIDER PHYSIOLOGICAL FACTORS

Abstract

In this paper, we study scheduling jobs and breaks for a single worker. The processing times of jobs increases as the worker tires. In this study, we assume that conventional machine scheduling models don’t always work for humans. So while determining the break time, we consider the workers physiological factors. The two objectives considered are total flow time and makespan. An exact mathematical model is presented with some physiological constraints. We prove the problem is $NP-hard$ by a reduction from Equal-Size-Partition. Thus, we develop a heuristic algorithm to solve large problems. Numerical examples are presented for understanding and analyzing the performance of the mathematical model and the heuristic.

1. Background

In classical scheduling problems, the optimal sequence of jobs on a single machine is equivalent to the optimal sequence of jobs performed by a single worker. But this is not the case in real life because the algorithm ignores constraints related to a human’s physiological factors and limitations. Workers get tired both physically and mentally while they are doing their job. So, this situation causes reduced performance and productivity of the workers.

Fatigue is one important reason for decreasing performance. It can be caused by extended working hours, inadequate rest periods and unsuitable working conditions. According to Fatigue Management System Guidelines (FMSG), a fatigued worker's ability to perform their task may be lost or impaired. Fatigued workers can have;

- Reduced motivation
- Decreased speed of task
- Increase in memory errors
- Inability to concentrate
- Incorrect action

Konz (1998) has declared that fatigue increases exponentially with time. Therefore, it is important to get rest before the fatigue level becomes too high. To prevent fatigue, workers need adequate rest periods during work periods. Breaks are designed to provide time for workers to overcome the fatigue arising from the work. Resting time is classified by Konz and Johnson (2004) as formal breaks (lunch, coffee), informal break (interruptions, training) and micro breaks (short pauses of a minute). When a break is given, the worker's performance is expected to increase due to recovery. The recovery depends on how fatigued the worker is when the rest begins and the length of the rest. If the length of the break is small, the incremental amount of recovery is less than the incremental amount of time. For example, 4 breaks of 5 minutes are often more useful than 1 break of 20 minutes for two reasons; fatigue will not be increased as much and the recovery will be greater.

In this study, we use rate- modifying activities (*rma*) as a kind of resting activity for workers. Rma was first introduced in the literature by Lee and Leon (2001). They defined rma as an activity which alters the production rates of machines. The rma plays an important role in work-rest scheduling in the human factors literature. The main idea of work-rest scheduling is to obtain the number, place and duration of rest periods. While determining those decisions, productivity, safety and comfort are considered.

In the literature, the objectives of the work-rest problem are to minimize fatigue and to provide recovery of the worker while not reducing productivity. Boucsein and Thum (1997) found that short breaks are more effective in promoting recovery from both mental and emotional recovery**.** Table 11 shows several researchers who studied the work-rest problem for various physiological variables such as heart rate, blood pressure, oxygen uptake and electromyography (EMG) signals. They found that resting time is important to recover from fatigue.

Table 11. Work-Rest Problems

Veltman and Gaillard (1993)	Heart rate, blood pressure
Boucsein (1993)	Heart rate variability (HRV), EMG signals, electro dermal activity
Imbeau et al. (1995)	Heart rate, Maximum aerobic capacity (%$VO_{2\max}$)
Wu and Wang (2002)	%$VO_{2\max}$, Heart Rate
Tiwari and Gite (2006)	Heart rate
Hsie et al. (2009)	% $VO_{2\max}$

Our scheduling model is based on one developed by Ozturkoglu et al. (2011). Their model is to determine the work sequence, the number of breaks and the place of each break without considering the physiological factors of the worker. In this study, we consider the human characteristics while determining the work sequence, the number of breaks and the optimal break schedule.

This study differs from existing research in several ways. First, we define rate-modifying activities as a resting period for workers; whereas all other studies use rate-modifying activities as machine maintenance and repair time. Therefore, a rate modifying activity is a break given to workers in order to let them recover. Second, there is no research that considered the workers physiological factors when scheduling the jobs. Third, in our study, deteriorated jobs are caused by fatigue of the worker, but in the literature, jobs deteriorate while waiting to process. Thus, tiredness or fatigue of the worker is described as the deterioration rate of jobs. When we consider the industrial side, this study can provide insight on increasing worker efficiency to assigned job tasks. To the best of our knowledge, there is no research that proposed a task sequencing approach which combines deteriorating jobs, rma and considers the human characteristics.

The chapter is organized as follows. In Section 2, we propose a mathematical model and test it computationally. In Section 3, we determine the computational complexity of the problem which is introduced in Section 1. We present a heuristic algorithm for larger problems and experimental results of the heuristic in sections 4 and 5 respectively. Lastly, we offer a summary in Section 6.

2. The Mathematical Model

This model is motivated by the problem of manual order picking activities in warehousing systems. The problem is to determine the sequence in which the orders should be picked in minimum time and within acceptable levels of human physiological factors. In addition, breaks can be scheduled to allow workers to recover during the order picking sequence. We consider sequencing n orders (jobs) on a single worker (processor) with varying processing speed due to the effects of various ergonomic factors. We assume that a break time (rma) can be scheduled to improve the worker state any time after the first task is scheduled. During the resting period, the worker stops doing his/her job. After an rma is scheduled, the speed of a worker is expected to return to normal limits.

Ozturkoglu et al. (2011) determined the optimal job sequence with the optimal number of breaks and the place of each break. But that model does not consider the human's physiological factors. Therefore, we add new constraints which consider physiological factors to obtain the optimal job schedule for a single worker.

The mathematical model decides the sequence in which jobs should be scheduled, how many breaks to use to avoid fatigue, if any, and where to position them in the schedule. Those decisions are affected by human factors such as heart rate, oxygen consumption, and blood pressure. The worker may not recover completely after a break. Hence, we assume that every job consumes the worker's physiological capacity at some level. The recovery rate changes with the position of the job. Also the types of jobs affect the recovery rate of the worker. Therefore, heavy jobs need more effort than light jobs.

Assumptions

- Breaks should be taken during work shifts.
- The worker may not recover completely after breaks.
- Consider at least one physiological factor.
- Worker can not process two or more jobs simultaneously.

Parameters

n indicates the number of jobs to be sequenced

i indicates the position number which is from 1 to n

k indicates the position number which is from 0 to n (k=0 is initial position)

j indicates the job number which is from 1 to n

F indicates the number of physiological factors

α is the constant deterioration rate of jobs for $0 < \alpha \leq 1$ when delayed by one position.

q is the fixed period of time to perform the rma

p_j is the initial processing time of job j before deterioration

p_{ji} is the processing time of job j if done i positions after an rma or initial position, i.e.

$$p_{ji} = (1+\alpha)^{i-1} p_j \tag{2.1}$$

r_f is the recovery rate of physiological factor f

a_{fj} is the usage rate of physiological factor f by job j

s_{fi} is the change in cumulative physiological factor f caused by the i^{th} rma

$\underline{R}_f \left(\overline{R_f}\right)$ is a lower (upper) limit on physiological factor f

Decision Variables

$$x_{ijk} = \begin{cases} 1 & \text{if job j is in the ith position after an rma which is placed before position } k+1, \\ 0 & \text{otherwise} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if an rma is placed before position i} \\ 0 & \text{otherwise} \end{cases}$$

C_i = Completion time of the job in position *i*.

R_{fi} = Cumulative value of physiological factor *f* at position *i*,

Objective Functions

To solve the problem, the objective of minimizing total flow time or makespan is used.

Minimize $Z = \sum_{i=1}^{n} C_i \qquad i = 1,..,n$ (2.2)

Minimize $Z = C_{\max}\,, \qquad C_{\max} \geq C_i \qquad i = 1,..,n$ (2.3)

.

Mathematical Model

$$C_1 = \sum_{j=1}^{n} p_{j1} x_{1j0} \tag{2.4}$$

$$C_i = C_{i-1} + \sum_{k=1}^{i} \sum_{j=1}^{n} p_{jk} x_{i,j,i-k} + q\, y_i \qquad i = 2,.....,n \tag{2.5}$$

$$\sum_{i=1}^{n} \sum_{k=0}^{i-1} x_{ijk} = 1 \qquad j = 1,.....,n \tag{2.6}$$

$$\sum_{k=0}^{i-1} \sum_{j=1}^{n} x_{ijk} = 1 \qquad i = 1,.....,n \tag{2.7}$$

$$R_{f1} \geq \sum_{j=1}^{n} \left(r_f - a_{fj}\right) p_{j1} x_{1j0} \qquad f = 1,.....,F \tag{2.8}$$

$$R_{fi} \geq R_{f,i-1} + \sum_{k=1}^{i} \sum_{j=1}^{n} \left(r_f - a_{fj}\right) p_{jk} x_{i,j,i-k} + s_{fi} y_i \qquad i = 2,.....,n \quad f = 1,.....,F \tag{2.9}$$

$$\underline{R}_f \leq R_{fi} \leq \overline{R}_f \qquad i = 1,.....,n \quad f = 1,.....,F \tag{2.10}$$

$$x_{kji} \leq y_{i+1} \qquad k = 2.....,n \quad j = 1,.....,n \quad i = 1,.....,k-1 \quad (2.11)$$

$$x_{ijk} \in \{0,1\} \qquad i = 1,.....,n \quad j = 1,.....,n \quad k = 0,.....,n \quad (2.12)$$

$$y_k \in \{0,1\} \qquad k = 2,.....,n \quad (2.13)$$

$$C_i \geq 0 \qquad i = 1,.....,n \quad (2.14)$$

In constraint (2.4), the completion time of the job in position one is equal to the processing time of the job assigned to position one. Before the first position, there is no rma ($y_1 = 0$). In constraint (2.5), the completion time of the job in position i is equal to the completion time of the job in position $i-1$ plus the processing time of the job assigned to position i plus the rma time if assigned. In constraint (2.6), each job is assigned to exactly one position. In constraint (2.7), each position is scheduled for only one job. Constraint (2.8) computes the cumulative level of a physiological factor of the worker at the end of position 1. Constraint (2.9) determines the cumulative level of a physiological factor of the worker at the end of position *i*. Constraint (2.10) ensures that the level of cumulative physiological factor is always within prescribed acceptable limits. Constraint (2.11) requires an rma to be placed in the related position if jobs are scheduled after the rma and to control the sequence of the rma. Lastly, constraints (2.12), (2.13) and (2.14) indicate the decision variables are binary and all other variables are non-negative.

Performance of the Model

A computational experiment is conducted to test the performance of the mathematical model with different numbers of jobs, deterioration rates and the rma time. Table 10 summarizes the model factors and each of their respective levels. In the model, we use three different physiological factors which are oxygen consumption ($VO_{2\max}$), heart rate ($HR_{\max}$) and blood pressure (BP). These parameters are generated from a uniform distribution. Tables 12 through Table 16 summarize the physiological factor parameters.

Table 12. The Parameters of the Problem

Parameters	Values
p_j	U~ [1-50]
# of jobs	25, 50
α	0.02,0.04 and 0.08
rma time	5, 10, 15 (min.)

Table 13. Parameter Settings for Recovery Rate

Physiological factors	**Recovery rate (r_f)**
$HR_{\max}$	r_f ~U[1-20] %
$VO_{2\max}$	r_f ~U[1-15] %
BP	r_f ~U[1-20] %

Table 14. Parameter Settings for Usage Rate

Physiological factors	**Usage rate (a_{fj})**
$HR_{\max}$	a_{fj} ~U[1-15] %
$VO_{2\max}$	a_{fj} ~U[1-15] %
BP	a_{fj} ~U[1-15] %

Table 15. Parameter Settings for Cumulative Changes

Physiological factors	**Changes based on factors (s_{fi})**
$HR_{\max}$	s_{fi} ~U[1-10] %
$VO_{2\max}$	s_{fi} ~U[1-10] %
BP	s_{fi} ~U[1-10] %

Table 16. Parameter Settings for Acceptable Limits

Physiological factors	**Acceptable limits ($\underline{R}_f\left(\overline{R_f}\right)$)**
$HR_{\max}$	160 bpm- 220 bpm
$VO_{2\max}$	15 % - 33%
BP	75 mmHg -115 mmHg

The proposed model is coded using AMPL and solved by CPLEX 9.1 on a computer with a Pentium IV 2.8 GHz processor and 1GB of RAM. For each parameter set, we run the model 10 times; 180 replications were run for both total completion time and makespan objective functions. The average computation times are given in Tables 17 and 18.

Table 17. Average Run Time for Total Completion Time (sec.)

Jobs numb.	Det. Rate	Rma Time	Ave. Comp. Time (sec.)
25	0.02	5	8.7
	0.02	10	14.2
	0.02	15	19.5
	0.04	5	8.5
	0.04	10	11.0
	0.04	15	20.2
	0.08	5	8.1
	0.08	10	10.3
	0.08	15	18.7
50	0.02	5	33.7
	0.02	10	42.5
	0.02	15	51.8
	0.04	5	21.9
	0.04	10	37.2
	0.04	15	51.9
	0.08	5	11.1
	0.08	10	28.2
	0.08	15	43.7

Table 18.Average Run Time for Makespan (sec.)

Jobs numb.	Det. Rate	Rma Time	Ave. Comp. Time (sec.)
25	0.02	5	12.5
	0.02	10	15.1
	0.02	15	19.9
	0.04	5	10.3
	0.04	10	15.6
	0.04	15	21.2
	0.08	5	9.4
	0.08	10	16.7
	0.08	15	23.8
50	0.02	5	41.2
	0.02	10	47.9
	0.02	15	54.5
	0.04	5	38.3
	0.04	10	46.4
	0.04	15	51.4
	0.08	5	18.3
	0.08	10	29.9
	0.08	15	48.1

The efficacy of the model is based on the average run time in seconds. The average time for total completion and the makespan problem is 24.5 seconds and 28.91 seconds respectively. As expected, run time increases as the number of jobs increase. The run time for the problem of both objectives decreases as the deterioration rate increases. Also, run time increases as the rma time goes up. These results are similar to the results of Ozturkoglu et al. (2011). When we compare run times, this model is computed faster than their model, because it is more constrained and the feasible region gets smaller. Another difference between the models is the place and the number of the breaks. Breaks are given more frequently in this model. This is expected because of the worker's physiological factors. As a result, this model reflects a real industry situation.

3. Complexity Result

In this section, we show $1 | p_{ij} = (1+\alpha_j)^{i-1} p_j, rm, phy | C_{\max}$ is $NP-hard$. The reduction is based upon Equal-Size-Partition which is ordinary NP-complete.

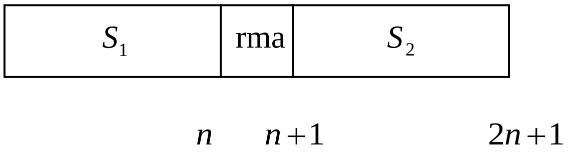
Equal-Size-Partition (ESP): Given a multi-set $S \ \{a_1, a_2, ..., a_{2n}\}$, $a_j \geq 0$ with $\sum_{j=1}^{2n} a_j = 2B$ is there a partition S_1 and S_2 such that $\sum_{i \in S_1}^{n} a_i = \sum_{i \varepsilon S_2}^{n} a_i = B$ and $|S_1| = |S_2| = n$?

Theorem 1. $1 | p_{ij} = (1+\alpha_j)^{i-1} p_j, rm, phy | C_{\max}$ is $NP-hard$.

Proof: Given an instance of ESP, we construct an instance of $1 | p_{ij} = (1+\alpha_j)^{i-1} p_j, rm, phy | C_{\max}$ so that its optimal solution is $2n+1$ if and only if the ESP has a solution.

For each element of S we create a job with $p_j = 1$; there are $2n$ jobs. Let the deterioration rate $\alpha = 0$ and the rma time $q = 1$. We only need one physiological factor, so we omit the subscript f. The recovery rate $r = 0$ and the changing factor caused by the rma $s = B$. The usage rate for job j is a_j, $j \in S$. Let $\underline{R} = 0$ and $\overline{R} = B$.

If the ESP has a solution, then $1 | p_{ij} = (1+\alpha_j)^{i-1} p_j, rm, phy | C_{\max}$ has the following schedule;



The makespan of this schedule is minimal since there must be at least one rma since $\sum_{j \in S} a_j > B_j$ and it is feasible since $0 \le R_i \le B,\ i = 1,..,n,\ \ R_n = 0$ and $R_{n+1} = B$ due to the rma.

If the ESP has no solution, then there must be at least 2 rma's, resulting in $C_{\max} \ge 2n+2$.

Lemma. $1 | p_{ij} = (1+\alpha_j)^{i-1} p_j, rm, phy | \sum C$ is $NP-hard$.

Proof: The proof is identical to the makespan proof except the optimal value of the total completion time is;

$$\frac{n(n+1)}{2} + n(n+1) + \frac{n(n+1)}{2} = 2n(n+1)$$

4. Heuristic Algorithm

In this section, we present a polynomial time heuristic algorithm for $1 | p_{ij} = (1+\alpha)^{i-1} p_j, rm, | C_{\max}$. In addition to the complexity of the problem, the larger sizes present some difficulty. Therefore, to solve large problems, a heuristic algorithm may be needed.

In the proposed heuristic algorithm, we first provide the worker's basic information such as gender, age, height and weight. Then, sort the jobs according to the SPT rule. The job which has the largest processing time is assigned to position 1. After assigning the job, calculate the fatigue rate of the worker. Fatigue can be used as an upper limit for energy expenditure during the work. Next, we determine an upper limit from the Borg scale. If this amount is larger than acceptable limits, then insert an rma and assign the next largest job to the position immediately after the rma. Otherwise assign the smallest job to the current position without doing an rma. Stop the heuristic when all jobs are assigned.

Santos and Resnick (1999) developed an equation which calculates a worker's predicted fatigue rate. We use fatigue as the single physiological factor, other factors could be used as well. Let us define a set of given jobs as J= {J_1, J_2, J_3… J_n}.

The parameters and the equation are given below;

$$F = -2.25 + 0.63t + 0.53m + 0.08pr \tag{4.1}$$

F The fatigue rate of the worker

m The mass of the handled load in kg

t The time into shift in minutes

pr The production rate in units per minute

b job with largest processing time within the set of unassigned jobs

s job with smallest processing time within the set of unassigned jobs

i Position number after rma or initial position

$J_{[i]}$ Job in position i

$$y_i = \begin{cases} 1 & \text{if an rma is placed before position i} \\ 0 & \text{otherwise} \end{cases}$$

p_{ij} Processing time of job *j* in position *i* as $p_{ij} = (1+\alpha)^{i-1} p_j$

C_i Completion time of the job in position i

The procedure is described as follows:

Step 1. Determine the basic information that will be used.

(e.g. mass of the load, production rate and time)

Step 2. Sequence the jobs in SPT order.

Step 3. Schedule the job with largest processing time in position one and no rma at the beginning.

$y_1 = 0$, $J_{[1]}$ =n

Set $C_1 = p_{1n}$,$b = b-1,\ s = 1,\ k = 2$.

Step 4. Calculate the ‘ *Fatigue* ’;

$$F = -2.25 + 0.63t + 0.53m + 0.08pr$$

Step 5. Determine the acceptable limit of the worker’s fatigue.

Step 6. Decide if fatigue is within the acceptable limits

If $F < \overline{R_f}$

Then schedule the jobs with smallest processing time in that position and do not insert an rma.

$y_i = 0$, $J_{[k]} = s$

Set $C_i = C_{i-1} + p_{is}$, $s = s+1$ and $k = k+1$;

Step 7. If $F \geq \overline{R_f}$

Then schedule the job with largest processing time in that position and insert an rma after that position.

$y_i = 1$, $J_{[k]} = b$

Set $C_i = C_{i-1} + p_j + q$

$b = b-1$, $k = k+1$

If $i < n$, go to Step 4.

Otherwise go to Step 8.

Step 8. If $k = n$, Stop the algorithm, Makespan $= C_i$.

5. Numerical Example of Heuristic

In this section, we use several numerical examples to illustrate the behavior of the model. In all the examples we assume parameters values from Table 19 and Table 20.

Table 19. Basic Information for the Numerical Examples

Mass of the hand load	U~[3-20] kg
Time into the shifts	U~[1-10]minutes
Production Rate	U~[1-10]unit /minute

Table 20. Experimental Factors

Factors	Levels
Number of jobs	10, 25, 50
Processing time	U~[10-100]
Deterioration rate	0.04, 0.06, 0.08
Rma time	5, 10, 15

To analyze the performance of the heuristic in terms of the error percentages, we compare heuristic solutions to an optimal solution on an instance taken randomly from the solution sets. We use the following equation:

$$e = \frac{F_h - F_{opt}}{F_{opt}} * 100\% \tag{5.1}$$

In this equation, F_h is the makespan value of the heuristic model, F_{opt} is the makespan value of the optimal solution and e is the average performance of the heuristic. The heuristic is replicated 10 times for each problem case. It is coded in JAVA and solved with a Pentium IV, 2.8 GHz processor and 1GB of RAM.

The number of problems solved in each combination data set is 10. We obtained the average percentage error of the heuristic for makespan and total flow time objectives which are given in Tables 21 and 22. The experimental results show that the heuristic requires small run times for large problems. In general, the heuristic can be applied to any problem size.

Also, the heuristic algorithm finds near optimal solutions for all problems tested. The average error percentage is % 0.37 for total flow time and % 0.64 for makespan.

Table 21. Comparison of Heuristic and Mathematical Model for Makespan

Number of jobs	**α**	**Rma Time**	**Ave. Error of Heuristic (%)**
10	0.04	5	0.13
		10	0.32
		15	0.24
	0.06	5	0.33
		10	0.13
		15	0.23
	0.08	5	0.45
		10	0.92
		15	0.51
25	0.04	5	0.19
		10	0.54
		15	0.44
	0.06	5	0.69
		10	0.63
		15	0.16
	0.08	5	0.48
		10	0.58
		15	0.31
50	0.04	5	0.32
		10	0.17
		15	0.60
	0.06	5	0.41
		10	0.14
		15	0.48
	0.08	5	0.37
		10	0.18
		15	0.19

Table 22. Comparison of Heuristic and Mathematical Model for Total Completion Time

Number of jobs	α	Rma Time	Ave. Error of Heuristic (%)
10	0.04	5	0.34
		10	0.96
		15	0.33
	0.06	5	0.45
		10	0.92
		15	0.78
	0.08	5	1.02
		10	0.59
		15	0.54
25	0.04	5	0.79
		10	0.17
		15	0.81
	0.06	5	0.94
		10	0.63
		15	0.38
	0.08	5	0.68
		10	0.37
		15	0.42
50	0.04	5	0.62
		10	0.91
		15	0.25
	0.06	5	0.91
		10	0.43
		15	0.98
	0.08	5	0.37
		10	0.84
		15	0.93

6. Summary

This paper deals with scheduling rate-modifying-activities for a single worker. In order to reflect real industrial applications, we assume that the processing times of jobs deteriorate. This study is the first study to consider the physiological condition of workers while determining the number and timing of breaks. We minimize flow time and makespan. The model is relatively fast and can be solved in less than one minute with 50 jobs. Also, the usage of jobs and recovery rate affects the solution when compared to the result of Ozturkoglu et al. (2011). The model gives higher completion time or makespan values because of the changing recovery rate. In this model, the worker doesn’t recover completely. Thus, this increases the realized processing times of jobs, and thus, total completion time. Additionally, the number and placement of breaks is completely different from models which do not consider physiological factors. We proved that this problem is $NP-hard$ for both objectives. Therefore, we developed a heuristic algorithm for large problems. The computational results show that the heuristic gives near-optimal results in a reasonable time for the large problems.

References

1. Boucsein, W. (1993). Psychophysiology in the computer work-place goals and methods. *Elsevier North Holland*, Amsterdam, 135-139.

2. Boucsein, W. and Thum, M. (1997). Design of work/rest schedule for computer work based on psycho-physiological recovery measures. *International Journal of Industrial Ergonomics*, 20, 7-51.

3. Hancock, P.A. and Desmond, P.A. (2001). Stress, workload and fatugue: Human factors in transportation. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

4. Hsie, M., Hsiao, Cheng, T. and Chen, H. (2009). A model used in creating a work-rest schedule for laborers. *Automation in Construction*, 18, (6), 762-769.

5. Imbeau, D., Desjardins, L., Dessureault, P.C., Riel, P. and Fraser, R. (1995). Oxygen consumption during scaffold assembling and disassembling work: Comparison between field measurements and estimation from heart rate. *International Journal of Industrial Ergonomics*, 15, 247-259.

6. Konz, S. (1998). Work/rest: Part II - The scientific basis (knowledge base) for the guide. *International Journal of Industrial Ergonomics*, 22, 73–99.

7. Konz, S. and Johnson, S. (2004). Work design. Sixth edition, Scottsdale, AZ: Holcomb Hathaway.

8. Lee, C.Y. and Leon, V.J. (2001). Machine scheduling with a rate-modifying activity. *European Journal of Operational Research*, 128, 119-128.

9. Ozturkoglu, Y., Bulfin, R. L. and Lodree E. (2011). A Model for Scheduling Deteriorating Jobs with Rate-Modifying-Activities on a Single Machine. Working paper.

10. Santos, E. and Resnick, M.L. (1999). The effects of fatigue on quality and productivity in repetitive tasks, Florida International University.

11. Tiwari, P.S. and Gite, L.P. (2006). Evaluation of work-rest schedules during operation of a rotary power tiller. *International Journal of Industrial Ergonomics*, 36, 203-210.

12. Veltman, J.A. and Gaillard, A.W.K. (1993). Indices of mental workload in a complex task environment. *Neuropsychobiology*, 28, 72-75.

13. Wu, H.C. and Wang, M.J. (2002). Relationship between maximum acceptable work time and physical workload. *Ergonomics*, 45, 280–289.

CHAPTER 4

A BI-CRITERIA SINGLE MACHINE SCHEDULING WITH RATE-MODIFYING-ACTIVITY

Abstract

We consider a single machine scheduling problem with two criteria: minimizing both total flow time with total tardiness and minimizing maximum tardiness with a number of tardy jobs. We present a mathematical model which is based on a model developed by Ozturkoglu et al. (2011) to find the optimal schedule. In the mathematical model, job processing times are assumed to deteriorate. Besides deteriorated jobs, we also consider rate-modifying-activities. To analyze the mathematical model, we use three different approaches. According to computational results, up to 50 jobs can be solved in less than one minute.

1. Introduction

In real life applications, customers and producers may have completely different perspectives. For instance, a producer wants to maximize his selling price of the product whereas a customer wants to minimize cost of product. To satisfy both customers and production effectiveness at the same time, minimizing both total completion time and number of tardy jobs is appropriate.

So, managers should consider more than one measure when they try to find the best schedule for their production process. Therefore, bi-criteria scheduling is becoming more attractive. In most applications it is beneficial and necessary to measure the objectives of a schedule with respect to different criteria. This provides more flexibility to the decision maker by allowing him/her to consider multiple schedules corresponding to the different solutions.

Research on bi-criteria scheduling is rare compared to research in single criterion scheduling. Almost all bi-criteria research assumes job processing times are constant. But in a real life situation, job processing times may deteriorate while jobs are waiting to be processed. Both machine and operator may cause this deterioration. For instance, a machine or tools may wear and processing time and quality of the jobs will change. Or the operator's physical condition may cause the processing speed to change over time. Browne and Yechiali (1990) introduced deteriorated processing times in the scheduling literature. They assumed that the processing time of a job grows linearly, depending on the start time of the job.

To prevent job deterioration, repair or maintenance, called a rate-modifying activities (rma), are needed. The rma, an activity which affects and changes the production rate of the machines, was first introduced by Lee and Leon (2001).

This research addresses bi-criteria scheduling problems involving a single machine with deteriorating jobs and rate-modifying-activities. We study two bi-criteria models; total flow time, total tardiness, maximum tardiness and number of tardy jobs. Minimizing total flow time leads to manufacturer satisfaction, while tardiness leads to customer dissatisfaction. To satisfy both sides by using these performance measures, we propose mathematical models and present experimental results for both$1/p_{ij} = (1+\alpha)^{i-1} p_j, rm/T_{\max} \mid \sum U$ and$1/p_{ij} = (1+\alpha)^{i-1} p_j, rm/\sum F \mid \sum T$.

2. Literature Review

In the bi-criteria literature, some criteria are used very often. Therefore, we divide the literature review based on performance measures. We classify the most common criteria studied in bi-criteria single machine scheduling problems by researchers. But before giving information about previous research, we explain major approaches to solve bi-criteria problems. There are three approaches.

1) Bi-criteria Approach

Generate the Pareto curve for all non-dominated schedules. Using Graham et al. (1979) three field notation, we denote single machine bi-criteria problems under basic assumptions as $1 // \gamma_1, \gamma_2$.

2) Secondary Objective Approach

First try to optimize the primary criterion and then solve the remaining secondary criterion subject to the primary criterion remaining optimal. Denote the single machine bi-criteria problems under basic assumptions as $1 // \gamma_2 \mid \gamma_1$.

3) The Weighting Method

In this method a weighted combination of both objectives is optimized. Mathematically, the weighting method can be stated as follows:

$$\min z(x) = w_1 z_1(x) + (1 - w_1) z_2(x) \qquad (2.1)$$

In this study, we use three approaches to analyze our mathematical model. Various combinations of the criteria are considered and analyzed as primary and secondary criteria in the literature. We give a brief literature review of the most commonly used performance measures.

Total Completion Time and Maximum Tardiness

Smith (1956) developed a polynomial time algorithm to use secondary objective approach scheduling problems with these two objectives. Afterwards, Heck and Roberts (1972), Emmons (1975), Van Wassenhove and Gelders (1980) extended Smith's study and developed some algorithms for the secondary approach. Chen and Bulfin (1993) proved that flow time with maximum tardiness bi-criteria problems are NP-hard.

Later, Kondakci et al. (1996) presented an algorithm to produce all efficient schedules for any given non decreasing function of the total completion tine and maximum tardiness objectives. Chen (2007) developed a heuristic to an objective of minimizing total flow time with maximum tardiness objectives.

Weighted Completion Time and Maximum Tardiness

Burns (1976) presented an algorithm that provide to a local optimum for both the weighted and unweighted problems in bi-criteria objectives. For the secondary approach, Bansal (1980) extended Burns (1976) algorithm and applied a branch and bound algorithm to find a globally optimal solution. In his algorithm, he found a locally optimal solution for the problem of minimizing weighted sum of completion times subject to the condition that every job be completed by its due date. Miyazaki (1982) solved bi-criteria problem with different approach which one of the criteria as objective and the other as a constraint. He developed a necessary condition under which the local and global solutions are different and developed an algorithm to obtain an improved schedule based on the locally optimal schedule. Shanthikumar and Buzacott (1982), Potts and Van Wassenhove (1983) developed some heuristics for problem $1//T_{\max} \mid F$. Posner (1985) and Bacghi and Ahmadi (1987) considered with these two performance measures with deadlines and they found tightest bound for same objectives. Chen and Bulfin (1993) proved that weighted flow time with maximum tardiness bi-criteria problems $1// w_T \mid T_{\max}$ and $1// w_u \mid T_{\max}$ are NP-hard.

Total Completion Time and Number of Tardy Job

Emmons (1975) was the first to study the secondary approach and presented a branch and bound algorithm for it. Then, Nelson et al. (1986) presented branching procedures for the bi-criteria approach. Chen and Bulfin (1993) proved that these objectives are NP-hard.

Maximum Tardiness and Number of Tardy Job

Firstly, Shanthikumar (1983) developed a branch and bound algorithm for the problem $1 \| T_{\max} \mid \sum U_j$. Later on, Nelson et al. (1986) and Chen and Bulfin (1994) proposed both heuristic and branch and bound algorithms for problem $1 \| \sum U_j \mid T_{\max}$. Huo et al. (2007) considered complexity relationship of single machine problems $1 \| \sum U_j \mid \max\{w_j T_j\}$ and $1 \| \max\{w_j T_j\} \mid \sum U_j$ with weighted tardiness. Also they proposed several fast heuristics.

Most papers mentioned above assume processing time is constant. To the best of our knowledge, there is no study which combines bi-criteria scheduling problems with deteriorating jobs. So this paper is the first study which combines deteriorating jobs with rate-modifying-activity in bi-criteria objectives.

3. Problem Description

As stated earlier, this study focuses on the single machine bi-criteria scheduling problem. There are n jobs to be processed on a single machine. The jobs are available at time zero and are independent of each other. Preemption is not allowed. The machine can handle one job at a time. Each job has a base processing time p_j before deterioration, a due date d_j and an actual processing time p_{ji} which is the processing time of job j if done i positions after an rma or the initial position. We calculate p_{ji} by;

$$p_{ji} = (1+\alpha)^{i-1} p_j \tag{3.1}$$

where α is the deterioration rate of jobs for $0 < \alpha \leq 1$ when delayed by one position. This is non-linear deterioration based on position rather than start time. And, q is the fixed period of time to perform the rma.

Decision Variables

$$x_{ijk} = \begin{cases} 1 & \text{if job j is in the ith position after an rma which is done before position k} + 1, \\ 0 & \text{otherwise} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if an rma is done before position i} \\ 0 & \text{otherwise} \end{cases}$$

C_i = Completion time of the job in position *i*.

Our model is based on Ozturkoglu et al. (2011). The constraints for our model are;

$$C_1 = \sum_{j=1}^{n} p_{j1} x_{1j0} \tag{3.2}$$

$$C_i = C_{i-1} + \sum_{k=1}^{i} \sum_{j=1}^{n} p_{jk} x_{i,j,i-k} + q\, y_i \qquad i = 2,\ldots..,n \tag{3.3}$$

$$\sum_{i=1}^{n} \sum_{k=0}^{i-1} x_{ijk} = 1 \qquad j = 1,\ldots..,n \tag{3.4}$$

$$\sum_{k=0}^{i-1} \sum_{j=1}^{n} x_{ijk} = 1 \qquad i = 1,\ldots..,n \tag{3.5}$$

$$x_{kji} \le y_{i+1} \qquad k = 2\ldots..,n; \quad j = 1,\ldots..,n; \quad i = 1,\ldots..,k-1 \tag{3.6}$$

$$x_{ijk} \in \{0,1\} \qquad i, j = 1,\ldots..,n; \quad k = 0,\ldots..,n \tag{3.7}$$

$$y_k \in \{0,1\} \qquad k = 2,\ldots..,n \tag{3.8}$$

$$C_i \ge 0 \qquad i = 1,\ldots..,n \tag{3.9}$$

In constraint (3.2), the completion time of the job in position one is equal to the processing time of the job assigned to position one. Before the first position, there is no rma ($y_1 = 0$). In constraint (3.3), the completion time of the job in position i is equal to the completion time of the job in position $i-1$ plus the processing time of the job assigned to position i plus the rma time if assigned. In constraint (3.4), each job is assigned to exactly one position. In constraint (3.5), each position is scheduled for only one job. Constraint (3.6) requires an rma to be done in the related position if jobs are scheduled after rma and to control the sequence of the rma. Lastly, constraints (3.7), (3.8) and (3.9) indicate the decision variables are binary and all other variables are non-negative.

4. Criterion

In this paper we focus on four different objectives which are to minimize total flow time, total tardiness, maximum tardiness and number of tardy jobs. Different combinations of two of these criteria are studied. The mathematical formulation for each criterion is given below.

4.1. **Total Completion Time**

Minimizing flow time keeps the work-in-process inventory at a low level. Also, it minimizes completion times, lateness and job waiting times. It is defined as:

$$\min z = \sum_{i=1}^{n} C_i \qquad (4.1)$$

Subject to:

Constraint sets (3.2), (3.3), (3.4), (3.5), (3.6), (3.7), (3.8) and (3.9) respectively.

4.2. Total Tardiness

Minimizing total tardiness reduces penalties caused by late jobs. Let T_j be the tardiness of job j .

$$\min z = \sum_{j=1}^{n} T_j \qquad (4.2)$$

Subject to:

Constraint sets (3.2), (3.3), (3.4), (3.5), (3.6), (3.7), (3.8), (3.9) and

$$T_i \geq C_i - d_i \qquad i = 1,..,n \tag{4.3}$$

4.3. Maximum Tardiness

Minimizing maximum tardiness is a measure of customer satisfaction based on due dates.

$$\min z = T_{\max} \tag{4.4}$$

Subject to:

Constraint sets (3.2), (3.3), (3.4), (3.5), (3.6), (3.7), (3.8), (3.9) and

$$T_{\max} \geq C_i - d_i \qquad i = 1,..,n \tag{4.5}$$

4.4. Number of Tardy Jobs

Often used in real applications, we try to finish as many jobs as possible on time because of the penalty costs.

$$\min z = \sum_{i=1}^{n} N_i \tag{4.6}$$

Subject to:

Constraint sets (3.2), (3.3), (3.4), (3.5), (3.6), (3.7), (3.8), (3.9) and

$$MN_i \geq T_i \qquad i = 1,..,n \tag{4.7}$$

$$N_i \in \{0,1\} \qquad i = 1,..,n \tag{4.8}$$

M is a very big number.

5. Computational Experiments

To understand the behavior of the mathematical model, three approaches are used. The proposed mathematical model is coded using AMPL and solved by CPLEX 9.1 on a computer with Pentium IV 2.8 GHz processor and 1GB of RAM. We perform an empirical study of the three bi-criteria approach. In the next subsection, we describe how we generate the data. And then we give results and analysis of experiments.

5.1. Data Generation

In our experiments, we consider 25 and 50 jobs. Job processing times are generated from a uniform distribution on the interval [1-50]. To generate the due dates, we use τ and R based on Huo et al. (2007) which denote the due date range and tardiness factor respectively. To generate each job's due date, we use a discrete uniform distribution with intervals $\sum_{j=1}^{n} p_j(1-\tau-R/2)\ and \sum_{j=1}^{n} p_j(1-\tau+R/2)$. Table 23 gives problem parameters.

Table 23. The Parameter of the Problem

Parameter	Values
p_j	U~ [1-50]
$\#of\ jobs$	25 and 50
α	0.025, 0.05, 0.075
q	2, 5, 8 (min.)
τ	0.25, 075
R	0.25, 0.50
w_i	0.25,0.50, 0.75

Ten replications of each of the possibilities were run for each combination of performance measure. Totally, 540 instances were generated. The results and analysis of experiments are given in the below.

5.2. Bi-criteria Approach

One of the commonly used methods in bi-criteria is the Pareto curve. In this method, solution s_1 is dominated by solution s_2, if s_2 is not worse than s_1 in all objectives and if s_2 is strictly better than s_1 for at least one of the objectives. This solution is called non-dominated and the set of non-dominated solutions in the feasible problem space is the Pareto optimal set.

To try to find efficient Pareto curve we have plotted the 25 points which are our objective functions. These points lie on the objective function line. To obtain these points, we use 25 jobs with 0.025 deterioration rate. All other parameters are the same.

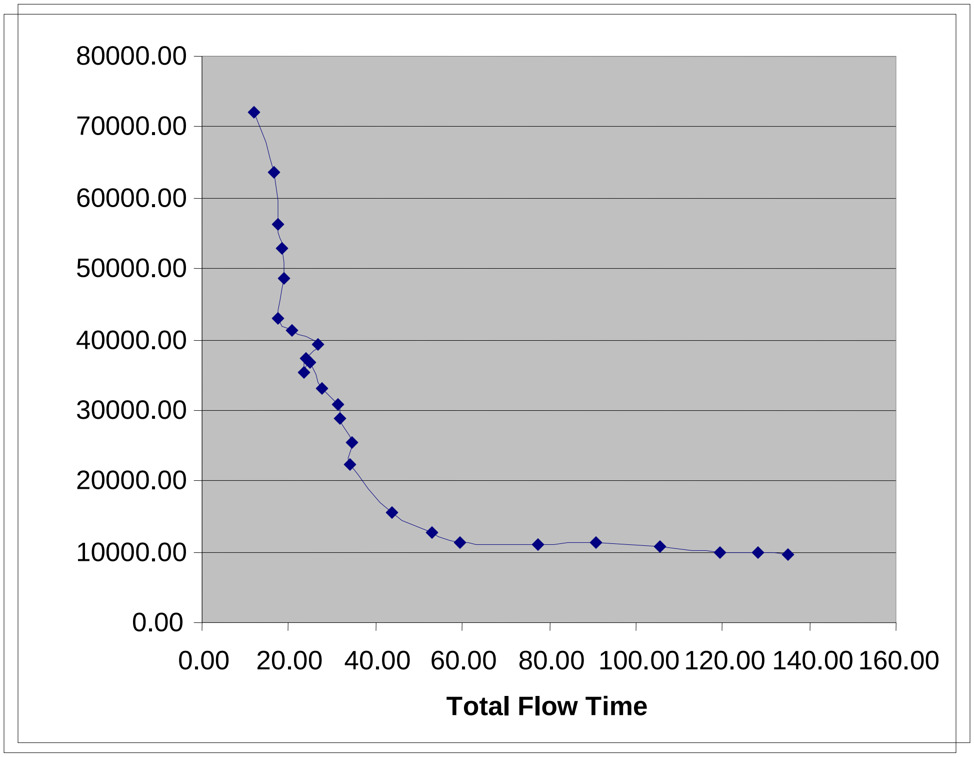


Figure 1. Pareto Curve for $1/_{ij} = (1+\alpha)^{i-1} p_j, rm/\sum F, \sum T$

Non-dominated means that there is no other solution in which one objective function can be improved without a simultaneous detriment to the other objective. In Figure 1, each of these points determines the extreme points of the dominated set in the decision space. All points are equally acceptable as the solution to the bi-criteria optimization problem. But, the decision maker should select only one solution for practical purposes. Therefore, the decision maker may choose a schedule that provides a more balanced performance of the two criteria employed.

5.3. Secondary Objective Approach

For the secondary objective, we solve in two stages. First, optimize the primary criterion and then solve the secondary criterion with the constraint that value of the primary criterion is equal to its optimal value.

Tables 24 and Table 25 show the computational time and given an rma of the $1/_{ij} = (1+\alpha_j)^{i-1} p_j, rm/T_{\max} \mid \sum U$ and $1/_{ij} = (1+\alpha_j)^{i-1} p_j, rm/\sum T \mid \sum F$ respectively.

Table 24. Average Run Time (sec.) for $1/_{ij} = (1+\alpha_j)^{i-1} p_j, rm/T_{\max} \mid \sum U$

Jobs numb.	Det. Rate	Rma time	Ave. Comp. Time (sec.)	Num. of Rma
25	0.025	2	5.7	9
	0.025	5	5.6	9
	0.025	8	5.9	9
	0.05	2	6.2	10
	0.05	5	6.3	10
	0.05	8	6.1	10
	0.075	2	8.6	11
	0.075	5	8.5	12
	0.075	8	8.8	12
	0.025	2	18.7	20
	0.025	5	19.0	21
	0.025	8	19.4	21

50	0.05	2	22.2	22
	0.05	5	24.3	22
	0.05	8	24.1	23
	0.075	2	28.5	24
	0.075	5	28.7	24
	0.075	8	29.4	24

Table 25. Average Run Time (sec.) for $1/_{ij} = (1+\alpha_j)^{i-1} p_j, rm/\sum T \,|\sum F$

Jobs numb.	Det. Rate	Rma time	Ave. Comp. Time (sec.)	Num. of Rma
25	0.025	2	19.1	8
	0.025	5	19.8	8
	0.025	8	20.2	8
	0.05	2	22.9	8
	0.05	5	23.3	8
	0.05	8	23.4	9
	0.075	2	26.6	9
	0.075	5	27.1	10
	0.075	8	28.5	10
50	0.025	2	32.9	17
	0.025	5	32.7	17
	0.025	8	34.3	16
	0.05	2	36.8	17
	0.05	5	37.8	18
	0.05	8	39.1	18
	0.075	2	38.4	19
	0.075	5	39.9	19
	0.075	8	40.9	19

Based on the tables, the number of rmas is based on both deterioration rate and rma time. A larger deterioration rate results in more rmas. It is obvious that for larger rma times, larger computation time is needed.

5.4. Weighted Method

The two objectives can be optimized at the same time by assigning proper weights in the weight method. Mathematically, the weighting method can be stated as follows:

$$\min z(x) = w_1 z_1(x) + (1 - w_1) z_2(x) \qquad (5.1)$$

We use three different values of w_i $(0.25, 0.50, 0.75)$ in our calculations. The extension of weight ranges is important for the stability of solution. Before using Equation 5.1, we normalize our objective function values to obtain reliable results.

Normalization

In practice, multiple objectives have different dimensions and it is difficult to compare different objective types. The individual preferences of the objectives are described by weights. These weights are assigned by the decision maker. But assigning proper weights is difficult and may cause problems. To prevent problems, normalization of objectives is necessary to get reliable solutions. Normalization of different objectives permits comparison of various dimensions and to find the relationship between different objectives.

In our data set, normalization is necessary. We transform the data into a range between 0 and 1. The normalization of the objective function values are found by using;

$$t_i = \frac{f_i(x) - z_i^L}{z_i^U - z_i^L} \qquad (5.2)$$

where

$f_i(x) =$ original value ,

$z_i^U \quad = \max\,(f_i(x))$,

$z_i^L \quad = \min\,(f_i(x))$,

$t_i \quad =$ transformed value.

This value provides the best normalization results as we normalize the objective functions by the true intervals of their variation over the Pareto Optimal set. After normalization, we run our new data set and obtain Tables 26 and 27.

Table 26. Weighted Method for $1/_{ij} = (1+\alpha_j)^{i-1}\; p_j, rm/\; f(T,F)$

Jobs numb .	Det. Rate	Rma time	$w_1 = 0.25$	$w_1 = 0.5$	$w_1 = 0.75$	Ave.Flow.T (sec.)	Num.of rma
25	0.025	2	6007.1	4257.2	2507.3	2.2	9
	0.025	5	6192.9	4388.5	2584.1	3.1	5
	0.025	8	6322.3	4480.5	2638.6	3.3	4
	0.05	2	6295.4	4446.4	2597.3	3.5	10
	0.05	5	6575.01	4643.1	2711.2	3.6	7
	0.05	8	6770.1	4780.6	2790.8	3.6	4
	0.075	2	6467.4	4573.8	2680.1	3.9	19
	0.075	5	6808.1	4811.8	2818.5	4.1	13
	0.075	8	7034.2	4973.48	2912.2	4.0	9
50	0.025	2	24789.9	17060.5	9331	18.3	16
	0.025	5	25499.6	17547.8	9595.9	21.5	14
	0.025	8	25943.7	17851.7	9759.6	19.5	10
	0.05	2	16395.8	11315.5	6235.2	18.2	20
	0.05	5	17197.9	11866.8	6532.7	23.6	13
	0.05	8	17717.2	12261.3	6751.2	21.1	10
	0.075	2	14993.8	9674.1	4949.4	21.6	28
	0.075	5	15108.7	11716.9	5432.7	22.7	22
	0.075	8	15473.9	11920.3	5748.1	24.1	16

Table 27. Weighted Method for $1/_{ij} = (1+\alpha_j)^{i-1} p_j, rm/ f(T_{\max}, U)$

Jobs numb.	Det. Rate	Rma time	$w_1 = 0.25$	$w_1 = 0.5$	$w_1 = 0.75$	Ave. Comp. Time (sec.)	Num. of rma
25	0.025	2	8857.2	6675.3	4964.2	6.2	8
	0.025	5	8909.1	6818.1	4997.6	6.1	6
	0.025	8	9121.7	6908.6	5271.9	9.7	5
	0.05	2	7995.3	5881.3	4495.4	14.5	12
	0.05	5	8093.1	5934.2	4214.6	20.1	10
	0.05	8	8380.6	5534.9	3984.3	20.2	9
	0.075	2	7307.8	4983.4	3439.5	9.9	15
	0.075	5	7495.5	4811.8	3692.4	20.2	13
	0.075	8	7693.9	4731.5	3934.5	21.2	10
50	0.025	2	25822.2	16346.7	13488.7	58.3	20
	0.025	5	26981.7	16984.6	14989.1	89.5	18
	0.025	8	27349.4	17964.2	15349.1	89.5	15
	0.05	2	22901.6	11594.7	8320.9	28.2	26
	0.05	5	23714.5	13688.7	9341.3	56.6	25
	0.05	8	26341.8	14143.8	9918.6	61.1	22
	0.075	2	19737.0	9985.2	8374.9	11.6	30
	0.075	5	23813.7	10671.1	8964.1	32.7	26
	0.075	8	25749.1	13462.7	10438.6	45.1	25

Tables 26 and 27 show a summary of the computational results. Problems with $w = 0.75$ have the smallest objective function. As rma time and deterioration rate get bigger, the objective function gets bigger for the same weights. If we fix rma time, the objective function increases as the deterioration rate increases. The decision manager can make his/her decisions quickly and control their manufacturing systems by choosing proper weights.

6. Conclusions

This is the first study on bi-criteria scheduling with deteriorating jobs and rate-modifying-activity. We address a real-life scheduling problem with periodic maintenance activity. In reality, scheduling maintenance will result in some jobs being tardy and a larger flow time. Thus, the bi-criteria used in this study are minimize total flow time with total tardiness, and minimize maximum tardiness with number of tardy jobs. Generally, mathematical models have not been used extensively for scheduling problems. In this study, all combinations are studied with the proposed mathematical programming model.

This is the first study which uses all three approaches to analyze the efficiency of the mathematical model with bi-criteria objectives. First, we use the model to find the Pareto Curve for both objectives, so a manager can make his decision from points on the curve. Then we use secondary objective method. Computational results show that the solution of the problem is dependent on the number of jobs and the other parameters of the problem. Although, 50 job problems are solved in around one minute, exponential growth in solution times makes larger problems much harder to solve. Lastly, we use weighted method to analyze model.

References

1. Azizoglu, M., Kondakci, S. and Kirca, O. (1991). Bicriteria scheduling problem involving total tardiness and total earliness penalties. *International Journal of Production Economics*, 23, 17-24.

2. Bacghi,U. and Ahmadi, R. H. (1987). An improved lower bound for minimizing weighted completion times with deadlines. *Operations Research*, 35, 311-313.

3. Baker K.R. and Scudder G.D. (1990). Sequencing with earliness and tardiness penalties: A Review. *Operations Research*, 38 (1), 22-36.

4. Bansal, S. P. (1980). Single machine scheduling to minimize the weighted sum of completion times with secondary criterion: A branch and bound approach. *European Journal of Operational Research*, 5, 177-181.

5. Browne S. and Yechiali U. (1990). Scheduling deteriorating jobs on a single processor. *Operations Research*, 38, 495-498.

6. Burns, R. N. (1976). Scheduling to minimize weighted sum of completion times with secondary criteria. *Naval Research Logistics Quarterly*, 23-1,125-129.

7. Chen, W.-J. (2007). Minimizing total flow time and maximum tardiness with periodic maintenance. *Journal of Quality in Maintenance Engineering*, 13-3, 293-303.

8. Chen, C. L. and Bulfin, R. L. (1993). Complexity of single machine, multi-criteria scheduling problems. *European Journal of Operational Research*, 70, 115-125.

10. C.L. Chen and R. L. Bulfin (1994). Scheduling a single machine to minimize two criteria: Maximum tardiness and number of tardy jobs. *IIE Transactions*, 26, 76-84.

11. Emmons, H. (1975). A note on a scheduling problem with dual criteria. *Naval Research Logistics Quarterly*, 22, 615-616.

12. Emmons, H. (1975). One machine sequencing to minimize mean flow time with minimum tardy. *Naval Research Logistics Quarterly*, 22, 585-592.

13. Graham, R.L., Lawler, E.L., Lenstra, J.K. and Rinnooy, K. A.H.G. (1979). Optimization and approximation in deterministic sequencing and scheduling: A Survey. *Annual Discrete Mathematics*, 5, 287–326.

14. H. Heck and Roberts S. (1972). A note on the extension of a result on scheduling with secondary criteria. *Naval Research Logistics Quarterly*, 19, 403-405.

15. Hino C.M., Ronconi D.P., Mendes A.B. (2005). Minimizing earliness and tardiness penalties in a single machine problem with a common due date. *European Journal of Operational Research*, 160,190-201.

16. Huo Y., Leung, J.Y.T. and Zhao, H. (2007). Bi-criteria scheduling problems: Number of tardy jobs and maximum weighted tardiness. *European Journal of Operational Research*, 177, 116-134.

17. Kondakci, S., Azizoglu, M. and Koksalan, M. (1996). Note: Bicriteria scheduling for minimizing flow time and maximum tardiness. *Naval Research Logistics*, 43, 929-36.

18. Lee, C.Y and Lin, C.S. (2001). Single-machine scheduling with maintenance and repair rate-modifying activities. *European Journal of Operational Research*, 135, 493-513.

17. Miyazaki, S. (1981). One machine scheduling problem with dual criteria. *Journal of the Operations Research Society of Japan*, 24-1, 37-50.

19. Mondal S.A and Sen A. (2001). Single machine weighted earliness-tardiness penalty problem with a common due date. *Computers and Operations Research*, 28, 649-669.

20. Nelson, R.T., Sarin, R.K. and Daniels, R.L. (1986). Scheduling with multiple performance measures: The one-machine case. *Management Science,* 32, 464-480.

21. Posner, M. E., (1985). Minimizing weighted completion times with deadlines. *Operations Research*, 33, 562-574.

22. Potts, C. N. and Van Wassenhove,L. N. (1983). An algorithm for single machine sequencing with deadlines to minimize total weighted completion time. *European Journal of Operational Research*, 12, 379-387.

23. Shanthikumar, J. G. and Buzacott, J. A. (1982). On the use of decomposition approaches in a single machine scheduling problem. *Journal of the Operations Research Society of Japan*, 25, 29-47.

24. Shanthikumar, J. G. (1983). Scheduling n jobs on one machine to minimize the maximum tardiness with minimum number tardy. *Computers and Operations Research*, 10, 255-266.

25. Smith, W.E. (1956). Various optimizers for single state production. *Naval Research Logistics Quarterly*, 3, 59-66.

26. Van Wassenhove, L. N. and Gelders, L.F. (1980). Solving a bicriterion scheduling problem. *European Journal of Operational Research*, 4, 42-8.

27. Wen-Jinn, C. (2007). Minimizing total flow time and maximum tardiness with periodic maintenance. *Journal of Quality in Maintenance Engineering*, 13 (3), 293-303.

CHAPTER 5

CONCLUSIONS

Minimizing completion time and makespan are important objectives for managers. The purpose of this dissertation is to increase producer and customer satisfaction by decreasing completion time of the jobs while considering some real life assumptions. To do this, we need to decide job positions and, if needed, when to do maintenance.

In this dissertation, we studied scheduling jobs and preventive maintenance under two new phenomena in the scheduling literature. First is *a deteriorated job.* Deteriorating jobs are tasks which need more time and effort to complete the process than when they are done earlier. Deterioration of the jobs is commonly due to machine wear. To prevent wear there is an activity called a *rate-modifying activity* (rma). Rma is also a new phenomena in scheduling problems. We presented a simple integer programming formulation to minimize makespan, total completion time, or total weighted completion to consider these two phenomena. In real life problems there may be many jobs, so we proposed a number of heuristic algorithms to solve larger problems in reasonable computational time.

In Chapter 2, the main concepts of our problem are introduced and a detailed overview of its literature is given. The main mathematical model is developed. We presented polynomial time algorithms for makespan objective when the jobs have equal processing times. We developed heuristics for both objectives. Three different experiments are done to analyze performance of the mathematical model and heuristics. As the computational experiments show, heuristics are very fast and provides close-to optimal solutions for both objectives. The behavior of the proposed heuristics performs well in terms of the execution time and the size of the problems.

In Chapter 3, an extended mathematical model is proposed. In this model, our processor is a worker who tires. We consider the physiological factors of the worker and added some new physiological constraints to our model. We proved that the problem and $1|\, p_{ij} = (1+\alpha_j)^{i-1} p_j, rm, phy \,|\, C_{\max}$ and $1|\, p_{ij} = (1+\alpha_j)^{i-1} p_j, rm, phy \,|\, \sum C_i$ is $NP-hard$. The reduction is based on the Equal-Size-Partition problem. For large problems, we developed an efficient heuristic for both makespan and total completion time objectives. According to numerical experiments, the heuristic is very effective for small to large size of problems and it yields much shorter run time when the problem size is large.

In Chapter 4, we provided a bi-criteria mathematical model under the same assumptions. The mathematical model is studied for two different objective sets; $1/\, p_{ij} = (1+\alpha_j)^{i-1} p_j, rm/T_{\max} \,|\, \sum U$ and $1/\, p_{ij} = (1+\alpha_j)^{i-1} p_j, rm/\sum\ F \,|\, \sum T$. To analyze the efficacy of the mathematical model, we use three different approaches. In the bi-criteria approach, the extreme points of the dominated set in the decision space on a Pareto Curve are generated. Then, a secondary approach is used to compare both objectives. Lastly, the weighted method is used to analyze the mathematical model.

In this dissertation, we have shown promising results for special scheduling problems. There is, of course, more research to be done. We would like to extend our problem to consider learning effect phenomena. Also, we could extend the problems to multiple machine settings. Lastly, possible research directions should be including other scheduling objectives and other processor environments.