A Model for Scheduling Deteriorating Jobs with
Rate-Modifying-Activities on a Single Machine

Yucel Ozturkoglu[1], Robert L. Bulfin[2], Emmett Lodree[3]

[1,2,3]Dept. of Industrial and Systems Engineering, Auburn University, Auburn, AL 36849 USA

Abstract

In this study we examine 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 increases or decreases depending on its initial processing time and other activities such as maintenance or repair of machine, break for workers, etc. Since we assume workers as processors in our study, a rate modifying activity (RMA) is a break given to workers in order to let them recover. The general problem we deal with in this study is to determine the work sequence, the number of breaks and their 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 an efficient heuristic algorithm to solve large size problems.

Keywords: Scheduling, Deteriorated Jobs, Rate-modifying-activity, Single Machine

---

This work is motivated by order picking in a distribution center. Most picking is done manually and ergonomic problems, exacerbated by fatigue, are often present. According to the Bureau of Labor Statistics (2009), in the "Warehousing and Storage" industry 'Sprains and Strains' type injuries averaged 46% of the total injuries within this industry (2003-2007). In terms of the event or exposure that led to all injuries within warehousing, "Overexertion" accounted for an average of 35% and was the most noted cause for injury. By scheduling breaks at appropriate times, the fatigue factor can be reduced, allowing workers to be more efficient and reduce ergonomic problems.
1

[1] Corresponding author Tel: 334 5522184 E-mail address:ozturyu@auburn.edu

1. Introduction

In the last four decades, scheduling researchers have 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.

The 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 caused by tiredness or fatigue of the worker is described as the deterioration rate of jobs. In the course of time, workers get tired both physically and mentally while they are doing their job.

The assumption about constant speed of machines or fixed process times has been changed by rate-modifying activities (RMA). The RMA, an activity which affects and changes the production rate of the machines, 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 instead of ordinary constant speed processing time. In our study we define RMA as a resting period in which workers stop doing their jobs for a given period of time. After an RMA is scheduled, the speed of a worker 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 $1 \mid p_{ij} = (1+\alpha_j)\, p_j, rm \mid \sum_{i=1}^{n} C_i$ and $1 \mid p_{ij} = (1+\alpha_j)\, p_j, rm \mid C_{\max}$. In these notations p, α, rm and C represent processing time, deterioration rate, RMA and completion time of jobs respectively.

The remainder of this paper is organized in six sections. In Section 2.2, a brief literature review is given. The problem definition is in Section 2.3. A model and a heuristic are found in Sections 2.4 and 2.5, respectively. Computational results are given in Section 2.6. 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 activity. 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 processing of a job is delayed then the required time to process that job increases linearly based on its initial process time. Hence, they constructed a scheduling problem with minimizing 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 with minimizing maximum lateness under linear deterioration. They presented two heuristics and they proved that the maximum lateness problem is NP-complete. 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.

RMA is a new phenomenon in scheduling problems in the last decade. In the scheduling literature, RMA is defined as 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. Therefore, one of the main decisions their model makes is to whether to stop the machine and fix the problem or to let it work with a lower production rate. Hence, they considered 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 breakdown. 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 were minimizing 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 obtain 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 equipments. These resources may not be available all the time. Thus, they considered the problem with a restricted RMA. They analyzed the computational complexity of both makespan and total completion 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 for 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 in the scheduling literature. Lodree and Geiger (2009) 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 (2009), the scheduling problem with the effects of deterioration and RMA has not been studied in the literature.

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 for a single worker. All jobs are available for processing at all times. The worker can do only one job at a time. Each job has a deterioration rate α which reflects a 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 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 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 deteriorated job j if done i positions after a RMA or initial position, i.e.

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

Model Variables:

x_{ijk} = 1 if job j is in the i^{th} position after RMA which is done just before position $k+1$, otherwise zero.

y_i = 1 if an RMA is done before position i, otherwise zero.

C_i Completion time of the job in position i.

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

- There is only one 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 worker recovers completely after an RMA (100% recovery).
- Deterioration process is the same after an RMA is scheduled.

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 breaks are needed?
3) Where are these breaks in the schedule?

The problem is $NP-hard$ since Kubiak and Velde (1998) proved that scheduling deteriorating jobs on a single machine with objective of minimize makespan is NP-hard. They defined a nonlinear deterioration for the jobs based on their positions. But, on the other hand, they did not consider rate-modifying-activity in their model. Hence, their proof is only based on the nonlinear deteriorated jobs. By comparing with their problem, our problem is special version of their problem which we also have RMA in our model. While trying to solve the optimal sequence of jobs to minimize makespan (or completion time) in our problem, we also try to find whether we assign any RMA or not. Also, if there is RMA, how many and where they should be placed. So, we can say that our problem is as hard as their problem because of additional decision processes. As a result, our problem is at least a $NP-hard$ problem.

4. The Mathematical Model

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

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

or

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

The related constraints with our model are given as follows.

$$C_1 = \sum_{j=1}^{n} p_{j1} x_{1j0} \tag{2.1}$$

$$C_i = C_{i-1} + \sum_{k=1}^{i} \sum_{j=1}^{n} p_{jk} x_{i,j,i-k} + q\, y_i \qquad i = 2,.....,n \tag{2.2}$$

$$\sum_{i=1}^{n} \sum_{k=0}^{i-1} x_{ijk} = 1 \qquad j = 1,.....,n \tag{2.3}$$

$$\sum_{k=0}^{i-1} \sum_{j=1}^{n} x_{ijk} = 1 \qquad i = 1,.....,n \tag{2.4}$$

$$x_{kji} \leq y_{i+1} \qquad k = 2.....,n; \quad j = 1,.....,n; \quad i = 1,.....,k-1 \tag{2.5}$$

$$x_{ijk} \in \{0,1\} \qquad i, j = 1,.....,n; \quad k = 0,.....,n \tag{2.6}$$

$$y_k \in \{0,1\} \qquad k = 2,.....,n \tag{2.7}$$

$$C_i \geq 0 \qquad i = 1,.....,n \tag{2.8}$$

The model given above decides how many RMAs should be done by. We explain all the model constraints below.

- In constraint (2.1), 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 (2.2), 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 (2.3), each job is assigned to exactly one position.
- In constraint (2.4), each position is scheduled for only one job.
- Constraint (2.5) 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 (2.6), (2.7) and (2.8) show that the variables are binary and non-negative.

5. The Heuristic Model for Problem of Makespan

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 efforts. Therefore, we propose a heuristic algorithm to solve large problems of minimizing makespan.

Let us define a set of given n jobs as $J=\{J_1, J_2, J_3, \ldots, J_n\}$.

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

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 Position number after RMA or initial position

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

$$R_{[k]} = \begin{cases} 1, & if\ an\ RMA\ is\ assigned\ to\ position\ k \\ 0, & otherwise \end{cases}$$

$P_{j[k]}$ Processing time of job j in position k as $P_{j[k]} = (1+\alpha)^{k-1} P_{j[1]}$

$C_{[k]}$ Completion time of the job in position k

The proposed heuristic is as follows:

Step 1: Order the jobs in ascending order (SPT) of processing times p_j

Step 2: Assign the job with largest processing time to position one and no RMA at the beginning

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

Set $C_{[1]} = P_{n[1]}$

$b = n - 1$ and $s = 1$.

Step 3: For $k = 2$ calculate:

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

If $D \geq q$ then; $R_{[k]} = 1$, $J_{[k]} = b$

Set $C_{[k]} = C_{[k-1]} + P_{b[1]} + q$,

$b = b - 1$ and $m = 1$

Otherwise go to step 4.

Step 4: $R_{[k]} = 0$, $J_{[k]} = s$

Set $C_{[k]} = C_{[k-1]} + P_{b[m]}$,

$s = s + 1$ and $m = m + 1$

Step 5: If $k = n$, Stop the algorithm, Makespan $= C_{[n]}$

This algorithm is $O(n \log n)$ where n is the number of the jobs. It is clear that Step1, which requires order the jobs based on SPT 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 happens just once and this iteration can be completed in $O(1)$.

The proposed heuristic model first applies the SPT 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 smallest processing time job. If this amount is greater than RMA time, 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.

6. Computational Results

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

To conduct our analyses on the proposed models, we identify four experimental factors: deterioration rate (α), RMA time (q), mean of 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 interactions and one four-factor interaction. 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. These are [18, 22], [10, 30], [1, 40], [36-44], [20-60], [1-80], [72-88], [40-120], [1-160]. We test our models for 50 jobs, and 810 instances (10 replications) are run 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 Pentium IV 2.8 GHz processor and 1GB of RAM. Ten replications of each of the possibilities were run for each performance measure. The average computational time of each of the possibilities for each problem is given in Table 6.1 and Table 6.2, respectively.

Table 6.1 Average Run Time (sec.) for Total Completion Time (n=50.)

α	**0.02**			**0.04**			**0.08**		
RMA time (q)	**5**	**10**	**15**	**5**	**10**	**15**	**5**	**10**	**15**
Avg. Run Time	58.3	91.5	119.5	28.2	56.6	61.1	11.6	32.7	45.1

Table 6.2 Average Run Time (sec.) for Makespan (n=50)

α	**0.02**			**0.04**			**0.08**		
RMA time (q)	**5**	**10**	**15**	**5**	**10**	**15**	**5**	**10**	**15**
Avg. Run Time	78	109.4	101.9	51.2	83.6	93.8	25.5	51.8	66.8

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 56.06 seconds and 73.55 seconds. As seen in the tables, when RMA time increases, run time for both models increases. We expect this result because increasing RMA time would let jobs deteriorate more. Hence, the algorithm compares the total required time, caused by deterioration, of more jobs than with smaller RMA times. For example, if RMA time is very small compared to the time of deteriorated jobs, we expect the model to give many RMAs. When the deterioration rate increases with RMA time constant, run time for both models decreases because of the same reason. Also, the problem of minimizing total completion time requires less time than that of makespan.

Experiment 2: Performance of the Heuristic Algorithm

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

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

In this equation, F_h is the makespan of the heuristic model and F_{opt} is the makespan of the optimal solution which is obtained by the mathematical model. When we test our models for 10 replications for each specific set of conditions, we obtain the average percentage error of the heuristic model which is given in the Table 6.3.

Table 6.3 Comparison of Proposed Heuristic and Mathematical Model for Makespan

			Ave. Error of Heuristic (%)		
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.

Table 6.4 Comparison of Proposed Heuristic and Mathematical Model for Total Completion Time

			Ave. Error of Heuristic (%)		
Mean	**Ranges of Process Time**	**α**	**RMA time**		
			5	**10**	**15**
		0.02	1.01	3.72	3.78
		0.04	0.74	2.01	2.82
	[18,22]	**0.08**	0.36	0.39	0.94
		0.02	1.05	2.09	4.12
		0.04	1.99	3.48	3.92
	[10,30]	**0.08**	0.51	0.53	2.99
		0.02	2.36	3.18	4.01
		0.04	1.2	4.27	3.09
20	**[1,40]**	**0.08**	0.43	0.48	2.19
		0.02	1.39	2.98	3.06
		0.04	0.21	1.09	3.28
	[36,44]	**0.08**	1.05	1.65	1.88
		0.02	0.49	4.27	4.39
		0.04	0.53	1.59	2.16
	[20,60]	**0.08**	0.19	2.19	3.14
		0.02	1.53	3.79	3.8
		0.04	1.02	2.91	3.15
40	**[1,80]**	**0.08**	0.57	3.06	3.05
		0.02	0.91	1.21	2.99
		0.04	1.32	1.48	2.14
	[72,88]	**0.08**	0.11	0.46	1.73
		0.02	0.95	1.97	2.22
		0.04	0.24	0.73	1.78
	[40,120]	**0.08**	0.12	0.61	1.39
		0.02	0.05	1.84	3.01
		0.04	0.31	1.16	1.98
80	**[1,160]**	**0.08**	0.34	0.88	2.05

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 RMA time increases, scheduling the jobs based on the order gives more error than scheduling the job by investigating each job separately. On the other hand, if the RMA is fixed and the deterioration rate raises, 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 is getting close. Hence, to decide the place of the RMA becomes less sensitive. Additionally, when the variation of the mean of processing time increases, the average percentage error of all instances in that group goes up.

Table 6.5 and Table 6.6 show the comparison of computational time for both 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 spends much less computation time than the proposed mathematical model.

Table 6.5 Comparison of Run Times for Makespan with 50 Jobs

		RMA time=10	
Ranges of Process Time	**α**	**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 6.6 Comparison of Run Times for Total Completion Time with 50 Jobs

		RMA time=10	
Ranges of Process Time	**α**	**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 test both models for 5 replications for 100 jobs. In this simulation, we assume that the deterioration rate is 0.08, the RMA time is 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.

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 6.7 shows the result of the experimental design for the problem of minimizing completion time.

Table 6.7 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	significant
RMA Time (q)	2	9015.83	4507.92	9343.96	0.00<0.05	significant
Mean (M)	2	13148.7	6574.35	13627.2	0.00<0.05	significant
Variance (V)	2	159.9	79.95	165.72	0.00<0.05	significant
α *q	4	508.9	127.23	263.71	0.00<0.05	significant
α *M	4	878.39	219.6	455.18	0.00<0.05	significant
α *V	4	433.77	108.44	224.78	0.00<0.05	significant
q *M	4	441	110.25	228.52	0.00<0.05	significant
q * V	4	311.49	77.87	161.41	0.00<0.05	significant
M* V	4	368.91	92.23	191.17	0.00<0.05	significant
α *q*M	8	67.8	8.48	17.57	0.00<0.05	significant
α *q*V	8	731.8	91.48	189.61	0.00<0.05	significant
α *M*V	8	918.42	114.8	237.96	0.00<0.05	significant
q *M*V	8	772.51	96.56	200.16	0.00<0.05	significant
α *q*M*V.	16	1490.8	93.18	193.13	0.00<0.05	significant
Error	729	351.7	0.48			
Total	809	45100				
	R^2 = 99.22%					

Table 6.7 clearly indicates that 99.22% of the variation in number of RMA, which is a main variable of the model, is explained by all factors. According to the experimental design results, there are significant differences among deterioration rate, RMA time, mean of processing times, variance of mean of processing times and their interactions at the level of significance 0.05. If, one of the factors is changed, the optimal number of RMA and their place in the sequence changes, too. Also, the change in the interactions between the model factors affects the result to a lesser degree.

Table 6.8 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	significant
RMA Time (q)	2	258126	129063	72.90	0.00<0.05	significant
Mean (M)	2	488619	244310	138.00	0.00<0.05	significant
Variance (V)	2	89859	44930	25.38	0.00<0.05	significant
α *q	4	7338	1834	1.04	0.388<0.05	significant
α *M	4	5565	1391	0.79	**0.535**	Not significant
α *V	4	12511	3128	1.77	0.134<0.05	significant
q *M	4	4678	1169	0.66	**0.620**	Not significant
q * V	4	16207	4052	2.29	0.058	significant
M* V	4	71023	17756	10.03	0.00<0.05	significant
α *q*M	8	27955	3494	1.97	0.047<0.05	significant
α *q*V	8	10868	1358	0.77	**0.632**	Not significant
α *M*V	8	43040	5380	3.04	0.002<0.05	significant
q *M*V	8	19171	2396	1.35	0.214<0.05	significant
α *q*M*V.	16	55393	3462	1.96	0.014<0.05	significant
Error	729	1290563	1770			
Total	809	2833916				

R^2 = **54.46**%

Table 6.8 shows that 54.46% of the variation in total run time of the model is explained by all factors. But the interaction of some factors is not significantly different.

7. Summary

This paper investigates a scheduling problem with deteriorating jobs and RMA 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. Secondly, we propose a heuristic model for makespan. Then, we present some computational experiments. According to the experimental results, the performance of the proposed model and the heuristic are quite satisfactory. We show that, when the number of jobs increases the computational time to solve the problem increases dramatically for the mathematical model. Besides, the heuristic model gives a reasonable error for 50 jobs when compared to the mathematical model.

REFERENCES

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.

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

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

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.

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

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

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

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

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

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

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

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.

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

Lodree, E. J. and Geiger, C.D. (2009) A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration,European Journal of Operational Research, In Press..

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

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

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.

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, Ann. Discrete Math**.,** 5, 287–326.