Scheduling Problem with Human Energy Recovery Function under Nonlinear Time-dependent Deterioration

Hua-Ping Wu, *Member, IAENG,* Min Huang, Vincent Cho, W. H. Ip, and Qun-Lin Fan

Abstract—**In this paper, minimizing makespan for single-machine scheduling problem with human energy recovery function under nonlinear time-dependent deterioration is considered. Firstly, a human recovery function is proposed according to the length of a rate-modifying activity (RMA) (i.e. the length of the rest time). Meanwhile, it is the first paper to introduce the recovery function into a scheduling problem. And the problem is proved to be an NP-hard problem. Then, a special case of the problem is proved to be solved in polynomial time.**

Index Terms—**scheduling, rate-modifying activity, time-dependent deterioration, human recovery function, makespan, NP-hard problem.**

I. Introduction

In this paper, minimizing makespan for single-machine scheduling problem with human energy recovery function under nonlinear time-dependent deterioration is considered. In classical scheduling problems, a machine always works efficiently and effectively. However, this does not fit in the life, even in the case with using the human operators. For example, in the process of loading and unloading cargos in ramp service, the processing time of a job is affected by human operators energy as this process is usually operated by human. It is easy to find that the total processing time of previous jobs and the deterioration rate of human operators have direct effects upon the processing time of later jobs. Hence, it should be more reasonable to describe the processing time of a job as a given constant that follows a nonlinear deterioration instead of linear one. More importance is that the length of break time plays a key role in the process of recovery. By affecting directly the extent of recovery it affects the efficiency of the whole services finally. Therefore, in this paper, the human energy recovery function is the first proposed. The scheduling problem with human energy recovery function under a nonlinear time-dependent deterioration is modeled as a single machine scheduling problem with recovery processing, i.e., a group of human operators is treated as a single machine, while airlines cargos are regarded as jobs which need to be processed. Moreover, for the general scheduling problems, the earlier the service is completed, the more efficient it is. So, the decision maker would more likes to seek minimal makespan as their goal.

Hence, in this paper, a single machine scheduling problem with human energy recovery function under nonlinear time-dependent deterioration is considered. A rational human recovery function is the first proposed to be integrated into deterioration jobs, RMA, and human operators fatigued to minimize makespan. Obviously, this research provides a reference for other practical scheduling problems particularly involved labors, such as maintenance scheduling, cleaning assignments and so on.

The remainders of this paper are organized as follows. Literature reviews are given in section 2. In section 3, the problem and the recovery function are presented and formulated. In section 4, the problem of minimizing makespan for single-machine scheduling with human energy recovery function under nonlinear time-dependent deterioration is proved to be an NP-hard. The special case is considered in section 5. And the conclusions are given in the last section.

Manuscript received February 1, 2015; revised May 15, 2015. This work was supported by the National Science Foundation for Distinguished Young Scholars of China under Grant No. 71325002, No. 61225012; the National Natural Science Foundation of China under Grant No.71071028; the Fundamental Research Funds for State Key Laboratory of Synthetical Automation for Process Industries under Grant No.2013ZCX11, the Humanity and Social Science Youth foundation of Ministry of Education of China under Grant No. 14YJC630032, the Scientific and Technological Research Program of Chongqing Municipal Education Commission under Grant No. KJ1400913, the Spark Program of Youth Research Program of Chongqing University of Technology, and the Scientific Research Foundation of Chongqing University of Technology (20014Z007).

H.P. Wu is with Accounting R&D Center, Chongqing University of Technology, Chongqing, 400054, China, e-mail: hpwu@cqut.edu.cn.

M. Huang is with School of Information Science and Engineering, Northeastern University. V. Cho is with the Department of Management and Marketing, The Hong Kong Polytechnic University. W. H. Ip is with Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University. Q.L. Fan is with School of Management, Chongqing University of Technology.

II. Literature Reviews

Deterioration jobs scheduling problem is first introduced independently by Gupta and Gupta[1] and by Browne and Yechiali[2]. Since then, related models have been extensively studied from variety of perspectives. For instance, Wang et al.[3] present single-machine scheduling with deteriorating jobs in which the jobs are subject to a series-parallel graph constraint and prove that the problem of minimizing the makespan and the total weighted completion time can be solved in polynomial time. Toksari and Güner[4] focus on analyzing parallel machine earliness/tardiness scheduling problem with linear deterioration and simultaneous effects of learning. Cheng and Sun[5] consider the problem where the processing time of a job is a linear function of its starting time and jobs can be rejected by paying penalties. They show that the problems of minimizing the makespan, the total weighted completion time and the maximum lateness/tardiness plus the total penalty of the rejected jobs are NP-hard. Hence, they apply dynamic programming to solve the related problem. Li et al.[6] present polynomial-time algorithms to solve the problem that its objective is to determine the optimal due dates and the processing sequence simultaneously to minimize costs for earliness, due date assignment and weighted number of tardy jobs, in which the actual processing time of a job is a linear increasing function of its starting time. Ng et al.[7] also study the problem of scheduling deteriorating jobs with release dates on a single machine. These papers all focus on linear deterioration jobs. A few of papers refer to the nonlinear deterioration jobs. Cheng et al.[8] investigate the problem of common due-window assignment and scheduling of deteriorating jobs and a maintenance activity simultaneously on a single-machine, in which the job actual processing times are assumed to follow a linear time-dependent deteriorating model. And they provide polynomial time solutions for the objective of the problem to simultaneously minimize the earliness, tardiness, due-window starting time and due-window size costs. Kuo and Yang[9] introduce a time-dependent learning effect to a single-machine scheduling problem based on the notion that the more one practices, the better one learns. In this regard, the processing times of the subsequent jobs are smaller than their normal processing times because of the learning effect. They define a time-dependent learning effect as follows. Let p_{ir} be the actual processing time of $J_i(i = 1, 2, \cdots, n)$ if it was scheduled in position r in a sequence. $p_{[r]}$ is the normal processing time of a job if scheduled in the r^{th} position of a sequence. And p_i is the normal (sequence-independent) processing time of job J_i. Then $p_{ir} = (1 + p_{[1]} + p_{[2]} + \cdots + p_{[r-1]})^a p_i = (1 + \sum_{k=1}^{r-1} p_{[k]})^a p_i$, where $a \leq 0$ is a constant learning index. According to a time-dependent learning effect introduced by Kuo and Yang[9], Wang et al.[10] consider a single machine scheduling problem with a nonlinear time-dependent deterioration. They define the actual processing times as follows. $p_{ir} = (1 + p_{[1]} + p_{[2]} + \cdots + p_{[r-1]})^a p_i = (1 + \sum_{k=1}^{r-1} p_{[k]})^a p_i$, where $a \geq 0$ is a constant deterioration index. p_{ir} is the actual processing time of $J_i(i = 1, 2, \cdots, n)$ if it is scheduled at position r in a sequence. $p_{[r]}$ is the normal processing time of a job if scheduled in the r^{th} position in a sequence. And p_i is the normal (sequence-independent) processing time of job J_i. They show that the single-machine makespan problem remains polynomially solvable under the proposed model. Wu and Cheng[11] address that the actual processing time of a job is a decreasing function of the sum of processing times based learning or increasing function of the sum of processing times based deteriorating effect to solve a two-agent single-machine scheduling problem. Li and Fan[12] address the nonresumable version of the scheduling problem with proportionally deteriorating jobs subject to availability constraints to minimize the total weighted completion time. Wu and Huang [13] investigate an NP-hard problem of single-machine scheduling problem with deteriorating jobs and different due dates to minimize total tardiness, and propose EDA to solve it. Wang et al.[14] consider single-machine scheduling problem with deterioration jobs in which the actual processing time of a job is a function of its position in a sequence, its starting time, and its resource allocation. Lee[15] provide the optimal schedules for the problem of single-machine scheduling with past-sequence-dependent setup times and general effects of deterioration and learning. And, besides that, another scholars study other scheduling problem with variety of processing time. For example, Wang and Choi[16] considered makespan minimization of a flexible flow shop scheduling problem with stochastic processing times. Lee[17] studied the scheduling problem with learning effect and setup time. Lai and Lee[18] discussed the scheduling problems with learning effect and forgetting effect.

Due to the characteristics of time-dependent deterioration in the loading and unloading cargos process, having a rest should be also considered. Moreover, having a rest implies that it modifies a performance rate of the machine. In this paper, it is also regarded as a rate-modifying activity (RMA). Lee and Leon[19] first consider single machine scheduling with a rate-modifying activity for a problem commonly found in electronic assembly lines. They provided polynomial and pseudo-polynomial algorithms for a number of objective functions. Motivated by a RMA, Wang and Wang[20] suggest single machine SLK due date assignment scheduling problem with a rate-modifying activity. A R-MA is introduced into a scheduling problem with job-dependent learning effects by Ji and Cheng[21] and parallel machine scheduling problem by Wang and Wei[22]. Rebai et al.[23] consider maintenance tasks the same as RMA assigned to multiple single machines . Lodree and Geiger[24] integrate time-dependent processing time/RMA framework into machine environments characterized by deteriorating machine performance and rate-modifying maintenance activities. They assume that the normal processing time of each job is equal to 1, the task actual processing times are represented as a variation of simple linear deterioration and a single machine always fully restores after a break time. Zhao and Tang[25] consider a single machine scheduling and due-window assignment problem, where the normal processing time is a fixed value and the actual processing time of a job is a linear function of its starting time and the job-independent deterioration rates are identical for all jobs. The objective is to schedule the jobs, the due-widow and the rate-modifying activity so as to minimize the sum of earliness, tardiness and due-window starting time and due-window size costs. Öztürkoğlu et al.[26] determined an optimal sequence with the optimal number and the positions of RMAs.

However, research on the detailed recovery processing has not been studied until now. To the best of our knowledge, this paper is the first study to propose a rational recovery function integrated into deterioration jobs, RMA, and human operators fatigued.

III. Problem Formulation

The problem in this paper can be formally described as follows: There are n independent jobs $J = \{J_1, J_2, \cdots, J_n\}$ to be processed non-preemptively on a single machine. Assume that all jobs are available at the same time, and have the same processing time if they were processed at the beginning of a sequence. Otherwise, in order to improve the efficiency of the machine, a resting time is inserted into a sequence. The problem is where and how long the resting time should be. The detailed notations are as follows:

n: The number of independent jobs to be processed on a single machine;

$J = \{J_1, J_2, \cdots, J_n\}$: The set of jobs;

p_i: The normal processing time for every job ($p_i > 0$);

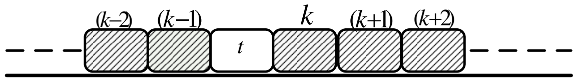

Fig. 1. The position of a maintaining time

p_{ir}: The actual processing time of job $J_i (J_i \in J)$ if it is scheduled in position r $(1 \le r \le n)$ in a given sequence;

$p_{[r]}$: The normal processing time for job scheduled in position r, when J_i is scheduled in position r, $p_i = p_{[r]}$.

b: The deterioration rate ($b > 0$);

k: A resting time is inserted in the front of the k^{th} job processed in a sequence (such as Figure 1);

T: The duration of resting for a single machine to be fully restored;

t: The duration of a resting time ($0 \le t \le T$);

$C_i(k,t)$: The completion time of job $J_i (1 \le i \le n)$ associated with scheduling an RMA in the k^{th} position of a sequence, where $1 < k \le n$, because it is never optimal to schedule the RMA in the first sequence position $k = 1$ [24];

$C_{\max}(k,t)$: The makespan associated a resting time in the position k $(1 < k \le n)$ of a sequence.

According to Wang et al.[10], if the resting time was not considered, the actual processing time of job J_{ir} could be described as a nonlinear function as follows:

$$p_{ir} = p_i[1 + p_{[1]} + \ldots + p_{[r-1]}]^b \ , \qquad 0 < r \le n \qquad (1)$$

However, if the resting time was scheduled, the processing times of jobs after resting will be affected by the duration t of resting. Suppose that the machine will fully recover if and only if $t = T$ where T depends on the nature of the problem. Then the recovery function can be defined.

Definition 3.1. The recovery function is defined as:

$$\begin{array}{c} F(t) = \frac{t}{T} p_{[r]}[(1 + p_{[k]} + p_{[k+1]} + \ldots + p_{[r-1]})^b - \\ (1 + p_{[1]} + p_{[2]} + \ldots + p_{[r-1]})^b] \end{array} \qquad (2)$$

Where $k \le r \le n$.

Obviously, it is a linear function of recovery time t, that is

(1) If $t = 0$, then $F(t) = 0$. It implies that the machine has no chance for recovering capacity. Namely, there is no rest time set;

(2) If $t = T$, then $F(t) = p_{[r]}[(1 + p_{[k]} + p_{[k+1]} + \ldots + p_{[r-1]})^b - (1 + p_{[1]} + p_{[2]} + \ldots + p_{[r-1]})^b]$. It indicates that the capacity of a machine can recover entirely when the length of a rest time T is scheduled in the sequence;

(3) If $0 < t < T$, then $F(t) = \frac{t}{T} p_{[r]}[(1 + p_{[k]} + p_{[k+1]} + \ldots + p_{[r-1]})^b - (1 + p_{[1]} + p_{[2]} + \ldots + p_{[r-1]})^b]$. It describes the amount of recovery that the machine captured when the length of a rest time t $(0 < t < T)$ is scheduled in the sequence. It is clear that $F(T) < F(t) < 0$.

Hence, according to the recovery function, when $0 \le t \le T$, the actual processing time of job p_{ir} can be described as follows:

$$p_{ir} = \begin{cases} p_i[1 + p_{[1]} + \ldots + p_{[r-1]}]^b & 0 < r < k \\ p_i[1 + p_{[1]} + \ldots + p_{[r-1]}]^b + F(t) & k \le r \le n \end{cases} \qquad (3)$$

Based on Equation (3), we have the following scenarios:

(1) When $t = 0$, $p_{ir} = p_i(1 + p_{[1]} + p_{[2]} + \ldots + p_{[r-1]})^b$. It denotes that a resting time is not inserted in a sequence and the actual processing times of jobs are increasing according to the job sequence;

(2) When $0 < t < T$, $p_{ir} = p_i[1 + p_{[1]} + p_{[2]} + \ldots + p_{[r-1]}]^b + F(t)$. It implies all the jobs after the resting time will be processed more efficiently;

(3) When $t = T$, $p_{ir} = p_i[1 + p_{[k]} + p_{[k+1]} + \ldots + p_{[r-1]}]^b$. When the resting duration is equal to T, then the machine is fully restored and the processing time of the first job after rest is equal to the normal processing time of it.

So the problem is how to assign jobs, how to schedule t, how long t should be and where t should be located to optimize the given objectives.

In the following sections, we consider the problem of minimizing the makespan with recovery function on a single machine under nonlinear time-dependent deterioration. We denote it as $1\left|p_{ir}, rm, rp\right|C_{\max}$ and $1\left|p_{ir}, rm\right|\sum_{i=1}^{n} C_i$ by using the three-field notation scheme $\alpha\left|\beta\right|\gamma$ introduced by Graham et al.[27], where represents the resting time or time of maintenance that would modify a processing rate of a machine, i.e. an RMA, and rp the recovery processing.

IV. The Problem of $1\left|p_{ir}, rm, rp\right|C_{\max}$

In this section, the problem of $1\left|p_{ir}, rm, rp\right|C_{\max}$ is proved to be an NP-hard problem.

Theorem 4.1. The problem $1\left|p_{ir}, rm\right|C_{\max}$ is an NP-hard problem.

Proof. We transform the partition problem to our problem.

Partition problem

Instance. A finite set of positive integers $x_i \in X$ $(i = 1, 2, \ldots 2n)$ and A with $\sum_{i=1}^{2n} x_i = 2A$, where $\frac{A}{2n} < x_i < \frac{3A}{2n}$ and $A > \sqrt{2}$.

Question. Can this set X be partitioned into two disjoint subsets X_1 and X_2 such that the sum of each subset is equal to $(\sum_{x_i \in X_1} x_i = \sum_{x_i \in X_2} x_i = A)$?

We construct the following instance of the scheduling problem. These are $2n + 2$ jobs. Among them there are $2n$ partition jobs $a_i (i = 1, 2, \ldots, 2n)$, and enforcer jobs a_{e1} and a_{e2} with the following parameters:

$$a_i = H + x_i, i = 1, 2, \ldots, 2n, \ x_i \in X$$

$$a_{e1} = a_{e2} = 3A$$

where $H = A$, the duration time of an RMA $t = 1$, and the deterioration rate $b = 2$.

Calculating $y = 2nH + 3A + 6A(1 + A)^2 + W$, where

$$\begin{array}{l} W{=}2 \sum\limits_{x_i, x_j \in X \wedge i<j} (H + x_i)(H + x_j) + \\ \qquad \sum\limits_{x_i, x_j \in X \wedge i<j} (H + x_i)^2 (H + x_j) \end{array}$$

We show that the partition problem has a solution if and only if there exists a solution to construct the instance of $1\left|p_{ir}, rm\right|C_{\max}$ with value $C_{\max} \le y$.

Only if ($\Rightarrow$)

Assume that a partition exists. Let X_1 and X_2 denote the disjoint subsets, i.e., $\sum_{x_i \in X_1} x_i = \sum_{x_i \in X_2} x_i = A$. There are

l jobs in subset X_1. Assign a_{e1} and a_{e2} as the last jobs in subsets X_1 and X_2, respectively. In such a schedule, we have

$$\begin{aligned} C_{\max} &= a_{[1]} + a_{[2]}(1+a_{[1]})^2 + \ldots + \\ & a_{[l]}(1+a_{[1]}+\ldots+a_{[l-1]})^2 + \\ & a_{e1}(1+\sum_{i=1}^{l} a_i)^2 + 1 + a_{[l+2]} + \\ & a_{[l+3]}(1+a_{[l+2]})^2 + \ldots + \\ & a_{[2n+1]}(1+a_{[l+2]}+\ldots+a_{[2n]})^2 + a_{e2}(1+\sum_{i=l+2}^{2n+1} a_i)^2 \\ &= 2nH + 2A + W + 1 + 6A(1+A)^2 \\ &< 2nH + 3A + W + 1 + 6A(1+A)^2 \end{aligned} \tag{4}$$

Hence, $C_{\max} < 2nH + 3A + W + 1 + 6A(1+A)^2 = y$ for $i = 1, 2, \ldots, 2n$.

If ($\Leftarrow$)

Assume that the partition problem has no solution. Then, we have

$$\begin{aligned} C_{\max} &= a_{[1]} + a_{[2]}(1+a_{[1]})^2 + \ldots + \\ & a_{[l]}(1+a_{[1]}+a_{[2]}+\ldots+a_{[l-1]})^2 + \\ & a_{e1}(1+\sum_{i=1}^{l} a_i)^2 + 1 + a_{[l+2]} + \\ & a_{[l+3]}(1+a_{[l+2]})^2 + \ldots + \\ & a_{[2n+1]}(1+a_{[l+2]}+\ldots+a_{[2n]})^2 + \\ & a_{e2}(1+\sum_{i=l+2}^{2n+1} a_i)^2 \\ &= 2nH + 2A + W + 1 + 3A(1+\sum_{x_i\in X_1} x_i)^2 + \\ & 3A(1+\sum_{x_i\in X_2} x_i)^2 \end{aligned} \tag{5}$$

Suppose that $\sum_{x_i\in X_1} x_i = D\,(D \neq A)$, $\sum_{x_i\in X_2} x_i = D'\,(D' \neq A)$, then $D + D' = A$ and Equation (5) can be expressed as follows:

$$\begin{aligned} C_{\max} &= 2nH + 2A + W + 1 + \\ & 3A(1+D)^2 + 3A(1+D')^2 \\ &= y + A[(D^2 + D'^2) - 1] + \tfrac{A(D-D')^2}{2}. \end{aligned} \tag{6}$$

Since $A > \sqrt{2}$, $D^2 + D'^2 > 1$.

Therefore, $C_{\max} = y + A[(D^2 - D'^2) - 1] + \frac{A(D-D')^2}{2} > y$.

Since the makespan is greater than the required value y, our scheduling problem has no solution, which ends the proof.

Therefore, the partition problem has a solution if and only if there exist a solution to the construct instance of $1\,|p_{ir}, rm\,|C_{\max}$ with value $C_{\max} \le y$.

According to the above Theorem 4.1, we obtain the following corollary.

Corollary 4.1. The problem $1\,|p_{ir}, rm, rp\,|C_{\max}$ is an NP-hard problem.

Proof. When $t = 0$, the problem $1\,|p_{ir}, rm, rp\,|C_{\max}$ becomes $1\,|p_{ir}, rm\,|C_{\max}$. Clearly, $1\,|p_{ir}, rm\,|C_{\max}$ is a special instance of $1\,|p_{ir}, rm, rp\,|C_{\max}$. Since the problem $1\,|p_{ir}, rm\,|C_{\max}$ is an NP-hard problem, the problem $1\,|p_{ir}, rm, rp\,|C_{\max}$ is also an NP-hard problem.

The proof is end.

V. The Special Case of $1\,|p_{ir}, rm, rp, p_i = p\,|C_{\max}$

In this section, we derive the optimal policy for assigning a suitable length of t (a resting time) in an appropriate position k when all job identical normal processing time and the objective of minimizing the makespan. In subsection 5.1, preliminary analysis is given. In subsection 5.2, the polynomial algorithm is presented, followed by numerical experiments in subsection 5.3.

A. Preliminary Analysis

In a sequence, while inserting an RMA at position k in a sequence, the makespan can be expressed as follows:

$$\begin{aligned} C_{\max}(k,t) &= p\sum_{s=0}^{k-2}(1+sp)^b + \tfrac{t}{T}p\sum_{s=0}^{n-k}(1+sp)^b + \\ & t + (1-\tfrac{t}{T})p\sum_{s=k-1}^{n-1}(1+sp)^b \end{aligned} \tag{7}$$

Here, $p\sum_{s=0}^{k-2}(1+sp)^b$ denotes the total processing times of the front $(k-1)$ jobs; t is a resting time; $\frac{t}{T}p\sum_{s=0}^{n-k}(1+sp)^b + (1-\frac{t}{T})p\sum_{s=k-1}^{n-1}(1+sp)^b$ describes the total processing times of the rest $(n-k+1)$ jobs. Clearly, the makespan is related to a position k and a resting time t. The problem can be accomplished by determining the value of k and t to minimize Equation (7). To determine the value of k and t, we propose the following propositions and theorems.

Proposition 5.1. For any k $(1 < k \le n)$ and t $(0 < t \le T)$, if $C_{\max}(k,t) < C_{\max}(k+1,t)$, then $C_{\max}(k,t) < C_{\max}(k+1,t) < \cdots < C_{\max}(n,t)$; if $C_{\max}(k,t) < C_{\max}(k-1,t)$, then $C_{\max}(k,t) < C_{\max}(k-1,t) < \cdots < C_{\max}(2,t)$.

Proof. For decided k and t, the makespan can be expressed as follows:

$$\begin{aligned} C_{\max}(k,t) &= p\sum_{s=0}^{k-2}(1+sp)^b + \tfrac{t}{T}p\sum_{s=0}^{n-k}(1+sp)^b + \\ & t + (1-\tfrac{t}{T})p\sum_{s=k-1}^{n-1}(1+sp)^b \end{aligned} \tag{8}$$

$$\begin{aligned} C_{\max}(k+1,t) &= p\sum_{s=0}^{k-1}(1+sp)^b + \tfrac{t}{T}p\sum_{s=0}^{n-k-1}(1+sp)^b + \\ & t + (1-\tfrac{t}{T})p\sum_{s=k}^{n-1}(1+sp)^b \end{aligned} \tag{9}$$

$$\begin{aligned} C_{\max}(k+2,t) &= p\sum_{s=0}^{k}(1+sp)^b + \tfrac{t}{T}p\sum_{s=0}^{n-k-2}(1+sp)^b + \\ & t + (1-\tfrac{t}{T})p\sum_{s=k+1}^{n-1}(1+sp)^b \end{aligned} \tag{10}$$

Using Equations (8), (9) and (10), we obtain:

$C_{\max}(k,t) - C_{\max}(k+1,t) = \frac{t}{T}p[(1+(n-k)p)^b - (1+(k-1)p)^b]$ and $C_{\max}(k+1,t) - C_{\max}(k+2,t) = \frac{t}{T}p[(1+(n-k-1)p)^b - (1+kp)^b]$.

If $C_{\max}(k,t) < C_{\max}(k+1,t)$, we have

$$\frac{t}{T}p(1+(n-k)p)^b < \frac{t}{T}p(1+(k-1)p)^b.$$

Since $T > 0$, $b > 0$, $p > 0$ and $t > 0$, then $\frac{t}{T}p(1+(n-k-1)p)^b < \frac{t}{T}p(1+(n-k)p)^b < \frac{t}{T}p(1+(k-1)p)^b < \frac{t}{T}p(1+kp)^b$, i.e., $\frac{t}{T}p(1+(n-k-1)p)^b < \frac{t}{T}p(1+kp)^b$ and $C_{\max}(k+1,t) < C_{\max}(k+2,t)$.

Similarly, we can obtain $C_{\max}(k,t) < C_{\max}(k+1,t) < \cdots < C_{\max}(n,t)$.

An analogous proof holds if $C_{\max}(k,t) < C_{\max}(k-1,t)$, then $C_{\max}(k,t) < C_{\max}(k-1,t) < \cdots < C_{\max}(2,t)$.

This concludes the proof.

Proposition 5.2. If an RMA is scheduled, then $T < \underset{1<k\leq n}{Min}\{p\sum_{s=k-1}^{n-1}[(1+sp)^b - (1+(s-(k-1))p)^b)]\}$.

Proof. Through the previous description, the RMA should be assigned to position k in a sequence only if the resulting makespan less than that of associating with not scheduling the RMA, the makespan associated with not scheduling an RMA is:

$$\begin{aligned} C_{\max}(k,t=0) = p + p(1+p)^b + p(1+2p)^b + \\ \cdots + p(1+(n-1)p)^b \end{aligned} \tag{11}$$

Therefore, the RMA is scheduled only if $C_{\max}(k,t) < C_{\max}(k,t=0)$. Using Equations (7) and (11), $C_{\max}(k,t) < C_{\max}(k,t=0)$ becomes

$$T < \underset{1<k\leq n}{Min}\{p\sum_{s=k-1}^{n-1}[(1+sp)^b - (1+(s-(k-1))p)^b)]\}.$$

Hence, the proposition is proved.

Theorem 5.1. For any t $(0 < t \leq T)$, the optimal policy for scheduling an RMA of length t for all jobs is as follows: If n is an even integer and $T < p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b - (1+(s-\frac{n}{2})p)^b]$, assign the RMA to sequence position $k = \frac{n}{2}+1$; If n is an odd integer and $T < p\sum_{s=\frac{n+1}{2}}^{n-1}[(1+sp)^b - (1+(s-\frac{n+1}{2})p)^b]$, assign the RMA to sequence position $k = \frac{n+1}{2}$ or $k = \frac{n+1}{2}+1$. Otherwise, do not schedule the RMA.

Proof. (For an even n) According to Equation (7),

$$\begin{aligned} C_{\max}(\tfrac{n}{2}+1,t) = p\sum_{s=0}^{\frac{n}{2}-1}(1+sp)^b + t + \\ \tfrac{t}{T}p\sum_{s=0}^{\frac{n}{2}-1}(1+sp)^b + \\ (1-\tfrac{t}{T})p\sum_{s=\frac{n}{2}}^{n-1}(1+sp)^b \end{aligned} \tag{12}$$

$$\begin{aligned} C_{\max}(\tfrac{n}{2}+2,t) = p\sum_{s=0}^{\frac{n}{2}}(1+sp)^b + t + \\ \tfrac{t}{T}p\sum_{s=0}^{\frac{n}{2}-2}(1+sp)^b + \\ (1-\tfrac{t}{T})p\sum_{s=\frac{n}{2}+1}^{n-1}(1+sp)^b \end{aligned} \tag{13}$$

$$\begin{aligned} C_{\max}(\tfrac{n}{2},t) = p\sum_{s=0}^{\frac{n}{2}-2}(1+sp)^b + t + \\ \tfrac{t}{T}p\sum_{s=0}^{\frac{n}{2}}(1+sp)^b + \\ (1-\tfrac{t}{T})p\sum_{s=\frac{n}{2}-1}^{n-1}(1+sp)^b \end{aligned} \tag{14}$$

Since $T>0$, $b>0$, $p>0$ and $t>0$, then $C_{\max}(\frac{n}{2}+1,t) - C_{\max}(\frac{n}{2}+2,t) = \frac{t}{T}p[(1+(\frac{n}{2}-1)p)^b - (1+\frac{n}{2}p)^b] < 0$, i.e., $C_{\max}(\frac{n}{2}+1,t) < C_{\max}(\frac{n}{2}+2,t)$; and $C_{\max}(\frac{n}{2}+1,t) - C_{\max}(\frac{n}{2},t) = \frac{t}{T}p[(1+(\frac{n}{2}-1)p)^b - (1+\frac{n}{2}p)^b] < 0$, i.e., $C_{\max}(\frac{n}{2}+1,t) < C_{\max}(\frac{n}{2},t)$.

By Proposition 5.1, we have

$$\begin{aligned} C_{\max}(\tfrac{n}{2}+1,t) < C_{\max}(\tfrac{n}{2}+2,t) < \\ C_{\max}(\tfrac{n}{2}+3,t) < \cdots C_{\max}(n,t) \end{aligned} \tag{15}$$

$$\begin{aligned} C_{\max}(\tfrac{n}{2}+1,t) < C_{\max}(\tfrac{n}{2},t) < \\ C_{\max}(\tfrac{n}{2}-1,t) < \cdots < C_{\max}(2,t) \end{aligned} \tag{16}$$

Equations (15) and (16) imply that the minimum makespan obtains when $k = \frac{n}{2}+1$.

Now the RMA should be assigned to position k in a sequence only if the resulting makespan is less than the makespan associated with not scheduling the RMA.

Therefore, the RMA is scheduled only if $C_{\max}(k,t) < C_{\max}(k,t=0)$. Using Equations (11) and (12), $C_{\max}(k,t) < C_{\max}(k,t=0)$ becomes

$$T < p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b - (1+(s-\tfrac{n}{2})p)^b].$$

The proof for odd is analogous. This proof ends.

Theorem 5.2. For any t $(0 < t \leq T < p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b - (1+(s-\frac{n}{2})p)^b])$ and k, the optimal policy for scheduling an RMA of length t for all jobs is as follows: If n is an even integer, assign the RMA to position $k = \frac{n}{2}+1$ in a sequence and the length $t = T$; If n is an odd integer, assign the RMA to position $k = \frac{n+1}{2}$ or $k = \frac{n+1}{2}+1$ in a sequence and the length $t = T$. Otherwise, do not schedule the RMA.

Proof. (For an even n) According to Equation (7), the makespan of scheduling an RMA in a sequence is:

$$\begin{aligned} C_{\max}(k,t) &= p\sum_{s=0}^{k-2}(1+sp)^b + t + \\ &\tfrac{t}{T}p\sum_{s=0}^{n-k}(1+sp)^b + \\ &t + (1-\tfrac{t}{T})p\sum_{s=k-1}^{n-1}(1+sp)^b \\ &= \{1 + \tfrac{1}{T}p\sum_{s=0}^{n-k}(1+sp)^b - \\ &\tfrac{1}{T}p\sum_{s=k-1}^{n-1}(1+sp)^b\}t + \\ &p[\sum_{s=0}^{k-2}(1+sp)^b + \sum_{s=k-1}^{n-1}(1+sp)^b] \\ &= \{1 - \tfrac{1}{T}p\sum_{s=k-1}^{n-1}[(1+sp)^b - \\ &(1+(s-(k-1))p)^b]\} + \\ &p\sum_{s=0}^{n-1}(1+sp)^b \end{aligned} \tag{17}$$

Based on Theorem 5.1, for $0 < t \leq T$ and $T < p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b - (1+(s-\frac{n}{2})p)^b]$, we have $C_{\max}(\frac{n}{2}+1,t) < C_{\max}(k,t)$, $(k = 2,\cdots,n.\ \ and\ \ k \neq \frac{n}{2}+1)$. So $C_{\max}(\frac{n}{2}+1,t)$ is minimal. Thus, the makespan of scheduling the RMA in a sequence is represented as follows:

$$C_{\max}(\tfrac{n}{2}+1,t)=\{1-\tfrac{1}{T}p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b-(1+(s-\tfrac{n}{2})p)^b]\}t+p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b+(1+(s-\tfrac{n}{2})p)^b] \tag{18}$$

Let $K=1-\frac{1}{T}p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b-(1+(s-\frac{n}{2})p)^b]$ and $B=p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b+(1+(s-\frac{n}{2})p)^b]$, then $C_{\max}(\frac{n}{2}+1,t)=Kt+B$.

Since $b>0$, $p>0$ and $0<t\le T<p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b-(1+(s-\frac{n}{2})p)^b]$, then $K<0$.

Obviously, $C_{\max}(\frac{n}{2}+1,t)$ is decreasing function when $0<t\le T$. Therefore, the minimum of $C_{\max}(k,t)$ is $C_{\max}(\frac{n}{2}+1,T)$.

The proof for odd n is analogous. This proof ends.

B. Algorithm

Based on above propositions and theorems, we give the following algorithm. Let $U=p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b-(1+(s-\frac{n}{2})p)^b]$ and $V=p\sum_{s=\frac{n+1}{2}}^{n-1}[(1+sp)^b-(1+(s-\frac{n+1}{2})p)^b]$. The procedure of the algorithm for solving the problem $1\left|p_{ir},rm,rp,p_i=p\right|C_{\max}$ is as follows:

Step 1. For an even integer n, computer whether T is less than U or not (For an odd integer n , computer whether T is less than V or not). If it is not, then do not schedule an RMA. Otherwise, move to Step 2.

Step 2. Assign the RMA in position $(\frac{n}{2}+1)$ and Computer $C_{\max}(\frac{n}{2}+1,T)$ (For an odd integer n, assign the RMA in position $(\frac{n+1}{2})$ or $(\frac{n+1}{2}+1)$ and computer $C_{\max}(\frac{n+1}{2},T)$), which is the minimal value of the makespan.

Because the time complexity of $p\sum_{s=\frac{n}{2}}^{n-1}[(1+sp)^b-(1+(s-\frac{n}{2})p)^b]$ is $O(\frac{n}{2})$, clearly, the time complexity of Algorithm 1 is $O(n)$.

C. Numerical Experiments

Example. Through investigating a certain large airport in China, an example is given here. There are 6 airplanes ($n=6$) which arrives at an airport at time 0. The normal processing time for each airplane is 35min ($p=35$), the deterioration rate b and the fully recovery time T are respectively designed as different values to solve the problem, i.e., $b=0.054, 0.056, 0.059, 0.100$, and $T=10\,\text{min}, 20\,\text{min}, 30\,\text{min}$.

All airplanes are available for ramp services at 0 and n is an even integer equal to 6. According to Proposition 5.1, Proposition 5.2 and Theorem 5.1, when $T<U$, an RMA will be scheduled.

According to the algorithm in 5.2, we have the following results of comparisons in Table 1, in which RMAC denotes it is considered whether an RMA is scheduled or not, and RMANC denotes no considering the RMA. Besides, the value of k is obtained by the algorithm in 5.2 and it is the position of scheduling an RMA. denotes that there is not an RMA. Yes indicates that an RMA should be scheduled for minimizing the makespan or the total completion time of jobs. And No is just contrary to it.

TABLE I
THE RESULTS OF COMPARISONS

		Makespan			
		RMAC			RAMNC
T	b	$T<U$?	k	$C_{\max}$	$C_{\max}$
10	0.054	Yes	4	253.06	258.55
	0.056	Yes	4	254.42	260.58
	0.059	Yes	4	256.49	263.68
	0.100	Yes	4	287.37	310.59
20	0.054	No	_	258.55	258.55
	0.056	No	_	260.58	260.58
	0.059	No	_	263.68	263.68
	0.100	Yes	4	297.37	310.59
30	0.054	No	_	258.55	258.55
	0.056	No	_	260.58	260.58
	0.059	No	_	263.68	263.68
	0.100	Yes	4	307.37	310.59

Analyses.

From Table 1 we can find that:

(1) Scheduling an RMA in a sequence can decrease the makespan. On the other hand, the makespan will increase with the deteriorating rate b. This is because that the processing time of jobs become increasing if the labors are easily tired.

(2) The value of T has an effect on whether an RMA is scheduled or not.

a) When $T<U$ for the minimal makespan problem, an RMA should be scheduled. Under these situations, for a certain deteriorating rate b, he longer the fully recovery time T of a group labors needs, the larger the objective value is. However, compared with no RMA scheduled situation, it will decrease.

b) When $T>U$ for the minimal makespan problem, an RMA should not be scheduled because scheduling with a shorter RMA has weakly impact on the recovery of workers and scheduling with a longer recovery time will make the objective value exceeding the one under the situation without an RMA.

c) While $T=U$ for the minimal makespan problem, the schedule assigned with an RMA is the same as that without an RMA. In this case, scheduling an RMA is encouraged considering the workers healthy.

VI. CONCLUSIONS

This paper firstly considers a single machine scheduling problem with human energy recovery function under nonlinear time-dependent deterioration. Firstly, a recovery function is firstly proposed. And the problem is proved to be an

NP-hard problem. Then, the special case of the problem is considered and solved in polynomial time. It provides a scheduling framework for other practical scheduling problems referring to labors including maintenance scheduling and cleaning assignments. In the future, the research will be extended to the problem with release dates and due dates, which is the more general situation. In this situation, the scheduling problems need to be solved by heuristic algorithm or intelligent optimization algorithm such as random search algorithm[28].

ACKNOWLEDGMENTS

The authors would like to thank the reviewer for their insightful comments. addressing them has improved the quality of the paper.

REFERENCES

[1] J. N. D. Gupta and S. K. Gupta, “Single facility scheduling with nonlinear processing times,” *Computers and Industrial Engineering*, vol. 14, no. 4, pp. 387–393, 1988.

[2] S. Browne and U. Yechiali, “Scheduling deteriorating jobs on a single processor,” *Operations Research*, vol. 38, no. 3, pp. 495–498, 1990.

[3] J. B. Wang, C. T. Ng, and T. C. E. Cheng, “Single-machine scheduling with deterioration jobs under a series-parallel graph constraint,” *Computers & Operations Research*, vol. 35, no. 8, pp. 2684–2693, 2008.

[4] M. D. Toksari and E. Güner, “Parallel machine scheduling problem to minimize the earliness/tardiness costs with learning effect and deteriorating jobs,” *Journal of Intelligent Manufacturing*, vol. 21, no. 6, pp. 843–854, 2010.

[5] Y. S. Cheng and S. J. Sun, “Scheduling linear deteriorating jobs with rejection on a single machine,” *European Journal of Operational Research*, vol. 194, no. 1, pp. 18–27, 2009.

[6] S. S. Li, C. T. Ng, and J. J. Yuan, “Scheduling deteriorating jobs with con/slk due date assignment on a single machine,” *International Journal of Production Economics*, vol. 131, no. 2, pp. 747–751, 2011.

[7] C. T. Ng, S. Li, T. C. E. Cheng, and J. Yuan, “Preemptive scheduling with simple linear deterioration on a single machine,” *Theoretical Computer Science*, vol. 411, no. 40-42, pp. 3578–3586, 2010.

[8] T. C. E. Cheng, S. J. Yang, and D. L. Yang, “Common due-window assignment and scheduling of linear time-dependent deteriorating jobs and a deteriorating maintenance activity,” *International Journal of Production Economics*, vol. 135, no. 1, pp. 154–161, 2012.

[9] W. H. Kuo and D. L. Yang, “Minimizing the total completion time in a single-machine scheduling problem with a time-dependent learning effect,” *European Journal of Operational Research*, vol. 174, no. 2, pp. 1184–1190, 2006.

[10] J. B. Wang, L. Y. Wang, D. Wang, and X. Y. Wang, “Single-machine scheduling with a time-dependent deterioration,” *The International Journal of Advanced Manufacturing Technology*, vol. 43, no. 7-8, pp. 805–809, 2009.

[11] W. H. Wu and S. R. Cheng, “Ant colony algorithms for a two-agent scheduling with sum-of-processing times-based learning and deteriorating considerations,” *Journal of Intelligent Manufacturing*, vol. 23, no. 5, pp. 1985–1993, 2012.

[12] S. S. Li and B. Q. Fan, “Single-machine scheduling with proportionally deteriorating jobs subject to availability constraints,” *Asia-Pacific Journal of Operational Research*, vol. 29, no. 4, pp. 287–326, 2012.

[13] H. P. Wu and M. Huang, “Improved estimation of distribution algorithm for the problem of single-machine scheduling with deteriorating jobs and different due dates,” *Computational and Applied Mathematics*, vol. 33, no. 3, pp. 557–573, 2014.

[14] J. B. Wang, M. Z. Wang, and P. Ji, “Scheduling jobs with processing times dependent on position, starting time, and allotted resource,” *Asia-Pacific Journal of Operational Research*, vol. 29, no. 5, pp. 287–326, 2012.

[15] W. C. Lee, “Single-machine scheduling with past-sequence-dependent setup times and general effects of deterioration and learning,” *Optimization Letters*, vol. 8, no. 1, pp. 135–144, 2014.

[16] K. Wang and S. Choi, “Decomposition-based scheduling for makespan minimisation of flexible flow shop with stochastic processing times,” *Engineering Letters*, vol. 18, no. 1, pp. 75–84, 2010.

[17] W.-C. Lee, “A note on scheduling with learning effect and past-sequence-dependent setup time,” *Lecture Notes in Engineering and Computer Science: Proceedings of the International MultiConference of Engineers and Computer Scientists 2011, IMECS 2011, 16-18 March, 2011, Hong Kong*, pp. 1362–1365.

[18] P.-J. Lai and W.-C. Lee, “Scheduling problems with integrable learning and forgetting effects,” *Lecture Notes in Engineering and Computer Science: Proceedings of the International MultiConference of Engineers and Computer Scientists, IMECS 2014, 12-14 March, 2014, Hong Kong*, pp. 1019–1024.

[19] C. Y. Lee and V. J. Leon, “Machine scheduling with a rate-modifying activity,” *European Journal of Operational Research*, vol. 128, no. 1, pp. 119–128, 2001.

[20] X. Y. Wang and M. Z. Wang, “Single machine common flow allowance scheduling with a rate-modifying activity,” *Computers and Industrial Engineering*, vol. 59, no. 4, pp. 898–902, 2010.

[21] M. Ji and T. C. E. Cheng, “Scheduling with job-dependent learning effects and multiple rate-modifying activities,” *Information Processing Letters*, vol. 110, no. 11, pp. 460–463, Jul. 2010.

[22] J. B. Wang and C. M. Wei, “Parallel machine scheduling with a deteriorating maintenance activity and total absolute differences penalties,” *Applied Mathematics and Computation*, vol. 217, no. 20, pp. 8093–8099, 2011.

[23] M. Rebai, I. Kacem, and K. H. Adjallah, “Earliness-tardiness minimization on a single machine to schedule preventive maintenance tasks: metaheuristic and exact methods,” *Journal of Intelligent Manufacturing*, vol. 23, no. 4, pp. 1207–1224, 2012.

[24] E. J. Lodree and C. D. Geiger, “A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration,” *European Journal of Operational Research*, vol. 201, no. 2, pp. 644–648, 2010.

[25] C. Zhao and H. Tang, “A note to due-window assignment and single machine scheduling with deteriorating jobs and a rate-modifying activity,” *Computers and Operations Research*, vol. 39, no. 6, pp. 2161–2169, 2010.

[26] Y. Öztürkoğlu, R. Bulfin, and E. Lodree, “A model for scheduling deteriorating jobs with rate-modifying-activities on a single machine,” *Working paper*, vol. 37, no. 1, pp. 1–10, 2010.

[27] R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. R. Kan, “Optimization and approximation in the deterministic sequencing and scheduling: a survey,” *Annals of Discrete Mathematics*, no. 5, pp. 287–326, 2010.

[28] S. Tabandeh and H. Michalska, “Random search techniques for optimal bidding in auction markest,” *IAENG International Journal of Applied Mathematics*, vol. 39, no. 4, pp. 276–285, 2009.