Computers & Industrial Engineering 137 (2019) 106072

Contents lists available at ScienceDirect

Computers & Industrial Engineering

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

An energy-efficient single machine scheduling problem with machine reliability constraints

Lu Chen*, Jinfeng Wang, Xianyang Xu

School of Mechanical Engineering, Shanghai Jiao Tong University, China

ARTICLE INFO

Keywords:
Energy-efficient scheduling
Energy consumption
Reliability
Ant colony optimization
Single machine scheduling

ABSTRACT

Considering the impact of machine conditions on processing energy is critical in energy-efficient scheduling. In this study, machine conditions are evaluated by machine reliability. The relationship between reliability and processing energy consumption was developed. A mathematical programming model was formulated for a single machine scheduling problem with the objectives to minimize tardiness cost and energy cost. Modified Emmons rules were proposed and embedded into an ant colony algorithm to solve real-world problems from a rotor production workshop. The effectiveness and efficiency of the algorithm were demonstrated by computational experiments. The impact of initial reliability and due date tightness on scheduling decisions were analyzed through sensitivity analyses, which provides useful managerial insights for real shop scheduling.

1. Introduction

Energy is one of the most indispensable sources to satisfy the development of modern society. Manufacturing facilities are responsible for a huge amount of energy consumption. In countries like China and the US, the energy consumption of manufacturing industry contributes nearly half of the national energy consumption (Conti, Holtberg, Diefenderfer, & LaRose, 2016; Wang, Wang, & Lin, 2017; Wang, Zhang, & Zhu, 2017).

In considerations of reducing production costs and policy pressure, it is inevitable for manufacturing factories to reduce the energy consumption under the premise of production capacity. There has been a growing interest in energy-efficient production. The research on minimizing energy consumption has been conducted mainly in two perspectives. The first one focuses on machine-level optimization, including designing more energy-efficient machines and reducing energy of components in various ways like modifying acceleration modes (Li, Zein, Kara, & Herrmann, 2011; Mori, Fujishima, Inamasu, & Oda, 2011; Neugebauer, Wabner, Rentzsch, & Ihlenfeldt, 2011). However, these methods need huge investment on time and money, which brings pressure on operations management, especially for small-sized business. The other strategy is energy-efficient scheduling, which reduces energy cost with less investment. Energy-efficient scheduling is proven to be an efficient way to reduce production energy (Gahm, Denz, Dirr, & Tuma, 2016; Liu et al., 2014; Yan & Fei, 2010).

A single machine scheduling problem is studied in this paper. This work is motivated by a real rotor factory in *Shanghai Electric*. Rotor spindle processing includes five main processes. We selected the CNS boring machine in this study due to its long processing time and high energy consumption. In addition, energy consumption was observed to be higher with machine condition degradation. Records show that breakdown happened in 145 work shifts (8 h/shift) during the year of 2016, which makes up to 1160 h of down time. Production is organized under make-to-order mode that incurs heavy pressure on delivery performance. The factory also needs to pay attention to energy consumption with stricter environmental regulations (Notice by the State Council, 2017) being put forward in China. Thus, tardiness cost and energy consumption cost are the two optimization objectives of the single machine scheduling problem addressed in this study. The challenge is to explore problem formulation that properly defines the energy consumption with regard to machine reliability, as well as to achieve a trade-off between the two costs.

The remainder of this paper is organized as follows. In Section 2, energy consumption with different machine reliability is calculated and the energy-efficient scheduling problem is formulated. In Section 3, the modified Emmons rules are proposed and embedded into an ant colony optimization algorithm to solve the problem. The performance of the algorithm is evaluated by numerical experiments in Section 4. Finally, Section 5 presents the conclusions and future perspective.

* Corresponding author.
E-mail address: chenlu@sjtu.edu.cn (L. Chen).

https://doi.org/10.1016/j.cie.2019.106072
Received 15 March 2019; Received in revised form 17 June 2019; Accepted 16 September 2019
Available online 17 September 2019

2. Literature review

We first review work on energy-efficient scheduling and then discuss scheduling with reliability consideration.

2.1. Energy-efficient scheduling

Various strategies in energy-efficient scheduling have been developed in recent research, such as on-off strategy and speed-scaling strategy. These strategies took different influencing factors and different trade-offs into classical scheduling model in order to minimize energy consumption.

The on-off strategy is to decide whether and when to shut down the machine or to keep it idle (Mouzon & Yildirim, 2008). The idea is to reduce the energy consumption incurred by machine idling between two jobs. A number of energy-efficient scheduling problems have been investigated, such as single-machine scheduling (Liu et al., 2014; Shrouf, Ordieres-Meré, García-Sánchez, & Ortega-Mier, 2014; Yin, Li, Lu, & Gao, 2016), parallel machine scheduling (Liang, Yang, Liu, & Guo, 2015), and flow shop scheduling (Dai, Tang, Giret, Salido, & Li, 2013; Lu, Gao, Li, Pan, & Wang, 2017). In some studies, the minimal idle time to turn off the machine needs to be decided. Che, Wu, Peng, and Yan (2017) considered both turning-on energy and shutting-down energy to define the minimal time interval. They formulated a multi-objective model to minimize tardiness and energy consumption. Yildirim and Mouzon (2012) considered set-up energy and developed a mixed integer programming model to minimize both makespan and energy consumption. Cheng, Chu, Liu, Wu, and Xia (2017) took only turning-on energy into account to build a model under the time-of-use mechanism.

Speed-scaling strategy was first studied by Yao, Demers, and Shenker (1995). A higher processing speed contributes less processing time, but it also leads to more energy consumption. By adjusting the processing speeds to a proper level when processing specific jobs, energy consumption decreases. Speed-scaling strategy has also been applied in various production environments, such as single machine scheduling (Bampis, Dürr, Kacem, & Milis, 2012; Fang, Uhan, Zhao, & Sutherland, 2016; Gong, De Pessemier, Joseph, & Martens, 2015), parallel machine scheduling (Fang & Lin, 2013), and flow shop scheduling (Dai et al., 2013; Ding, Song, & Wu, 2016; Fang, Uhan, Zhao, & Sutherland, 2013; Luo, Du, Huang, Chen, & Li, 2013; Mansouri, Aktas, & Besikci, 2016; Yin, Li, Gao, Lu, & Zhang, 2017).

Other parameters that related to energy consumption such as cutter type, feed rate, cutting speed, and cutting depth could have also be considered in scheduling problems (Kibira, Hatim, Kumara, & Shao, 2015). Wang, Wang, et al. (2017) and Wang, Zhang, et al. (2017) took into account the cutting conditions and proposed a two-stage heuristic algorithm. He, Li, Wu, and Sutherland (2015) tried to select appropriate type of machine tools to save production energy. Bulsara, Hingu, and Vaghasiya (2016) found that reducing machine vibration could save energy consumption.

2.2. Scheduling with reliability constraints

Machine reliability has a remarkable influence on energy consumption, which significantly affects energy-efficient scheduling. Yan and Hua (2010) found a strong mapping relationship between energy consumption and reliability through experimental analysis. However, current research on scheduling with machine reliability constraints focuses mainly on maintenance optimization (Basri, Abdul Razak, Ab-Samat, & Kamaruddin, 2017).

Optimizing the maintenance operation includes the maintenance assignment and decisions on optimal maintenance intervals. For example, Moghaddam and Usher (2010) built a model to determine the optimal preventive maintenance (PM) interval to minimize total cost subjected to a constraint on system reliability. Lu, Zhou, and Li (2016) developed an integrated reliability model. An optimal PM schedule is obtained to minimize the total production cost.

Some studies considered joint optimization of scheduling and maintenance. Wang and Liu (2015) took two kinds of resources (machines and molds) into account. The authors simultaneously minimized makespan and machines unavailability by determining job assignment and PM planning. Berrichi and Yalaoui (2013) developed a mathematical model to minimize total tardiness and unavailability of the production system. The start times of PM and job sequence were decided by ant colony optimization.

2.3. Summary and contributions of the paper

Our work on single machine scheduling problem with machine reliability constraints represents a contribution to a new field of research. Previous studies on energy-efficient scheduling regard machine either at *perfect* condition with rated power consumption or at *down* condition without availability. However, this is never the case in real workshops. The energy consumption of jobs processing may increase due to machine degradation (Dahmus & Gutowski, 2004). Ignorance of machine conditions on energy consumption would make production schedule less efficient. Our contribution is to define energy consumption with regard to machine reliability and to incorporate it into the existing scheduling model. In contrast to most existing studies, our model minimizes a total cost function composed of tardiness cost and energy consumption cost (a function of machine reliability).

Single machine scheduling problem is of importance both for practical and theoretical points of view. Not only do single machine scheduling problems have wide practical applications in many industries, but also the methods and results developed may provide fundamentals for examining scheduling problems in other production environments, such as parallel machines and flow shop.

Other contributions of this paper are: (1) Special attention has been drawn to address the practical constraints when we developed the algorithm. Classical Emmons rules have been adapted and embedded into the algorithm in order to improve the efficiency of the algorithm. (2) An extensive computational analysis has been conducted based on the real data obtained from the rotor production workshop. Valuable managerial insights for the decision-makers are provided.

3. Problem formulation

In the addressed problem, the machine is featured with reliability, which is related with the actual energy consumption rate of processing. Before formulating the problem, the relationship between energy consumption and reliability is defined.

3.1. Calculation of energy consumption

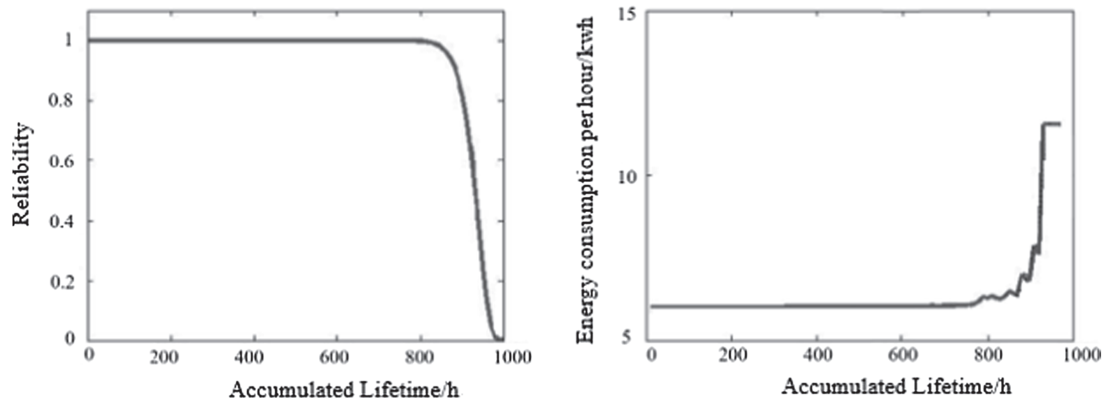
The reliability $R(L)$ of the machine is assumed to follow an exponential distribution. L is defined as the accumulated lifetime since last as-good-as-new maintenance.

$$R(L) = \exp(-\lambda L),\ L \geq 0 \tag{1}$$

The relationship between energy consumption and reliability is shown in Fig. 1 (Yan & Hua, 2010). Fig. 1(a) shows that reliability decreases with the increase of accumulated lifetime. Fig. 1(b) shows that the energy consumption rate increases with the increase of accumulated lifetime.

Denote r as the reliability of the machine. The actual machine energy consumption rate (kw/h), $E(r)$, is defined as:

$$E(r) = \begin{cases} E & r_{th1} \leq r \leq 1 \\ E + \omega(r_{th1} - r) & r_{th2} \leq r \leq r_{th1} \\ \infty & 0 \leq r \leq r_{th2} \end{cases} \tag{2}$$



a. Relationship between reliability and accumulated lifetime

b. Relationship between actual energy consumption and accumulated lifetime

Fig. 1. Relationship between energy consumption and reliability (Yan & Hua, 2010).

where E is the nominal value of the energy consumption rate, r_{th1} and r_{th2} are predefined threshold values, and ω is the increment of energy consumption rate due to reliability decrease. When the reliability is larger than r_{th1}, the machine processes jobs at nominal energy consumption rate. When the reliability is smaller than r_{th1}, the actual energy consumption increases proportionally with machine reliability. When reliability is smaller than r_{th2}, the machine is unable to use due to a large possibility of breaking down. Thus, the energy consumption rate is defined to be infinite.

For the CNS machine addressed in our study, the mean time between failure (*MTBF*) is estimated to be 3000 working hours. With exponential distribution, we have:

$$MTBF = \int_0^\infty e^{-\lambda t} dt = \frac{1}{\lambda}$$

and $\lambda = 0.0003$. Thus, the reliability threshold given by Eq. (1) is 0.4. Once machine reliability exceeds this threshold, a preventive maintenance is triggered. Thus, we set the value of r_{th2} to 0.4. With $r_{th2} = 0.4$, r_{th1} is estimated as $r_{th1} = (r_{th2} + 1)/2 = 0.7$.

From the factory we obtained the maximum energy consumption rate (E_{max}) and the nominal energy consumption rate (E) respectively. Thus, ω is calculated as:

$$\omega = \frac{E_{max} - E}{r_{th1} - r_{th2}} = 100$$

ω is assumed to be constant when the machine processes different jobs.

3.2. Mathematical model

The single machine scheduling problem is formally defined as follows. A set of jobs (denoted as N) needs to be processed on a single machine with its initial lifetime identified. The scheduling problem is to sequence the jobs on the machine to minimize energy consumption cost and tardiness cost. We assume that:

(i) The machine can handle one job at a time and preemption is not allowed;
(ii) The jobs are all available at time 0;
(iii) The reliability of the machine will not change until a job processing has completed;
(iv) Maintenance is not considered.

Notations are defined as follows:

H	Set of positions, $H = \{1, \cdots, \lvert N\rvert\}$
p_i	Processing time of job i, $i \in N$
d_i	Due date of job i, $i \in N$
L_0	Accumulated lifetime of machine at time 0
M	A large positive number
γ	Unit tardiness cost
ε	Unit energy cost
Decision variables are:	
x_{ik}	$= 1$, if job i is processed in position k, $i \in N$, $k \in H$; 0, otherwise
C_i	Completion time of job i, $i \in N$
Z_k	Completion time of the job in position k, $k \in H$
L_i	The accumulated lifetime of the machine before job i is processed, $i \in N$

The problem is formulated as:

$$\text{Mininize}\left\{\varepsilon \sum_{i\in N} p_i \cdot E(\exp(-\lambda L_i)) + \gamma \sum_{i\in N} \max(C_i - d_i, 0)\right\} \tag{3}$$

s.t.

$$\sum_{i\in N} x_{ik} = 1, \forall\ k \in H \tag{4}$$

$$\sum_{k\in H} x_{ik} = 1, \forall\ i \in N \tag{5}$$

$$\sum_{i\in N} x_{i1} \cdot p_1 \leq Z_1 \tag{6}$$

$$\sum_{i\in N} x_{ik} \cdot p_i + Z_{k-1} \leq Z_k, \forall\ k \in H\backslash\{1\} \tag{7}$$

$$C_i + M(1 - x_{ik}) \geq Z_k, \forall\ i \in N, k \in H \tag{8}$$

$$L_i - L_0 + M(1 - x_{i1}) \geq 0, \forall\ i \in N \tag{9}$$

$$L_i - L_0 + M(1 - x_{ik}) \geq Z_{k-1}, \forall\ i \in N, k \in H\backslash\{1\} \tag{10}$$

$$x_{ik} \in \{0, 1\}, C_i, Z_k, L_i \geq 0, \forall\ i \in N, k \in H \tag{11}$$

The objective function (3) minimizes the summation of energy consumption cost and tardiness cost. Constraints (4) ensure that for a given position, only one job can be processed in. Constraints (5) ensure that for a given job, it can be allocated to only one position. Constraints

(6) and (7) define the lower bound of the completion time of all the positions. Constraints (8) calculate the lower bound of the completion time of all the jobs. Constraints (9) calculate the lower bound of accumulated lifetime before job start processing. Constraints (10) define the relationship between accumulated lifetime and completion time of all the positions. Constraints (11) define the nature of the decision variables.

Commercial solvers fail to obtain optimal solution in limited time for real-size problems. For example, an instance with eight jobs took Lingo solver more than one hours to solve to optimality. Thus, an ant colony optimization (ACO) algorithm was developed.

4. Algorithm design

In this section, an ACO algorithm was developed. A construction heuristic is firstly proposed to generate an initial solution. The classical Emmons rules were modified and applied to the local search process. It decreases the randomness of search and improves the solution quality.

4.1. Solution construction

Define a vector including $|N| + 1$ elements to represent a solution, with the first element being 0. Solution construction is a process to select a job to be processed in the next position until all the jobs are scheduled. A heuristic approach is applied to select the next-processing job.

Define τ_{ij} to be the pheromone trails to indicate the favorability of processing job j after job i. τ_{ij} is set to 0 initially and will be updated at each iteration.

Denote U to be the set of unscheduled jobs and $\overline{p}$ to be the average processing time of the jobs in U. Define heuristic factor between η_{ij} of any two jobs i and j as:

$$\eta_{ij} = \frac{\delta}{p_j} + \frac{1-\delta}{p_j}\cdot\exp\left(-\frac{\max(d_j - p_j - C_{max}(S), 0)}{\xi\cdot\overline{p}}\right)$$

where $C_{max}(S)$ is the makespan of the current partial schedule S, δ and ξ are two scale parameters to be determined by experiments.

Define q_0 to be a predefined parameter indicating the exploitation preference, $0 \le q_0 \le 1$. Initially, generate a random number q from the uniform distribution [0, 1]. If $q \le q_0$, then an ant will select a job that maximizes the value of $(\tau_{i^*j})^{\alpha}\cdot(\eta_{i^*j})^{\beta}$. α and β are the coefficients that can adjust the importance of pheromone and heuristic factor. That is, j^*=argmax$\{(\tau_{i^*j})^{\alpha}\cdot(\eta_{i^*j})^{\beta}\}$. If $q > q_0$, then an ant randomly selects a job with a probability P_j defined as:

$$P_j = \frac{(\tau_{i^*j})^{\alpha}\cdot(\eta_{i^*j})^{\beta}}{\sum_{l\in N}(\tau_{i^*l})^{\alpha}\cdot(\eta_{i^*l})^{\beta}}$$

where i^* is the last job in the current partial sequence.

The complete solution construction process is described as follows. The process terminates with a job sequence S.

Algorithm 1 (*Solution construction*).

Set $U = N$, $S = \{0\}$, q_0
While $U \ne \emptyset$ **do**
  Generate a random number q from uniform distribution [0,1]
  If $q \le q_0$ **then**
    $j^* = \text{argmax}\{(\tau_{i^*j})^{\alpha}\cdot(\eta_{i^*j})^{\beta}\}$
  Else
    Select j^* based on the probabilities P_j
  Endif
  Insert j^* at the end of S
  $U = U/\{j^*\}$
Endwhile
Return S

4.2. Modified Emmons rules

Emmons Rules were put forward to minimize total tardiness for single machine scheduling (Emmons, 1969; Kanet, 2007). Suppose N is the set of jobs to be processed. Define $Q^{'} = N - Q$ for any subset $Q \subset N$. $Q^{'}$ is the complement of Q. Define $P(Q)$ to be the summation of the processing times of the jobs in subset Q. $P(Q) = \sum_{i\in Q} p_i$. Define B_i, A_i ($i \in N$) to be the sets of jobs processed preceding and following job$i$. $j{\prec}k$ means that job j precedes job k in some optimum sequences.

The original Emmons rules and related corollaries are given as theorem as follows.

Theorem 1. *For any job* j, k, *if* $p_j \le p_k$ *and* $d_j \le max\{d_k, P(B_k) + p_k\}$, *then* $j{\prec}k$.

Theorem 2. *For any job* j, k, *if* $d_k \ge max\{d_j, P(A_j^{'}) - p_k\}$, *then* $j{\prec}k$.

The above theorems are modified in order to be applicable to solve the energy-efficient scheduling problem addressed in our study.

Theorem 1 is modified as follows:

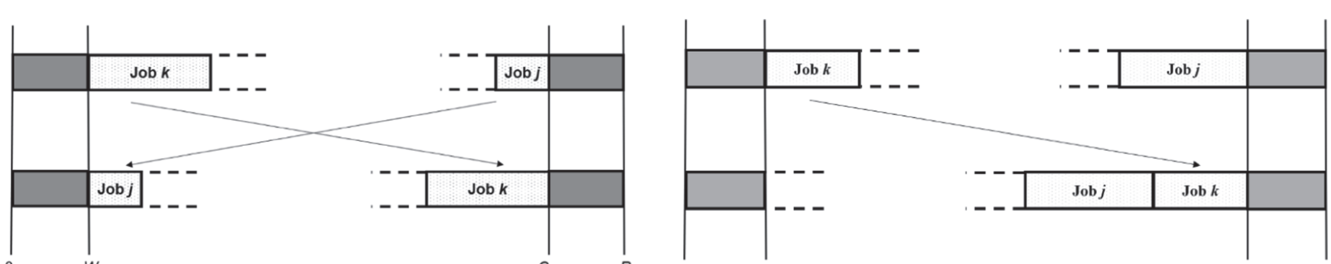
Theorem 3. *For any two jobs j and k, assume that k precedes j. If (1)* $p_j \le p_k$, *(2)* $d_j \le max\{d_k, P(B_k) + p_k\}$ *and (3) the total energy consumption incurred by swapping job k and job j is negative, then* $j{\prec}k$.

From Theorem 1, if properties (1) and (2) *are satisfied, then the total tardiness is minimized with* $j{\prec}k$. *And if property* (3) *is also satisfied, then the total energy consumption will decrease by swapping job k and job j.*

Theorem 2 is modified as follows:

Theorem 4. *For any two jobs j and k, if (1)* $d_k \ge max\{d_j, P(A_j^{'}) - p_k\}$ *and (2) the total energy consumption incurred by inserting job k after job j is negative, then* $j{\prec}k$.

From Theorem 2, if property (1) *is satisfied, then the total tardiness is minimized with* $j{\prec}k$. *If property* (2) *is also satisfied, then the total energy consumption will decrease by inserting job k after job j.*



a) Swap operation between job j and job k b) Insert job k after job j

Fig. 2. The two local search operations.

4.3. Local search

Theorem 3 and Theorem 4 give sufficient conditions for job precedence relations in optimal sequences. Therefore, they are applied to the local search process of the ACO algorithm to improve the search efficiency. The two local search rules are defined based on the two theorems.

LS1: Swap job k and job j if the three properties in Theorem 3 are satisfied.

LS2: Insert Job k after job j if the two properties in Theorem 4 are satisfied.

The two operations are illustrated in Fig. 2.

The complete process of local search is described as:

Algorithm 2 (*Local Search*).

For $m = 2$ to $|N| + 1$ **do**
  Set $n = m + 1$
  Get job i in the m^{th} position
  While $n \le |N| + 1$ **do**
    Get job j in the n^{th} position
    If jobs i and j meet the conditions in Theorem 3/Theorem 4 **then**
      Swap job i and job j/Insert job i after job j
      Exit
    Else
      $n = n + 1$
    Endif
  Endwhile
  $m = m + 1$
Endfor

4.4. Pheromone update

When all ants in the colony finish solution construction, pheromone trails of paths belonging to good solutions are updated to make *good* paths more desirable in the next iteration. Pheromone update includes pheromone evaporation and pheromone increase.

The evaporation rate is used to implement a forgetting mechanism, which helps the algorithm to forget bad solutions and to explore new regions. Define global best solution to be S^{Best} and its objective function vale to be Obj^{Best}. The best solution in current iteration is defined to be $S^{LocalBest}$, with the objective function value $Obj^{LocalBest}$. Define ρ to be the evaporation rate and a as the current iteration. The pheromone trail is updated as:

$$\tau_{ij}(a+1) = (1-\rho)\tau_{ij}(a) + \Delta\tau_{ij}$$

$\Delta\tau_{ij}$ is calculated under different situations:

(1) For the paths belonging to the global best solution, increase the pheromone trail with a significant value: $\Delta\tau_{ij}=0.8 * C/Obj^{Best}$.
(2) For the paths belonging to the best solution in current iteration, increase the pheromone trail with a smaller value: $\Delta\tau_{ij}= 0.3 * C/Obj^{LocalBest}$ to avoid a too-rapid convergence toward a suboptimal region.
(3) For other paths, no pheromone value is added.

The above description is defined as:

$$\Delta\tau_{ij} = \begin{cases} 0.8 * \frac{C}{Obj^{Best}}, & i \text{ precedes } j \text{ in } S^{Best} \\ 0.3 * \frac{C}{Obj^{LocalBest}}, & i \text{ precedes } j \text{ in } S^{LocalBest} \\ 0, & Otherwise \end{cases}$$

where C is a constant and will be determined through experiments.

4.5. The ACO algorithm

Define the solution constructed by current ant to be S^{ants} and its objective function value to be Obj^{ants}. The ACO algorithm discussed above is summarized as follows. ITER and ANTS represent the total number of iterations and the total number of ants, respectively.

Algorithm 3. (*ACO algorithm based on improved Emmons Rules*).

Initialize algorithm parameters α, β, q_0, ρ, ITER, ANTS
For iter = 1 to ITER **do**
  For ants = 1 to ANTS **do**
    Algorithm 1
    Apply local search in **Algorithm 2**
    If ($Obj^{ants} < Obj^{LocalBest}$), **then**
      $Obj^{LocalBest} = Obj^{ants}$, $S^{LocalBest} = S^{ants}$
    Endif
    ants = ants + 1
  Endfor
  If ($Obj^{LocalBest} < Obj^{Best}$), **then**
    $Obj^{Best} = Obj^{LocalBest}$, $S^{Best} = S^{LocalBest}$
  Endif
  iter = iter + 1
Endfor

5. Computational experiments and results

Several computational experiments were conducted and the experimental results are presented in this section. The ACO algorithm was implemented by Python 2.7.1 and run on a computer with Intel(R) Core (TM) i5-7200U (2.50 GHz) processor and 8 GB RAM.

5.1. Description of instances

There is a series of processes in the rotor production workshop. The CNS boring machine is selected in our experiment. The process is to bore a hole. There are five different types of rotors in terms of power level of the gas turbine. The processing time and the energy consumption rate for each type of rotor are different as shown in Table 1.

Different instances were randomly generated based on the information we had obtained from the factory. Parameters of the instances are described as follows:

(1) A sequence of jobs with the 5 different types is randomly generated. The number of jobs ranges from 3 to 90.
(2) Each jobs has a due date, which was generated from uniform distribution $\left[\left(1-T-\frac{D}{2}\right)\sum_{i\in N} p_i, \left(1-T+\frac{D}{2}\right)\sum_{i\in N} p_i\right]$ (Potts & Van Wassenhove, 1982). T is the average tardiness factor, and D is the relative range of due date.
(3) In the objective function (3), unit tardiness cost $\gamma = 10$; unit energy cost $\varepsilon = 0.4$.

Some preliminary experiments on the parameter calibration for the ACO algorithm were conducted. The associated parameters are set as shown in Table 2.

5.2. Small-scale experiments

We first evaluate the performance of ACO algorithm with small-scale problems by comparing the solutions obtained from the ACO algorithm with the optimal solutions obtained from the commercial

Table 1
Processing time and energy consumption rate for different jobs.

Rotor Types	I	II	III	IV	V
Processing Duration/h	10	6	4	12	8
Nominal value of energy consumption rate (kw/h)	30	28	30	28	32

Table 2
Algorithm-related parameter settings.

Parameter	δ	ξ	q_0	α	β	ρ	ANTS	ITER
Value	0.5	0.8	0.85	1.5	2.5	0.5	1.5|N|	40

Table 3
The results of small-scale experiments.

|N|	Lingo		ACO		Dev
	Obj^{Lingo}	CPU/s	Obj^{ACO}	CPU/s	(%)
3	1280.99	1.39	1280.99	0.21	0
4	1806.23	5.06	1806.23	0.54	0
5	2212.31	33.25	2212.31	0.68	0
6	2891.12	480.14	2891.12	0.73	0
7	3403.95	2131.87	3403.95	0.75	0
8	3905.26	3600.00	3895.96	0.81	0.24
9	4830.7	3600.00	4794.7	1.44	0.75
10	6073.01	3600.00	5722.02	1.18	5.78
11	6888.82	3600.00	6310.83	1.42	8.39
12	7925.52	3600.00	7127.52	1.65	10.07

solver.

Since CPLEX and Gurobi are not able to deal with the exponent function in the objective function (Eq. (3)), Lingo 17.0 was used to solve small-scale problems optimally. The computational time for Lingo is set to be 3600 s. If the global optimal objective cannot be obtained with 3600 s, the current local optimal solution will be retrieved as the final solution. The number of jobs ranges from 3 to 12. For each problem, ten instances are generated and solved. The average objective values are calculated and compared. The deviation between the solutions from the ACO algorithm and the solutions from Lingo is defined as follow. The results are given in Table 3.

$$\text{Dev} = \frac{Obj^{Lingo} - Obj^{ACO}}{Obj^{Lingo}} \times 100\%$$

It is observed from Table 3 that global optimal solutions can be obtained by Lingo within the specified time when the number of jobs is less than seven. And the ACO algorithm can get the optimal solutions with much less computational time. With the increase of the number of jobs, Lingo cannot solve the problems to optimum within the specified time period. And the solution obtained by the ACO algorithm is far better than the local optimal solution obtained by Lingo. The proposed ACO algorithm is proven to be effective and efficient in solving problems with small-scale jobs.

5.3. Medium and large-scale experiments

In order to evaluate the performance of the ACO algorithm for solving medium and large-scale instances. Different configurations of the ACO algorithm are tested. They are: (1) the ACO algorithm without any local search process (standard ACO); (2) the ACO algorithm with LS1; (3) the ACO algorithm with LS2; (4) the ACO algorithm with both LS1 and LS2.

Different groups of instances were generated and solved by the four ACO algorithms respectively. The objective function value obtained from the standard ACO algorithm is used as the benchmark (denoted as *Obj*) for the other three configurations (denoted as Obj_{LS}). The deviation is calculated as:

$$\text{Dev} = \frac{Obj - Obj_{LS}}{Obj} \times 100\%$$

Ten instances were solved in each group, and the average deviation is calculated and compared. Table 4 summarize the results.

The following observations can be made:

Table 4
Comparison of different configurations of the ACO algorithm.

|N|	ACO with LS1		ACO with LS2		ACO with LS1 and LS2	
	Dev/%	CPU/s	Dev/%	CPU/s	Dev/%	CPU/s
20	1.28	4.15	1.42	5.36	1.95	6.30
30	2.50	12.15	2.63	14.31	3.43	17.59
40	3.55	20.01	4.42	32.21	5.74	40.36
50	3.88	35.31	4.89	62.66	6.35	80.63
60	3.66	45.55	5.19	100.04	7.26	122.47
70	4.14	82.37	6.89	190.91	8.45	223.56

(1) The ACO algorithm with local search outperforms the standard ACO algorithm. And the improvement becomes more apparent as the number of jobs increases. This tendency is the same for the three configurations of the ACO algorithm.
(2) If there is no limit on the computational time, the ACO algorithm with both local search rules performs the best. The deviation from the standard ACO is between 6.35% and 10.79%.
(3) Local search rule LS2 has a more significant impact on the solution quality, comparing with the LS1. Also, the ACO with LS2 takes more computational time to return a global best solution. The reason is that the LS2 rule with loose conditions can discover more available exchanges in the local search process, which contributes to the improving of solution quality but to some extents prolongs the computational time.

The above observations demonstrate the effectiveness of the modified Emmons rules in improving the solution quality of the ACO algorithm. The one with both local search rules is used in the following sensitivity analyses.

5.4. Sensitivity analyses

Sensitivity analyses were conducted by varying the each of the following two parameters: (1) initial lifetime of the machine, and (2) due date tightness. Medium sized (MS) and large sized (LS) instances were used in this set of experiments.

5.4.1. Initial lifetime

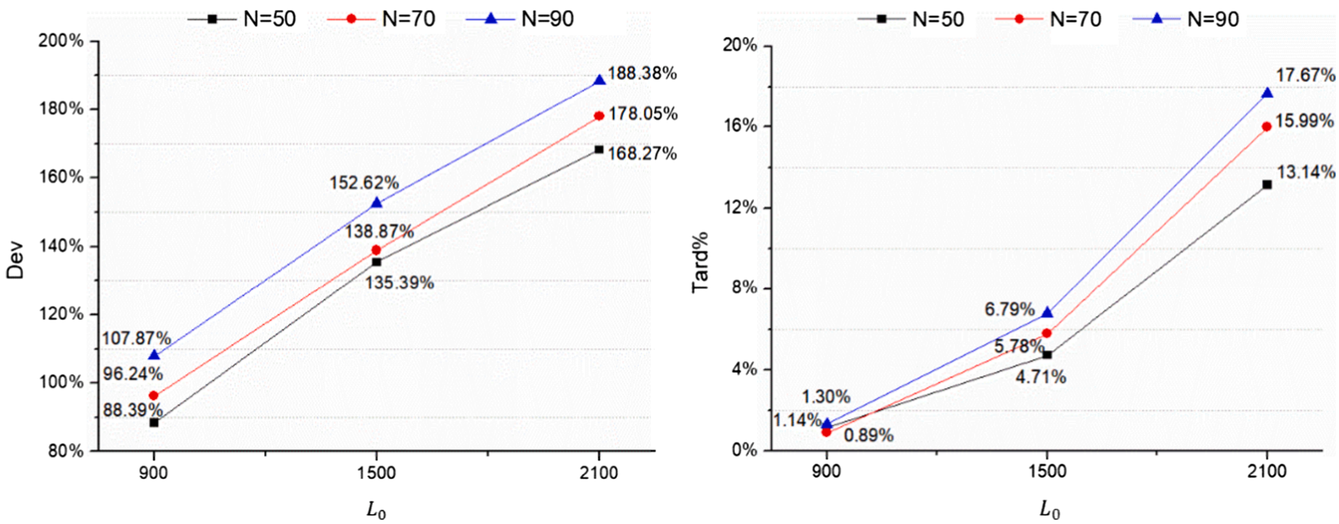
Given that the total available time for the machine is 3000 working hours since the last as-good-as-new maintenance, three levels of initial lifetime are considered. They are short initial lifetime ($L_0 = 900$), medium initial lifetime ($L_0 = 1500$), and long initial lifetime ($L_0 = 2100$). Set $T = 0.3$, and $D = 0.8$ for due date. Medium and large size problems were randomly generated and solved by the ACO algorithm.

The result is shown in Fig. 3. Fig. 3(a) illustrates how the energy consumption cost (C_E) increases as the initial lifetime increases. Fig. 3(b) illustrates how the ratio of tardiness cost to total cost changes as the initial lifetime increases. The energy consumption cost with zero initial lifetime is used as the benchmark (denoted as $C_{\overline{E}}$). The deviation is calculated as:

$$\text{Dev} = \frac{C_E - C_{\overline{E}}}{C_{\overline{E}}} \times 100\%$$

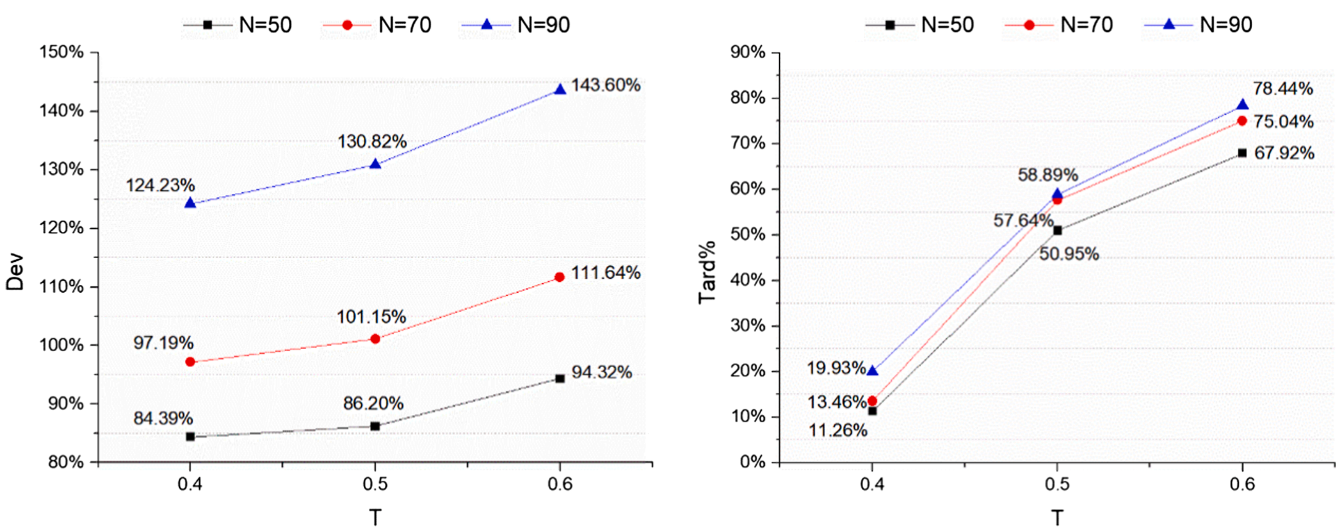
The figure shows that:

(1) With the increase of initial life time (L_0), the energy consumption cost sees a sharp rise due to the extra energy consumption from reliability decrease. This tendency gets more apparent with larger problem size.
(2) The ratio of tardiness cost to total cost also increases with the increase of L_0. This is because sacrifice of tardiness cost needs to be made to acquire lower total cost. Thus, when initial lifetime is large



(a) Energy consumption cost (b) Tardiness cost

Fig. 3. Sensitivity analysis for initial lifetime.



(a) Energy consumption cost (b) Tardiness cost

Fig. 4. Sensitivity analysis for due date.

(which means that the machine is at a worse condition), reducing energy consumption cost contributes much more in reducing the total production cost.

5.4.2. Due date tightness

Three different levels of due date are considered, namely loose due date ($T = 0.4$), medium tight due date ($T = 0.5$), and tight due date ($T = 0.6$). The machine's initial lifetime is set to be 900 in this set of experiments.

The result is shown in Fig. 4. Fig. 4(a) illustrates how the energy consumption cost increases as the due date gets tight. Fig. 4(b) illustrates how the ratio of tardiness cost to total cost changes with different levels of due date. The energy consumption cost with $T = 0.3$, $D = 0.8$, and $L_0 = 900$ is used as the benchmark (denoted as $C_{\overline{E}}$). The deviation is calculated as:

$$\text{Dev} = \frac{C_E - C_{\overline{E}}}{C_{\overline{E}}} \times 100\%$$

where C_E is the energy consumption cost with different levels of due date.

The figure shows that:

(1) Tardiness cost sharply increases when due date gets tighter. This tendency gets more apparent with larger problem size.
(2) Energy consumption costs also increases with tighter due date. This is because when due date is tight, reducing tardiness cost has higher priority to obtain a solution with lower total cost. However, this can only be achieved when the machine is at a good condition with short initial life time.

6. Conclusions

We have studied an energy-efficient single machine scheduling problem derived from a rotor manufacturing company. Energy consumption is assumed to be dependent on machine reliability. A mathematical model is formulated to minimize energy consumption cost and tardiness cost. An ant colony algorithm is developed for real-world problems. The classical Emmons rules are adapted and embedded into the ACO algorithm.

Computational analyses demonstrate the efficiency of the ACO algorithm. Sensitivity analyses provide valuable information about the impact of the machine's initial lifetime and due date tightness on scheduling. Results show that when the machine is at a worse condition with large initial lifetime, the energy cost increases up to 188% while the increase of tardiness cost is limited (around 17%). It reveals that reducing energy consumption cost contributes much more in reducing the total production cost. On the other hand, when production delivery pressure is high with tight due dates, keeping the machine at a good condition is important. The managerial implication is twofold: (1) the results give the managers insight into the magnitude of the additional cost due to reliability decrease; (2) it is helpful for the decision-makers to determine appropriate scheduling strategies under different scenarios.

The methodology proposed in this study can be easily adapted to the scheduling of other machines in the workshop. An interesting extension of this research would be to integrate maintenance decision into the model. A joint optimization of scheduling and maintenance could be achieved.

Acknowledgement

This work was supported by National Natural Science Foundation of China (Project number: 51775347). Thanks are also due to the referees for their valuable comments.

References

Bampis, E., Dürr, C., Kacem, F., & Milis, I. (2012). Speed scaling with power down scheduling for agreeable deadlines. *Sustainable Computing: Informatics and Systems, 2*(4), 184–189.
Basri, E. I., Abdul Razak, I. H., Ab-Samat, H., & Kamaruddin, S. (2017). Preventive maintenance (PM) planning: A review. *Journal of Quality in Maintenance Engineering, 23*(2), 114–143.
Berrichi, A., & Yalaoui, F. (2013). Efficient bi-objective ant colony approach to minimize total tardiness and system unavailability for a parallel machine scheduling problem. *The International Journal of Advanced Manufacturing Technology, 68*(9–12), 2295–2310.
Bulsara, M. A., Hingu, A. D., & Vaghasiya, P. S. (2016). Energy loss due to unbalance in rotor–shaft system. *Journal of Engineering, Design and Technology, 14*(2), 277–285.
Che, A., Wu, X., Peng, J., & Yan, P. (2017). Energy-efficient bi-objective single-machine scheduling with power-down mechanism. *Computers & Operations Research, 85*, 172–183.
Cheng, J., Chu, F., Liu, M., Wu, P., & Xia, W. (2017). Bi-criteria single-machine batch scheduling with machine on/off switching under time-of-use tariffs. *Computers & Industrial Engineering, 112*, 721–734.
Conti, J., Holtberg, P., Diefenderfer, J., & LaRose, A. (2016). *International Energy Outlook 2016.* Washington, DC: U.S. Energy Information Administration (EIA).
Dahmus, J. B., & Gutowski, T. G. (2004). An environmental analysis of machining. *ASME 2004 international mechanical engineering congress and exposition* (pp. 643–652). Dahm: American Society of Mechanical Engineers.
Dai, M., Tang, D., Giret, A., Salido, M. A., & Li, W. D. (2013). Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm. *Robotics and Computer-Integrated Manufacturing, 29*(5), 418–429.
Ding, J.-Y., Song, S., & Wu, C. (2016). Carbon-efficient scheduling of flow shops by multi-objective optimization. *European Journal of Operational Research, 248*(3), 758–771.
Emmons, H. (1969). One-machine sequencing to minimize certain functions of job tardiness. *Operations Research, 17*(4), 701–715.
Fang, K.-T., & Lin, B. M. (2013). Parallel-machine scheduling to minimize tardiness penalty and power cost. *Computers & Industrial Engineering, 64*(1), 224–234.
Fang, K., Uhan, N. A., Zhao, F., & Sutherland, J. W. (2013). Flow shop scheduling with peak power consumption constraints. *Annals of Operations Research, 206*(1), 115–145.
Fang, K., Uhan, N. A., Zhao, F., & Sutherland, J. W. (2016). Scheduling on a single machine under time-of-use electricity tariffs. *Annals of Operations Research, 238*(1–2), 199–227.
Gahm, C., Denz, F., Dirr, M., & Tuma, A. (2016). Energy-efficient scheduling in manufacturing companies: A review and research framework. *European Journal of Operational Research, 248*(3), 744–757.
Gong, X., De Pessemier, T., Joseph, W., & Martens, L. (2015). An energy-cost-aware scheduling methodology for sustainable manufacturing. *Procedia CIRP, 29*, 185–190.
He, Y., Li, Y., Wu, T., & Sutherland, J. W. (2015). An energy-responsive optimization method for machine tool selection and operation sequence in flexible machining job shops. *Journal of Cleaner Production, 87*, 245–254.
Kanet, J. J. (2007). New precedence theorems for one-machine weighted tardiness. *Mathematics of Operations Research, 32*(3), 579–588.
Kibira, D., Hatim, Q., Kumara, S., & Shao, G. (2015). Integrating data analytics and simulation methods to support manufacturing decision making. *Proceedings of the 2015 Winter Simulation Conference* (pp. 2100–2111). Huntington Beach, California, USA: IEEE Press.
Li, W., Zein, A., Kara, S., & Herrmann, C. (2011). An investigation into fixed energy consumption of machine tools. *Glocalized solutions for sustainability in manufacturing* (pp. 268–273). Springer.
Liang, P., Yang, H.-d., Liu, G.-S., & Guo, J.-H. (2015). An ant optimization model for unrelated parallel machine scheduling with energy consumption and total tardiness. *Mathematical Problems in Engineering,* 1–8. http://www.hindawi.com/journals/mpe/2015/907034/https://doi.org/10.1155/2015/907034.
Liu, C., Yang, J., Lian, J., Li, W., Evans, S., & Yin, Y. (2014). Sustainable performance oriented operational decision-making of single machine systems with deterministic product arrival time. *Journal of Cleaner Production, 85*, 318–330.
Lu, B., Zhou, X., & Li, Y. (2016). Joint modeling of preventive maintenance and quality improvement for deteriorating single-machine manufacturing systems. *Computers & Industrial Engineering, 91*, 188–196.
Lu, C., Gao, L., Li, X., Pan, Q., & Wang, Q. (2017). Energy-efficient permutation flow shop scheduling problem using a hybrid multi-objective backtracking search algorithm. *Journal of Cleaner Production, 144*, 228–238.
Luo, H., Du, B., Huang, G. Q., Chen, H., & Li, X. (2013). Hybrid flow shop scheduling considering machine electricity consumption cost. *International Journal of Production Economics, 146*(2), 423–439.
Mansouri, S. A., Aktas, E., & Besikci, U. (2016). Green scheduling of a two-machine flowshop: Trade-off between makespan and energy consumption. *European Journal of Operational Research, 248*(3), 772–788.
Moghaddam, K. S., & Usher, J. S. (2010). Optimal preventive maintenance and replacement schedules with variable improvement factor. *Journal of Quality in Maintenance Engineering, 16*(3), 271–287.
Mori, M., Fujishima, M., Inamasu, Y., & Oda, Y. (2011). A study on energy efficiency improvement for machine tools. *CIRP Annals-Manufacturing Technology, 60*(1), 145–148.
Mouzon, G., & Yildirim, M. B. (2008). A framework to minimise total energy consumption and total tardiness on a single machine. *International Journal of Sustainable Engineering, 1*(2), 105–116.
Neugebauer, R., Wabner, M., Rentzsch, H., & Ihlenfeldt, S. (2011). Structure principles of energy efficient machine tools. *CIRP Journal of Manufacturing Science and Technology, 4*(2), 136–147.
Notice by the State Council (2017). Central People's Government of the People's Republic of China. http://www.gov.cn/zhengce/content/2017-01/05/content_5156789.htm.
Potts, C. N., & Van Wassenhove, L. (1982). A decomposition algorithm for the single machine total tardiness problem. *Operations Research Letters, 1*(5), 177–181.
Shrouf, F., Ordieres-Meré, J., García-Sánchez, A., & Ortega-Mier, M. (2014). Optimizing the production scheduling of a single machine to minimize total energy consumption costs. *Journal of Cleaner Production, 67*, 197–207.
Wang, S., & Liu, M. (2015). Multi-objective optimization of parallel machine scheduling integrated with multi-resources preventive maintenance planning. *Journal of Manufacturing Systems, 37*, 182–192.
Wang, Y.-C., Wang, M. J., & Lin, S. C. (2017). Selection of cutting conditions for power constrained parallel machine scheduling. *Robotics and Computer-Integrated Manufacturing, 43*, 105–110.
Wang, Y., Zhang, Y., & Zhu, H. (2017). *China energy statistical yearbook 2017.* Beijing: China Statistics Press.
Yan, H., & Fei, L. (2010). Methods for integrating energy consumption and environmental impact considerations into the production operation of machining processes. *Chinese Journal of Mechanical Engineering, 23*(4), 428–435.
Yan, J., & Hua, D. (2010). Energy consumption modeling for machine tools after preventive maintenance. *2010 IEEE international conference on industrial engineering and engineering management* (pp. 2201–2205). Macao, China: IEEE.
Yao, F., Demers, A., & Shenker, S. (1995). A scheduling model for reduced CPU energy. *Proceedings of IEEE 36th annual foundations of computer science* (pp. 374–382). Milwaukee, WI, USA: IEEE.
Yildirim, M. B., & Mouzon, G. (2012). Single-machine sustainable production planning to minimize total energy consumption and total completion time using a multiple objective genetic algorithm. *IEEE Transactions on Engineering Management, 59*(4), 585–597.
Yin, L., Li, X., Gao, L., Lu, C., & Zhang, Z. (2017). A novel mathematical model and multi-objective method for the low-carbon flexible job shop scheduling problem. *Sustainable Computing: Informatics and Systems, 13*, 15–30.
Yin, L., Li, X., Lu, C., & Gao, L. (2016). Energy-efficient scheduling problem using an effective hybrid multi-objective evolutionary algorithm. *Sustainability, 8*(12), 1268.