Heuristic and exact algorithms for single-machine scheduling problems with general truncated learning effects

Ji-Bo Wang[1] · Li-Han Zhang[1] · Zheng-Guo Lv[1] · Dan-Yang Lv[1] · Xin-Na Geng[2] ·
Xinyu Sun[2]

Abstract

This paper addresses single-machine scheduling problems with truncated learning effects. The objective is to determine the optimal job schedule such that the makespan, the total weighted completion time and the maximum lateness are to be minimized. All the considered problems are NP-hard; hence, for each problem, we propose the heuristic and branch-and-bound algorithms. Extensive numerical experiments validate the efficiency of the proposed solution algorithms on a set of randomly generated instances.

Keywords Scheduling · Branch-and-bound · Learning effect · Tabu search · Simulated annealing

Mathematics Subject Classification 90B35 · 90C26

1 Introduction

In many real-manufacturing systems, the job processing times can be shortened by the learning effects, such as firms and employees perform a job (task) over and over, so they learn how to perform more efficiently (Biskup 1999; Wang 2010; Wang et al. 2010; Wu et al. 2013;

J.-B. Wang, D.-Y. Lv, and X.-N. Geng have contributed equally to this work.

✉ Ji-Bo Wang
  wangjibo75@163.com

✉ Dan-Yang Lv
  2536959388@qq.com

✉ Xin-Na Geng
  2394940761@qq.com

[1] School of Computer Science, Shenyang Aerospace University, Shenyang, People's Republic of China

[2] School of Management, Xi'an Jiaotong University, Xi'an, People's Republic of China

Cheng et al. 2014; Wang et al. 2022). For details on the scheduling problems with learning effects, the reader may refer to the comprehensive survey (Azzouz et al. 2018). Kuo and Yang (2007), Lee (2011), and Soroush (2014) considered single-machine scheduling with learning effects and past sequence dependent setup times. Hsu et al. (2011) studied unrelated parallel machine scheduling with learning effects and past sequence-dependent setup times. Niu et al. (2015) addressed scheduling problems with a more general learning effect. They proved that some single-machine scheduling problems remain polynomially solvable. They also proved that some special cases of the flow shop scheduling problems remain polynomially solvable. Liang et al. (2019) considered scheduling problems with learning effects. Under flow shop setting, for some regular objectives, they proposed heuristic algorithms to solve these problems. Wang et al. (2020) examined single-machine total completion time minimization with a position-weighted learning effect. Under job release dates, they proposed a branch-and-bound algorithm and a heuristic algorithm to solve the problem. Sun et al. (2021) studied flow shop problem with general position weighted learning effects. For the total weighted completion time minimization, they proposed several heuristics and a branch-and-bound algorithm to solve the problem. Several recent articles, Liu and Jiang (2020b) and Lu et al. (2021) studied resource allocation single-machine scheduling problems with learning effects. Geng et al. (2019), Liu and Jiang (2020a), Zhao (2021) and Lv and Wang (2021) considered resource allocation flow shop scheduling problems with learning effects. Zhao (2022) addressed single-machine scheduling with general truncated learning effects and past-sequence-dependent setup times. They proved that some regular function minimizations remain polynomially solvable.

However, subject to uncontrolled learning effects (see above articles), the actual job processing time will plummet to zero dramatically due to the increasing number of jobs already processed. Hence, Wu et al. (2011), Wu et al. (2012), Li et al. (2013), Wang et al. (2013), Wu et al. (2013), and Lu et al. (2015) proposed scheduling models with truncated learning effects. Wang et al. (2019) studied flow shop problems with truncated learning effects. For the makespan and total weighted completion time minimizations, they proposed several heuristics and a branch-and-bound algorithm. Wang et al. (2021) addressed single-machine resource allocation scheduling with truncated learning effects. Under convex resource allocation, a bicriteria analysis was provided for the total weighted flow time and total resource consumption cost. Recently, Jiang et al. (2022) investigated single-machine scheduling problems with the sum of actual processing times based and a job-dependent truncation parameter based learning effects. The objective is to determine the optimal job schedule such that the makespan, the total weighted completion time and the maximum lateness are to be minimized. They proved that these three problems are NP-hard. They proved that some special cases of the problems remain polynomial solvable. For the general case of the problems, they only proposed approximation algorithms. It is natural and interesting to continue the work of Jiang et al. (2022) but study the heuristic and exact algorithms to solve the general problems with the sum of actual processing times based and a job-dependent truncation parameter based learning effects. The contributions of this research are summarized as follows: (1) we study single-machine scheduling problems with general truncated learning effects. (2) For three regular objective costs (i.e., the makespan, total weighted completion time and maximum lateness), we propose some solution algorithms (including the heuristic, tabu search, simulated annealing and branch-and-bound algorithms). (3) Extensive numerical experiments are tested to evaluate the accuracy and efficiency of the proposed algorithms.

This paper is organized as follows. The scheduling model and notations are presented in Sect. 2. Sections 3 and 4 address the branch-and-bound algorithm and meta-heuristic

algorithms. Section 4 conducts the numerical experiments. The conclusions are presented in the last section.

2 Problem description

There is a set of $\ddot{n}$ independent and non-preemptive jobs $\widetilde{J} = \{\ddot{J}_1, \ddot{J}_2, \ldots, \ddot{J}_{\ddot{n}}\}$ to be processed on a single-machine and that are available for processing at time zero. Let $[h]$ denote hth position in a schedule, $P_{[h]}^A$ be the actual processing time of job $\ddot{J}_h$. As in Jiang et al. (2022), if job $\ddot{J}_h$ is scheduled in the lth position, its actual processing time is:

$$P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\}, \tag{1}$$

where $\tilde{P}_h$ is the normal processing time of job $\ddot{J}_h$ (i.e., the processing time of a job without any learning effects), a_h ($a_h \leq 0$) is a learning factor of job $\ddot{J}_h$ and b_h ($0 \leq b_h \leq 1$) is a truncation parameter of job $\ddot{J}_h$.

For a given sequence $\tilde{\pi} = [\ddot{J}_1, \ddot{J}_2, \ldots, \ddot{J}_{\ddot{n}}]$, let $\tilde{C}_h = \tilde{C}_h(\tilde{\pi})$ (d_h, w_h) be the completion time (due date, weight) of job $\ddot{J}_h$, then $\tilde{C}_{\max} = \max\{\tilde{C}_h, h = 1, 2, \ldots, \ddot{n}\}$ is the makespan of all jobs, $\sum w_h \tilde{C}_h$ is the total weighted completion time and $L_{\max} = \max\{\tilde{C}_h - d_h | h = 1, 2, \ldots, \ddot{n}\}$ is the maximum lateness. Our goal is to determine a schedule such that γ ($\gamma \in \{\tilde{C}_{\max}, \sum w_h \tilde{C}_h, L_{\max}\}$) is minimized. The problem can be denoted by $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| \gamma$, where $\gamma \in \{\tilde{C}_{\max}, \sum w_h \tilde{C}_h, L_{\max}\}$, where the first field (i.e., 1) denotes the single-machine, the second field (i.e., $P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\}$) denotes the job characteristics, and the last field (i.e., γ) is the objective cost.

3 Branch-and-bound algorithm

From Jiang et al. (2022), the problem $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| \gamma$ ($\gamma \in \{\tilde{C}_{\max}, \sum w_h \tilde{C}_h, L_{\max}\}$) is NP-hard. Hence, branch-and-bound (denoted by B&B, where we need a lower bound and an upper bound) and heuristic algorithms are chosen to solve the problem $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| \gamma$. If $a_h = a, b_h = b$, some lemmas are given.

Lemma 1 (Jiang et al. 2022). *For the problem* $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^a, b \right\} \right|$ $\tilde{C}_{\max}$, *an optimal sequence can be obtained by the SPT rule, i.e., sequencing the jobs in non-decreasing order of* $\tilde{P}_h$.

Lemma 2 (Jiang et al. 2022). *For the problem* $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^a, b \right\} \right|$ $\sum \tilde{C}_h$, *an optimal sequence can be obtained by the SPT rule, i.e., sequencing the jobs in non-decreasing order of* $\tilde{P}_h$.

By Lemma 1 and the pairwise interchange method, we have the following results.

Lemma 3 *For the problem* $1 \left| P_{h,l}^A = \tilde{P}_h \max\left\{\left(1 + \sum_{h=1}^{l-1} P_{[h]}^A\right)^a, b\right\} \right| \sum w_h \tilde{C}_h$, *if* $\tilde{P}_h \leq \tilde{P}_j$
implies $w_h \geq w_j$ *for all jobs* $\ddot{J}_h$ *and* $\ddot{J}_j$, *an optimal sequence can be obtained by the WSPT
rule, i.e., sequencing the jobs in non-decreasing order of* $\frac{\tilde{P}_h}{w_h}$.

Lemma 4 *For the problem* $1 \left| P_{h,l}^A = \tilde{P}_h \max\left\{\left(1 + \sum_{h=1}^{l-1} P_{[h]}^A\right)^a, b\right\} \right| L_{\max}$, *if* $\tilde{P}_h \leq \tilde{P}_j$
implies $d_h \leq d_j$ *for all jobs* $\ddot{J}_h$ *and* $\ddot{J}_j$, *an optimal sequence can be obtained by the EDD
rule, i.e., sequencing the jobs in non-decreasing order of* d_h.

Lemma 5 (Hardy et al. 1967). *The term* $\sum_{j=1}^n x_j y_j$ *gets the minimum, when sequence* x_j
and sequence y_j *are arranged in opposite monotonous order.*

3.1 Lower bounds

3.1.1 Criterion $C_{\max}$

Let $\tilde{\pi} = (\tilde{\pi}_{SP}, \tilde{\pi}_{UP})$ be a sequence, where $\tilde{\pi}_{SP}$ ($\tilde{\pi}_{UP}$) is the scheduled (unscheduled) part,
and there are θ jobs in $\tilde{\pi}_{SP}$; hence, the completion time of the $\theta + 1$th job is

$$\tilde{C}_{[\theta+1]}(\tilde{\pi}_{UP}) = \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) + \tilde{P}_{[\theta+1]} \max\left\{\left(1 + \sum_{h=1}^{\theta} P_{[h]}^A\right)^{a_{[\theta+1]}}, b_{[\theta+1]}\right\}.$$

Similarly,

$$\tilde{C}_{[\theta+j]}(\tilde{\pi}_{UP}) = \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) + \sum_{v=1}^{j} \tilde{P}_{[\theta+v]}$$
$$\times \max\left\{\left(1 + \sum_{h=1}^{\theta} P_{[h]}^A + \sum_{h=\theta+1}^{\theta+v-1} P_{[h]}^A\right)^{a_{[\theta+v]}}, b_{[\theta+v]}\right\},$$

(2)

where $\sum_{h=k}^{k-1} P_{[h]}^A = 0$ and $1 \leq j \leq \ddot{n} - \theta$.

Let $a_{\min} = \min\{a_h | h = \theta+1, \theta+2, \ldots, \ddot{n}\}$ and $b_{\min} = \min\{b_h | h = \theta+1, \theta+2, \ldots, \ddot{n}\}$,
from Eq. (2), we have

$$\tilde{C}_{\max}(\tilde{\pi}) = \tilde{C}_{[\ddot{n}]}(\tilde{\pi})$$
$$\geq \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) + \sum_{v=1}^{\ddot{n}-\theta} \tilde{P}_{[\theta+v]} \max\left\{\left(1 + \sum_{h=1}^{\theta} P_{[h]}^A + \sum_{h=\theta+1}^{\theta+v-1} P_{[h]}^A\right)^{a_{\min}}, b_{\min}\right\}. (3)$$

From Eq. (3), the terms $\tilde{C}_{[\theta]}(\tilde{\pi}_{SP})$ and $\sum_{h=1}^{\theta} P_{[h]}^A$ are fixed constants, the term
$\sum_{v=1}^{\ddot{n}-\theta} \tilde{P}_{[\theta+v]} \max\left\{\left(1 + \sum_{h=1}^{\theta} P_{[h]}^A + \sum_{h=\theta+1}^{\theta+v-1} P_{[h]}^A\right)^{a_{\min}}, b_{\min}\right\}$ can be minimized by the
SPT rule of the unscheduled jobs (see Lemma 1). Therefore, the lower bound of
$1 \left| P_{h,l}^A = \tilde{P}_h \max\left\{\left(1 + \sum_{h=1}^{l-1} P_{[h]}^A\right)^{a_h}, b_h\right\} \right| \tilde{C}_{\max}$ can be obtained:

$$LB_{\tilde{C}_{\max}} = \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) + \sum_{v=1}^{\ddot{n}-\theta} \tilde{P}_{<\theta+v>} \max\left\{\left(1 + \sum_{h=1}^{\theta} P_{[h]}^A + \sum_{h=\theta+1}^{\theta+v-1} P_{<h>}^A\right)^{a_{\min}}, b_{\min}\right\}, (4)$$

where $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$ and $P^A_{<h>}$ $(h = \theta + 1, \theta + 2, \ldots, \ddot{n})$ is the actual processing time of hth position by the order of $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$.

3.1.2 Criterion $\sum w_h \tilde{C}_h$

From Eq. (2), we have

$$\sum w_h \tilde{C}_h(\tilde{\pi})$$

$$= \sum_{h=1}^{\theta} w_{[h]} \tilde{C}_{[h]}(\tilde{\pi}_{SP}) + \sum_{h=\theta+1}^{\ddot{n}} w_{[h]} \tilde{C}_{[h]}(\tilde{\pi}_{UP})$$

$$= \sum_{h=1}^{\theta} w_{[h]} \tilde{C}_{[h]}(\tilde{\pi}_{SP}) + \sum_{h=\theta+1}^{\ddot{n}} \tilde{C}_{[\theta]} w_{[h]}$$

$$+ \sum_{h=\theta+1}^{\ddot{n}} w_{[h]} \sum_{v=1}^{h-\theta} \tilde{P}_{[\theta+v]} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{j=\theta+1}^{\theta+v-1} P^A_{[j]} \right)^{a_{[\theta+v]}}, b_{[\theta+v]} \right\}. \quad (5)$$

Let $a_{\min} = \min\{a_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$, $b_{\min} = \min\{b_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$ and $w_{\min} = \min\{w_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$, from Eq. (5), we have

$$\sum w_h \tilde{C}_h(\tilde{\pi})$$

$$\geq \sum_{h=1}^{\theta} w_{[h]} \tilde{C}_{[h]}(\tilde{\pi}_{SP}) + \sum_{h=\theta+1}^{\ddot{n}} \tilde{C}_{[\theta]} w_{[h]}$$

$$+ w_{\min} \sum_{h=\theta+1}^{\ddot{n}} \sum_{v=1}^{h-\theta} \tilde{P}_{[\theta+v]} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{j=\theta+1}^{\theta+v-1} P^A_{[j]} \right)^{a_{\min}}, b_{\min} \right\}. \quad (6)$$

From Eq. (6), the terms $\sum_{h=1}^{\theta} w_{[h]} \tilde{C}_{[h]}(\tilde{\pi}_{SP})$, $\sum_{h=\theta+1}^{\ddot{n}} \tilde{C}_{[\theta]} w_{[h]}$ and $\sum_{h=1}^{\theta} P^A_{[h]}$ are fixed constants, the term $\sum_{h=\theta+1}^{\ddot{n}} \sum_{v=1}^{h-\theta} \tilde{P}_{[\theta+v]} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{j=\theta+1}^{\theta+v-1} P^A_{[j]} \right)^{a_{\min}}, b_{\min} \right\}$ can be minimized by the SPT rule of the unscheduled jobs (see Lemma 2). Therefore, the first lower bound of $1 \left| P^A_{h,l} = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P^A_{[h]} \right)^{a_h}, b_h \right\} \right| \sum w_h \tilde{C}_h$ can be obtained:

$$LB^1_{\sum w_h \tilde{C}_h}$$

$$= \sum_{h=1}^{\theta} w_{[h]} \tilde{C}_{[h]}(\tilde{\pi}_{SP}) + \tilde{C}_{[\theta]} \sum_{h=\theta+1}^{\ddot{n}} w_h$$

$$+ w_{\min} \sum_{h=\theta+1}^{\ddot{n}} \sum_{v=1}^{h-\theta} \tilde{P}_{<\theta+v>} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{j=\theta+1}^{\theta+v-1} P^A_{<j>} \right)^{a_{\min}}, b_{\min} \right\}, \quad (7)$$

where $_{<h>}$ denotes hth position in a sequence such that $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$ and $P^A_{<h>}$ $(h = \theta + 1, \theta + 2, \ldots, \ddot{n})$ is the actual processing time of hth position by the order of $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$.

From Eq. (5), the term $\sum_{h=\theta+1}^{\ddot{n}} w_{[h]} \sum_{v=1}^{h-\theta} \tilde{P}_{[\theta+v]} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{j=\theta+1}^{\theta+v-1} P^A_{[j]} \right)^{a_{[\theta+v]}}, b_{[\theta+v]} \right\} = \sum_{h=\theta+1}^{\ddot{n}} w_{[h]} \sum_{v=1}^{h-\theta} P^A_{[\theta+v]}$. According to HLP rule (see Lemma

[5](#)), we have

$$\sum_{h=\theta+1}^{\ddot{n}} w_{[h]} \sum_{v=1}^{h-\theta} P_{[\theta+v]}^{A}$$

$$= w_{[\theta+1]} P_{[\theta+1]}^{A} + w_{[\theta+2]} \left(P_{[\theta+1]}^{A} + P_{[\theta+2]}^{A} \right) + \cdots$$

$$+ w_{[n]} \left(P_{[\theta+1]}^{A} + P_{[\theta+2]}^{A} + \cdots + P_{[n]}^{A} \right)$$

$$\geq w_{(\theta+1)} P_{[\theta+1]}^{A} + w_{(\theta+2)} \left(P_{[\theta+1]}^{A} + P_{[\theta+2]}^{A} \right) + \cdots$$

$$+ w_{(n)} \left(P_{[\theta+1]}^{A} + P_{[\theta+2]}^{A} + \cdots + P_{[n]}^{A} \right),$$

where $w_{(\theta+1)} \geq w_{(\theta+2)} \geq \cdots \geq w_{(\ddot{n})}$.

Let $a_{\min} = \min\{a_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$, $b_{\min} = \min\{b_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$, according to HLP rule (see Lemma [5](#)), we have

$$w_{(\theta+1)} P_{[\theta+1]}^{A} + w_{(\theta+2)} \left(P_{[\theta+1]}^{A} + P_{[\theta+2]}^{A} \right) + \cdots + w_{(n)} \left(P_{[\theta+1]}^{A} + P_{[\theta+2]}^{A} + \cdots + P_{[n]}^{A} \right)$$

$$= P_{[\theta+1]}^{A} \left(w_{(\theta+1)} + w_{(\theta+2)} + \cdots + w_{(n)} \right)$$

$$+ P_{[\theta+2]}^{A} \left(w_{(\theta+2)} + w_{(\theta+3)} + \cdots + w_{(n)} \right) + \cdots + P_{[n]}^{A} w_{(n)}$$

$$\geq P_{<\theta+1>}^{A} \left(w_{(\theta+1)} + w_{(\theta+2)} + \cdots + w_{(n)} \right)$$

$$+ P_{<\theta+2>}^{A} \left(w_{(\theta+2)} + w_{(\theta+3)} + \cdots + w_{(n)} \right) + \cdots + P_{<n>}^{A} w_{(n)},$$

where $P_{<\theta+v>}^{A} = \tilde{P}_{<\theta+v>} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P_{[h]}^{A} + \sum_{j=\theta+1}^{\theta+v-1} P_{<j>}^{A} \right)^{a_{\min}}, b_{\min} \right\}$, $v = 1, 2, \ldots, \ddot{n} - \theta$, $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$.

Therefore, the second lower bound of $1 \left| P_{h,l}^{A} = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^{A} \right)^{a_h}, b_h \right\} \right| \sum w_h \tilde{C}_h$ can be obtained:

$$LB_{\sum w_h \tilde{C}_h}^{2}$$

$$= \sum_{h=1}^{\theta} w_{[h]} \tilde{C}_{[h]}(\tilde{\pi}_{SP}) + \tilde{C}_{[\theta]} \sum_{h=\theta+1}^{\ddot{n}} w_h$$

$$+ \sum_{h=\theta+1}^{\ddot{n}} w_{(h)} \sum_{v=1}^{h-\theta} \tilde{P}_{<\theta+v>} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P_{[h]}^{A} + \sum_{j=\theta+1}^{\theta+v-1} P_{<j>}^{A} \right)^{a_{\min}}, b_{\min} \right\}, \quad (8)$$

where $w_{(\theta+1)} \geq w_{(\theta+2)} \geq \cdots \geq w_{(\ddot{n})}$, $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$ (note that $\tilde{P}_{<h>}$ and $w_{(h)}$ do not necessarily correspond to the same job), $P_{<h>}^{A} = \tilde{P}_{<h>} \max \left\{ \left(1 + \sum_{j=1}^{\theta} P_{[j]}^{A} + \sum_{j=\theta+1}^{h-1} P_{<j>}^{A} \right)^{a_{\min}}, b_{\min} \right\}$, $h = \theta + 1, \theta + 2, \ldots, \ddot{n}$, and $P_{<h>}^{A}$ ($h = \theta + 1, \theta + 2, \ldots, \ddot{n}$) is the actual processing time of hth position by the order of $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$.

Remark If $\tilde{P}_h$ and w_h ($h = \theta + 1, \theta + 2, ..., \ddot{n}$) satisfy agreeable condition (i.e., $\tilde{P}_h \leq \tilde{P}_j$ implies $w_h \geq w_j$), then $\tilde{P}_{<h>}$ and $w_{(h)}$ correspond to the same job. Otherwise, $\tilde{P}_{<h>}$ and $w_{(h)}$ do not necessarily correspond to the same job.

Combining $LB^1_{\sum w_h \tilde{C}_h}$ and $LB^2_{\sum w_h \tilde{C}_h}$, the lower bound of $1 \left| P^A_{h,l} = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P^A_{[h]} \right)^{a_h}, b_h \right\} \right| \sum w_h \tilde{C}_h$ is

$$LB_{\sum w_h \tilde{C}_h} = \max \left\{ LB^1_{\sum w_h \tilde{C}_h}, LB^2_{\sum w_h \tilde{C}_h} \right\}. \tag{9}$$

3.1.3 Criterion $L_{\max}$

Similar to Sects. 3.1.1 and 3.1.2, let $a_{\min} = \min\{a_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$, $b_{\min} = \min\{b_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$ and $d_{\max} = \max\{d_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$, it follows that

$$L_{[\theta+j]}(\tilde{\pi}) = \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) + \sum_{v=1}^{j} \tilde{P}_{[\theta+v]} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{h=\theta+1}^{\theta+v-1} P^A_{[h]} \right)^{a_{[\theta+v]}}, b_{[\theta+v]} \right\}$$
$$-d_{[\theta+j]}, \tag{10}$$

where $\sum_{h=k}^{k-1} P^A_{[h]} = 0$ and $1 \le j \le \ddot{n} - \theta$.

Hence,

$$L_{[\theta+j]}(\tilde{\pi})$$
$$= \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) + \sum_{v=1}^{j} \tilde{P}_{[\theta+v]} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{h=\theta+1}^{\theta+v-1} P^A_{[h]} \right)^{a_{[\theta+v]}}, b_{[\theta+v]} \right\} - d_{[\theta+j]}$$
$$\ge \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) + \sum_{v=1}^{j} \tilde{P}_{[\theta+v]} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{h=\theta+1}^{\theta+v-1} P^A_{[h]} \right)^{a_{\min}}, b_{\min} \right\} - d_{\max}, \tag{11}$$

where $1 \le j \le \ddot{n} - \theta$.

It is noticed that $\sum_{v=1}^{j} \tilde{P}_{[\theta+v]} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{h=\theta+1}^{\theta+v-1} P^A_{[h]} \right)^{a_{\min}}, b_{\min} \right\}$ can be minimized by the SPT rule (see Lemma 1). Therefore, the first lower bound of the problem $1 | P^A_{h,l} = \tilde{P}_h \times \max \left\{ \left(1 + \sum_{h=1}^{l-1} P^A_{[h]} \right)^{a_h}, b_h \right\} | L_{\max}$ is:

$$LB^1_{L_{\max}} = \max \left\{ \tilde{C}_{[1]} - d_{[1]}, \tilde{C}_{[2]} - d_{[2]}, \ldots, \tilde{C}_{[\theta]} - d_{[\theta]}, \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) \right.$$
$$\left. + \sum_{v=1}^{\ddot{n}-\theta} \tilde{P}_{<\theta+v>} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P^A_{[h]} + \sum_{h=\theta+1}^{\theta+v-1} P^A_{<h>} \right)^{a_{\min}}, b_{\min} \right\} - d_{\max} \right\}, \tag{12}$$

where $\tilde{P}_{<\theta+1>} \le \tilde{P}_{<\theta+2>} \le \cdots \le \tilde{P}_{<\ddot{n}>}$ and $P^A_{<h>}$ ($h = \theta + 1, \theta + 2, \ldots, \ddot{n}$) is the actual processing time of hth position by the order of $\tilde{P}_{<\theta+1>} \le \tilde{P}_{<\theta+2>} \le \cdots \le \tilde{P}_{<\ddot{n}>}$.

According to the definition of maximum lateness, we have $L_{\max} = \max\{\tilde{C}_{[1]} - d_{[1]}, \tilde{C}_{[2]} - d_{[2]}, \ldots, \tilde{C}_{[\theta]} - d_{[\theta]}, \tilde{C}_{[\theta+1]} - d_{[\theta+1]}, \ldots, \tilde{C}_{[\ddot{n}]} - d_{[\ddot{n}]}\}$. Assuming that the completion time $\tilde{C}_{[\theta+j]}$ ($j = 1, 2, \ldots, \ddot{n} - \theta$) is a fixed value, according to EDD rule (see Lawler 1973), we have

$$\max\{\tilde{C}_{[\theta+1]} - d_{[\theta+1]}, \tilde{C}_{[\theta+2]} - d_{[\theta+2]}, \ldots, \tilde{C}_{[\ddot{n}]} - d_{[\ddot{n}]}\}$$
$$\ge \max\{\tilde{C}_{[\theta+1]} - d_{(\theta+1)}, \tilde{C}_{[\theta+2]} - d_{(\theta+2)}, \ldots, \tilde{C}_{[\ddot{n}]} - d_{(\ddot{n})}\},$$

where $d_{(\theta+1)} \leq d_{(\theta+2)} \leq \cdots \leq d_{(\ddot{n})}$.

Let $a_{\min} = \min\{a_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$, $b_{\min} = \min\{b_h | h = \theta + 1, \theta + 2, \ldots, \ddot{n}\}$, according to Lemma 1, we have

$$\max_{j=1}^{\ddot{n}-\theta} \left\{ \tilde{C}_{[\theta+j]} - d_{(\theta+j)} \right\}$$

$$= \max_{j=1}^{\ddot{n}-\theta} \left\{ \tilde{C}_{[\theta]} + \sum_{v=1}^{j} P_{[\theta+v]}^A - d_{(\theta+j)} \right\}$$

$$\geq \tilde{C}_{[\theta]} + \max_{j=1}^{\ddot{n}-\theta} \left\{ \sum_{v=1}^{j} P_{<\theta+v>}^A - d_{(\theta+j)} \right\},$$

where $P_{<\theta+v>}^A = \tilde{P}_{<\theta+v>} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P_{[h]}^A + \sum_{h=\theta+1}^{\theta+v-1} P_{<h>}^A \right)^{a_{\min}}, b_{\min} \right\}$, $v = 1, 2, \ldots, j$, $j = 1, 2, \ldots, \ddot{n} - \theta$ and $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$.

Therefore, the second lower bound of the problem $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| L_{\max}$ is

$$LB_{L_{\max}}^2 = \max \left\{ \tilde{C}_{[1]} - d_{[1]}, \tilde{C}_{[2]} - d_{[2]}, \ldots, \tilde{C}_{[\theta]} - d_{[\theta]}, \tilde{C}_{[\theta]}(\tilde{\pi}_{SP}) \right.$$
$$\left. + \max_{j=1}^{\ddot{n}-\theta} \left\{ \sum_{v=1}^{j} \tilde{P}_{<\theta+v>} \max \left\{ \left(1 + \sum_{h=1}^{\theta} P_{[h]}^A + \sum_{h=\theta+1}^{\theta+v-1} P_{<h>}^A \right)^{a_{\min}}, b_{\min} \right\} - d_{(\theta+j)} \right\} \right\},$$
(13)

where $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$, $d_{(\theta+1)} \leq d_{(\theta+2)} \leq \cdots \leq d_{(\ddot{n})}$ (note that $\tilde{P}_{<h>}$ and $d_{(h)}$ do not necessarily correspond to the same job) and $P_{<h>}^A$ ($h = \theta + 1, \theta + 2, \ldots, \ddot{n}$) is the actual processing time of hth position by the order of $\tilde{P}_{<\theta+1>} \leq \tilde{P}_{<\theta+2>} \leq \cdots \leq \tilde{P}_{<\ddot{n}>}$.

Combining $LB_{L_{\max}}^1$ and $LB_{L_{\max}}^2$, the lower bound of $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| L_{\max}$ is

$$LB_{L_{\max}} = \max \left\{ LB_{L_{\max}}^1, LB_{L_{\max}}^2 \right\}.$$
(14)

3.2 Upper bound

For the optimization of minimizing regular objective costs, any feasible solution can be proposed as an upper bound. Obviously, the SPT rule is a heuristic for $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| \tilde{C}_{\max}$, the WSPT rule is a heuristic for $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| \sum w_h \tilde{C}_h$, the EDD rule is a heuristic for $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| L_{\max}$. In addition, an improvement scheme (i.e., an efficient constructive scheme for the m-machine flow shop problem $Fm||\sum \tilde{C}_h$, see Framinan and Leisten 2003) can be adopt for $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| \gamma$,

where $\gamma \in \{\tilde{C}_{\max}, \sum w_h \tilde{C}_h, L_{\max}\}$. Details of the upper bound (UB) algorithm is presented as follows:

Algorithm 1 (UB)

Step 1. Arrange the jobs by the SPT/WSPT/EDD rule (SPT for $\tilde{C}_{\max}$, WSPT for $\sum w_h \tilde{C}_h$, and EDD for $L_{\max}$).

Step 2. Set $r = 2$. Select the first two jobs from the sorted list and select the better of the two possible sequences.

Step 3. Increment r, $r = r + 1$ (where r denotes the rth job from the sorted list using Step 1). Select the rth job from the sorted list and insert it into r possible positions of the best partial sequence obtained so far. Among the r sequences, the best r-job partial sequence is selected based on minimum $\tilde{C}_{\max}$ ($\sum w_h \tilde{C}_h$, $L_{\max}$). Next, determine all possible sequences by interchanging jobs in positions i and j of the above partial sequence for all i, j ($1 \leq i < r, i < j \leq r$). Select the best partial sequence among $r(r-1)/2$ sequences having minimum $\tilde{C}_{\max}$ ($\sum w_h \tilde{C}_h$, $L_{\max}$).

Step 4. If $r = \ddot{n}$, then STOP; otherwise, go to Step 3.

3.3 Branch and bound algorithm

B&B algorithm search follows a depth-first strategy, this algorithm assigns jobs in a forward manner starting from the first job position (assign a job to a node).

Algorithm 2 (B&B)

Step 1. (Upper bound) Calculate the initial upper bound by Algorithm 1 (UB).

Step 2. (Bounding) Calculate the lower bound (see Eq. (4) for $\tilde{C}_{\max}$, Eq. (9) for $\sum w_h \tilde{C}_h$, and Eq. (14) for $L_{\max}$) for the node. If the lower bound for an unfathomed partial sequence of jobs is larger than or equal to the value of $\tilde{C}_{\max}$ ($\sum w_h \tilde{C}_h$ and $L_{\max}$) of the initial solution, eliminate the node and all the nodes following it in the branch. Calculate the value $\tilde{C}_{\max}$ ($\sum w_h \tilde{C}_h$ and $L_{\max}$) of the completed sequence, if it is less than the initial solution, replace it as the new solution; otherwise, eliminate it.

Step 3. (Termination) Continue until all nodes have been explored.

For better understanding of B&B algorithm, an example is proposed to explain the B&B algorithm as follows.

Example 1 Consider the problem $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| \tilde{C}_{\max}$, where $\ddot{n} = 5$ and the other corresponding parameters are given in Table 1.

Solution

According to the SPT rule, the sequence is $\ddot{J}_1 \rightarrow \ddot{J}_2 \rightarrow \ddot{J}_3 \rightarrow \ddot{J}_4 \rightarrow \ddot{J}_5$, and the corresponding objective function value is $\tilde{C}_{\max} = 28.4000$. For the $\tilde{C}_{\max}$ minimization, the search tree of the B&B algorithm can be seen from Fig. 1. The numbers in Fig. 1 represent the lower bound values, and J_0 is defined as the level 0.

Table 1 Data for Example 1	$\ddot{J}_h$	$\ddot{J}_1$	$\ddot{J}_2$	$\ddot{J}_3$	$\ddot{J}_4$	$\ddot{J}_5$
	$\tilde{P}_h$	4	7	8	9	10
	a_h	−0.1	−0.25	−0.2	−0.15	−0.22
	b_h	0.9	0.7	0.65	0.7	0.8

Table 2 CPU time (ms) results of the algorithms for $\tilde{C}_{max}$

| n	a	SPT		UB		TS		SA		B&B	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
9	−0.05	0.05	1.00	0.80	2.00	4088.45	4467.00	21.30	27.00	76.80	265.00
	−0.1	0.05	1.00	0.75	1.00	4332.35	4535.00	21.65	27.00	158.80	424.00
	−0.15	0.20	1.00	0.80	1.00	4265.35	4383.00	21.30	26.00	233.80	541.00
	−0.2	0.00	0.00	0.80	1.00	4314.00	4457.00	21.65	26.00	272.90	599.00
	−0.25	0.00	0.00	0.80	1.00	4446.65	4998.00	22.70	31.00	205.50	435.00
10	−0.05	0.00	0.00	0.95	8.00	7263.75	7601.00	27.60	35.00	509.55	1896.00
	−0.1	0.00	0.00	0.85	2.00	4384.50	8173.00	21.05	31.00	1820.40	4183.00
	−0.15	0.00	0.00	0.95	2.00	4425.80	9796.00	24.35	40.00	2468.10	8396.00
	−0.2	0.00	0.00	0.55	2.00	7287.60	7810.00	29.25	45.00	2313.55	5081.00
	−0.25	0.00	0.00	0.85	3.00	4589.30	10421.00	22.20	41.00	2659.85	5284.00
11	−0.05	0.15	1.00	0.95	2.00	8463.30	8598.00	26.00	33.00	6423.35	33551.00
	−0.1	0.05	1.00	0.85	2.00	8459.10	8618.00	25.75	29.00	11930.60	29289.00
	−0.15	0.00	0.00	1.00	2.00	8584.20	8824.00	26.60	29.00	12655.75	38784.00
	−0.2	0.05	1.00	0.80	1.00	8584.65	8705.00	26.55	29.00	24435.93	54631.00
	−0.25	0.10	1.00	0.90	2.00	8576.95	8771.00	26.95	34.00	22694.35	47719.00
12	−0.05	0.10	1.00	1.10	2.00	13192.50	13768.00	31.70	37.00	69807.40	269376.00
	−0.1	0.10	1.00	1.10	2.00	13206.40	13736.00	31.55	39.00	128750.40	428109.00
	−0.15	0.10	1.00	1.30	2.00	13671.15	17272.00	34.40	48.00	178991.30	542462.00
	−0.2	0.05	1.00	1.30	2.00	13265.40	13628.00	31.20	35.00	22256.15	665503.00
	−0.25	0.10	1.00	1.05	2.00	13111.85	13746.00	31.15	35.00	292446.05	693448.00
13	−0.05	0.05	1.00	1.05	2.00	13155.60	17916.00	25.60	37.00	308557.90	1635260.00
	−0.1	0.10	1.00	1.10	3.00	17608.60	47163.00	35.30	74.00	1040881.50	3760330.00
	−0.15	0.05	1.00	1.05	2.00	15980.20	22494.00	32.65	51.00	2199233.70	6097080.00
	−0.2	0.05	1.00	1.00	4.00	13390.20	22233.00	26.80	42.00	2274058.50	7848020.00
	−0.25	0.10	1.00	0.85	2.00	11160.20	22214.00	25.35	47.00	3328439.00	9932900.00
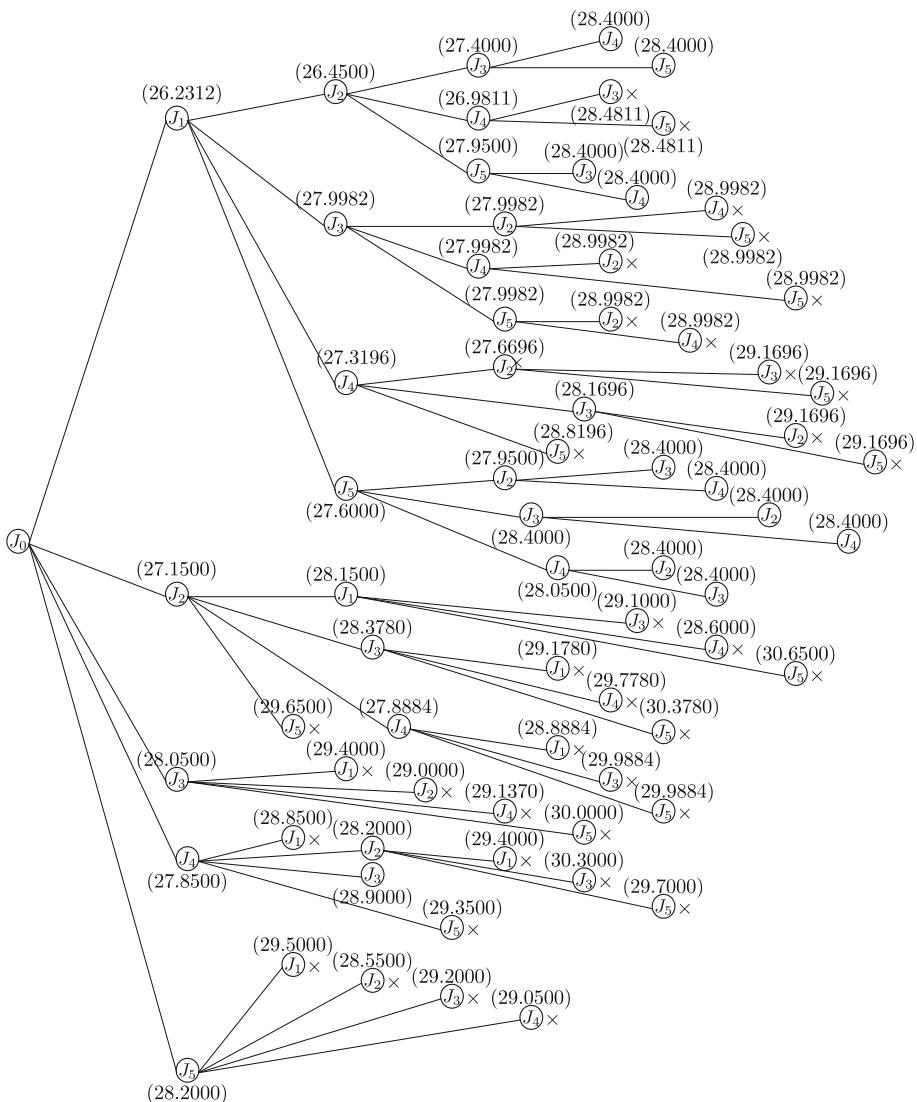
Fig. 1 Search tree of the B&B algorithm for Example 1 ($\times$ denotes pruning)

At level 1, for job $\ddot{J}_1$: $\theta = 1$, from Eq. (4), we have

$$
LB(\ddot{J}_1) = \tilde{C}_{[1]} + \sum_{v=1}^{4} \tilde{P}_{<\theta+v>} \max \left\{ \left(1 + \sum_{h=1}^{1} P_{[h]}^{A} + \sum_{h=2}^{v} P_{<h>}^{A}\right)^{a_{\min}}, b_{\min} \right\}
$$

$$
= 4 + 7 \times \max \left\{(1+4)^{-0.25}, 0.65\right\} + 8 \times \max \left\{(1+4+4.6812)^{-0.25}, 0.65\right\}
$$

$$
+ 9 \times \max \left\{(1+4+4.6812+5.2)^{-0.25}, 0.65\right\}
$$

$$
+ 10 \times \max \left\{(1+4+4.6812+5.2+5.85)^{-0.25}, 0.65\right\}
$$

Table 3 Error results of the algorithms for $\tilde{C}_{\max}$

n	a	SPT		UB		TS		SA	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max
9	−0.05	1.02091	1.04482	1.00142	1.00503	1.00603	1.02865	1.00016	1.00066
	−0.1	1.02472	1.05324	1.00169	1.00503	1.00721	1.02865	1.00015	1.00066
	−0.15	1.03974	1.08096	1.00222	1.00824	1.01222	1.04506	1.00021	1.00117
	−0.2	1.04438	1.08592	1.00169	1.01028	1.01148	1.02956	1.00026	1.00212
	−0.25	1.05442	1.09847	1.00354	1.02038	1.00237	1.04860	1.00097	1.00834
10	−0.05	1.00593	1.01862	1.00032	1.00149	1.00088	1.00720	1.00010	1.00045
	−0.1	1.02931	1.04773	1.00158	1.00949	1.00643	1.02068	1.00027	1.00114
	−0.15	1.05097	1.18224	1.00236	1.01205	1.01558	1.05284	1.00057	1.00267
	−0.2	1.05960	1.10623	1.00373	1.01667	1.02074	1.05344	1.00092	1.00477
	−0.25	1.06462	1.12497	1.00392	1.01814	1.02593	1.05177	1.00083	1.00410
11	−0.05	1.00856	1.02254	1.00036	1.00257	1.00208	1.01324	1.00013	1.00045
	−0.1	1.02931	1.04876	1.00173	1.00723	1.00938	1.02689	1.00031	1.00144
	−0.15	1.03570	1.06274	1.00201	1.00703	1.01466	1.03613	1.00082	1.00469
	−0.2	1.04695	1.08463	1.00362	1.01988	1.02216	1.04144	1.00088	1.00434
	−0.25	1.04525	1.08746	1.00381	1.01509	1.02479	1.04668	1.00210	1.00730
12	−0.05	1.00892	1.01963	1.00039	1.00222	1.00220	1.00915	1.00027	1.00099
	−0.1	1.02262	1.04439	1.00196	1.00823	1.01096	1.03905	1.00081	1.00243
	−0.15	1.03915	1.06274	1.00221	1.00835	1.01741	1.03848	1.00102	1.00318
	−0.2	1.04601	1.06381	1.00255	1.02351	1.02405	1.05194	1.00104	1.00485
	−0.25	1.04936	1.08155	1.00247	1.00646	1.02338	1.04788	1.00167	1.00930
13	−0.05	1.01542	1.02594	1.00082	1.00385	1.00238	1.01512	1.00046	1.00190
	−0.1	1.02911	1.04215	1.00262	1.00861	1.01124	1.01926	1.00106	1.00283
	−0.15	1.04746	1.06791	1.00181	1.00836	1.01916	1.04081	1.00189	1.00697
	−0.2	1.05762	1.10187	1.00429	1.01673	1.02197	1.04428	1.00465	1.01476
	−0.25	1.07245	1.10073	1.00400	1.02564	1.03164	1.04985	1.00371	1.01307

$$= 4 + 4.6812 + 5.2 + 5.85 + 6.5$$
$$= 26.2312.$$

The calculation process of the remaining lower bounds and values of $\tilde{C}_{\max}$ can be obtained similarly. From Fig. 1, we can obtain that the optimal job sequence is $\pi^* = (\ddot{J}_1, \ddot{J}_2, \ddot{J}_3, \ddot{J}_4, \ddot{J}_5), (\ddot{J}_1, \ddot{J}_2, \ddot{J}_3, \ddot{J}_5, \ddot{J}_4), (\ddot{J}_1, \ddot{J}_2, \ddot{J}_5, \ddot{J}_3, \ddot{J}_4), (\ddot{J}_1, \ddot{J}_2, \ddot{J}_5, \ddot{J}_4, \ddot{J}_3),$ $(\ddot{J}_1, \ddot{J}_5, \ddot{J}_2, \ddot{J}_3, \ddot{J}_4), (\ddot{J}_1, \ddot{J}_5, \ddot{J}_2, \ddot{J}_4, \ddot{J}_3), (\ddot{J}_1, \ddot{J}_5, \ddot{J}_3, \ddot{J}_2, \ddot{J}_4), (\ddot{J}_1, \ddot{J}_5, \ddot{J}_3, \ddot{J}_4, \ddot{J}_2), (\ddot{J}_1, \ddot{J}_5, \ddot{J}_4,$ $\ddot{J}_2, \ddot{J}_3), (\ddot{J}_1, \ddot{J}_5, \ddot{J}_4, \ddot{J}_3, \ddot{J}_2)$, and the optimal value $\tilde{C}_{\max} = 28.4000$.

4 Meta-heuristic algorithms

4.1 Tabu search

In this subsection, the tabu search (TS) algorithm is used to find a near-optimal solution. The initial sequence used in the TS algorithm is chosen by arranging the jobs by the

Table 4 CPU time (ms) results of the algorithms for $\sum w_h \tilde{C}_h$

n	a	WSPT		UB		TS		SA		B&B	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
9	−0.05	0.00	0.00	0.70	1.00	4055.42	4429.00	21.60	25.00	17.20	76.00
	−0.1	0.00	0.00	0.95	1.00	4265.05	4390.00	22.10	27.00	16.35	57.00
	−0.15	0.00	0.00	0.70	1.00	4248.85	4395.00	22.20	27.00	34.00	200.00
	−0.2	0.00	0.00	0.75	2.00	4307.10	4429.00	21.80	28.00	75.55	708.00
	−0.25	0.00	0.00	0.75	1.00	4477.95	5298.00	21.95	30.00	28.70	57.00
10	−0.05	0.00	0.00	0.45	1.00	7325.20	7641.00	27.65	35.00	59.10	182.00
	−0.1	0.00	0.00	0.80	2.00	4347.80	7581.00	23.00	37.00	89.90	373.00
	−0.15	0.00	0.00	0.95	2.00	4455.55	9740.00	22.90	35.00	85.55	242.00
	−0.2	0.00	0.00	0.45	1.00	7303.55	7861.00	28.35	48.00	312.42	1103.00
	−0.25	0.00	0.00	0.70	2.00	4623.75	10911.00	22.95	41.00	217.85	639.00
11	−0.05	0.05	1.00	0.90	1.00	8490.00	8608.00	26.60	29.00	272.30	1349.00
	−0.1	0.05	1.00	0.75	1.00	8511.60	8627.00	26.50	29.00	388.40	1183.00
	−0.15	0.00	0.00	0.80	1.00	8627.25	8746.00	26.25	28.00	502.15	1774.00
	−0.2	0.05	1.00	0.70	1.00	8646.50	8742.00	26.45	28.00	1141.05	7578.00
	−0.25	0.00	0.00	0.90	2.00	8673.05	8861.00	26.80	29.00	1279.55	4991.00
12	−0.05	0.00	0.00	1.20	2.00	13358.10	14051.00	32.90	42.00	852.70	6113.00
	−0.1	0.00	0.00	1.05	2.00	13258.70	14224.00	32.50	38.00	1818.95	5198.00
	−0.15	0.00	0.00	1.30	5.00	13771.11	17482.00	33.70	49.00	3969.10	15815.00
	−0.2	0.00	0.00	1.05	2.00	13320.35	14172.00	33.20	43.00	5053.55	18225.00
	−0.25	0.00	0.00	1.05	2.00	13280.95	13903.00	31.75	40.00	12027.70	121611.00
13	−0.05	0.00	0.00	1.00	2.00	13311.35	18180.00	25.70	37.00	9730.10	138380.00
	−0.1	0.00	0.00	1.45	4.00	17677.35	46005.00	35.45	106.00	16251.45	135126.00
	−0.15	0.00	0.00	1.15	2.00	15976.50	22600.00	31.40	45.00	42399.55	233938.00
	−0.2	0.00	0.00	1.05	2.00	13463.75	22125.00	27.15	64.00	26000.00	146695.00
	−0.25	0.00	0.00	0.85	2.00	15044.93	22383.00	24.90	46.00	42033.00	167010.00

Table 5 Error results of the algorithms for $\sum w_h \tilde{C}_h$

n	a	WSPT		UB		TS		SA	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max
9	−0.05	1.00241	1.00414	1.00028	1.00550	1.00035	1.00551	1.00047	1.00434
	−0.1	1.00256	1.01312	1.00067	1.00551	1.00103	1.00780	1.00223	1.01103
	−0.15	1.00351	1.01049	1.00096	1.00664	1.00180	1.00934	1.00228	1.01103
	−0.2	1.00962	1.05493	1.00278	1.01871	1.00368	1.02719	1.00650	1.04869
	−0.25	1.01387	1.04462	1.00230	1.01691	1.00348	1.01309	1.00793	1.02854
10	−0.05	1.01342	1.10812	1.00008	1.00144	1.00006	1.00094	1.00031	1.00285
	−0.1	1.00902	1.01993	1.00029	1.00420	1.00087	1.00984	1.00139	1.00984
	−0.15	1.00906	1.01512	1.00004	1.00080	1.00060	1.00472	1.00133	1.00730
	−0.2	1.01502	1.03549	1.00208	1.00867	1.00371	1.01356	1.00559	1.02237
	−0.25	1.01952	1.04937	1.00245	1.01663	1.00546	1.02234	1.00700	1.01632
11	−0.05	1.00142	1.00356	1.00007	1.00125	1.00002	1.00031	1.00002	1.00031
	−0.1	1.00353	1.00631	1.00032	1.00185	1.00034	1.00222	1.00039	1.00222
	−0.15	1.00982	1.01765	1.00130	1.00762	1.00296	1.01124	1.00590	1.01563
	−0.2	1.01557	1.03458	1.00144	1.00782	1.00538	1.02485	1.00847	1.02323
	−0.25	1.02421	1.04442	1.00620	1.02204	1.00930	1.03466	1.01282	1.03039
12	−0.05	1.00222	1.00509	1.00004	1.00081	1.00011	1.00156	1.00015	1.00113
	−0.1	1.00442	1.01093	1.00002	1.00020	1.00050	1.00336	1.00085	1.00612
	−0.15	1.01338	1.05293	1.00212	1.00926	1.00334	1.01142	1.00553	1.01875
	−0.2	1.01703	1.04918	1.00516	1.02363	1.00731	102456	1.01181	1.03412
	−0.25	1.02488	1.05086	1.00592	1.02171	1.00943	1.02162	1.01697	1.04238
13	−0.05	1.00711	1.01313	1.00001	1.00009	1.00001	1.00011	1.00001	1.00011
	−0.1	1.00762	1.01633	1.00032	1.00143	1.00043	1.00133	1.00129	1.00730
	−0.15	1.00989	1.01844	1.00155	1.00550	1.00212	1.01088	1.00306	1.00975
	−0.2	1.01493	1.03522	1.00176	1.01701	1.00294	1.01300	1.00509	1.02124
	−0.25	1.02048	1.04052	1.00590	1.01430	1.00676	1.01079	1.00785	1.01291

SPT/WSPT/EDD rule (SPT for $\tilde{C}_{max}$, WSPT for $\sum w_h \tilde{C}_h$, and EDD for L_{max}), and the maximum number of iterations for the TS algorithm is set at $1000\ddot{n}$. The same as Wu et al. (2014), the implementation of the TS algorithm is given below:

Algorithm 3 (TS)

> **Step 1.** Let the tabu list be empty and the iteration number be zero.
> **Step 2.** Set the initial sequence of the TS algorithm, calculate its objective function and set the current sequence as the best solution $\tilde{\pi}^*$.
> **Step 3.** Search the associated neighborhood (the neighborhood is generated by the random exchange of any two jobs) of the current sequence and resolve if there is a sequence $\tilde{\pi}^{**}$ with the smallest objective function in associated neighborhoods and it is not in the tabu list.
> **Step 4.** If $\tilde{\pi}^{**}$ is better than $\tilde{\pi}^*$, then let $\tilde{\pi}^* = \tilde{\pi}^{**}$. Update the tabu list and the iteration number.
> **Step 5.** If there is not a sequence in associated neighborhoods but it is not in the tabu list or the maximum number of iterations is reached, then output the local optimal sequence

Table 6 CPU time (ms) results of the algorithms for L_{max}

n	a	EDD		UB		TS		SA		B&B	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
9	−0.05	0.00	0.00	0.75	1.00	3969.32	4316.00	21.95	26.00	47.30	158.00
	−0.1	0.00	0.00	0.90	2.00	4141.50	4427.00	22.85	28.00	115.60	469.00
	−0.15	0.00	0.00	0.80	2.00	4174.25	4446.00	22.80	32.00	155.05	349.00
	−0.2	0.00	0.00	0.65	3.00	4169.75	4388.00	22.15	27.00	128.00	354.00
	−0.25	0.00	0.00	1.00	3.00	4219.45	4817.00	22.35	52.00	66.80	210.00
10	−0.05	0.00	0.00	0.70	1.00	7349.90	7576.00	28.70	34.00	527.20	1997.00
	−0.1	0.00	0.00	1.15	5.00	4451.55	8687.00	23.12	40.00	844.35	2231.00
	−0.15	0.00	0.00	1.10	5.00	4535.00	9837.00	23.10	43.00	691.00	3035.00
	−0.2	0.00	0.00	0.55	1.00	7310.95	7611.00	29.20	39.00	1120.70	6652.00
	−0.25	0.00	0.00	1.00	3.00	4645.25	10926.00	24.50	52.00	831.65	4020.00
11	−0.05	0.05	1.00	1.00	2.00	8530.31	8646.00	27.10	29.00	3139.70	16172.00
	−0.1	0.00	0.00	0.85	1.00	8540.25	8612.00	27.15	29.00	4835.70	10424.00
	−0.15	0.05	1.00	0.85	1.00	8703.05	9081.00	27.85	33.00	5315.80	22512.00
	−0.2	0.00	0.00	0.95	2.00	8638.25	8778.00	28.45	32.00	7639.95	22197.00
	−0.25	0.00	0.00	1.05	2.00	8656.65	8793.00	27.95	29.00	6972.95	29085.00
12	−0.05	0.05	1.00	1.15	2.00	13332.10	13951.00	34.00	42.00	26780.50	132485.00
	−0.1	0.00	0.00	1.10	2.00	13368.00	14738.00	33.95	42.00	59404.45	242965.00
	−0.15	0.00	0.00	1.15	2.00	13813.20	17420.00	34.40	52.00	73017.30	484337.00
	−0.2	0.00	0.00	1.10	2.00	13430.40	14299.00	34.10	43.00	72420.00	389122.00
	−0.25	0.00	0.00	1.15	2.00	13210.80	14564.00	33.25	42.00	69097.75	207347.00
13	−0.05	0.00	0.00	1.00	2.00	13290.95	18101.00	26.20	37.00	237912.60	1058660.00
	−0.1	0.10	1.00	1.55	4.00	17897.75	47088.00	34.95	77.00	455595.45	3192910.00
	−0.15	0.00	0.00	1.15	3.00	17573.45	51076.00	33.05	50.00	748584.05	2152910.00
	−0.2	0.05	1.00	1.20	4.00	13424.35	21047.00	26.50	41.00	463489.60	2867830.00
	−0.25	0.00	0.00	0.95	3.00	10821.20	20794.00	26.15	51.00	473180.35	1736950.00

Table 7 Error results of the algorithms for $L_{\max}$

n	a	EDD		UB		TS		SA	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max
9	−0.05	1.02321	1.06342	1.00284	1.01018	1.01315	1.05018	1.00688	1.05004
	−0.1	1.01956	1.05782	1.00416	1.02344	1.02041	1.07860	1.00792	1.05004
	−0.15	1.04261	1.08462	1.01609	1.08073	1.03867	1.10119	1.01421	1.06656
	−0.2	1.06123	1.16772	1.01016	1.04374	1.04705	1.14776	1.01813	1.10360
	−0.25	1.07619	1.13902	1.01514	1.10103	1.04621	1.11355	1.02336	1.08616
10	−0.05	1.02211	1.04246	1.00238	1.01757	1.00714	1.02754	1.00224	1.01208
	−0.1	1.03615	1.08016	1.00960	1.03496	1.01874	1.05397	1.00759	1.02480
	−0.15	1.05993	1.10352	1.01076	1.07090	1.02550	1.06118	1.01218	1.03630
	−0.2	1.01332	1.14042	1.01772	1.06045	1.03245	1.09674	1.02115	1.06765
	−0.25	1.09925	1.21912	1.01845	1.06800	1.04186	1.13906	1.03158	1.08292
11	−0.05	1.01422	1.02973	1.00264	1.01248	1.00561	1.02168	1.00408	1.02099
	−0.1	1.03472	1.06632	1.01082	1.03966	1.02664	1.05674	1.01117	1.03987
	−0.15	1.05001	1.10942	1.01107	1.06064	1.03804	1.08916	1.01686	1.09305
	−0.2	1.05601	1.12310	1.01502	1.07147	1.03386	1.08749	1.02752	1.07082
	−0.25	1.07241	1.14763	1.02070	1.09308	1.04972	1.13119	1.03830	1.12817
12	−0.05	1.00991	1.03362	1.00164	1.01105	1.00386	1.02714	1.00358	1.01759
	−0.1	1.03175	1.08622	1.00657	1.02808	1.02002	1.05054	1.01653	1.08168
	−0.15	1.05042	1.10442	1.01189	1.04435	1.03685	1.07654	1.02171	1.06058
	−0.2	1.06623	1.11015	1.01611	1.05519	1.04305	1.07510	1.04508	1.10239
	−0.25	1.06982	1.13422	1.02318	1.08626	1.04898	1.10317	1.03397	1.11060
13	−0.05	1.02419	1.06170	1.00353	1.01358	1.00820	1.02151	1.00517	1.01748
	−0.1	1.04102	1.06263	1.00617	1.02906	1.01789	1.03273	1.01519	1.02557
	−0.15	1.05281	1.09342	1.01207	1.03591	1.01873	1.03451	1.01723	1.03500
	−0.2	1.06772	1.12703	1.01759	1.03320	1.02434	1.04096	1.02257	1.06294
	−0.25	1.07152	1.11782	1.01225	1.03386	1.02714	1.04101	1.02900	1.05328

$\tilde{\pi}$ and objective function value $\tilde{C}_{\max}$ ($\sum w_h \tilde{C}_h$, $L_{\max}$). Otherwise, update the tabu list and go to Step 3.

4.2 Simulated annealing

Simulated annealing is a good choice to solve $1 \left| P_{h,l}^A = \tilde{P}_h \max \left\{ \left(1 + \sum_{h=1}^{l-1} P_{[h]}^A \right)^{a_h}, b_h \right\} \right| \gamma$, where $\gamma \in \{\tilde{C}_{\max}, \sum w_h \tilde{C}_h, L_{\max}\}$. The details of the SA algorithm is summarized as follows:

Algorithm 4 (SA)

> **Step 1**. Initial schedule: Arranging the jobs by the SPT/WSPT/EDD rule (SPT for $\tilde{C}_{\max}$, WSPT for $\sum w_h \tilde{C}_h$, and EDD for $L_{\max}$)
> **Step 2**. The pairwise interchange (PI) neighborhood generation method was used in the algorithms.

Table 8 Computational results of the heuristics for $\tilde{C}_{max}$

n	a	CPU time of SPT		CPU time of UB		CPU time of TS		CPU time of SA		$\dfrac{\tilde{C}_{max}(UB)}{\tilde{C}_{max}(SPT)}$		$\dfrac{\tilde{C}_{max}(TS)}{\tilde{C}_{max}(SPT)}$		$\dfrac{\tilde{C}_{max}(SA)}{\tilde{C}_{max}(SPT)}$	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
100	−0.05	0.10	1.00	874.30	919.00	9704.42	11223.00	1577.00	1607.00	0.98394	0.98650	0.99643	0.99743	0.99575	0.99928
	−0.1	0.30	1.00	3695.00	5336.00	9439.61	15859.00	1617.33	1669.00	0.96675	0.97006	0.99443	0.99569	0.99036	0.99643
	−0.15	0.30	1.00	873.60	960.00	7263.50	11173.00	1605.50	1646.00	0.95458	0.96141	0.99236	0.99397	0.98146	0.98702
	−0.2	0.00	0.00	4441.00	5815.00	9829.70	17011.00	1590.83	1619.00	0.95609	0.96571	0.99152	0.99393	0.98259	0.99531
	−0.25	0.00	0.00	851.20	874.00	7777.00	11399.00	1574.67	1605.00	0.96032	0.96772	0.99165	0.99469	0.98129	0.99168
125	−0.05	0.20	1.00	7718.50	8633.00	17077.10	21136.00	2408.67	2444.00	0.98521	0.98750	0.99744	0.99813	0.99615	0.99863
	−0.1	0.20	1.00	7812.30	9560.00	15476.20	20153.00	2448.00	2481.00	0.96729	0.97255	0.99513	0.99614	0.99300	0.99774
	−0.15	0.00	0.00	2159.40	2428.00	11066.60	20871.00	2423.33	2437.00	0.95990	0.96604	0.99412	0.99517	0.98836	0.99586
	−0.2	0.00	0.00	2176.60	2234.00	16801.20	20624.00	2409.50	2477.00	0.95824	0.96373	0.99335	0.99470	0.98651	0.99999
	−0.25	0.20	1.00	2081.40	2234.00	14412.40	20045.00	2380.33	2421.00	0.96077	0.97022	0.99297	0.99515	0.98338	0.98864
150	−0.05	0.20	1.00	5892.20	7248.00	32337.40	38738.00	2084.50	2195.00	0.98512	0.98806	0.99761	0.99805	0.99723	0.99993
	−0.1	0.10	1.00	5772.30	8550.00	28465.10	37031.00	2114.83	2287.00	0.96800	0.98286	0.99588	0.99639	0.99155	0.99785
	−0.15	0.20	1.00	15252.20	18785.00	26577.80	46595.00	2228.50	2654.00	0.96161	0.99807	0.99494	0.99577	0.98967	0.99871
	−0.2	0.00	0.00	4398.90	4599.00	24078.40	32616.00	2016.33	2076.00	0.96152	0.96862	0.99365	0.99499	0.98945	0.99384
	−0.25	0.10	1.00	4188.11	4248.00	25835.30	29686.00	2022.50	2092.00	0.96678	0.97302	0.99437	0.99539	0.98901	0.99818
175	−0.05	0.00	0.00	9621.10	15678.00	33711.60	51641.00	2734.50	2881.00	0.98360	0.98621	0.99790	0.99830	0.99628	0.99786
	−0.1	0.20	1.00	10341.20	12572.00	19011.90	26524.00	2861.17	3266.00	0.96883	0.97209	0.99633	0.99669	0.99383	0.99746
	−0.15	0.00	0.00	10074.80	16124.00	30873.10	51504.00	2877.67	3185.00	0.96073	0.96634	0.99527	0.99561	0.98941	0.99352
	−0.2	0.20	1.00	9004.20	12526.00	27208.60	50855.00	2742.67	2917.00	0.96361	0.97386	0.99511	0.99670	0.98779	0.99336
	−0.25	0.10	1.00	15536.80	30848.00	25677.90	42254.00	2668.33	2769.00	0.96533	0.97077	0.99450	0.99537	0.98680	0.99650
200	−0.05	0.20	2.00	19156.10	31818.00	41499.00	72009.00	3657.17	4002.00	0.98329	0.98542	0.99805	0.99826	0.99612	0.99853
	−0.1	0.20	1.00	18089.50	27193.00	51656.50	72323.00	3504.33	3600.00	0.96703	0.97098	0.99672	0.99718	0.99281	0.99771
	−0.15	0.00	0.00	16876.70	19711.00	51434.40	70493.00	3453.50	3536.00	0.96111	0.96509	0.99571	0.99610	0.99053	0.99731
	−0.2	0.00	0.00	18177.00	21756.00	34311.80	69736.00	3469.00	3619.00	0.96085	0.96645	0.99518	0.99567	0.98909	0.99696
	−0.25	0.00	0.00	14713.10	25900.00	55768.50	67535.00	3789.67	4347.00	0.96813	0.97537	0.99524	0.99640	0.98848	0.99409

Table 9 Computational results of the heuristics for $\sum w_h \tilde{C}_h$

n	a	CPU time of WSPT		CPU time of UB		CPU time of TS		CPU time of SA		$\dfrac{\sum w_h \tilde{C}_h(UB)}{\sum w_h \tilde{C}_h(WSPT)}$		$\dfrac{\sum w_h \tilde{C}_h(TS)}{\sum w_h \tilde{C}_h(WSPT)}$		$\dfrac{\sum w_h \tilde{C}_h(SA)}{\sum w_h \tilde{C}_h(WSPT)}$	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
100	−0.05	0.00	0.00	902.50	997.00	7234.80	11078.00	1583.67	1593.00	0.99682	0.99853	0.99754	0.99906	0.99801	0.99933
	−0.1	0.00	0.00	3049.00	4257.00	7389.90	12602.00	1597.50	1617.00	0.99293	0.99688	0.99684	0.99864	0.99818	0.99949
	−0.15	0.10	1.00	920.70	1002.00	8011.60	11434.00	1597.33	1637.00	0.99149	0.99371	0.99633	0.99763	0.99884	0.99994
	−0.2	0.20	1.00	3207.20	4155.00	10314.40	17628.00	1569.71	1602.00	0.98905	0.99324	0.99558	0.99697	0.99882	0.99999
	−0.25	0.00	0.00	879.70	891.00	8856.20	11440.00	1565.50	1606.00	0.98703	0.98957	0.99267	0.99527	0.99822	0.99995
125	−0.05	0.20	1.00	7083.00	8376.00	12324.80	19028.00	2371.17	2399.00	0.99874	0.99922	0.99956	0.99978	0.99990	0.99999
	−0.1	0.20	1.00	6380.00	8784.00	14615.40	19865.00	2377.67	2425.00	0.99527	0.99667	0.99861	0.99887	0.99972	0.99999
	−0.15	0.00	0.00	2183.50	2387.00	15207.80	19337.00	2368.00	2419.00	0.99255	0.99501	0.99720	0.99774	0.99959	0.99997
	−0.2	0.00	0.00	2168.60	2249.00	16764.50	19983.00	2350.83	2369.00	0.99032	0.99194	0.99497	0.99618	0.99883	0.99999
	−0.25	0.20	1.00	2098.40	2149.00	14833.50	18886.00	2360.50	2397.00	0.98883	0.99156	0.99357	0.99560	0.99867	0.99998
150	−0.05	0.10	1.00	5576.10	6370.00	28881.00	42606.00	2015.83	2090.00	0.99827	0.99910	0.99922	0.99972	0.99948	0.99999
	−0.1	0.10	1.00	5642.20	7150.00	32458.00	36764.00	2093.67	2317.00	0.99497	0.99689	0.99786	0.99920	0.99868	0.99987
	−0.15	0.10	1.00	14210.90	17749.00	25687.20	43337.00	2134.50	2454.00	0.98811	0.99988	0.99287	0.99830	0.99931	0.99991
	−0.2	0.00	0.00	4481.30	4666.00	24572.10	33352.00	2006.50	2100.00	0.99051	0.99311	0.99500	0.99673	0.99894	0.99999
	−0.25	0.10	1.00	4294.40	4357.00	24824.10	30027.00	2033.00	2108.00	0.98895	0.99228	0.99475	0.99506	0.99936	0.99999
175	−0.05	0.30	1.00	10086.40	16286.00	37110.40	52093.00	2738.17	2848.00	0.99874	0.99914	0.99968	0.99979	0.99993	0.99999
	−0.1	0.00	0.00	10868.70	12170.00	28000.20	54795.00	2835.83	3241.00	0.99568	0.99652	0.99871	0.99924	0.99979	0.99999
	−0.15	0.10	1.00	10629.40	18401.00	32164.50	51719.00	2918.00	3262.00	0.99300	0.99411	0.99766	0.99853	0.99920	0.99998
	−0.2	0.20	1.00	9233.60	12193.00	25870.80	49104.00	2702.50	2888.00	0.99150	0.99318	0.99622	0.99669	0.99916	0.99986
	−0.25	0.10	1.00	15775.40	35568.00	25859.80	42796.00	2706.17	2882.00	0.99056	0.99354	0.99439	0.99554	0.99880	0.99999
200	−0.05	0.10	1.00	19743.80	30292.00	35304.90	60768.00	3698.67	4241.00	0.99827	0.99917	0.99903	0.99962	0.99933	0.99991
	−0.1	0.00	0.00	19109.00	27372.00	60136.80	72461.00	3493.17	3641.00	0.99575	0.99639	0.99861	0.99923	0.99971	0.99999
	−0.15	0.00	0.00	17624.70	19299.00	40346.30	72653.00	3460.83	3618.00	0.99364	0.99534	0.99764	0.99832	0.99965	0.99999
	−0.2	0.10	1.00	19010.70	25830.00	40295.90	70359.00	3494.17	3611.00	0.99184	0.99517	0.99618	0.99728	0.99940	0.99999
	−0.25	0.00	0.00	14951.80	26141.00	57905.10	68050.00	3872.83	4489.00	0.99063	0.99470	0.99619	0.99892	0.99938	0.99998

Table 10 Computational results of the heuristics for L_{max}

n	a	CPU time of EDD		CPU time of UB		CPU time of TS		CPU time of SA		$\frac{L_{max}(UB)}{L_{max}(EDD)}$		$\frac{L_{max}(TS)}{L_{max}(EDD)}$		$\frac{L_{max}(SA)}{L_{max}(EDD)}$	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
100	−0.05	0.00	0.00	955.00	987.00	9693.90	11603.00	1609.33	1708.00	0.97930	0.98190	0.99240	0.99586	0.99501	0.99730
	−0.1	0.00	0.00	3217.10	5740.00	9362.40	29808.00	1707.83	1750.00	0.96045	0.96734	0.98987	0.99507	0.99333	0.99642
	−0.15	0.20	1.00	968.90	1019.00	8951.00	11384.00	1691.83	1709.00	0.94823	0.96052	0.98642	0.99333	0.98926	0.99977
	−0.2	0.10	1.00	3234.30	4251.00	6743.60	15580.00	1691.33	1712.00	0.95337	0.96452	0.98959	0.99324	0.99491	0.99999
	−0.25	0.10	1.00	947.90	962.00	8660.60	11449.00	1686.33	1717.00	0.95509	0.97960	0.98645	0.99336	0.99502	0.99999
125	−0.05	0.20	1.00	6762.00	7779.00	14292.60	19779.00	2324.83	2534.00	0.98443	0.98717	0.99704	0.99798	0.99915	0.99999
	−0.1	0.10	1.00	6820.00	8408.00	14880.00	20464.00	2367.00	2549.00	0.96303	0.96917	0.99406	0.99534	0.99545	0.99907
	−0.15	0.00	0.00	2376.40	2511.00	16270.00	19581.00	2409.33	2562.00	0.95677	0.96864	0.99190	0.99494	0.99507	0.99999
	−0.2	0.00	0.00	2383.10	2458.00	14412.30	20641.00	2473.83	2568.00	0.95537	0.96766	0.99011	0.99410	0.99491	0.99999
	−0.25	0.00	0.00	2319.10	2378.00	14210.10	19895.00	2415.50	2616.00	0.95877	0.97765	0.99049	0.99517	0.99679	0.99999
150	−0.05	0.20	1.00	6361.30	7385.00	28652.30	42815.00	2247.83	2414.00	0.98159	0.98625	0.99422	0.99781	0.99756	0.99999
	−0.1	0.00	0.00	6080.10	7032.00	30516.80	38761.00	2275.50	2565.00	0.96443	0.97197	0.99394	0.99643	0.99741	0.99992
	−0.15	0.00	0.00	14912.60	23192.00	23886.50	43901.00	2323.33	2965.00	0.96458	0.99800	0.99462	0.99634	0.99546	0.99968
	−0.2	0.20	1.00	4900.40	5177.00	23083.50	33965.00	2191.67	2258.00	0.96055	0.97598	0.99206	0.99572	0.99741	0.99999
	−0.25	0.00	0.00	4754.00	4913.00	28835.40	30029.00	2218.50	2310.00	0.96365	0.98007	0.99155	0.99682	0.99589	0.99999
100	−0.05	0.00	0.00	11631.80	18486.00	32587.00	52010.00	2954.50	3106.00	0.98246	0.98529	0.99755	0.99808	0.99893	0.99993
	−0.1	0.10	1.00	12319.00	18819.00	28147.20	59166.00	3118.33	3735.00	0.96582	0.97188	0.99591	0.99697	0.99770	0.99999
	−0.15	0.00	0.00	11620.60	19255.00	24429.80	52190.00	3090.17	3489.00	0.96010	0.96589	0.99409	0.99565	0.99701	0.99999
	−0.2	0.10	1.00	9957.40	14126.00	25581.30	51451.00	2989.67	3300.00	0.96466	0.97566	0.99352	0.99619	0.99603	0.99999
	−0.25	0.10	1.00	17592.70	37827.00	35107.80	43772.00	2907.33	3093.00	0.95983	0.96801	0.99226	0.99481	0.99584	0.99941
200	−0.05	0.00	0.00	21858.41	33421.00	42521.90	81714.00	3912.00	4395.00	0.98168	0.98451	0.99708	0.99959	0.99817	0.99905
	−0.1	0.10	1.00	21366.60	30715.00	55542.40	74591.00	3818.67	3995.00	0.96650	0.96965	0.99591	0.99820	0.99744	0.99999
	−0.15	0.00	0.00	19805.70	21370.00	42831.30	72927.00	3779.67	3953.00	0.95920	0.96659	0.99506	0.99629	0.99717	0.99999
	−0.2	0.00	0.00	21235.10	32514.00	32349.70	70971.00	3780.00	4015.00	0.95953	0.96654	0.99359	0.99559	0.99705	0.99999
	−0.25	0.00	0.00	16717.50	30474.00	54929.20	62984.00	3956.00	4275.00	0.97003	0.97509	0.99454	0.99724	0.99804	0.99999

Step 3. Acceptance probability: When a new schedule is generated, it is accepted if its objective value is smaller than that of the original schedule; otherwise, it is accepted with some probability that decreases as the process evolves. The probability of acceptance is generated from an exponential distribution,

$$P(accept) = exp(-\alpha \times \triangle WC/L),$$

where α is a control parameter and $\triangle WC/L$ is the change in the objective value (i.e., $\tilde{C}_{\max}, \sum w_h \tilde{C}_h$ and $L_{\max}$). In addition, the method was adopted to change α in the kth iteration as follows:

$$\alpha = \frac{k}{\delta}$$

where δ is an experimental constant. After preliminary trials, $\delta = 1$ was used in our experiments.

If $\tilde{C}_{\max}(\sum \mathcal{W}\tilde{C}_h, L_{\max})$) increases as a result of a random pairwise interchange, the new sequence is accepted when $P(accept) > \theta$, where θ is randomly sampled from the uniform distribution $U(0, 1)$.

Step 4. Stopping condition: The schedule is stable after $1000\ddot{n}$ iterations (see Lai et al. 2014).

5 Computational experiments

In this section, computational experiments are conducted to evaluate the accuracy and efficiency of the SPT, WSPT, EDD, UB, TS, SA and B&B algorithms. The C++ language was used to implement both the heuristic and B&B algorithms. Detailed programming and testing configurations are as follows.

- C++ version: Visual studio 2019, installed on Windows 10 system, occupies 800 MB–210 GB of free space on the hard disk, depending on the features installed. Typical installations require 20–50 GB of free space.
- Testing computer: One desktop workstation with one AMD R5-5500U CPU (2.1–4.0 GHz, 6 cores, 12 threads). The computer has 16G of running memory and 512G of solid-state hard disks.

Similar to Jiang et al. (2022), the following test parameters are randomly generated:

(1) $\tilde{P}_h$ is uniformly integers distributed over [1, 10];
(2) $a_h = a$, where $a = -0.05, -0.1, -0.15, -0.2, -0.25$;
(3) b_h is uniformly distributed over (0, 1);
(4) w_h is uniformly integers distributed over [1, 10];
(5) d_h is uniformly integers distributed over [1, 20];

- Small sized problem instances: $\ddot{n} = 9, 10, 11, 12, 13$, also use B&B to solve to global optimum;
- Large sized problem instances: $\ddot{n} = 100, 125, 150, 175, 200$, B&B was disabled.

For each parameters' combination ($\ddot{n}$ and a), 20 randomly generated instances were evaluated.

5.1 Small-sized instances

For $\tilde{C}_{\max}$, the error of the solution produced by heuristics is calculated as

$$\frac{\tilde{C}_{\max}(H)}{\tilde{C}_{\max}(H^*)},$$

where H is a sequence obtained by the SPT, UB, TS, SA and the optimal sequence H^* is obtained by the B&B algorithm. For $\sum \mathcal{W}\tilde{C}_h$, the error of the solution produced by heuristics is calculated as

$$\frac{\sum w_h \tilde{C}_h(H)}{\sum w_h \tilde{C}_h(H^*)},$$

where H is a sequence obtained by the WSPT, UB, TS, SA and the optimal sequence H^* is obtained by the B&B algorithm. For $L_{\max}$, the error of the solution produced by heuristics is calculated as

$$\frac{L_{\max}(H)}{L_{\max}(H^*)},$$

where H is a sequence obtained by the EDD, UB, TS, SA and the optimal schedule H^* is obtained by the B&B algorithm. It is also defined that "CPU time" as the running time of the SPT (WSPT, EDD), UB, TS, SA, B&B algorithm and time unit is milli-seconds (1/1000 seconds). The corresponding results are summarized in Tables 2, 3, 4, 5, 6 and 7. From Tables 2, 4 and 6, we can see that the CPU time increase rapidly, i.e., grows exponentially with the number of jobs ($\ddot{n}$). From Table 3, for the $\tilde{C}_{\max}$ minimization, the results of SA appear to be more accurate than the SPT, UB and TS. For the $\sum w_h \tilde{C}_h$ and $L_{\max}$ minimizations, Tables 5 and 7 show that UB appear to be more accurate than the SPT, SA and TS.

5.2 Large-sized instances

The performance of the heuristics SPT (WSPT, EDD), UB, TS and SA was verified by the ratios $\frac{\tilde{C}_{\max}(H)}{\tilde{C}_{\max}(SPT)}$, $\frac{\sum w_h \tilde{C}_h(H)}{\sum w_h \tilde{C}_h(WSPT)}$ and $\frac{L_{\max}(H)}{L_{\max}(EDD)}$, where H is a sequence obtained by the UB, TS and SA. The corresponding results are summarized in Tables 8, 9 and 10. Tables 8, 9 and 10 shows that the error of UB outperforms TS and SA. Tables 8, 9 and 10 also shows that the CPU time (ms) of the TS is much longer than the UB and AS. The running times for the heuristics SPT, WSPT and EDD are almost the same for $\ddot{n} = 100, 125, 150, 175, 200$.

6 Conclusions

In this article, we addressed a single-machine problem with truncated learning effects: how to minimise the makespan, total-weighted completion time and maximum lateness. We proposed a branch-and-bound algorithm and several heuristics. Numerical studies showed that the UB performs best for large sized instances. Future research can focus on flow shop or job shop scheduling problems with truncated learning effects or deterioration effects (see Huang and Wang 2015; Huang 2019). Given that the single-machine problems, effective exact algorithms will be considered.

Data Availability The data used to support the findings of this study are available from the corresponding author upon request.

Declarations

Conflict of interest There are none.

References

Azzouz A, Ennigrou M, Said LB (2018) Scheduling problems under learning effects: classification and cartography. Int J Prod Res 56:1642–1661

Biskup D (1999) Single-machine scheduling with learning considerations. Eur J Oper Res 115:173–178

Cheng TCE, Kuo W-H, Yang D-L (2014) Scheduling with a position-weighted learning effect. Optim Lett 8:293–306

Framinan JM, Leisten R (2003) An efficient constructive heuristic for flowtime minimization in permutation flow shops. Omega 31(4):311–317

Geng X-N, Wang J-B, Bai D (2019) Common due date assignment scheduling for a no-wait flowshop with convex resource allocation and learning effect. Eng Optim 51(8):1301–1323

Hardy G, Littlewood J, Polya G (1967) Inequalities. Cambridge University Press, Cambridge

Hsu C-J, Kuo W-H, Yang D-L (2011) Unrelated parallel machine scheduling with past-sequence-dependent setup time and learning effects. Appl Math Model 35:1492–1496

Huang X (2019) Bicriterion scheduling with group technology and deterioration effect. J Appl Math Comput 60(1–2):455–464

Huang X, Wang J-J (2015) Machine scheduling problems with a position-dependent deterioration. Appl Math Model 39(10–11):2897–2908

Jiang Z, Chen F, Zhang X (2022) Single-machine scheduling problems with general truncated sum-of-actual-processing-time-based learning effect. J Comb Optim 43:116–139

Kuo W-H, Yang D-L (2007) Single-machine scheduling with past-sequence-dependent setup times and learning effects. Inf Process Lett 102:22–26

Lai K, Hsu P-H, Ting P-H, Wu C-C (2014) A truncated sum of processing-times-based learning model for a two-machine flowshop scheduling problem. Human Factors Ergon Manuf Service Ind 24:152–160

Lawler EL (1973) Optimal sequencing of a single machine subject to precedence constraints. Manag Sci 19:544–546

Lee W-C (2011) A note on single-machine scheduling with general learning effect and past-sequence-dependent setup time. Comput Math Appl 62:2095–2100

Liang X-X, Zhang B, Wang J-B, Yin N, Huang X (2019) Study on flow shop scheduling with sum-of-logarithm-processing-times-based learning effects. J Appl Math Comput 61:373–388

Li L, Yang S-W, Wu Y-B (2013) Single machine scheduling jobs with a truncated sum-of-processing-times-based learning effect. Int J Adv Manuf Technol 67:261–267

Liu W-W, Jiang C (2020a) Flow shop resource allocation scheduling with due date assignment, learning effect and position-dependent weights. Asia-Pac J Oper Res 37(3):2050014

Liu W-W, Jiang C (2020b) Due-date assignment scheduling involving job-dependent learning effects and convex resource allocation. Eng Optim 52(1):74–89

Lu Y-Y, Teng F, Feng Z-X (2015) Scheduling jobs with truncated exponential sum-of-logarithm-processing-times based and position-based learning effects. Asia-Pac J Oper Res 32(3):1550026

Lu Y-Y, Wang T-T, Wang R-Q, Li Y (2021) A note on due-date assignment scheduling with job-dependent learning effects and convex resource allocation. Eng Optim 53(7):1273–1281

Lv D-Y, Wang J-B (2021) Study on resource-dependent no-wait flow shop scheduling with different due-window assignment and learning effects. Asia-Pac J Oper Res 38(6):2150008

Niu Y-P, Wan L, Wang J-B (2015) A note on scheduling jobs with extended sum-of-processing-times-based and position-based learning effect. Asia-Pac J Oper Res 32(2):1550001

Soroush HM (2014) Scheduling in bicriteria single machine systems with past-sequence-dependent setup times and learning effects. J Oper Res Soc 65:1017–1036

Sun X, Geng X-N, Liu F (2021) Flow shop scheduling with general position weighted learning effects to minimise total weighted completion time. J Oper Res Soc 72(12):2674–2689

Wang J-B (2010) Single-machine scheduling with a sum-of-actual-processing-time-based learning effect. J Oper Res Soc 61:172–177

Wang J-B, Gao M, Wang J-J, Liu L, He H (2020) Scheduling with a position-weighted learning effect and job release dates. Eng Optim 52(9):1475–1493

Wang J-B, Liu F, Wang J-J (2019) Research on m-machine flow shop scheduling with truncated learning effects. Int Trans Oper Res 26(3):1135–1151

Wang J-B, Lv D-Y, Xu J, Ji P, Li F (2021) Bicriterion scheduling with truncated learning effects and convex controllable processing times. Int Trans Oper Res 28(3):1573–1593

Wang J-B, Sun L-H, Sun L-Y (2010) Single machine scheduling with exponential sum-of-logarithm-processing-times based learning effect. Appl Math Model 34:2813–2819

Wang J-B, Wang X-Y, Sun L-H, Sun L-Y (2013) Scheduling jobs with truncated exponential learning functions. Optim Lett 7:1857–1873

Wang J-B, Zhang B, He H (2022) A unified analysis for scheduling problems with variable processing times. J Ind Manag Optim 18(2):1063–1077

Wu C-C, Lee W-C, Liou M-J (2013) Single-machine scheduling with two competing agents and learning consideration. Inf Sci 251:136–149

Wu C-C, Wu W-H, Wu W-H, Hsu P-H, Yin Y, Xu J (2014) A single-machine scheduling with a truncated linear deterioration and ready times. Inf Sci 256:109–125

Wu C-C, Yin Y, Cheng S-R (2011) Some single-machine scheduling problems with a truncation learning effect. Comput Ind Eng 60(4):790–795

Wu C-C, Yin Y, Cheng SR (2013) Single-machine and two-machine flowshop scheduling problems with truncated position-based learning functions. J Oper Res Soc 64(1):147–156

Wu C-C, Yin Y, Wu W-H, Cheng S-R (2012) Some polynomial solvable single-machine scheduling problems with a truncation sum-of-processing-times based learning effect. Eur J Ind Eng 6:441–453

Zhao S (2021) Resource allocation flowshop scheduling with learning effect and slack due window assignment. J Ind Manag Optim 17(5):2817–2835

Zhao S (2022) Scheduling jobs with general truncated learning effects including proportional setup times. Comput Appl Math 41:146