

# NDP约束下的最小化最大加权完工时间单机一致性在线排序问题研究

李文杰, 杜智慧

洛阳师范学院, 数学科学学院, 河南 洛阳

收稿日期: 2022年3月11日; 录用日期: 2022年4月14日; 发布日期: 2022年4月21日

## 摘要

本文在工件不能被强制推迟加工约束(即NDP约束)下研究最小化最大加权完工时间单台机器在线排序问题。每个工件  $J_j$  都具有一个释放时间  $r_j \geq 0$ , 一个加工时间  $p_j \geq 0$  和一个权重  $w_j \geq 0$ 。每两个工件  $J_i$  和  $J_j$  的释放时间与加工时间均具有一致性, 即若  $r_i \geq r_j$ , 则有  $p_i \geq p_j$ 。我们首先利用对手法构造出该排序问题的下界是1.5, 其次设计出一个在线算法SLF并采用最小反例法证明其争比是1.732。

## 关键词

在线排序, 在线算法, NDP约束, 一致性, 加权完工时间

# Online NDP-Constraint Scheduling of Jobs with Agreeable to Minimize the Maximum Weighted Completion Time

Wenjie Li, Zhihui Du

School of Mathematical Sciences, Luoyang Normal University, Luoyang Henan

Received: Mar. 11<sup>th</sup>, 2022; accepted: Apr. 14<sup>th</sup>, 2022; published: Apr. 21<sup>st</sup>, 2022

## Abstract

This paper studies the single-machine online schedule under the NDP-constraint to minimize the maximum weighted completion time. Each job  $J_j$  has a release time  $r_j \geq 0$ , a processing time  $p_j \geq 0$  and a weight  $w_j \geq 0$ . Any two jobs  $J_i, J_j$  have agreeable release times and processing

times (if  $r_i \geq r_j$ , then  $p_i \geq p_j$ ). We first present a lower bound of 1.5 by the adversary method, and then design an online algorithm SLF with the competitive ratio of 1.732.

## Keywords

Online Scheduling, Online Algorithm, NDP-Constraint, Agreeable, Weighted Completion Time

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

在线排序是现代排序领域中发展最为迅速的研究方向之一, 并被广泛地应用于生产加工、货运物流、航班调度、医疗服务等众多领域。在线排序中决策者在制定排序策略时只掌握当前已经被释放工件的信息(例如, 释放时间、加工时间、权重等), 人们称这种在线为时间在线。列表在线、不可测在线是另外两种常见的在线排序模型[1] [2] [3] [4]。

一般用竞争比来衡量一个在线算法关于对应在线排序问题的性能。我们称算法  $H$  的竞争比是  $\rho_H$ , 如果满足  $\rho_H = \sup_{I>0} \{H(I)/OPT(I) : OPT(I) > 0\}$ , 其中  $I$  是排序问题的工件实例,  $H(I)$  表示算法  $H$  关于  $I$  产生的目标值,  $OPT(I)$  表示离线最优算法产生的目标值。当一个排序问题在线算法的竞争比与排序问题的下界相吻合时, 则称该算法是最优在线算法。

本文在 NDP 约束下研究最小化最大加权完工时间单机在线排序问题。NDP 约束是指在任何机器出现空闲的时刻, 如果有还没被安排加工的工件存在, 则空闲机器必须立刻选择工件加工。换言之, 工件不能被强制推迟加工。NDP 约束是 Li 和 Yuan 在文献[5]中根据实际背景以及排序模型拓展提出的一种工件加工约束条件。该排序模型在实际生活中确有体现。例如市政服务大厅的办事窗口(抽象为机器), 当窗口开放并在给客户(抽象为工件)办理业务时, 后面到达的客户就会自觉地取号排队等待服务, 这些已到达还未办理业务的客户最不愿意看到窗口突然暂停服务的情况出现。因为每个客户都希望自己的满意度最大化都不愿意再去排队等待。这里将本文在 NDP 约束下所要研究的具体排序问题描述如下:

设  $I$  是一个工件集合(简称实例)包含  $n$  个工件  $J_1, J_2, \dots, J_n$ 。对每一个实例  $I$  中的工件  $J_i$ , 它具有一个释放时间(又称到达时间)  $r_i \geq 0$ , 一个加工时间(又称长度)  $p_i \geq 0$ , 一个权重  $w_i \geq 0$ 。要求实例  $I$  中任意两个工件  $J_i, J_k$  的释放时间和加工时间之间满足: 若  $r_i \geq r_k$ , 则有  $p_i \geq p_k$ 。通常称之为  $I$  中工件的释放时间和加工时间具有一致性。排序目标: 确定一个工件不能被强制延迟加工(即满足 NDP 约束)的可行排序最小化所有工件的最大加权完工时间( $WC_{\max}$ ), 其中  $WC_{\max} = \max \{w_i C_i : i = 1, 2, \dots, n\}$ ,  $C_i$  为工件  $J_i$  的完工时间。方便起见, 现将 NDP 约束下的工件的释放时间和加工时间具有一致性的最小化最大加权完工时间单机在线排序问题简记为(P)。用 Graham 等人在文献[6]中定义的三参数表示法, 本文所研究的排序问题(P)可表示为:

$$1 | \text{online}, r_i, \text{agreeable}(r_i, p_i), \text{NDP} | WC_{\max},$$

其中 “agreeable( $r_i, p_i$ )” 表示关于问题(P)的任一实例中工件的释放时间和加工时间具有一致性。

关于无 NDP 约束的最小化最大加权完工时间单机在线排序问题, Chai 等人在文献[7]中给出了一个 2-竞争的最优在线算法。在 NDP 约束下, Li 和 Yuan 在文献[5]中研究了该问题。他们首先给出问题的下

界为 2, 其次设计出了一个竞争比为 2.618 的在线算法。关于最小化最大加权完工时间在线排序问题的其它相关文献可参见[8] [9] [10] [11] [12]。

本文将在第二节构造出问题(P)的竞争比下界为 1.5, 在第三节中设计并证明该问题的一个 1.732-竞争的在线算法 SLF。

## 2. 问题的下界

本节将通过构造特殊实例证明问题(P)的竞争比下界是 1.5。用  $WC_{on}(I)$  和  $WC_{opt}(I)$  分别表示任一在线算法和离线最优算法关于实例  $I$  生成的排序所达到的目标函数值。

**定理 2.1** 对于问题  $1|online, r_i, agreeable(r_i, p_i), NDP|WC_{max}$ , 不存在竞争比小于 1.5 的在线算法。

证明 设  $H$  是问题(P)的任意一个在线算法。令  $\varepsilon$  是一个任意小的正实数,  $K$  是一个充分大的正实数。下面用对手法构造的特殊实例  $I$  至多含有 3 个工件。

在 0 时刻, 工件  $J_1$  和  $J_2$  被释放并且它们的加工时间和权重分别为  $p_1 = 1, w_1 = 1, p_2 = 1 + \varepsilon, w_2 = 0$ 。由 NDP 约束可知, 在线算法  $H$  必须在 0 时刻选择一个工件开工。如果算法  $H$  在 0 时刻选择加工  $J_2$ , 则之后不再有新工件被释放。此情形  $I = \{J_1, J_2\}$ 。易知在线算法  $H$  和离线最优算法 OPT 关于  $I$  生成的排序分别为  $(J_2, J_1)$  和  $(J_1, J_2)$ 。于是可得  $C_{on}(I) = w_1(p_2 + p_1) = 2 + \varepsilon, C_{opt}(I) = w_1 p_1 = 1$ 。进而可得

$$\frac{WC_{on}(I)}{WC_{opt}(I)} = 2 + \varepsilon > 1.5。$$

如果算法  $H$  在 0 时刻安排  $J_1$  开工, 则第三个工件  $J_3$  将在  $1 + \frac{1}{2}\varepsilon$  时刻被释放, 其加工时间和权重分别为  $p_3 = 1 + \varepsilon, w_3 = K$ 。此情形  $I = \{J_1, J_2, J_3\}$ 。由 NDP 约束和  $r_3 = 1 + 0.5\varepsilon > 1$  可知, 算法  $H$  只能安排  $J_3$  在继  $J_2$  完工之后开工。于是可得  $C_{on}(I) = w_3(p_1 + p_2 + p_3) = K(3 + 2\varepsilon)$ 。注意到, 最优算法 OPT 则可以先安排  $J_2$  在 0 时刻开工, 然后安排  $J_3$  在  $J_2$  的完工时刻开工(可以这样安排的原因是  $r_3 = 1 + 0.5\varepsilon < 1 + \varepsilon$ ), 最后安排  $J_1$  在  $2 + 2\varepsilon$  时刻开工。可得离线最优序下的目标值为  $C_{opt}(I) = w_3(p_2 + p_3) = K(2 + 2\varepsilon)$ 。因此可以推得

$$\frac{WC_{on}(I)}{WC_{opt}(I)} = \frac{3 + 2\varepsilon}{2 + 2\varepsilon} \rightarrow 1.5, \text{ 当 } K \rightarrow +\infty \text{ 时。}$$

综上, 定理 2.1 得证。 □

## 3. 在线算法及其竞争比分析

本节将设计关于问题(P)的在线算法 SLF (加工策略 “Shortest-length or Largest-weight Job First” 的简称)并用最小反例法分析其竞争比至多为 1.732。对于问题(P)的任意一个实例  $I$ , 令  $\sigma$  和  $\pi$  分别表示在线算法 SLF 和离线最优算法 OPT 关于实例  $I$  生成的排序。令  $t$  表示当前时刻, 下面的一系列记号将会在后面的讨论中被用到。

- $S_i(\sigma), S_i(\pi)$  分别表示  $I$  中工件  $J_i$  在排序  $\sigma$  和  $\pi$  中的开工时间。
- $C_i(\sigma), C_i(\pi)$  分别表示  $I$  中工件  $J_i$  在排序  $\sigma$  和  $\pi$  中的完工时间。
- $WC_{max}(\sigma), WC_{max}(\pi)$  分别表示排序  $\sigma$  和  $\pi$  关于实例  $I$  产生的目标值。
- $U(t)$  表示在  $t$  时刻已经被释放但还没有被安排加工的工件构成的集合。
- $J_{min}(t)$  表示  $U(t)$  中加工时间最小的工件。如果有多个这样的工件, 则定义为最早被释放的工件。
- $J_w(t)$  表示  $U(t)$  中权重最大的工件。如果有多个这样的工件, 则定义为最早被释放的工件。
- $p_i(t)$  表示  $U(t)$  中任一工件  $J_i$  的长度。特别的, 用  $p_{min}(t), p_w(t)$  分别表示  $J_{min}(t)$  和  $J_w(t)$  的长度。

设  $\lambda = \sqrt{3} - 1$ , 在线算法 SLF 的具体执行步骤可描述如下:

### 在线算法 SLF

**第 0 步:** 置  $t$  等于  $I$  中工件的最早释放时间。不妨假设  $I$  中工件的最小释放时间为 0, 即置  $t := 0$ 。

**第 1 步:** 若在  $t$  时刻  $U(t) \neq \emptyset$  且机器空闲, 则执行第 2 步或第 3 步; 否则重置  $t$  为下一个满足机器空闲且  $U(t)$  非空的最早时刻。

**第 2 步:** 若  $U(t)$  只含有一个工件  $J_i$ , 则立刻加工  $J_i$  并重置  $t := t + p_i(t)$ 。转到第 1 步; 否则,  $U(t)$  至少包含两个工件, 则转到第 3 步。

**第 3 步:** 若此时  $U(t)$  中权重最大的工件  $J_w(t)$  满足  $t \geq \lambda \cdot p_w(t)$ , 则立刻加工  $J_w(t)$  并重置  $t := t + p_w(t)$ 。转到第 1 步; 否则, 转到第 4 步。

**第 4 步:** 此时  $U(t)$  中权重最大的工件  $J_w(t)$  满足  $t < \lambda \cdot p_w(t)$ , 则分以下两种情形,

**(4.1):** 若此时  $U(t)$  中长度最小的工件  $J_{\min}(t)$  满足  $t + p_{\min}(t) \leq \lambda \cdot p_w(t)$ , 则立刻加工  $J_{\min}(t)$  并重置  $t := t + p_{\min}(t)$ 。转到第 1 步; 否则, 转到第 4.2 步。

**(4.2):** 可知  $U(t)$  中任意一个工件  $J_i$  的长度  $p_i(t)$  满足  $t + p_i(t) > \lambda \cdot p_w(t)$ , 则立刻加工  $J_w(t)$  并重置  $t := t + p_w(t)$ 。转到第 1 步。

接下来, 证明算法 SLF 关于问题(P)的竞争比  $\rho = \lambda + 1 = \sqrt{3} \approx 1.732$ 。反证法, 假设存在一个包含工件数量最小的实例  $I$  满足  $WC_{\max}(\sigma) > \rho \cdot WC_{\max}(\pi)$ , 其中  $\sigma$  和  $\pi$  分别是在线算法 SLF 和离线最优算法 OPT 关于实例  $I$  生成的排序。通常称此实例为关于问题(P)的一个最小反例。由算法 SLF 的第 0 步, 可以假设  $I$  中第一个工件在 0 时刻被释放并继续假设  $I$  中  $n$  个工件  $J_1, J_2, \dots, J_n$  在  $\sigma$  序下的开工时间分别满足  $S_1(\sigma) < S_2(\sigma) < \dots < S_n(\sigma)$ 。可得  $S_1(\sigma) = 0$ ,  $C_n(\sigma) = C_{\max}(\sigma)$ 。设  $J_e, 1 \leq e \leq n$  是  $I$  中第一个满足其加权完工时间等于目标值的工件, 即满足  $w_e C_e(\sigma) = WC_{\max}(\sigma)$ 。下面给出两个重要引理。

**引理 3.1** 在排序  $\sigma$  和  $\pi$  中所有工件都是被连续加工的, 即在 0 时刻与  $C_n(\sigma)$  时刻之间机器没有空闲时间段。

证明 首先考虑排序  $\sigma$ 。如果在  $J_e$  之前有工件  $J_a, 2 \leq a \leq e$  满足  $C_{a-1}(\sigma) < S_a(\sigma)$ , 即  $[C_{a-1}(\sigma), S_a(\sigma)]$  是空闲时间段, 则令  $I' = \{J_a, J_{a+1}, \dots, J_n\}$ 。如果在  $J_e$  之后有工件  $J_b, e \leq b \leq n-1$  满足  $C_b(\sigma) < S_{b+1}(\sigma)$ , 即  $[C_b(\sigma), S_{b+1}(\sigma)]$  是空闲时间段, 则令  $I'' = \{J_1, J_1, \dots, J_b\}$ 。由算法 SLF 的执行策略可知, 算法 SLF 关于实例  $I, I'$  及  $I''$  产生的目标值相同。再由 NDP 约束和实例  $I$  是最小反例, 可以推出  $I'$  或  $I''$  是更小的反例, 与实例  $I$  是最小反例相矛盾。于是可得在排序  $\sigma$  中所有工件被连续加工, 即机器无空闲时间段。由 NDP 约束及  $I$  中工件在排序  $\sigma$  中被连续加工, 可知离线最优序  $\pi$  中所有工件也是被连续加工的。□

**引理 3.2**  $w_i < w_e, i = e+1, e+2, \dots, n$ 。

证明 由  $WC_{\max}(\sigma) = w_e C_e(\sigma)$  以及  $C_i(\sigma) > C_e(\sigma), e+1 \leq i \leq n$  可直接推出  $w_i < w_e, e+1 \leq i \leq n$ 。□

下面将分两种情形证明最小反例  $I$  不存在, 进而得到算法 SLF 关于问题(P)的竞争比就是 1.732。

**情形 1**  $w_i \geq w_e, i = 1, 2, \dots, e$ 。则可得

$$WC_{\max}(\sigma) = w_e C_e(\sigma) = w_e \sum_{i=1}^e p_i \leq WC_{\max}(\pi) < \rho \cdot WC_{\max}(\pi)。$$

显然与  $WC_{\max}(\sigma) > \rho \cdot WC_{\max}(\pi)$  相矛盾。

**情形 2**  $\{J_1, J_2, \dots, J_{e-1}\}$  中存在权重小于  $w_e$  的工件。则令  $k = \max\{i : w_i < w_e, i = 1, 2, \dots, e-1\}$ 。于是可得  $w_i \geq w_e, i = k+1, k+2, \dots, e$ 。令  $r_{\min} = \min\{r_i : k+1 \leq i \leq e\}$ 。如果  $r_{\min} \leq S_k(\sigma)$ , 表明  $\{J_{k+1}, J_{k+2}, \dots, J_e\}$  中至少有一个工件在  $S_k(\sigma)$  时刻或  $S_k(\sigma)$  时刻之前到达。由于  $J_k$  不是  $S_k(\sigma)$  时刻已到达的工件中权重最大的工件而算法 SLF 选择在  $S_k(\sigma)$  时刻加工  $J_k$ , 表明  $J_k$  一定是加工时间最小的工件并且满足  $S_k(\sigma) + p_k \leq \lambda \cdot p_w(S_k(\sigma))$ 。根据引理 3.2 以及  $w_i \geq w_e, i = k+1, k+2, \dots, e$ , 可知  $S_k(\sigma)$  时刻已到达的工

件中权重最大的工件一定属于  $\{J_{k+1}, J_{k+2}, \dots, J_e\}$ , 即  $p_w(S_k(\sigma)) \in \{p_{k+1}, p_{k+2}, \dots, p_e\}$ 。因此可得

$$\begin{aligned} WC_{\max}(\sigma) &= w_e \left( S_k(\sigma) + p_k + \sum_{i=k+1}^e p_i \right) \leq w_e \left( \lambda \cdot p_k(S_k(\sigma)) + \sum_{i=k+1}^e p_i \right) \\ &\leq (1 + \lambda) w_e \sum_{i=k+1}^e p_i \leq \rho \cdot WC_{\max}(\pi). \end{aligned}$$

这显然与  $I$  是最小反例相矛盾。

如果  $r_{\min} > S_k(\sigma)$ , 即  $r_i \geq S_k(\sigma) \geq r_k, i = k+1, k+2, \dots, e$ , 则有  $WC_{\max}(\pi) \geq w_e \left( S_k(\sigma) + \sum_{i=k+1}^e p_i \right)$ 。由于  $I$  中工件的释放时间与加工时间具有一致性, 可知  $p_i \geq p_k, i = k+1, k+2, \dots, e$ 。若  $S_k(\sigma) \geq 0.5\lambda \cdot p_k$ , 则可得

$$\begin{aligned} \frac{WC_{\max}(\sigma)}{WC_{\max}(\pi)} &= \frac{w_e \left( S_k(\sigma) + p_k + \sum_{i=k+1}^e p_i \right)}{w_e \left( S_k(\sigma) + \sum_{i=k+1}^e p_i \right)} \\ &\leq \frac{S_k(\sigma) + (e-k+1)p_k}{S_k(\sigma) + (e-k)p_k} \\ &\leq \frac{0.5\lambda \cdot p_k + (e-k+1)p_k}{0.5\lambda \cdot p_k + (e-k)p_k} \\ &\leq \frac{\lambda+4}{\lambda+2} \\ &= \rho. \end{aligned}$$

结论与  $WC_{\max}(\sigma) > \rho \cdot WC_{\max}(\pi)$  相矛盾。

不妨假设  $S_k(\sigma) < 0.5\lambda \cdot p_k$ 。如果在  $S_k(\sigma)$  时刻只有一个工件  $J_k$ , 由 NDP 约束和引理 3.1, 可知在离线最优序  $\pi$  中  $J_k$  一定排在  $\{J_{k+1}, J_{k+2}, \dots, J_e\}$  中工件之前加工。于是可得  $\min\{S_i(\pi) : k+1 \leq i \leq e\} \geq C_k(\pi) \geq p_k$ 。进而可得

$$\begin{aligned} \frac{WC_{\max}(\sigma)}{WC_{\max}(\pi)} &= \frac{w_e \left( S_k(\sigma) + p_k + \sum_{i=k+1}^e p_i \right)}{w_e \left( \min\{S_i(\pi) : k+1 \leq i \leq e\} + \sum_{i=k+1}^e p_i \right)} \\ &\leq \frac{S_k(\sigma) + (e-k+1)p_k}{\min\{S_i(\pi) : k+1 \leq i \leq e\} + (e-k)p_k} \\ &\leq \frac{0.5\lambda \cdot p_k + (e-k+1)p_k}{p_k + (e-k)p_k} \\ &\leq \frac{\lambda+4}{4} \\ &< \rho. \end{aligned}$$

如果在  $S_k(\sigma)$  时刻至少有两个工件(包括  $J_k$ )已经到达, 即集合  $U(S_k(\sigma))$  中工件数大于等于 2。注意到  $r_{\min} > S_k(\sigma)$ , 再根据引理 3.1, 可知在离线最优序  $\pi$  下  $U(S_k(\sigma))$  至少有一个工件一定被排在  $\{J_{k+1}, J_{k+2}, \dots, J_e\}$  中工件之前加工。于是可得  $\min\{S_i(\pi) : k+1 \leq i \leq e\} \geq \min\{p_x : J_x \in U(S_k(\sigma))\}$ 。由算法 SLF 的执行策略可知, 在  $S_k(\sigma)$  时刻  $J_k$  要么是一个长度最小的工件, 要么是一个权重最大的工件。若  $J_k$  是一个长度最小的工件, 则有  $\min\{S_i(\pi) : k+1 \leq i \leq e\} \geq \min\{p_x : J_x \in U(S_k(\sigma))\} = p_k$ 。用上面式子的分析形式可直接推出  $WC_{\max}(\sigma) < \rho \cdot WC_{\max}(\pi)$ 。

若  $J_k$  是一个权重最大的工件, 则根据算法 SLF 的第 4.2 步可知集合  $U(S_k(\sigma))$  中的工件满足,  $\forall J_x \in U(S_k(\sigma)), S_k(\sigma) + p_x > \lambda \cdot p_k$ 。注意到  $S_k(\sigma) < 0.5\lambda \cdot p_k$ 。于是可得

$$\forall J_x \in U(S_k(\sigma)), p_x > 0.5\lambda \cdot p_k.$$

再根据  $r_{\min} > S_k(\sigma)$  以及根据引理 3.1, 可推得

$$\min\{S_i(\pi) : k+1 \leq i \leq e\} \geq \min\{p_x : J_x \in U(S_k(\sigma))\} > 0.5\lambda \cdot p_k.$$

因此可得

$$\begin{aligned} \frac{WC_{\max}(\sigma)}{WC_{\max}(\pi)} &= \frac{w_e(S_k(\sigma) + p_k + \sum_{i=k+1}^e p_i)}{w_e(\min\{S_i(\pi) : k+1 \leq i \leq e\} + \sum_{i=k+1}^e p_i)} \\ &\leq \frac{S_k(\sigma) + (e-k+1)p_k}{\min\{S_i(\pi) : k+1 \leq i \leq e\} + (e-k)p_k} \\ &\leq \frac{0.5\lambda \cdot p_k + (e-k+1)p_k}{0.5\lambda \cdot p_k + (e-k)p_k} \\ &\leq \frac{\lambda + 4}{\lambda + 2} \\ &= \rho. \end{aligned}$$

综上可知, 满足  $WC_{\max}(\sigma) > \rho \cdot WC_{\max}(\pi)$  的最小反例  $I$  不存在。表明对每个关于问题(P)的实例都有  $WC_{\max}(\sigma) \leq \rho \cdot WC_{\max}(\pi)$  恒成立。于是得到本文的主要结论如下:

**定理 3.3** 对于问题 1 | online,  $r_i$ , agreeable( $r_i, p_i$ ), NDP |  $WC_{\max}$ , 在线算法 SLF 的竞争比是 1.732。 □

## 基金项目

河南省自然科学基金项目(222300420503), 河南省高校重点科研项目(22A110015), 河南省高校青年骨干教师培养计划项目(2019GGJS202, 2018XJGGJS-10)。

## 参考文献

- [1] 唐国春, 张峰, 罗守成, 刘丽丽. 现代排序论[M]. 上海: 上海科学普及出版社, 2003.
- [2] 万国华. 排序与调度的理论、模型和算法[M]. 北京: 清华大学出版社, 2019.
- [3] Blazewicz, J., Ecker, K., Pesch, E., Schmidt, G., Sterna, M. and Weglarz, J. (2019) Handbook on Scheduling—From Theory to Practice. Springer Nature Switzerland AG, Cham. <https://doi.org/10.1007/978-3-319-99849-7>
- [4] Baker, K.R. (1974) Introduction to Sequencing and Scheduling. John Wiley & Sons, New York.
- [5] Li, W.J. and Yuan, J.J. (2021) Single-Machine Online Scheduling of Jobs with Non-Delayed Processing Constraint. *Journal of Combinatorial Optimization*, **41**, 830-843. <https://doi.org/10.1007/s10878-021-00722-4>
- [6] Graham, R.L., Lawer, E.L. and Lenstra, J.K. (1979) Optimization and Approximation in Deterministic Sequencing and Scheduling. *Annals of Discrete Mathematics*, **5**, 287-326. [https://doi.org/10.1016/S0167-5060\(08\)70356-X](https://doi.org/10.1016/S0167-5060(08)70356-X)
- [7] Chai, X., Lu, L.F., Li, W.H. and Zhang, L.Q. (2018) Best-Possible Online Algorithms for Single Machine Scheduling to Minimize the Maximum Weighted Completion Time. *Asia-Pacific Journal of Operational Research*, **35**, Article ID: 1850048. <https://doi.org/10.1142/S0217595918500483>
- [8] Li, W.H. and Chai, X. (2018) Online Scheduling on Bounded Batch Machines to Minimize the Maximum Weighted Completion Time. *Journal of the Operations Research Society of China*, **6**, 455-465. <https://doi.org/10.1007/s40305-017-0179-x>
- [9] Chen, R.B. and Yuan, J.J. (2020) Single-Machine Scheduling of Proportional-Linearly Deteriorating Jobs with Positional Due Indices. *4OR: A Quarterly Journal of Operations Research*, **18**, 177-196. <https://doi.org/10.1007/s10288-019-00410-4>
- [10] Yuan, J.J., Ng, C.T. and Cheng, T.C.E. (2020) Scheduling with Release Dates and Preemption to Minimize Multiple Max-Form Objective Functions. *European Journal of Operational Research*, **280**, 860-875. <https://doi.org/10.1016/j.ejor.2019.07.072>

- [11] Zhao, Q.L. and Yuan, J.J. (2020) Bicriteria Scheduling of Equal Length Jobs on Uniform Parallel Machines. *Journal of Combinatorial Optimization*, **39**, 637-661. <https://doi.org/10.1007/s10878-019-00507-w>
- [12] Feng, Q. and Yuan, J.J. (2007) NP-Hardness of a Multicriteria Scheduling on Two Families of Jobs. *OR Transactions*, **11**, 121-126.