RAIRO-Oper. Res. 58 (2024) 1093–1103
https://doi.org/10.1051/ro/2024017

**RAIRO Operations Research**
www.rairo-ro.org

# PRICE OF ANARCHY IN UNIFORM PARALLEL MACHINES SCHEDULING GAME WITH WEIGHTED COMPLETION TIME AS SOCIAL GOAL

Felipe T. Muñoz[1,*] and Marco A. Parra[2]

**Abstract.** This article investigates the efficiency of Nash equilibria in a utilitarian scheduling game, where each job, acting as an agent, strategically selects a machine for its processing. The study focuses on a uniform parallel machine environment, employing the Weighted Shortest Processing Time rule as the local policy and the total weighted completion time as the social goal. We establish upper and lower bounds for the Price of Anarchy, offering valuable insights for this scheduling game.

**Mathematics Subject Classification.** 91A10, 91B14, 90B35.

Received September 12, 2022. Accepted January 21, 2024.

## 1. Introduction

This article delves into the scheduling game, where a set of jobs must be processed by uniform parallel machines. In contrast to classical scheduling problems, where a decision-maker optimizes a central goal (objective function) by determining the optimal schedule, each job acts as an agent. Each job strategically selects a machine based on the minimization of its disutility, represented by the completion time associated with that decision. This decentralized decision-making can lead to a schedule where no agent (job) can unilaterally change its decision or strategy to reduce its disutility. Such a schedule is a Nash equilibrium. The value of the central objective for Nash equilibrium may be worse than that of the central objective for the optimal schedule (social optimum) obtained through centralized decision-making. This inefficiency is quantified by the Price of Anarchy (PoA), proposed by Koutsoupias and Papadimitriou [16]. In a given game, the PoA for an instance is defined as the worst-case ratio of the central objective in a Nash equilibrium to the social optimum. The PoA of a game represents the worst-case value of this ratio across all instances of the game [7].

This study analyzes the PoA for the worst-case pure Nash equilibrium and provides both upper and lower bounds. Each machine employs the Weighted Shortest Processing Time (WSPT) rule as the local policy. The WSPT rule sequences jobs in decreasing order of the $w_j/p_j$ ratio, where $p_j$ represents the processing requirement of job $j$ and $w_j$ represents the weight or relative importance of job $j$. An important feature of this rule is its ability to generate an optimal sequence for a set of jobs assigned to a single machine [21]. According to the classification scheme proposed by Lee *et al.* [17], this problem is represented by $Q_m(\text{WSPT})|ut = -C_j|\sum w_jC_j$. Meanwhile, the centralized problem is represented by $Q_m||\sum w_jC_j$ in the three-field scheduling notation [11].

*Keywords.* Price of anarchy, scheduling game, uniform parallel machines, weighted completion time.

[1] Department of Industrial Engineering, Universidad del Bío-Bío, Concepción, Chile.
[2] Faculty of Engineering, Universidad del Bío-Bío, Concepción, Chile.
*Corresponding author: fmunoz@ubiobio.cl

### 1.1. Related works

According to Immorlica *et al.* [14], large-scale autonomous systems have become increasingly common with the advent of the Internet. These systems consist of many independent and selfishly acting agents, all competing to share a common resource, such as bandwidth in a network or processing power in a parallel computing environment. Practical settings range from smart routing among stub autonomous systems in the Internet to selfish user association in wireless local area networks. In many systems of this kind, it is infeasible to impose some centralized control on the users. Rather, a centralized authority can only design protocols *a priori* and hope that the independent and selfish choices of the users combine to create socially desirable results. We encourage readers to refer to the introductory article on scheduling games by Heydenreich *et al.* [12] to comprehensively explore different models, techniques, and results in this domain.

From Koutsoupias and Papadimitriou work [16], makespan as a social goal has been studied by Azar *et al.* [2], Christodoulou *et al.* [6], Czumaj and Vöcking [9], Immorlica *et al.* [14], Yu *et al.* [22] and Caragiannis *et al.* [5]. In this scenario, the disutility of an agent (job) is determined by the total load of the machine on which it is processed. Hence, minimizing the makespan is a minimax social choice objective. On the other hand, the literature offers results for scheduling games where the minimization of the total weighted or unweighted completion time (minsum) is considered as the social goal [1, 3, 7, 8, 13, 17, 19, 23].

The literature provides some results related to the studied problem. In Theorem 3.2 of [7] and Theorem 1 of [1], it is established that the PoA of pure Nash equilibria for the $R(\mathrm{WSPT})|ut = -C_j|\sum w_j C_j$ problem is at most 4. On the other hand, in Theorem 9 of [8], the same bound is proved for a particular version of the $Q_m(\mathrm{WSPT})|ut = -C_j, \mathcal{M}_j|\sum w_j C_j$ problem, where $\mathcal{M}_j$ represents the presence of machine eligibility restrictions. Both problems represent a generalization of the studied problem. More precisely, the problem under study is a specific case of $Q_m(\mathrm{WSPT})|ut = -C_j, \mathcal{M}_j|\sum w_j C_j$ problem, where each job can be processed by any machine. Given this relationship, it can be established that the PoA for the $Q_m(\mathrm{WSPT})|ut = -C_j, \mathcal{M}_j|\sum w_j C_j$ problem serves as an upper bound for the $Q_m(\mathrm{WSPT})|ut = -C_j|\sum w_j C_j$ problem.

Other authors studied the problem without weights, equivalent to letting $w_j = 1$ for all jobs $j$. For these problems, the local policy used was the SPT (Shortest Processing Time first) rule. In Theorem 7 of [17], it is established that the PoA of pure Nash equilibrium for the $Q_2(\mathrm{SPT})|ut = -C_j|\sum C_j$ problem is in a range between $(3+\sqrt{3})/4 \approx 1.183$ and $(1+\sqrt{5})/2 \approx 1.618$. Later, in Theorem 4 of [23] a lower bound of 1.1875 is established. Results for the $Q_m(\mathrm{SPT})|ut = -C_j|\sum C_j$ problem are presented in [13, 23]. In Theorems 3 and 5 of [13], it is established that the PoA is in a range between $e/(e-1) \approx 1.582$ and 2, while in Theorem 3 of [23] it is established that the upper bound is $2 - 2/((n+m)(n+1))$, where $n$ and $m$ represent the number of jobs and machines, respectively.

Table 1 displays the PoA of the pure Nash equilibria for related problems. The performance guarantee is tight for problems where a single value is reported, representing the worst-case PoA for Nash equilibrium. In cases where a range of values is reported, these values serve as lower and upper bounds for the PoA. This inaccuracy highlights a research opportunity to establish a tighter PoA or narrow the upper and lower bounds gap.

### 1.2. Our results

The main results of this work include a lower bound on PoA of 2 and an upper bound depending on the machine speeds and the number of machines. Specifically, the upper bound is given by:

$$\frac{3}{2}\left(1 - \frac{1}{m}\right)\frac{s_{\max}}{s_{\min}} + \frac{1}{m},$$

where $m > 2$ is the number of machines, $s_{\max}$ and $s_{\min}$ represent the speed of the fastest and slowest machines, respectively. Additionally, for the $m = 2$ case, we establish an upper bound of:

$$\frac{s_{\max}}{s_{\min}} + \frac{1}{4}.$$

TABLE 1. Price of Anarchy for related problems.

| Problem | Price of Anarchy | Reference |
|---|---|---|
| $Q_2(\text{SPT})\|ut = -C_j\|\sum C_j$ | $[1.1830, 1.6180]$ | [17] |
| $Q_2(\text{SPT})\|ut = -C_j\|\sum C_j$ | PoA $\geq 1.1875$ | [23] |
| $Q(\text{SPT})\|ut = -C_j\|\sum C_j$ | $[1.5820, 2]$ | [13] |
| $Q_m(\text{SPT})\|ut = -C_j\|\sum C_j$ | PoA $\leq 2 - \frac{2}{(n+m)(n+1)}$ | [23] |
| $Q_2(\text{WSPT})\|ut = -C_j\|\sum w_jC_j$ | PoA $\leq \frac{s_{\max}}{s_{\min}} + \frac{1}{4}$ | Section 5.1 |
| $Q_m(\text{WSPT})\|ut = -C_j\|\sum w_jC_j$ | PoA $\leq \frac{3}{2}\left(1 - \frac{1}{m}\right)\frac{s_{\max}}{s_{\min}} + \frac{1}{m}$ | Section 5.1 |
| $Q(\text{WSPT})\|ut = -C_j\|\sum w_jC_j$ | PoA $\geq 2$ | Section 5.2 |
| $Q(\text{WSPT})\|ut = -C_j, \mathcal{M}_j\|\sum w_jC_j$ | 4 | [8] |
| $R(\text{WSPT})\|ut = -C_j\|\sum w_jC_j$ | 4 | [1,7] |

**Notes.** $s_{\max}$ and $s_{\min}$ denote the speed of the fastest and slowest machine, respectively. $m$ represents the number of machines, and $n$ represents the number of jobs.

Regarding methodological contributions, the proof technique used to establish the upper bound of the PoA is similar to that used in [13, 18, 23] and consists of establishing inequalities that describe the properties of the solutions of the centralized and decentralized problem. On the other hand, to set the lower bound for the PoA, an instance whose PoA is 2 is proposed. This instance was designed taking as a reference the instances proposed in [13, 20].

The remainder of the paper unfolds as follows: Section 2 provides the problem statement and introduces the notation used. Section 3 elucidates some properties of the optimal solution, while Section 4 delves into properties characterizing Nash equilibria. Section 5 presents the upper and lower bounds for the PoA. Finally, Section 6 encapsulates the main conclusions drawn from this study.

## 2. Problem statement

$\mathcal{J} = \{1, \ldots, n\}$ represents the set of jobs that need to be scheduled on a set $\mathcal{M} = \{1, \ldots, m\}$ of uniformly related parallel machines. Each job $j$ has a processing requirement $p_j$ and a weight $w_j$, which represents the relative importance of the job. Without losing generality, it is considered that the jobs are indexed according to the WSPT rule ([21], Thm. 3). When there are ties in the ratio, these are arbitrarily resolved. Then,

$$\frac{w_1}{p_1} \geq \frac{w_2}{p_2} \geq \cdots \geq \frac{w_n}{p_n}. \tag{2.1}$$

Jobs must be scheduled on a single machine, and each machine can only process a single job at a time without interruption. Machines operate at different speeds, where $\boldsymbol{s}$ is the vector representing the speeds of the machines and $s_i > 0$ denotes the speed of machine $i \in \mathcal{M}$. Then, if job $j$ is assigned to machine $i$, $p_j/s_i$ units of processing time will be required. Given a vector $\boldsymbol{s}$, $s_{\min}$ and $s_{\max}$ are the lowest and highest speeds, respectively. Without losing generality, it is considered that the index and speed of the machines are adapted (modified) in such a way that

$$1 = s_1 \leq s_2 \leq \cdots \leq s_m = \frac{s_{\max}}{s_{\min}}. \tag{2.2}$$

A solution to the problem or schedule corresponds to an assignment of jobs to machines represented by a vector $\boldsymbol{x}$, where $x_j$ gives the machine to which job $j$ is assigned, that is, $x_j = i$ if job $j$ is assigned to machine $i$ in schedule $\boldsymbol{x}$. $\mathcal{J}_i(\boldsymbol{x})$ is the set of jobs assigned to machine $i$ in schedule $\boldsymbol{x}$. Given a schedule $\boldsymbol{x}$, each job $j \in \mathcal{J}$ will have a completion time. Let $C_j(\boldsymbol{x}, \boldsymbol{s})$ be the completion time of job $j$ in schedule $\boldsymbol{x}$ with machines that operate (work) at speed $\boldsymbol{s}$. The jobs sequence is solved using the WSPT rule (this order is induced by assumption (2.1)). The completion time of job $j$ in schedule $\boldsymbol{x}$ can be expressed as

$$C_j(\boldsymbol{x}, \boldsymbol{s}) = \frac{p_j}{s_{x_j}} + \sum_{\substack{k \in \mathcal{J}_{x_j}(\boldsymbol{x}) \\ k<j}} \frac{p_k}{s_{x_j}}.$$

The Total Weighted Completion Time (TWCT) and the weighted sum of processing times of schedule $\boldsymbol{x}$ are defined as follows:

$$\begin{aligned} Z(\boldsymbol{x}, \boldsymbol{s}) &= \sum_{j \in \mathcal{J}} w_j C_j(\boldsymbol{x}, \boldsymbol{s}) = \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{J}_i(\boldsymbol{x})} w_j C_j(\boldsymbol{x}, \boldsymbol{s}), \\ \eta(\boldsymbol{x}) &= \sum_{j \in \mathcal{J}} \frac{w_j p_j}{s_{x_j}} = \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{J}_i(\boldsymbol{x})} \frac{w_j p_j}{s_i}. \end{aligned}$$

With these definitions, the following identities are immediate.

$$Z(\boldsymbol{x}, \boldsymbol{s}) = \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{J}_i(\boldsymbol{x})} \sum_{\substack{k \in \mathcal{J}_i(\boldsymbol{x}) \\ k \le j}} \frac{w_j p_k}{s_i} = \eta(\boldsymbol{x}) + \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{J}_i(\boldsymbol{x})} \sum_{\substack{k \in \mathcal{J}_i(\boldsymbol{x}) \\ k < j}} \frac{w_j p_k}{s_i} = \eta(\boldsymbol{x}) + \sum_{j \in \mathcal{J}} \sum_{\substack{k \in \mathcal{J}_{x_j}(\boldsymbol{x}) \\ k<j}} \frac{w_j p_k}{s_{x_j}}. \tag{2.3}$$

Hereinafter, $\boldsymbol{x}$ is a Nash equilibrium, and $\boldsymbol{x}^*$ is the social optimum.

## 3. Centralized solution, properties of social optimum

This section presents some properties of the centralized solution. These properties will later be used to prove an upper bound for the PoA.

Early on, Eastman *et al.* ([10], Thm. 1) established the following inequality,

$$\sum_{j \in \mathcal{J}} \sum_{\substack{k \in \mathcal{J} \\ k \le j}} w_j p_k \le m Z(\tilde{\boldsymbol{x}}, \mathbf{1}) - \frac{(m-1)}{2} \sum_{j \in \mathcal{J}} w_j p_j, \tag{3.1}$$

where $\tilde{\boldsymbol{x}}$ represents the optimal solution if all machines were to operate at unit speed, $Z(\tilde{\boldsymbol{x}}, \mathbf{1})$ represents the TWCT of $\tilde{\boldsymbol{x}}$ at unit-speed machines, and

$$\sum_{j \in \mathcal{J}} \sum_{\substack{k \in \mathcal{J} \\ k \le j}} w_j p_k,$$

is the TWCT when all the jobs are assigned to a single machine that operates at unit speed.

Next, we introduce two lemmas that provide insights into the properties of the TWCT for the centralized solution, often referred to as the social optimum. The first lemma is derived from [18], where the worst-case performance guarantee of locally optimal solutions is investigated for the centralized problem. The specific neighborhood considered in this study is the jump neighborhood, also known as insertion.

**Lemma 3.1** ([18], Lem. 2.1)**.** *The total weighted completion time of the social optimum $\boldsymbol{x}^*$ of the $Q_m(\mathrm{WSPT})|ut = -C_j|\sum w_j C_j$ problem satisfies the inequality:*

$$Z(\tilde{\boldsymbol{x}}, \mathbf{1}) \le Z(\boldsymbol{x}^*, \mathbf{1}) \le \left(\frac{s_{\max}}{s_{\min}}\right) Z(\boldsymbol{x}^*, \boldsymbol{s}).$$

*Proof.* If $\tilde{\boldsymbol{x}}$ is the optimal solution when the machines work at unit speed, the solution $\boldsymbol{x}^*$ reports a TWCT of at least $Z(\tilde{\boldsymbol{x}}, \mathbf{1})$ in that environment (unit-speed machines). Then, we have that $Z(\tilde{\boldsymbol{x}}, \mathbf{1}) \leq Z(\boldsymbol{x}^*, \mathbf{1})$.

On the other hand, starting from equation (2.3) and using equation (2.2), we have

$$Z(\boldsymbol{x}^*, \boldsymbol{s}) = \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{J}_i(\boldsymbol{x}^*)} \sum_{\substack{k \in \mathcal{J}_i(\boldsymbol{x}^*) \\ k \leq j}} \frac{w_j p_k}{s_i} \geq \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{J}_i(\boldsymbol{x}^*)} \sum_{\substack{k \in \mathcal{J}_i(\boldsymbol{x}^*) \\ k \leq j}} \frac{w_j p_k}{s_m} = \frac{Z(\boldsymbol{x}^*, \mathbf{1})}{s_m}.$$

□

**Lemma 3.2.** *The total weighted completion time of the social optimum* $\boldsymbol{x}^*$ *of the* $Q_m(\text{WSPT})|ut = -C_j|\sum w_j C_j$ *problem satisfies the inequality:*

$$\sum_{j \in \mathcal{J}} \sum_{\substack{k \in \mathcal{J} \\ k \leq j}} w_j p_k \leq m\left(\frac{s_{\max}}{s_{\min}}\right) Z(\boldsymbol{x}^*, \boldsymbol{s}) - \frac{(m-1)}{2} \sum_{j \in \mathcal{J}} w_j p_j.$$

*Proof.* Follows from equation (3.1) and Lemma 3.1. □

Further properties of the social optimum are detailed below:

$$\sum_{j \in \mathcal{J}} w_j p_j \leq Z(\boldsymbol{x}^*, \mathbf{1}) \tag{3.2}$$

$$\sum_{j \in \mathcal{J}} w_j p_j \geq \eta(\boldsymbol{x}^*). \tag{3.3}$$

## 4. Decentralized solution, Nash equilibrium property

This section presents an upper bound for the TWCT of any Nash equilibrium. This inequality relates the Nash equilibrium with the social optimum.

**Lemma 4.1.** *For any Nash equilibrium* $\boldsymbol{x}$ *of* $Q_m(\text{WSPT})|ut = -C_j|\sum w_j C_j$ *problem, its total weighted completion time satisfies*

$$Z(\boldsymbol{x}, \boldsymbol{s}) \leq \frac{\eta(\boldsymbol{x}^*)}{m} + \left(\frac{m-3}{2m}\right) \sum_{j \in \mathcal{J}} w_j p_j + \left(\frac{s_{\max}}{s_{\min}}\right) Z(\boldsymbol{x}^*, \boldsymbol{s}). \tag{4.1}$$

*Proof.* For any job $j \in \mathcal{J}$, let $i \in \mathcal{M}$ be one of its strategies. Then, if job $j$ takes strategy $i$ (move to machine $i$), it will get a completion time

$$\frac{p_j}{s_i} + \sum_{\substack{k \in \mathcal{J}_i(\boldsymbol{x}) \\ k < j}} \frac{p_k}{s_i}.$$

So that, a schedule $\boldsymbol{x}$ will be a Nash equilibrium if and only if

$$C_j(\boldsymbol{x}, \boldsymbol{s}) \leq \frac{p_j}{s_i} + \sum_{\substack{k \in \mathcal{J}_i(\boldsymbol{x}) \\ k < j}} \frac{p_k}{s_i}, \qquad \text{for all } j \in \mathcal{J}, i \in \mathcal{M}. \tag{4.2}$$

We multiply equation (4.2) by $w_j$ and sum over all $i \in \mathcal{M}$,

$$m w_j C_j(\boldsymbol{x}, \boldsymbol{s}) \leq \sum_{i \in \mathcal{M}} \frac{w_j p_j}{s_i} + \sum_{i \in \mathcal{M}} \sum_{\substack{k \in \mathcal{J}_i(\boldsymbol{x}) \\ k < j}} \frac{w_j p_k}{s_i}. \tag{4.3}$$

On the other hand, for the first term on the right-hand side of equation (4.3), we have that

$$\sum_{i\in\mathcal{M}} \frac{w_j p_j}{s_i} = \frac{w_j p_j}{s_{x_j^*}} + \sum_{i\in\mathcal{M}\setminus\{x_j^*\}} \frac{w_j p_j}{s_i} \leq \frac{w_j p_j}{s_{x_j^*}} + (m-1) w_j p_j. \tag{4.4}$$

By equations (4.4) and (4.3),

$$m w_j C_j(\boldsymbol{x}, \boldsymbol{s}) \leq \frac{w_j p_j}{s_{x_j^*}} + (m-1) w_j p_j + \sum_{i\in\mathcal{M}} \sum_{\substack{k\in\mathcal{J}_i(\boldsymbol{x})\\ k<j}} \frac{w_j p_k}{s_i}.$$

Summing over all $j \in \mathcal{J}$ and using equation (2.3), we have

$$mZ(\boldsymbol{x}, \boldsymbol{s}) \leq \eta(\boldsymbol{x}^*) + (m-1)\sum_{j\in\mathcal{J}} w_j p_j + \sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{M}} \sum_{\substack{k\in\mathcal{J}_i(\boldsymbol{x})\\ k<j}} \frac{w_j p_k}{s_i}. \tag{4.5}$$

On the other hand, for the last term of equation (4.5), we have that

$$\sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{M}} \sum_{\substack{k\in\mathcal{J}_i(\boldsymbol{x})\\ k<j}} \frac{w_j p_k}{s_i} = \sum_{j\in\mathcal{J}} \sum_{\substack{k\in\mathcal{J}\\ k<j}} \frac{w_j p_k}{s_{x_k}} \leq \sum_{j\in\mathcal{J}} \sum_{\substack{k\in\mathcal{J}\\ k<j}} w_j p_k = \sum_{j\in\mathcal{J}} \sum_{\substack{k\in\mathcal{J}\\ k\leq j}} w_j p_k - \sum_{j\in\mathcal{J}} w_j p_j. \tag{4.6}$$

By equations (4.6) and (4.5),

$$mZ(\boldsymbol{x}, \boldsymbol{s}) \leq \eta(\boldsymbol{x}^*) + (m-2)\sum_{j\in\mathcal{J}} w_j p_j + \sum_{j\in\mathcal{J}} \sum_{\substack{k\in\mathcal{J}\\ k\leq j}} w_j p_k.$$

Finally, we conclude the proof by using Lemma 3.2. $\square$

## 5. Price of Anarchy

This section establishes upper and lower bounds for the PoA of the $Q_m(\text{WSPT})|ut = -C_j|\sum w_j C_j$ problem.

### 5.1. Upper bound

Lemma 4.1 will be used to determine the upper bound for the PoA. Since the second term on the right-hand side of equation (4.1) is less than zero for $m = 2$ and greater than zero for $m > 3$, it is necessary to analyze these cases separately.

**Theorem 5.1.** *For $m > 2$, the PoA for the pure Nash equilibria of the $Q_m(\text{WSPT})|ut = -C_j|\sum w_j C_j$ problem is at most*

$$\frac{3}{2}\left(1 - \frac{1}{m}\right)\left(\frac{s_{\max}}{s_{\min}}\right) + \frac{1}{m}. \tag{5.1}$$

*Proof.* By Lemma 4.1, equation (3.2) and Lemma 3.1

$$Z(\boldsymbol{x}, \boldsymbol{s}) \leq \frac{\eta(\boldsymbol{x}^*)}{m} + \frac{3}{2}\left(\frac{m-1}{m}\right)\left(\frac{s_{\max}}{s_{\min}}\right) Z(\boldsymbol{x}^*, \boldsymbol{s}).$$

By equation (2.3) we have $\eta(\boldsymbol{x}^*) \leq Z(\boldsymbol{x}^*, \boldsymbol{s})$. Finally, we can conclude that,

$$Z(\boldsymbol{x}, \boldsymbol{s}) \leq Z(\boldsymbol{x}^*, \boldsymbol{s})\left(\frac{3}{2}\left(\frac{m-1}{m}\right)\left(\frac{s_{\max}}{s_{\min}}\right) + \frac{1}{m}\right).$$

$\square$

**Theorem 5.2.** *The PoA for the pure Nash equilibria of the* $Q_2(\text{WSPT})|ut = -C_j|\sum w_j C_j$ *problem is at most*

$$\left(\frac{s_{\max}}{s_{\min}}\right) + \frac{1}{4}.$$

*Proof.* Using $m = 2$ in Lemma 4.1 gives

$$Z(\boldsymbol{x}, \boldsymbol{s}) \leq \frac{\eta(\boldsymbol{x}^*)}{2} - \frac{1}{4}\sum_{j\in\mathcal{J}} w_j p_j + \left(\frac{s_{\max}}{s_{\min}}\right) Z(\boldsymbol{x}^*, \boldsymbol{s}).$$

By equation (3.3),

$$Z(\boldsymbol{x}, \boldsymbol{s}) \leq \frac{\eta(\boldsymbol{x}^*)}{4} + \left(\frac{s_{\max}}{s_{\min}}\right) Z(\boldsymbol{x}^*, \boldsymbol{s}).$$

By equation (2.3) we have $\eta(\boldsymbol{x}^*) \leq Z(\boldsymbol{x}^*, \boldsymbol{s})$. Finally, we can conclude that

$$Z(\boldsymbol{x}, \boldsymbol{s}) \leq Z(\boldsymbol{x}^*, \boldsymbol{s})\left(\frac{s_{\max}}{s_{\min}} + \frac{1}{4}\right).$$

□

The upper bounds presented in Theorems 5.1 and 5.2 have the particularity of including the value of the $s_{\max}/s_{\min}$ ratio, thus incorporating a measurable characteristic of the studied machine environment. However, the value of these bounds grows as the ratio grows. Consequently, they prove to be useful primarily for specific instances of the problem, specifically those where the ratio is not excessively large.

To provide a more refined delimitation of the PoA, we introduce the following theorem, utilizing both the parametric upper bound and a fixed bound of 4. This fixed bound is derived from a more general setting than the problem under study, as detailed in Table 1.

**Theorem 5.3.** *The PoA for the pure Nash equilibria of* $Q_m(\text{WSPT})|ut = -C_j|\sum w_j C_j$ *problem is no greater than*

$$\begin{aligned} &\min\left\{\left(\frac{s_{\max}}{s_{\min}}\right) + \frac{1}{4};\ 4\right\}, && \text{for } m = 2\\ &\min\left\{\frac{3}{2}\left(1 - \frac{1}{m}\right)\left(\frac{s_{\max}}{s_{\min}}\right) + \frac{1}{m};\ 4\right\}, && \text{for } m > 2. \end{aligned}$$

From the upper bound presented in Theorem 5.3, it can be determined that the upper bounds of Theorems 5.1 and 5.2 will be useful, provided that:

$$\begin{aligned} &\frac{s_{\max}}{s_{\min}} \leq \frac{15}{4} = 3.75, && \text{for } m = 2 \text{ machines},\\ &\frac{s_{\max}}{s_{\min}} \leq \frac{2}{3}\left(\frac{4m-1}{m-1}\right) = \frac{2}{3} + \frac{2m}{m-1}, && \text{for } m > 2 \text{ machines}. \end{aligned}$$

For $m = 3$, the last condition states that $s_{\max}/s_{\min} \leq 11/3 \approx 3.67$, and when $m$ increases, the right-hand side term decreases monotonically and converges to $8/3 \approx 2.67$.

Another interesting situation related to upper bound (5.1) arises when the ratio $s_{\max}/s_{\min} = 1$. For this case, the upper bound expression becomes equal to $(3m - 1)/2m$. This expression matches the robust PoA for $P_m(\text{SPT})|ut = -C_j|\sum C_j$ problem [19], and the performance guarantees of move-optimal schedules for $P||\sum w_j C_j$ problem [4].

## 5.2. Lower bound

To establish a lower bound for the PoA, an instance inspired by the work of [13, 20] is presented.

**Theorem 5.4.** *The PoA for* $Q(\text{WSPT})|ut = -C_j|\sum w_jC_j$ *problem is at least* 2.

*Proof.* We consider an instance $\mathcal{I}$ with $n$ jobs and $m = n-k+1$ machines. The first machine has speed $s \in \mathbb{N}$, with $s > 1$, and all the other machines have speed 1. Jobs have identical weight and processing requirement,

$$w_j = p_j = \begin{cases} 1 & \text{if } 1 \leq j \leq s-1 \\ r^{j-s} & \text{if } s \leq j \leq n \end{cases},$$

where $r = \frac{s}{s-1}$.

For this instance, we have a Nash equilibrium $\boldsymbol{x}$ when all jobs are scheduled on the fastest machine. Then, the completion time for jobs will be (consult Appendix A.1 for the details of this part of the proof)

$$C_j(\boldsymbol{x}) = \begin{cases} \dfrac{j}{s} & \text{for } 1 \leq j \leq s-1 \\ r^{j-s} & \text{for } s \leq j \leq n \end{cases}.$$

Schedule $\boldsymbol{x}$ is a Nash equilibrium, since Equation (4.2) can be verified for all jobs. More precisely, for this instance we have $C_j(\boldsymbol{x}) \leq p_j$ for all $j = 1, \ldots, n$, where $p_j$ would be the new completion time for job $j$ if it changed its strategy to any other machine (unit-speed machine). The TWCT for schedule $\boldsymbol{x}$ is (consult Appendix A.3 for the details of this part of the proof)

$$Z(\boldsymbol{x}) = \left(\frac{s-1}{2}\right) + \left(\frac{r^{2n-2s+2}-1}{r^2-1}\right).$$

Let $\hat{\boldsymbol{x}}$ be another schedule for the instance $\mathcal{I}$ where the jobs $j = 1, \ldots, n-k$ are assigned to unit-speed machines, and the remaining jobs, $j = n-k+1, \ldots, n$, are assigned to the machine that works at speed $s$. Note that the unit-speed machines will only process one job and $s < n-k$. The completion time of jobs in this schedule will be (consult Appendix A.2 for the details of this part of the proof)

$$C_j(\hat{\boldsymbol{x}}) = \begin{cases} 1 & \text{for } 1 \leq j \leq s-1 \\ r^{j-s} & \text{for } s \leq j \leq n-k \\ r^{j-s} - r^{n-k-s} & \text{for } n-k+1 \leq j \leq n \end{cases}.$$

The TWCT for schedule $\hat{\boldsymbol{x}}$ is (consult Appendix A.4 for the details of this part of the proof)

$$Z(\hat{\boldsymbol{x}}) = (s-1) + \left(\frac{r^{2n-2s+2}-1}{r^2-1}\right) + \left(\frac{r^{2n-2s+1-2k} - r^{2n-2s+1-k}}{r-1}\right).$$

Now, we determine the $\frac{Z(\boldsymbol{x})}{Z(\hat{\boldsymbol{x}})}$ ratio. After some algebraic considerations, it is determined that the ratio is:

$$\frac{Z(\boldsymbol{x})}{Z(\hat{\boldsymbol{x}})} = \frac{\left(\frac{\left(\frac{s-1}{2}\right)\left(r^2-1\right)-1}{r^{2n-2s+2}}\right)+1}{\left(\frac{(s-1)(r^2-1)-1}{r^{2n-2s+2}}\right)+1+\left(2-\frac{1}{s}\right)\left(r^{-2k}-r^{-k}\right)}.$$

For $n-s+1 \to \infty$, the ratio $\frac{Z(\boldsymbol{x})}{Z(\hat{\boldsymbol{x}})}$ converges to

$$\varphi = \frac{1}{1+\left(2-\frac{1}{s}\right)\left(r^{-2k}-r^{-k}\right)}.$$

Letting $k = \lceil s \ln 2 \rceil$,

$$\varphi = \frac{1}{1 + \left(2 - \frac{1}{s}\right)\left(r^{-2\lceil s \ln 2 \rceil} - r^{-\lceil s \ln 2 \rceil}\right)}.$$

For $s \to \infty$, we have that $\varphi$ converges to

$$\varphi' = \frac{1}{1 + 2(r^{-2s \ln 2} - r^{-s \ln 2})} = \frac{1}{1 + 2\Big(\big(r^s\big)^{-2 \ln 2} - \big(r^s\big)^{-\ln 2}\Big)}.$$

Since $r = 1 + \frac{1}{s-1}$, we have that

$$\varphi' = \frac{1}{1 + 2(e^{-2 \ln 2} - e^{-\ln 2})} = \frac{1}{1 + 2(4^{-1} - 2^{-1})} = 2.$$

Let $\boldsymbol{x}^*$ be the unknown social optimum and $Z(\boldsymbol{x}^*)$ be its TWCT. Since $\hat{\boldsymbol{x}}$ is a feasible solution of the instance, we have that $Z(\boldsymbol{x}^*) \leq Z(\hat{\boldsymbol{x}})$. Therefore

$$\text{PoA} = \frac{Z(\boldsymbol{x})}{Z(\boldsymbol{x}^*)} \geq \varphi' = 2.$$

□

Considering the presented instance and letting $w_j = 1$ for all $j \in \mathcal{J}$, we have an instance of the unweighted case of the studied problem, the $Q_m(\text{SPT})|ut = -C_j|\sum C_j$ problem. Then, by making $k = s$, we get the same instance presented in [13], which proved a lower bound of $e/(e-1) \approx 1.582$ for the PoA of this scheduling game. The upper bound for the PoA of this problem is 2.

Here, it can be seen that the PoA is between 1.582 and 2 for the unweighted case. Whereas for weighted case, the PoA is at least 2. Evidencing a separation (disjoint sets) in the range of values where the real value of the PoA can be found.

## 6. Concluding remarks

In this paper, the PoA of scheduling game $Q_m(\text{WSPT})|ut = -C_j|\sum w_j C_j$ was studied. Given the design of a worst-case instance, it is determined that the PoA for the problem is at least 2. Conversely, based on the PoA of a related problem, identified as a generalization of the studied problem, it is established that the PoA is at most 4. Additionally, a parametric upper bound for the PoA has been determined, enabling the establishment of upper bounds for the PoA for families of instances described by three parameters: the number of machines, the speed of the fastest machine, and the speed of the slowest machine. Both fixed and parametric bounds contribute to a better understanding of the efficiency achieved in this scheduling game.

Further research to determine a tight PoA for the problem or narrow the gap between upper and lower bounds opens an intriguing direction for future investigations. Identifying new parametric bounds for the PoA holds promise in advancing our understanding of the problem.

## Appendix A. Complement for the proof of Theorem 5.4

### A.1. Completion time of jobs for Nash equilibrium

For jobs $j = 1, \ldots, s-1$ we have that their completion time is

$$C_j(\boldsymbol{x}) = \sum_{h=1}^{j} \frac{1}{s} = \frac{j}{s}.$$

Then, the completion time of job $s-1$ is $\left(\frac{s-1}{s}\right)$. With this result and taking into account that $r=\frac{s}{s-1}$, the completion time for jobs $j=s,\ldots,n$ is

$$\begin{aligned}C_j(\boldsymbol{x}) &= \left(\frac{s-1}{s}\right)+\sum_{h=s}^{j}\frac{r^{h-s}}{s}=\left(\frac{s-1}{s}\right)+\frac{1}{s}\sum_{h=0}^{j-s}r^h\\&=\left(\frac{s-1}{s}\right)+\frac{1}{s}\left(\frac{r^{j-s+1}-1}{r-1}\right)=\left(\frac{s-1}{s}\right)+\left(\frac{s-1}{s}\right)\left(r^{j-s+1}-1\right)\\&=\left(\frac{s-1}{s}\right)\left(r^{j-s+1}\right)=\frac{1}{r}\left(r^{j-s+1}\right)=r^{j-s}.\end{aligned}$$

## A.2. Completion time of jobs for schedule $\hat{\boldsymbol{x}}$

Each job $j=1,\ldots,s-1$ is processed by unit-speed machines, so their completion time is $C_j(\hat{\boldsymbol{x}})=1$. Each job $j=s,\ldots,n-k$ is processed by unit-speed machines, so their completion time is $C_j(\hat{\boldsymbol{x}})=r^{j-s}$. Jobs $j=n-k+1,\ldots,n$ are processed by the s-speed machine, so their completion time is

$$\begin{aligned}C_j(\hat{\boldsymbol{x}}) &= \sum_{h=n-k+1}^{j}\frac{r^{h-s}}{s}=\sum_{h=n-k-s+1}^{j-s}\frac{r^h}{s}=\frac{1}{s}\left(\sum_{h=0}^{j-s}r^h-\sum_{h=0}^{n-k-s}r^h\right)\\&=\frac{1}{s}\left(\frac{r^{j-s+1}-1}{r-1}-\frac{r^{n-k-s+1}-1}{r-1}\right)=\frac{1}{s}\left(\frac{r^{j-s+1}-r^{n-k-s+1}}{r-1}\right)\\&=\left(\frac{s-1}{s}\right)\left(r^{j-s+1}-r^{n-k-s+1}\right)=\frac{1}{r}\left(r^{j-s+1}-r^{n-k-s+1}\right)=r^{j-s}-r^{n-k-s}.\end{aligned}$$

## A.3. Total weighted completion time of Nash equilibrium

$$\begin{aligned}Z(\boldsymbol{x}) &= \sum_{j=1}^{n}w_jC_j(\boldsymbol{x})=\sum_{j=1}^{s-1}\frac{j}{s}+\sum_{j=s}^{n}r^{2(j-s)}\\&=\frac{s(s-1)}{2s}+\sum_{j=0}^{n-s}r^{2j}=\left(\frac{s-1}{2}\right)+\left(\frac{r^{2(n-s+1)}-1}{r^2-1}\right).\end{aligned}$$

## A.4. Total weighted completion time of schedule $\hat{\boldsymbol{x}}$

$$\begin{aligned}Z(\hat{\boldsymbol{x}}) &= \sum_{j=1}^{n}w_jC_j(\hat{\boldsymbol{x}})=\sum_{j=1}^{s-1}1+\sum_{j=s}^{n-k}r^{2(j-s)}+\sum_{j=n-k+1}^{n}r^{j-s}\left(r^{j-s}-r^{n-s-k}\right)\\&=(s-1)+\sum_{j=s}^{n-k}r^{2(j-s)}+\sum_{j=n-k+1}^{n}r^{2(j-s)}-\sum_{j=n-k+1}^{n}r^{j+n-2s-k}\\&=(s-1)+\sum_{j=s}^{n}r^{2(j-s)}-\sum_{j=2n-2s+1-2k}^{2n-2s-k}r^j\\&=(s-1)+\sum_{j=0}^{n-s}r^{2j}-\left(\sum_{j=0}^{2n-2s-k}r^j-\sum_{j=0}^{2n-2s-2k}r^j\right)\\&=(s-1)+\left(\frac{r^{2(n-s+1)}-1}{r^2-1}\right)+\left(\frac{r^{2n-2s+1-2k}-r^{2n-2s+1-k}}{r-1}\right).\end{aligned}$$

## References


[1] F. Abed, J.R. Correa and C.-C. Huang, Optimal coordination mechanisms for multi-job scheduling games, in Algorithms-ESA 2014: 22th Annual European Symposium, Wroclaw, Poland, September 8–10, 2014. Proceedings 21. Springer Berlin Heidelberg (2014) 13–24.

[2] Y. Azar, L. Fleischer, K. Jain, V. Mirrokni and Z. Svitkina, Optimal coordination mechanisms for unrelated machine scheduling. *Oper. Res.* **63** (2015) 489–500.

[3] S. Bhattacharya, S. Im, J. Kulkarni and K. Munagala, Coordination mechanisms from (almost) all scheduling policies, in Proceedings of the 5th Conference on Innovations in Theoretical Computer Science (ITCS). ACM, NY, USA (2014) 121–134.

[4] T. Brueggemann, J.L. Hurink and W. Kern, Quality of move-optimal schedules for minimizing total weighted completion time. *Oper. Res. Lett.* **34** (2006) 583–590.

[5] I. Caragiannis, M. Flammini, C. Kaklamanis, P. Kanellopoulos and L. Moscardelli, Tight bounds for selfish and greedy load balancing. *Algorithmica* **61** (2011) 606–637.

[6] G. Christodoulou, E. Koutsoupias and A. Nanavati, Coordination mechanisms. *Theor. Comput. Sci.* **410** (2009) 3327–3336.

[7] R. Cole, J.R. Correa, V. Gkatzelis, V. Mirrokni and N. Olver, Decentralized utilitarian mechanisms for scheduling games. *Games Econ. Behav.* **92** (2015) 306–326.

[8] J.R. Correa and M. Queyranne, Efficiency of equilibria in restricted uniform machine scheduling with total weighted completion time as social cost. *Nav. Res. Logist.* **59** (2012) 384–395.

[9] A. Czumaj and B. Vöcking, Tight bounds for worst-case equilibria. *ACM Trans. Algorithms* **3** (2007) 1–17.

[10] W.L. Eastman, S. Even and I.M. Isaacs, Bounds for the optimal scheduling of $n$ jobs on $m$ processors. *Manage. Sci.* **11** (1964) 268–279.

[11] R.L. Graham, E.L. Lawler, J.K. Lenstra and A.H.G. Rinooy Kan, Optimization and approximation in deterministic sequencing and scheduling: a survey. *Ann. Discrete Math.* **5** (1979) 287–326.

[12] B. Heydenreich, R. Müller and M. Uetz, Games and mechanism design in machine scheduling – an introduction. *Prod. Oper. Manage.* **16** (2007) 437–454.

[13] R. Hoeksma and M. Uetz, The price of anarchy for utilitarian scheduling games on related machines. *Discrete Optim.* **31** (2019) 29–39.

[14] N. Immorlica, L.E. Li, V.S. Mirrokni and A.S. Schulz, Coordination mechanisms for selfish scheduling. *Theor. Comput. Sci.* **410** (2009) 1589–1598.

[15] T. Kawaguchi and S. Kyan, Worst case bound of an LRF schedule for the mean weighted flow-time problem. *SIAM J. Comput.* **15** (1986) 1119–1129.

[16] E. Koutsoupias and C.H. Papadimitriou, Worst-case equilibria. *Comput. Sci. Rev.* **3** (2009) 65–69.

[17] K. Lee, J.Y.-T. Leung and M.L. Pinedo, Coordination mechanisms for parallel machine scheduling. *Eur. J. Oper. Res.* **220** (2012) 305–313.

[18] F.T. Muñoz and A.A. Pinochet, Performance guarantee of the jump neighborhood for scheduling jobs on uniformly related machines. *RAIRO: Oper. Res.* **56** (2022) 1079–1088.

[19] M. Rahn and G. Schäfer, Bounding the inefficiency of altruism through social contribution games, in International Conference on Web and Internet Economics. Springer Berlin Heidelberg, Berlin, Heidelberg (2013) 391–404.

[20] O.L. Rivera, *Cotas para el precio de la anarquía de juegos de Scheduling*. Undergraduate thesis, Universidad de Chile, Chile (2012). https://repositorio.uchile.cl/handle/2250/111965.

[21] W.E. Smith, Various optimizers for single-stage production. *Nav. Res. Logist. Q.* **3** (1956) 59–66.

[22] L. Yu, K. She, H. Gong and C. Yu, Price of anarchy in parallel processing. *Inf. Process. Lett.* **110** (2010) 288–293.

[23] L. Zhang, Y. Zhang, D. Du and Q. Bai, Improved price of anarchy for machine scheduling games with coordination mechanisms. *Optim. Lett.* **13** (2019) 949–959.