Discrete Optimization 9 (2012) 241–248

Contents lists available at SciVerse ScienceDirect

Discrete Optimization

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

An asymptotically optimal online algorithm to minimize the total completion time on two multipurpose machines with unit processing times

Dvir Shabtay *, Shlomo Karhi

Department of Industrial Engineering and Management, Ben-Gurion University of the Negev Beer-Sheva, Israel

ARTICLE INFO

Article history:
Received 4 March 2012
Received in revised form 10 June 2012
Accepted 27 July 2012
Available online 18 August 2012

Keywords:
Multipurpose machine scheduling
Online scheduling
Total completion time
Competitive ratio

ABSTRACT

In the majority of works on online scheduling on multipurpose machines the objective is to minimize the makespan. We, in contrast, consider the objective of minimizing the total completion time. For this purpose, we analyze an online-list scheduling problem of n jobs with unit processing times on a set of two machines working in parallel. Each job belongs to one of two sets of job types. Jobs belonging to the first set can be processed on either of the two machines while jobs belonging to the second set can only be processed on the second machine. We present an online algorithm with a competitive ratio of $\rho_{LB} + O(\frac{1}{n})$, where ρ_{LB} is a lower bound on the competitive ratio of any online algorithm and is equal to $1 + \left(\frac{-\alpha+\sqrt{4\alpha^3-\alpha^2+2\alpha-1}}{2\alpha^2+1}\right)^2$ where $\alpha = \frac{1}{3} + \frac{1}{6}\left(116 - 6\sqrt{78}\right)^{1/3} + \frac{(58+3\sqrt{78})^{1/3}}{3(2)^{2/3}} \approx 1.918$. This result implies that our online algorithm is asymptotically optimal.

1. Introduction

We study an online scheduling problem with unit processing times on a set of two multipurpose machines where the objective is to minimize the total completion time. Our problem can be formally stated as follows. A set of n jobs $\mathcal{J} = \{J_1, J_2, \ldots, J_n\}$ is available at time zero to be processed nonpreemptively on a set of two machines $\mathcal{M} = \{M_1, M_2\}$ working in parallel. The jobs are categorized into two job types according to the set of machines that can process each job. Let $\mathcal{M}_l$ be the set of machines that can process jobs of type l for $l = 1, 2$. We consider the case where $\mathcal{M}_1 = \{M_1, M_2\}$ and $\mathcal{M}_2 = \{M_2\}$. Since machine M_2 can process both job types, it is referred to as a flexible machine, while machine M_1 is referred to as a non-flexible machine. All jobs have the same processing time and, without loss of generality, we assume that processing times are restricted to unity; that is $p_j = 1$ for $j = 1, 2, \ldots, n$ where p_j is the processing time of job J_j. We aim to assign the jobs to the machines such that the *total completion time*, $z = \sum_{j=1}^{n} C_j$, will be minimized, where C_j is the completion time of job J_j for $j = 1, 2, \ldots, n$.

Given an assignment of jobs to machines, let x_i be the number of jobs that have been assigned to machine M_i for $i = 1, 2$. Since all jobs have unit processing time, the completion time of the jth job to be processed on some machine M_i is exactly at time j for $j = 1, \ldots, x_i$. Thus, the total completion time is given by

$$z = \sum_{i=1}^{2}\sum_{j=1}^{x_i} j = \frac{x_1(x_1+1)}{2} + \frac{x_2(x_2+1)}{2}. \quad (1)$$

* Corresponding author. Tel.: +972 8 6461389; fax: +972 8 6472958.
E-mail addresses: dvirs@bgu.ac.il (D. Shabtay), shlomok@post.bgu.ac.il (S. Karhi).

1572-5286/$ – see front matter
doi:10.1016/j.disopt.2012.07.005

We assume that the online version of our problem follows the *online-list* paradigm, where the jobs are ordered in a list and are presented to the scheduler one by one. As soon as a job is presented to the scheduler he knows its type. Then, he has to assign the jobs according to an *online algorithm* where each job has to be irreversibly assigned to some machine before the next job is presented. It is commonly assumed in online scheduling that, in addition to the uncertainty about the job parameters, the scheduler does not know the number of jobs in the list.

In this paper we develop an online algorithm for minimizing the total completion time. In order to evaluate the quality of our online algorithm, the *competitive analysis* evaluation technique presented by Sleator and Tarjan [1] is used. Competitive analysis is a type of worst-case analysis in which the performance of an online algorithm is compared to that of an optimal *offline algorithm*. In *offline scheduling* the scheduler has access to the entire instance of the problem prior to making any scheduling decision. Thus, in our case the scheduler knows in advance both the number of jobs in the list and the type of each job. Therefore, he can apply an *offline algorithm* which takes all the data about the jobs into consideration when making any scheduling decision.

The competitive analysis evaluation technique can be described as follows. Let z be a criterion (objective function) that has to be minimized. For an online Algorithm A, let $z^A(\mathcal{I})$ denote the objective value produced by Algorithm A, for instance $\mathcal{I} \in I$, where I is the set of all possible instances. Further, let OPT be an optimal offline algorithm, and let $z^{\text{OPT}}(\mathcal{I})$ be the corresponding minimum objective value for instance $\mathcal{I}$. We say that Algorithm A is *ρ-competitive* if the condition that $z^A(\mathcal{I}) \leq \rho z^{\text{OPT}}(\mathcal{I})$ holds for any input instance $\mathcal{I} \in I$. Moreover, the *competitive ratio* of Algorithm A denoted by ρ_A is the infimum of ρ such that A is ρ-competitive. According to Sleator and Tarjan, an online scheduling problem has a lower bound ρ_{LB} if no online algorithm has a competitive ratio smaller than ρ_{LB}. Moreover, an online algorithm is called optimal if its competitive ratio matches the lower bound of the problem.

Different variants of the online-list scheduling problem on a set of multipurpose machines have been discussed in the literature (e.g., [2–10]), all of which consider the objective of minimizing the makespan. To the best of our knowledge, our paper is the first to consider the total completion time criterion in the context of online-list scheduling on multipurpose machines. Although our analysis is restricted to two machines with unit processing times, we believe that in future studies these results can be generalized.

The rest of the paper is organized as follows. In Section 2 we determine the optimal assignment of jobs to machines in an offline system. Moreover, we show that the problem has a lower bound of $\rho_{\text{LB}} = 1 + \left(\frac{-\alpha+\sqrt{4\alpha^3-\alpha^2+2\alpha-1}}{2\alpha^2+1}\right)^2$, where $\alpha = \frac{1}{3} + \frac{1}{6}\left(116 - 6\sqrt{78}\right)^{1/3} + \frac{\left(58+3\sqrt{78}\right)^{1/3}}{3(2)^{2/3}} \approx 1.918$. This implies that no online algorithm has a competitive ratio smaller than $\rho_{\text{LB}} \approx 1.1573$. In Section 3 we present an online algorithm with a competitive ratio of $\rho = \rho_{\text{LB}} + O\left(\frac{1}{n}\right)$. This result implies that our online algorithm is asymptotically optimal. A summary section concludes our paper.

2. A lower bound on the competitive ratio

The following lemma provides the minimal total completion time value and the optimal job assignment strategy for the *offline* version of our problem.

Lemma 1. *Given an instance $\mathcal{I} \in I$, the optimal job assignment to machines $x^* = (x_1^*, x_2^*)$ for an offline problem is*

$$x^* = \begin{cases} \left(\left\lfloor \frac{n}{2} \right\rfloor, \left\lceil \frac{n}{2} \right\rceil\right) & \text{if } n_1 \geq n_2 \\ (n_1, n_2) & \text{if } n_1 < n_2, \end{cases} \tag{2}$$

and the minimum value of the total completion time is

$$z^{\text{OPT}}(\mathcal{I}) = \begin{cases} \frac{\lfloor \frac{n}{2} \rfloor \left(\lfloor \frac{n}{2} \rfloor + 1\right)}{2} + \frac{\lceil \frac{n}{2} \rceil \left(\lceil \frac{n}{2} \rceil + 1\right)}{2} & \text{if } n_1 \geq n_2 \\ \frac{n_1 (n_1+1)}{2} + \frac{n_2 (n_2+1)}{2} & \text{if } n_1 < n_2, \end{cases} \tag{3}$$

where for instance $\mathcal{I}$, n_l represents the number of jobs of type l for $l = 1, 2$ in the set of n jobs.

Proof. The offline problem can be solved by minimizing Eq. (1) subject to $x_2 \geq n_2$ and $x_1 + x_2 = n$, where x_i is restricted to be a non-negative integer for $i = 1, 2$. Since $x_1 + x_2 = n$, we can rewrite Eq. (1) as

$$z(x_1) = \frac{x_1(x_1+1)}{2} + \frac{(n-x_1)(n-x_1+1)}{2} = (x_1)^2 - nx_1 + \frac{n(n+1)}{2}. \tag{4}$$

The lemma now follows from the fact that $z(x_1)$ is a convex function whose minimum is at the point where either $x_1 = \left\lceil \frac{n}{2} \right\rceil$ or $x_1 = \left\lfloor \frac{n}{2} \right\rfloor$. □

The next lemma provides a lower bound on the competitive ratio of any online algorithm. The idea behind the proof is to construct a "*hard*" instance which is accomplished in two phases. In the first phase only jobs of type 1 arrive such that any online algorithm will have to assign a portion of jobs to the flexible machine (machine M_2) in order to keep the solution value not too far from the optimal one. Then, in the second phase, only jobs of type 2 arrive all of which have to be assigned to machine M_2. This yields a non-balanced solution in which machine M_2 is overloaded.

Lemma 2. *Any online algorithm is at least* ρ_{LB}*-competitive with*

$$\rho_{\text{LB}} = \left(\frac{-\alpha + \sqrt{4\alpha^3 - \alpha^2 + 2\alpha - 1}}{2\alpha^2 + 1}\right)^2 + 1, \quad \textit{where} \tag{5}$$

$$\alpha = \frac{1}{3} + \frac{1}{6}\left(116 - 6\sqrt{78}\right)^{1/3} + \frac{\left(58 + 3\sqrt{78}\right)^{1/3}}{3(2)^{2/3}} \approx 1.918.$$

This yields that ρ_{LB} *is an irrational number of approximately* 1.1573.

Proof. The proof is by contradiction. Let us assume that there exists an online algorithm A which is ρ'-competitive with $\rho' < \rho_{\text{LB}} \approx 1.1573$. We consider an instance $\mathcal{I}$ of the problem where the first n_1 jobs in the list are of type 1 and n_1 is even such that $\lceil \frac{n_1}{2} \rceil = \lfloor \frac{n_1}{2} \rfloor = \frac{n_1}{2}$. According to Lemma 1, we have that $z^{\text{OPT}}(\mathcal{I}) = \frac{n_1}{2}\left(\frac{n_1}{2} + 1\right)$. Now let x be the number of jobs of type 1 that have been assigned to machine M_2. Since Algorithm A is ρ'-competitive we have that

$$z^A(\mathcal{I}) = \frac{1}{2}\left[(n_1 - x)(n_1 - x + 1) + x(x + 1)\right] \le \rho' \frac{n_1}{2}\left(\frac{n_1}{2} + 1\right),$$

which implies that

$$2x^2 - 2n_1 x + (n_1^2 + n_1 - 0.5\rho' n_1^2 - \rho' n_1) \le 0.$$

It is easy to verify that for $n_1 \to \infty$ the last inequality holds only if

$$\frac{n_1\left(1 - \sqrt{\rho' - 1}\right)}{2} \le x \le \frac{n_1\left(1 + \sqrt{\rho' - 1}\right)}{2}. \tag{6}$$

We now let instance $\mathcal{I}$ be further expanded to include $n_2 = \alpha n_1$ jobs of type 2 ($\alpha \ge 1$), which have to be assigned to machine M_2. Then

$$z^A(\mathcal{I}) = \frac{1}{2}\left[(n_1 - x)(n_1 - x + 1) + (\alpha n_1 + x)(\alpha n_1 + x + 1)\right],$$

and, according to Lemma 1,

$$z^{\text{OPT}}(\mathcal{I}) = \frac{n_1(n_1 + 1)}{2} + \frac{\alpha n_1(\alpha n_1 + 1)}{2} = \frac{1}{2}\left(n_1^2\left(\alpha^2 + 1\right) + n_1(\alpha + 1)\right).$$

Thus, since Algorithm A is ρ'-competitive

$$\frac{1}{2}\left[(n_1 - x)(n_1 - x + 1) + (\alpha n_1 + x)(\alpha n_1 + x + 1)\right] \le \rho' \frac{1}{2}\left(n_1^2\left(\alpha^2 + 1\right) + n_1(\alpha + 1)\right),$$

which further implies that

$$2x^2 + 2n_1 x(\alpha - 1) \le (\rho' - 1)\left(n_1^2\left(\alpha^2 + 1\right) + n_1(\alpha + 1)\right). \tag{7}$$

Combining (7) with (6), the following inequality holds for $n_1 \to \infty$:

$$n_1^2\left(1 - \sqrt{\rho' - 1}\right)^2 + 2n_1^2\left(1 - \sqrt{\rho' - 1}\right)(\alpha - 1) \le 2n_1^2(\rho' - 1)\left(\alpha^2 + 1\right). \tag{8}$$

By a simple mathematical manipulation, we can rewrite the inequality in (8) as

$$f(y) = 2\left(\alpha^2 + 0.5\right)y^2 + 2\alpha y - 2\alpha + 1 \ge 0, \tag{9}$$

where $y = \sqrt{\rho' - 1} \in \Re^+$. It is easy to show that the inequality in (9) holds only for

$$y = \sqrt{\rho' - 1} \ge \frac{-\alpha + \sqrt{4\alpha^3 - \alpha^2 + 2\alpha - 1}}{2\alpha^2 + 1}.$$

Thus,

$$\rho' \geq \left(\frac{-\alpha+\sqrt{4\alpha^3-\alpha^2+2\alpha-1}}{2\alpha^2+1}\right)^2+1=f(\alpha),$$

which contradicts our assumption that $\rho' < \rho = \left(\frac{-\alpha+\sqrt{4\alpha^3-\alpha^2+2\alpha-1}}{2\alpha^2+1}\right)^2+1$. Note that the contradiction holds for any $\alpha \geq 1$. The fact that the maximum of $f(\alpha)$ for $\alpha \geq 1$ is at the point for which $\alpha = \frac{1}{3}+\frac{1}{6}\left(116-6\sqrt{78}\right)^{1/3}+\frac{\left(58+3\sqrt{78}\right)^{1/3}}{3(2)^{2/3}} \approx 1.918$ completes our proof. □

3. An asymptotically optimal online algorithm

Below we present an online algorithm (Algorithm 1) for our problem and prove that it is a ρ-competitive algorithm with $\rho = \rho_{LB} + O\left(\frac{1}{n}\right)$. This algorithm assigns job J_i of type 1 to machine M_1 as long as this assignment does not violate the competitive ratio. Otherwise, it assigns job J_i to machine M_2. To help with the implementation of Algorithm 1, we let $\widetilde{x}_1^i$ and $\widetilde{x}_2^i$ represent the number of jobs assigned to machine M_1 and M_2, respectively, during the course of the algorithm, immediately after job J_i is assigned. Moreover, we let z_i^{OPT} represent the minimal total completion time value for the offline version of the problem. According to Lemma 1,

$$z_i^{OPT} = \begin{cases} \frac{\lfloor \frac{i}{2} \rfloor \left(\lfloor \frac{i}{2} \rfloor + 1\right)}{2} + \frac{\lceil \frac{i}{2} \rceil \left(\lceil \frac{i}{2} \rceil + 1\right)}{2} & \text{if } n_1^i \geq n_2^i \\ \frac{n_1^i(n_1^i+1)}{2} + \frac{n_2^i(n_2^i+1)}{2} & \text{if } n_1^i < n_2^i, \end{cases} \tag{10}$$

where n_l^i represents the number of jobs of type l for $l = 1, 2$ among the first i jobs in the list. The algorithm is formally presented as follows.

Algorithm 1 (*An Online Algorithm for Our Problem*)**.**

Initialization: $\widetilde{x}_1^0 = \widetilde{x}_2^0 = 0$ and $n_1^0 = n_2^0 = 0$.
For $i = 1, \ldots, n$ do:
  Job J_i of type l arrives: set $n_l^i = n_l^{i-1} + 1$ and calculate z_i^{OPT} by (10).
  If job J_i is of type 1 and $\frac{1}{2}\left[\left(\widetilde{x}_1^{i-1}+1\right)\left(\widetilde{x}_1^{i-1}+2\right)+\left(\widetilde{x}_2^{i-1}\right)\left(\widetilde{x}_2^{i-1}+1\right)\right] \leq \rho z_i^{OPT}$
    then assign job J_i to machine M_1 and set $\widetilde{x}_1^i = \widetilde{x}_1^{i-1} + 1$ and $\widetilde{x}_2^i = \widetilde{x}_2^{i-1}$.
  Else, assign job J_i to machine M_2 and set $\widetilde{x}_2^i = \widetilde{x}_2^{i-1} + 1$ and $\widetilde{x}_1^i = \widetilde{x}_1^{i-1}$.
  End if
End for.

The following lemma provides an upper bound to the number of jobs of type 1 that will be assigned to machine M_2 during the implementation of Algorithm 1. This will later help us prove that the algorithm is indeed asymptotically optimal.

Lemma 3. *At most* $\frac{(1-\sqrt{\rho-1})n_1^i}{2}+1$ *jobs of type* 1 *will be assigned to machine* M_2 *at any stage i of* Algorithm 1.

Proof. The proof is by contradiction. Let us assume that more than $\frac{(1-\sqrt{\rho-1})n_1^i}{2}+1$ jobs of type 1 will be assigned to machine M_2. Since among all instances, an instance where all jobs are of type 1 has the maximal number of jobs of type 1 that will be assigned to machine M_2, we restrict our proof to such an instance. Without loss of generality, we may assume that job J_i is of type 1 and has been assigned to machine M_2. This directly implies from Algorithm 1 that

$$\frac{1}{2}\left[(n_1^i-x+1)\left(n_1^i-x+2\right)+(x-1)x\right] > \rho\frac{n_1^i}{2}\left(\frac{n_1^i}{2}+1\right),$$

where x is the number of jobs of type 1 that have been assigned to machine M_2 right after the assignment of job J_i. Thus,

$$f(x) = 2x^2 - 2(n_1^i+2)x + \left(n_1^i\right)^2 + 3n_1^i + 2 - \rho n_1^i\left(\frac{n_1^i}{2}+1\right) > 0. \tag{11}$$

By setting $f(x)$ to zero we obtain that $x_{1,2} = \frac{(n_1^i+2)\pm\sqrt{(\rho-1)\left(n_1^i\right)^2+2(\rho-1)n_1^i}}{2}$. Since $f(x)$ is a convex function, the inequality in (11) holds for when either x is greater than $\frac{(n_1^i+2)+\sqrt{(\rho-1)\left(n_1^i\right)^2+2(\rho-1)n_1^i}}{2}$ or smaller than $\frac{(n_1^i+2)-\sqrt{(\rho-1)\left(n_1^i\right)^2+2(\rho-1)n_1^i}}{2}$. However, since

Algorithm 1 assigns a job of type 1 to machine M_2 only if $\widetilde{x}_1^{i-1} > \widetilde{x}_2^{i-1}$, the number of jobs that will be assigned to machine M_2 is not greater than $n/2 = n_1/2$. This implies that $x < \frac{(n_1^i+2)-\sqrt{(\rho-1)\left(n_1^i\right)^2+2(\rho-1)n_1^i}}{2} \le \frac{(n_1^i+2)-\sqrt{\rho-1}n_1^i}{2} = \frac{(1-\sqrt{\rho-1})n_1^i}{2} + 1$, which contradicts our assumption that more than $\frac{(1-\sqrt{\rho-1})n_1^i}{2} + 1$ jobs of type 1 will be assigned to machine M_2. □

Theorem 1. *Algorithm 1 is a ρ-competitive online algorithm with*

$$\rho = \left(\frac{-\alpha+\sqrt{4\alpha^3-\alpha^2+2\alpha-1}}{2\alpha^2+1}\right)^2 + 1 + O\left(\frac{1}{n}\right) = \rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right), \quad \textit{where} \tag{12}$$

$$\alpha = \frac{1}{3} + \frac{1}{6}\left(116 - 6\sqrt{78}\right)^{1/3} + \frac{\left(58+3\sqrt{78}\right)^{1/3}}{3(2)^{2/3}} \approx 1.918. \tag{13}$$

Proof. The proof is by contradiction. Let us assume that Algorithm 1 is not a ρ-competitive *online* algorithm, with ρ as given by Eqs. (12)–(13). Also, let $\widehat{I}$ be the set of instances for which $z_n^A(\mathcal{I}) > \rho z_n^{\mathrm{OPT}}(\mathcal{I})$ for any $\mathcal{I} \in \widehat{I}$, where $z_n^A(\mathcal{I})$ is the total completion time obtained by applying Algorithm 1 with instance $\mathcal{I}$. Among all instances that belong to set $\widehat{I}$, let $\mathcal{I}_{\min}$ be the one with the smallest number of jobs.

Since Algorithm 1 maintains the condition that $\frac{1}{2}\left[(\widetilde{x}_1^{i-1}+1)\left(\widetilde{x}_1^{i-1}+2\right)+(\widetilde{x}_2^{i-1})\left(\widetilde{x}_2^{i-1}+1\right)\right] \le \rho z_i^{\mathrm{OPT}}$ at any stage i for which job J_i is assigned to machine M_1, we have that for instance $\mathcal{I}_{\min} \in \widehat{I}$, job J_n is assigned to machine M_2. Next, we provide and prove two properties that hold for instance $\mathcal{I}_{\min}$. The first is that $\widetilde{x}_2^{n-1} \ge \widetilde{x}_1^{n-1}$ and the second is that job J_n is of type 2. We later use these properties to prove that $\mathcal{I}_{\min} \notin \widehat{I}$, which will imply that $\widehat{I} = \emptyset$ thereby completing the proof.

Let us first prove that for instance $\mathcal{I}_{\min}$ we have that $\widetilde{x}_2^{n-1} \ge \widetilde{x}_1^{n-1}$. By contradiction, let us assume that for instance $\mathcal{I}_{\min}$ we have that $\widetilde{x}_2^{n-1} < \widetilde{x}_1^{n-1}$. Since jobs of type 2 can only be processed on M_2, the fact that $\widetilde{x}_2^{n-1} < \widetilde{x}_1^{n-1}$ also implies that $n_2^{n-1} < n_1^{n-1}$ and therefore also that $n_2^n \le n_1^n$. Thus, according to Eq. (3),

$$z_{n-1}^{\mathrm{OPT}} = \frac{\left\lfloor \frac{n-1}{2}\right\rfloor\left(\left\lfloor \frac{n-1}{2}\right\rfloor+1\right)}{2} + \frac{\left\lceil \frac{n-1}{2}\right\rceil\left(\left\lceil \frac{n-1}{2}\right\rceil+1\right)}{2}, \tag{14}$$

and

$$z_n^{\mathrm{OPT}} = \frac{\left\lfloor \frac{n}{2}\right\rfloor\left(\left\lfloor \frac{n}{2}\right\rfloor+1\right)}{2} + \frac{\left\lceil \frac{n}{2}\right\rceil\left(\left\lceil \frac{n}{2}\right\rceil+1\right)}{2}. \tag{15}$$

Moreover, the fact that $\mathcal{I}_{\min}$ is the instance with the smallest number of jobs among all instances that belong to set $\widehat{I}$ implies that

$$\frac{1}{2}\left[(\widetilde{x}_1^{n-1})\left(\widetilde{x}_1^{n-1}+1\right)+(\widetilde{x}_2^{n-1})\left(\widetilde{x}_2^{n-1}+1\right)\right] \le \rho z_{n-1}^{\mathrm{OPT}},$$

and the fact that $\mathcal{I}_{\min} \in \widehat{I}$ implies that

$$\frac{1}{2}\left[(\widetilde{x}_1^{n-1})\left(\widetilde{x}_1^{n-1}+1\right)+(\widetilde{x}_2^{n-1}+1)\left(\widetilde{x}_2^{n-1}+2\right)\right] > \rho z_n^{\mathrm{OPT}}.$$

Thus, the following set of inequalities holds for instance $\mathcal{I}_{\min}$:

$$\rho z_n^{\mathrm{OPT}} - (\widetilde{x}_2^{n-1}+1) < \frac{1}{2}\left[(\widetilde{x}_1^{n-1})\left(\widetilde{x}_1^{n-1}+1\right)+(\widetilde{x}_2^{n-1})\left(\widetilde{x}_2^{n-1}+1\right)\right] \le \rho z_{n-1}^{\mathrm{OPT}},$$

and we have that

$$\widetilde{x}_2^{n-1}+1 > \rho\left(z_n^{\mathrm{OPT}} - z_{n-1}^{\mathrm{OPT}}\right) \ge z_n^{\mathrm{OPT}} - z_{n-1}^{\mathrm{OPT}}. \tag{16}$$

According to Eqs. (14)–(15),

$$z_n^{\mathrm{OPT}} - z_{n-1}^{\mathrm{OPT}} = \frac{\left\lfloor \frac{n}{2}\right\rfloor\left(\left\lfloor \frac{n}{2}\right\rfloor+1\right)}{2} + \frac{\left\lceil \frac{n}{2}\right\rceil\left(\left\lceil \frac{n}{2}\right\rceil+1\right)}{2} - \frac{\left\lfloor \frac{n-1}{2}\right\rfloor\left(\left\lfloor \frac{n-1}{2}\right\rfloor+1\right)}{2} - \frac{\left\lceil \frac{n-1}{2}\right\rceil\left(\left\lceil \frac{n-1}{2}\right\rceil+1\right)}{2}.$$

To complete the proof that $\widetilde{x}_2^{n-1} \ge \widetilde{x}_1^{n-1}$, we consider the following two possible cases. The first is where n is odd and the second is where n is even. If n is odd ($n-1$ is even), we have that $\left\lfloor \frac{n-1}{2}\right\rfloor = \left\lceil \frac{n-1}{2}\right\rceil = \left\lfloor \frac{n}{2}\right\rfloor = \frac{n-1}{2}$ and that $\left\lceil \frac{n}{2}\right\rceil = \frac{n+1}{2}$. Thus,

$$z_n^{\mathrm{OPT}} - z_{n-1}^{\mathrm{OPT}} = \frac{\frac{n+1}{2}\left(\frac{n+1}{2}+1\right)}{2} - \frac{\frac{n-1}{2}\left(\frac{n-1}{2}+1\right)}{2} = \frac{n-1}{2} + 1. \tag{17}$$

By inserting Eq. (17) into Eq. (16), we obtain that $\widetilde{x}_2^{n-1} > \frac{n-1}{2}$. Since $\widetilde{x}_1^{n-1} + \widetilde{x}_2^{n-1} = n-1$, we have that $\widetilde{x}_1^{n-1} < \frac{n-1}{2}$ and thus also that $\widetilde{x}_2^{n-1} > \widetilde{x}_1^{n-1}$, which contradicts our assumption that $\widetilde{x}_2^{n-1} < \widetilde{x}_1^{n-1}$. If n is even ($n-1$ is odd) we have that $\lfloor \frac{n-1}{2} \rfloor = \frac{n-2}{2}$ and that $\lceil \frac{n-1}{2} \rceil = \lfloor \frac{n}{2} \rfloor = \lceil \frac{n}{2} \rceil = \frac{n}{2}$. Thus,

$$z_n^{\mathrm{OPT}} - z_{n-1}^{\mathrm{OPT}} = \frac{\frac{n}{2}\left(\frac{n}{2}+1\right)}{2} - \frac{\frac{n-2}{2}\left(\frac{n-2}{2}+1\right)}{2} = \frac{n-2}{2} + 1. \tag{18}$$

By inserting (18) into (16), we obtain that $\widetilde{x}_2^{n-1} > \frac{n-2}{2} = \frac{n}{2} - 1$. The fact that n is even implies further that $\widetilde{x}_2^{n-1} \geq \frac{n}{2}$. Since $\widetilde{x}_1^{n-1} + \widetilde{x}_2^{n-1} = n-1$ we have that $\widetilde{x}_1^{n-1} \leq \frac{n}{2} - 1$ and thus also that $\widetilde{x}_2^{n-1} > \widetilde{x}_1^{n-1}$. This contradicts our assumption that $\widetilde{x}_2^{n-1} < \widetilde{x}_1^{n-1}$ and completes the proof that for instance $\mathcal{I}_{\min}$, $\widetilde{x}_2^{n-1} \geq \widetilde{x}_1^{n-1}$. This result together with the fact that Algorithm 1 assigns a job of type 1 to machine M_2 only if $\widetilde{x}_2^{i-1} < \widetilde{x}_1^{i-1}$ implies that for instance $\mathcal{I}_{\min}$ job J_n is of type 2.

Next, we will use the above two properties ($\widetilde{x}_2^{n-1} \geq \widetilde{x}_1^{n-1}$ and job J_n is of type 2) to prove $\mathcal{I}_{\min} \notin \widehat{I}$. This will result in a contradiction and complete our proof.

For any instance $\mathcal{I}$ of size n, we can represent the objective value obtained by applying Algorithm 1 as

$$\begin{aligned} z_n^A &= \frac{1}{2}\left(\widetilde{x}_1^n\left(\widetilde{x}_1^n+1\right) + \widetilde{x}_2^n\left(\widetilde{x}_2^n+1\right)\right) = \frac{1}{2}\left((n_1^n - x)\left(n_1^n - x + 1\right) + (\alpha n_1^n + x)\left(\alpha n_1^n + x + 1\right)\right) \\ &= \frac{1}{2}\left(2x^2 + 2n_1^n(\alpha-1)x + \left(n_1^n\right)^2\left(\alpha^2+1\right) + n_1^n(\alpha+1)\right), \end{aligned}$$

where $n_2^n = \alpha n_1^n$ and x is the number of jobs of type 1 that have been assigned to machine M_2. Since for instance $\mathcal{I}_{\min}$ we have that $\widetilde{x}_2^{n-1} \geq \widetilde{x}_1^{n-1}$ and that job J_n is of type 2 implies that $\widetilde{x}_2^n = \alpha n_1^n + x > \widetilde{x}_1^n = n_1^n - x$; i.e., that $x > \frac{n_1^n(1-\alpha)}{2}$. Moreover, since z_n^A is a convex function of x with its minimum at the point for which $x = \frac{n_1^n(1-\alpha)}{2}$, setting x to its maximal value maximizes z_n^A. Thus, by Lemma 3,

$$\begin{aligned} 2z_n^A(\mathcal{I}_{\min}) &= 2x^2 + 2n_1^n(\alpha-1)x + \left(n_1^n\right)^2\left(\alpha^2+1\right) + n_1^n(\alpha+1) \\ &\leq 2\left(\frac{\left(1-\sqrt{\rho-1}\right)n_1^n}{2} + 1\right)^2 + 2n_1^n(\alpha-1)\left(\frac{\left(1-\sqrt{\rho-1}\right)n_1^n}{2} + 1\right) + \left(n_1^n\right)^2\left(\alpha^2+1\right) + n_1^n(\alpha+1) \\ &= \frac{\left(n_1^n\right)^2\left(\rho + 2\alpha\left(\alpha + 1 - \sqrt{\rho-1}\right)\right)}{2} + n_1^n\left(3\alpha - 2\sqrt{\rho-1} + 1\right) + 2. \end{aligned} \tag{19}$$

In order to complete the proof that $I_{\min} \notin \widehat{I}$, we now consider two possible cases for instance $\mathcal{I}_{\min}$. The first is where $\alpha > 1$ (i.e., $n_2^n > n_1^n$) and the second is where $\alpha \leq 1$ (i.e., $n_2^n \leq n_1^n$).

(i) According to Eq. (3), if $\alpha > 1$,

$$z_n^{\mathrm{OPT}}(\mathcal{I}_{\min}) = \frac{n_1^n(n_1^n+1)}{2} + \frac{\alpha n_1^n(\alpha n_1^n + 1)}{2} = \frac{1}{2}\left(\left(n_1^n\right)^2\left(\alpha^2+1\right) + n_1^n(\alpha+1)\right). \tag{20}$$

Due to (19) and (20) and the facts that $\mathcal{I}_{\min} \in \widehat{I}$ and $\rho = \rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right)$,

$$\begin{aligned} &\frac{\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right) + 2\alpha\left(\alpha + 1 - \sqrt{\rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right) - 1}\right)\right)}{2} + n_1^n\left(3\alpha - 2\sqrt{\rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right) - 1} + 1\right) + 2 \\ &\geq 2z_n^A(\mathcal{I}_{\min}) > 2\left(\rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right)\right)z_n^{\mathrm{OPT}}(\mathcal{I}_{\min}) = \left(\rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right)\right)\left(\left(n_1^n\right)^2\left(\alpha^2+1\right) + n_1^n(\alpha+1)\right), \end{aligned}$$

which implies that

$$\begin{aligned} &\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}}\left(\alpha^2+1\right) - \frac{\rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right) + 2\alpha\left(\alpha + 1 - \sqrt{\rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right) - 1}\right)}{2}\right) \\ &+ n_1^n\left(\rho_{\mathrm{LB}}(\alpha+1) - 3\alpha - 1 + 2\sqrt{\rho_{\mathrm{LB}} + O\left(\frac{1}{n}\right) - 1}\right) - 2 + O\left(\frac{1}{n}\right)\left(\left(n_1^n\right)^2\left(\alpha^2+1\right) + n_1^n(\alpha+1)\right) < 0, \end{aligned}$$

or that,

$$\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}}\left(\alpha^2+0.5\right)+\alpha\sqrt{\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)-1}-\alpha\left(\alpha+1\right)-\frac{1}{2}O\left(\frac{1}{n}\right)\right)+n_1^n(\alpha+1)\left(\rho_{\mathrm{LB}}-1\right)$$
$$-2n_1^n\left(\alpha-\sqrt{\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)-1}\right)-2+O\left(\frac{1}{n}\right)\left(\left(n_1^n\right)^2\left(\alpha^2+1\right)+n_1^n(\alpha+1)\right)<0.$$

Since $n_1^n(\alpha+1)\left(\rho_{\mathrm{LB}}-1\right)\geq 0$, $2n_1^n\sqrt{\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)-1}\geq 0$ and $O\left(\frac{1}{n}\right)\geq 0$, the last inequality further implies that

$$\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}}\left(\alpha^2+0.5\right)+\alpha\sqrt{\rho_{\mathrm{LB}}-1}-\alpha\left(\alpha+1\right)\right)-\frac{1}{2}\left(n_1^n\right)^2O\left(\frac{1}{n}\right)-2n_1^n\alpha-2$$
$$+O\left(\frac{1}{n}\right)\left(\left(n_1^n\right)^2\left(\alpha^2+1\right)+n_1^n(\alpha+1)\right)<0.$$

Now, let $O\left(\frac{1}{n}\right)=\frac{b}{n}=\frac{b}{n_1^n(\alpha+1)}$, where b is a constant value. Then,

$$\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}}\left(\alpha^2+0.5\right)+\alpha\sqrt{\rho_{\mathrm{LB}}-1}-\alpha\left(\alpha+1\right)\right)-\frac{bn_1^n}{2(\alpha+1)}-2n_1^n\alpha-2$$
$$+\left(\frac{b}{n_1^n(\alpha+1)}\right)\left(\left(n_1^n\right)^2\left(\alpha^2+1\right)+n_1^n(\alpha+1)\right)<0,$$

or

$$\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}}\left(\alpha^2+0.5\right)+\alpha\sqrt{\rho_{\mathrm{LB}}-1}-\alpha\left(\alpha+1\right)\right)+n_1^n\left(\frac{b\left(\alpha^2+1\right)}{(\alpha+1)}-\frac{b}{2(\alpha+1)}-2\alpha\right)+b-2<0.$$

It is easy to show that for $b=1+\sqrt{3}$ we have that $b>2$ and that $\frac{b(\alpha^2+1)}{(\alpha+1)}-\frac{b}{2(\alpha+1)}-2\alpha\geq 0$ for any value of α. This implies that

$$\left(n_1^n\right)^2\left(\left(\rho_{\mathrm{LB}}-1\right)\left(\alpha^2+0.5\right)+\alpha\sqrt{\rho_{\mathrm{LB}}-1}-\alpha+0.5\right)<0.$$

Thus,

$$f(y)=y^2\left(\alpha^2+0.5\right)+\alpha y-(\alpha-0.5)<0, \tag{21}$$

where $y=\sqrt{\rho_{\mathrm{LB}}-1}\in\Re^+$. It is easy to show that the inequality in (21) holds only for

$$\rho_{\mathrm{LB}}<\left(\frac{-\alpha+\sqrt{4\alpha^3-\alpha^2+2\alpha-1}}{2\alpha^2+1}\right)^2+1=\rho_{\mathrm{LB}}, \tag{22}$$

which results in a contradiction and implies that $I_{\min}\notin\widehat{I}$ and thus also that $\widehat{I}=\emptyset$. This further contradicts our assumption that Algorithm 1 is not a ρ-competitive *online* algorithm with ρ as given by Eqs. (12)–(13) for the case where $\alpha>1$.

(ii) According to Eq. (3), if $\alpha\leq 1$,

$$z_n^{\mathrm{OPT}}(\mathcal{I}_{\min})=\frac{\left\lfloor\frac{n}{2}\right\rfloor\left(\left\lfloor\frac{n}{2}\right\rfloor+1\right)}{2}+\frac{\left\lceil\frac{n}{2}\right\rceil\left(\left\lceil\frac{n}{2}\right\rceil+1\right)}{2}\geq\frac{n}{2}\left(\frac{n}{2}+1\right)=\frac{n_1^n(\alpha+1)}{2}\left(\frac{n_1^n(\alpha+1)}{2}+1\right). \tag{23}$$

Moreover, due to (19) and (23) and since $\mathcal{I}_{\min}\in\widehat{I}$ and $\rho=\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)$,

$$\frac{\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)+2\alpha\left(\alpha+1-\sqrt{\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)-1}\right)\right)}{2}+n_1^n\left(3\alpha-2\sqrt{\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)-1}+1\right)+2$$
$$\geq 2z_n^A(\mathcal{I}_{\min})>2\rho z_n^{\mathrm{OPT}}(\mathcal{I}_{\min})=\left(\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)\right)n_1^n(\alpha+1)\left(\frac{n_1^n(\alpha+1)}{2}+1\right),$$

which further implies that

$$\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}}(\alpha+1)^2-\rho_{\mathrm{LB}}+2\alpha\sqrt{\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)-1}-2\alpha\,(\alpha+1)\right)+2n_1^n(\alpha+1)\,(\rho_{\mathrm{LB}}-1)$$
$$-4n_1^n\alpha+4n_1^n\sqrt{\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)-1}-\left(n_1^n\right)^2O\left(\frac{1}{n}\right)+O\left(\frac{1}{n}\right)n_1^n(\alpha+1)\,(n_1(\alpha+1)+2)-4<0.$$

Since $n_1^n\sqrt{\rho_{\mathrm{LB}}+O\left(\frac{1}{n}\right)-1}\geq 0$, $2n_1^n(\alpha+1)\,(\rho_{\mathrm{LB}}-1)>0$ and $O\left(\frac{1}{n}\right)\geq 0$, the last inequality further implies that

$$\left(n_1^n\right)^2\left(\rho_{\mathrm{LB}}\alpha\,(\alpha+2)+2\alpha\sqrt{\rho_{\mathrm{LB}}-1}-2\alpha\,(\alpha+1)\right)$$
$$-4n_1^n\alpha-\left(n_1^n\right)^2O\left(\frac{1}{n}\right)+O\left(\frac{1}{n}\right)n_1^n(\alpha+1)\,(n_1(\alpha+1)+2)-4<0.$$

Now, if we let $O\left(\frac{1}{n}\right)=\frac{b}{n}=\frac{b}{n_1^n(\alpha+1)}$ then

$$\left(n_1^n\right)^2\left((\rho_{\mathrm{LB}}-1)\alpha\,(\alpha+2)-\alpha^2+2\alpha\sqrt{\rho_{\mathrm{LB}}-1}\right)-4n_1^n\alpha-\frac{n_1^nb}{(\alpha+1)}+bn_1^n(\alpha+1)+2b-4<0.$$

It is easy to show that for $b=1+\sqrt{3}$ we have that $2b>4$ and that $b(\alpha+1)-4\alpha-\frac{b}{(\alpha+1)}>0$ for any value of $\alpha<1$. This implies that

$$y^2\,(2+\alpha)+2y-\alpha<0, \tag{24}$$

where $y=\sqrt{\rho_{\mathrm{LB}}-1}\in\Re^+$. It is easy to show that the inequality in (24) holds only for

$$\rho_{\mathrm{LB}}<\left(\frac{-1+\sqrt{1+(2+\alpha)\,\alpha}}{2+\alpha}\right)^2+1=\left(\frac{-1+\sqrt{(\alpha+1)^2}}{2+\alpha}\right)^2+1=\left(\frac{\alpha}{2+\alpha}\right)^2+1=f(\alpha). \tag{25}$$

Since the maximum of $f(\alpha)$ for $\alpha\leq 1$ is at the point for which $\alpha=1$ and is equal to $10/9<1.112$ and $\rho_{\mathrm{LB}}\approx 1.1573$, we have a contradiction. This contradiction implies that $I_{\min}\notin\widehat{I}$ which further implies that $\widehat{I}=\emptyset$ which contradicts our assumption that Algorithm 1 is not a ρ-competitive *online* algorithm with ρ as given by Eqs. (12)–(13) also for the case where $\alpha\leq 1$. □

4. Summary

We study an online list scheduling problem on a set of two multipurpose machines with unit processing times. In our model there are two job types where the first machine can process only jobs of type 1 while the second machine can process both job types. The number of jobs in the list and the type of each job is not known in advance to the scheduler. Our objective is to assign the jobs to the machines such that the total completion time criterion will be minimized. We suggest using an online algorithm which is based on exploiting the concept of machine flexibility where, as long as a desired competitive ratio is not violating, a job is assigned to the less flexible machine that can process it. We prove that the suggested algorithm has a competitive ratio of $\rho_{\mathrm{LB}}+O(\frac{1}{n})$, where ρ_{LB} is a lower bound on the competitive ratio of online algorithm and is equal to $1+\left(\frac{-\alpha+\sqrt{4\alpha^3-\alpha^2+2\alpha-1}}{2\alpha^2+1}\right)^2$, where $\alpha=\frac{1}{3}+\frac{1}{6}\left(116-6\sqrt{78}\right)^{1/3}+\frac{\left(58+3\sqrt{78}\right)^{1/3}}{3(2)^{2/3}}\approx 1.918$.

References

[1] D.D. Sleator, R.E. Tarjan, Amortized efficiency of list update and paging rules, Communications of the ACM 28 (1985) 202–208.
[2] Y. Azar, J. Naor, R. Rom, The competitiveness of on-line assignments, Journal of Algorithms 18 (2) (1995) 221–237.
[3] H.C. Hwang, S.Y. Chang, Y. Hong, A posterior competitiveness for list scheduling algorithm on machines with eligibility constraints, Asia Pacific Journal of Operational Research 21 (1) (2004) 117–125.
[4] Y.W. Jiang, Y. He, C.M. Tang, Optimal online algorithms for scheduling on two identical machines under a grade of service, Journal of Zhejiang University Science A 7 (3) (2006) 309–314.
[5] J. Park, S.Y. Chang, K. Lee, Online and semi-online scheduling of two machines under a grade of service provision, Operations Research Letters 34 (6) (2006) 692–696.
[6] Y. Jiang, Online scheduling on parallel machines with two GoS levels, Journal of Combinatorial Optimization 16 (2008) 28–38.
[7] K. Lim, K. Lee, S.Y. Chang, Improved bounds for online scheduling with eligibility constraints, Theoretical Computer Science 412 (2011) 5211–5224.
[8] D. Shabtay, S. Karhi, Online scheduling of two job types on a set of multipurpose machines with unit processing times, Computers and Operations Research 39 (2) (2012) 405–412.
[9] M. Mandelbaum, D. Shabtay, Scheduling unit length jobs on parallel machines with lookhead information, Journal of Scheduling 14 (4) (2011) 335–350.
[10] A. Zhang, Y. Jiang, Z. Tan, Online parallel machines scheduling with two hierarchies, Theoretical Computer Science 410 (2009) 3597–3605.