Operations Research Letters 43 (2015) 489–494

Contents lists available at ScienceDirect

Operations Research Letters

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

Pseudo lower bounds for online parallel machine scheduling

Zhiyi Tan *, Rongqi Li

Department of Mathematics, Zhejiang University, Hangzhou 310027, PR China

ARTICLE INFO

Article history:
Received 6 December 2014
Received in revised form
7 July 2015
Accepted 8 July 2015
Available online 15 July 2015

Keywords:
Scheduling
Online
Competitive ratio
Lower bound

ABSTRACT

We present pseudo lower bounds for the online scheduling problems on parallel and identical machines, which is the infimum of the competitive ratio of an online algorithm that can be proved by using three lower bounds on the optimum makespan. Pseudo lower bounds for fixed m machines, which match the competitive ratio of the current best algorithm when $m = 4, 5, 6$, are obtained in this paper.

1. Introduction

In this paper we consider classical online scheduling problems on parallel and identical machines. We are given a sequence $\mathcal{J} = \{J_1, J_2, \ldots, J_n\}$ of independent jobs with positive processing times $p_1, p_2, \ldots, p_n$, which should be non-preemptively scheduled on m identical machines $M_1, M_2, \ldots, M_m$. The jobs arrive online in a list, i.e., each job should be assigned to a machine before the next job is revealed. The goal is to minimize the makespan, which is the maximum completion time among the machines.

The quality of the performance of an online algorithm is measured by its *competitive ratio*. For a job sequence $\mathcal{J}$ and an algorithm A, let $C^A(\mathcal{J})$ denote the makespan produced by A and let $C^*(\mathcal{J})$ denote the optimal makespan. Then the competitive ratio of A is defined as $\sup_{\mathcal{J}}\{\frac{C^A(\mathcal{J})}{C^*(\mathcal{J})}\}$. An online scheduling problem has a *lower bound* ρ if no online algorithm has a competitive ratio smaller than ρ. An online algorithm A is called *optimal* if its competitive ratio matches the lower bound of the problem.

In 1966, Graham proposed the algorithm List Scheduling (LS for short) and proved that the competitive ratio of LS is $2 - \frac{1}{m}$ for m machines [7]. List Scheduling simply assigns each job one by one to the machine where it can be completed earliest. Twenty years later, LS was proved to be the optimal algorithm for two and three identical machines [3]. With the algorithm given by Galambos and Woeginger in 1993 which has a smaller worst-case ratio than that of LS for any $m \geq 4$ [5], it is recognized that LS cannot be the optimal algorithm when $m \geq 4$. From then on, algorithms and lower bounds have been refined bit by bit. The current best algorithm with competitive ratio $1 + \sqrt{\frac{1+\ln 2}{2}} \approx 1.9201$ was given by Fleischer and Wahl [4]. The current best lower bound for general number of machines is 1.85358, which is obtained by exhaustive search using computers [6]. When the number of machines is small, Chen et al. [2] presented an algorithm with competitive ratio at most $\max\left\{\frac{4m^2-3m}{2m^2-2}, \frac{2(m-1)^2+\sqrt{1+2m(m-1)}-1}{(m-1)^2+\sqrt{1+2m(m-1)}-1}\right\}$. Chen et al. [2] also gave some lower bounds, which are still the best ones for many fixed and small values of m. For the case of $m = 4$, Rudin III and Chandrasekaran [8] improved the lower bound to $\sqrt{3}$.

Though endless effort has been performed to design optimal algorithms, a gap between lower bounds and competitive ratios of online algorithms still exists, even for the case of $m = 4$. One reason is the following. By the definition of the competitive ratio, the makespan produced by an online algorithm is compared with the optimal makespan. But when proving the competitive ratio, the makespan produced by an online algorithm is in fact compared with the lower bounds for the optimal makespan. All the previous studies of online algorithms only use the following three lower bounds for the optimum makespan [1]:

(i) The total processing time of all jobs divided by m.
(ii) The largest processing time among all jobs.
(iii) The total processing time of the $(km - k + 1)$-st to $(km + 1)$-st largest jobs among all jobs, where $k = 1, \ldots, \lfloor \frac{n-1}{m} \rfloor$.

Taken such difference into consideration, Albers [1] introduced a new measurement to reflect the limit when designing and analyzing the online algorithms, which we call *pseudo lower bound* in this paper. That is, the infimum that the competitive ratio of an

* Corresponding author.
E-mail address: tanzy@zju.edu.cn (Z. Tan).

http://dx.doi.org/10.1016/j.orl.2015.07.002
0167-6377/

algorithm can be proved when using certain lower bounds widely known on the optimum. Albers [1] proved that the pseudo lower bound of online parallel machine scheduling for general number of machines is 1.917, which is much closer to the competitive ratio of the current best online algorithm than the lower bound. It indicates that to find a new lower bound of the optimum makespan is much more urgent and essential than to design an improved algorithm.

In this paper, we adopt the idea of [1], and give the pseudo lower bounds of online parallel machine scheduling for fixed m machines. The pseudo lower bounds for $m = 4, 5, 6$ machines match the competitive ratio of the current best algorithm given in [2]. It follows that as for the case of general number of machines, to obtain optimal algorithms for small number of machines, including $m = 4$, also requires more involved estimation of optimum makespan.

The structure of the paper is as follows. Section 2 presents some preliminary results. The pseudo lower bounds are proved in Section 3.

2. Preliminaries

In this section, we will present some preliminary results which will be used in the construction of the job sequence and proof of the pseudo lower bound.

For any given $m \geq 4$, define

$$f_i^{(m)}(x) = \left(\frac{m-1-x}{m}\right)^i, \quad i = 1, \dots, m, \tag{1}$$

and

$$g_i^{(m)}(x) = \frac{1}{i}\left(\sum_{j=1}^{i} f_j^{(m)}(x) - (m-1-mx)\right), \quad i = 1, \dots, t, \tag{2}$$

where $t = \lfloor \frac{m}{2} \rfloor$. By the definition of $g_i^{(m)}(x)$, it is easy to see that for any $1 \leq i < t$,

$$(i+1)g_{i+1}^{(m)}(x) - ig_i^{(m)}(x) = f_{i+1}^{(m)}(x). \tag{3}$$

By the definition of $f_i^{(m)}(x)$, for any $1 \leq i < m$,

$$\begin{aligned}
\sum_{j=1}^{i} f_j^{(m)}(x) &= \frac{\frac{m-1-x}{m}\left(1 - \left(\frac{m-1-x}{m}\right)^i\right)}{1 - \frac{m-1-x}{m}} \\
&= \frac{\frac{m-1-x}{m} - f_{i+1}^{(m)}(x)}{\frac{1+x}{m}} \\
&= \frac{m}{1+x}\left(\frac{m-1-x}{m} - f_{i+1}^{(m)}(x)\right).
\end{aligned} \tag{4}$$

Substituting (4) into (2), we have

$$\begin{aligned}
g_i^{(m)}(x) &= \frac{1}{i}\left(\frac{m}{1+x}\left(\frac{m-1-x}{m} - f_{i+1}^{(m)}(x)\right) - (m-1-mx)\right) \\
&= \frac{m}{(1+x)i}\left(x^2 - f_{i+1}^{(m)}(x)\right).
\end{aligned} \tag{5}$$

The following two lemmas state some basic properties of $f_i^{(m)}(x)$ and $g_i^{(m)}(x)$.

Lemma 2.1. (i) *For any* $1 \leq i \leq m$, $f_i^{(m)}(x)$ *is a strictly decreasing function on* $[0, 1]$,

(ii) *For any* $1 \leq i \leq t$, $g_i^{(m)}(x)$ *is a strictly increasing function on* $[0, 1]$.

Proof. Clearly, $(f_i^{(m)}(x))' = -\frac{i}{m}\left(\frac{m-1-x}{m}\right)^{i-1}$. Hence, $-1 \leq (f_i^{(m)}(x))' < 0$ for any $1 \leq i \leq m$ and $0 \leq x \leq 1$. Thus

$$(g_i^{(m)}(x))' = \frac{1}{i}\left(\sum_{j=1}^{i}(f_j^{(m)}(x))' + m\right) \geq \frac{1}{i}(-i+m) > 0$$

for any $1 \leq i \leq t$ and $0 \leq x \leq 1$. The lemma follows. □

Lemma 2.2. *For any* $1 \leq i \leq t$, *the equation* $g_i^{(m)}(x) = \frac{1}{2}$ *has a unique solution on* $\left(\frac{5}{7}, \frac{m+t}{2m-t}\right)$.

Proof. Recall that $m \geq 4$. Thus

$$\begin{aligned}
g_i^{(m)}\left(\frac{5}{7}\right) &= \frac{1}{i}\left(\sum_{j=1}^{i} f_j^{(m)}\left(\frac{5}{7}\right) - \left(\frac{2m}{7} - 1\right)\right) \\
&= \frac{1}{i}\left(\sum_{j=1}^{i}\left(\frac{m - \frac{12}{7}}{m}\right)^j - \frac{2m-7}{7}\right) \\
&< \frac{1}{i}\left(i\frac{7m-12}{7m} - \frac{2m-7}{7}\right) \\
&\leq \frac{7m-12}{7m} - \frac{4m-14}{7m} = \frac{3m+2}{7m} \leq \frac{1}{2}.
\end{aligned}$$

Since $g_i^{(m)}(x)$ is strictly increasing and continuous, we only need to show that $g_i^{(m)}\left(\frac{m+t}{2m-t}\right) > \frac{1}{2}$.

By (5),

$$\begin{aligned}
&g_i^{(m)}\left(\frac{m+t}{2m-t}\right) \\
&= \frac{m}{\left(1 + \frac{m+t}{2m-t}\right)i}\left(\left(\frac{m+t}{2m-t}\right)^2 - f_{i+1}^{(m)}\left(\frac{m+t}{2m-t}\right)\right) \\
&= \frac{2m-t}{3i}\left(\left(\frac{m+t}{2m-t}\right)^2 - \left(\frac{m-1-\frac{m+t}{2m-t}}{m}\right)^{i+1}\right) \\
&= \frac{2m-t}{3i}\left(\left(\frac{m+t}{2m-t}\right)^2 - \left(1 - \frac{3}{2m-t}\right)^{i+1}\right).
\end{aligned}$$

Thus $g_i^{(m)}\left(\frac{m+t}{2m-t}\right) > \frac{1}{2}$ is equivalent to

$$\left(1 - \frac{3}{2m-t}\right)^{i+1} < \left(\frac{m+t}{2m-t}\right)^2 - \frac{3i}{2(2m-t)}. \tag{6}$$

It is well known that for any $n \geq 1$ and $0 \leq x \leq 1$,

$$(1-x)^n \leq 1 - nx + \frac{n(n-1)}{2}x^2.$$

Hence,

$$\begin{aligned}
\left(1 - \frac{3}{2m-t}\right)^{i+1} &\leq 1 - (i+1)\frac{3}{2m-t} + \frac{i(i+1)}{2}\left(\frac{3}{2m-t}\right)^2 \\
&= \frac{(m+t)^2 + (3m^2 - 6mt)}{(2m-t)^2} - \frac{3i + (3i+6)}{2(2m-t)} \\
&\quad + \frac{9i(i+1)}{2(2m-t)^2} \\
&= \left(\frac{m+t}{2m-t}\right)^2 - \frac{3i}{2(2m-t)} + \frac{3m^2 - 6mt}{(2m-t)^2} \\
&\quad - \frac{3i+6}{2(2m-t)} + \frac{9i(i+1)}{2(2m-t)^2}.
\end{aligned}$$