*ICIC Express Letters*
*Part B: Applications*
Volume 7, Number 1, January 2016

ICIC International ©2016 ISSN 2185-2766
pp. 181–187

# MINIMIZING THE TOTAL STRETCH IN TWO PARALLEL MACHINES WITH GOS LEVELS

Jang Rae Lee[1] and Suk-Hun Yoon[2,*]

[1]Department of Industrial Engineering
Seoul National University
No. 1, Gwanak-ro, Gwanak-gu, Seoul 08826, Korea
future33@nate.com

[2]Department of Industrial and Information Systems Engineering
Soongsil University
No. 369, Sangdo-ro, Dongjak-gu, Seoul 06978, Korea
*Corresponding author: yoon@ssu.ac.kr

Received June 2015; accepted August 2015

Abstract. *We consider a non-preemptive scheduling problem to minimize the total stretch in two parallel machines with two grade of service (GoS) levels. The stretch of a job is defined as the ratio of the flow time to its processing time. When jobs have different release times, the problem of minimizing the total stretch is NP-complete even for a single machine. In this paper, we assume that the release times of jobs are all zeros. We provide some optimality conditions and show that the problem can be solved in polynomial time.*
**Keywords:** Non-preemptive scheduling, Parallel machines, Total stretch, Grade of service

1. **Introduction.** In service industries, it is a common practice that service providers offer differential services to special customers. For example, in a superstore like Wal-Mart, some counters are reserved for customers with small purchases to accelerate the checkout process, which reduces waiting times of the customers. The stretch is defined as the ratio of a job response time to its processing time, that is, a special case of weighted flowtime, in which the weight is defined by reciprocal of the job processing time. The stretch measure relates the customers' waiting times to their demands. It reflects their psychological expectations that they are willing to wait longer for larger requests [1].

For several decades, grade of service (GoS) models have been studied extensively. Hwang et al. [2] considered parallel machines with job and machine GoS levels to minimize the makespan, where jobs can be processed by lower or the same GoS level machines. They proposed an algorithm with worst case performance of 5/4 and $2 - 1/(m-1)$ for $m = 2$ and $m \geq 3$, respectively. Luo et al. [3] considered semi-online scheduling problems on parallel machines with two GoS levels and unit processing time to minimize the makespan. Lee et al. [4] considered semi-online scheduling problems on parallel machines under GoS eligibility constraints to minimize the makespan. They provided lower bounds of the competitive ratio for any algorithm. Online and semi-online versions of minimizing the makespan in two machines with different GoS levels have been investigated [5-8]. Bender et al. [9] proposed offline and online algorithms to minimize maximum stretch in a single processor. Bender et al. [10] proposed PTAS for minimizing the total stretch. Legrand et al. [11] showed that the minimization problems of the total stretch in single machine without preemption and unrelated parallel machines under the divisible load are NP-complete.

In this paper, we consider minimization of the total stretch in parallel machines. In Section 2, we define our model and present some notations and assumptions. In Section 3, we show that a two-machine problem under two GoS levels is polynomially solvable. Finally, we provide the summary and concluding remarks.

2. **Notations and Problem Definition.** Let $J = \{J_1, J_2, J_3, \cdots, J_n\}$ be the job set. Let $p_j$ be the processing time of job $J_j$, $r_j$ its release time, $a_j$ its starting time, $C_j$ its completion time, and $F_j$ its flow time. Then, the stretch of job $J_j$ is defined by $s_j = (C_j - r_j)/p_j$ or $s_j = F_j/p_j$. Assume that the release times of jobs are all zeros. Therefore, $s_j = C_j/p_j$. In parallel machines with GoS levels, job $J_j$ and machine $M_i$ are labeled with GoS levels $g(J_j)$ and $g(M_i)$, respectively. Job $J_j$ is allowed to be processed on machine $M_i$ only when $g(J_j) \geq g(M_i)$.

We consider two parallel identical machines with two GoS levels, where GoS levels are classified by job processing times. If a job processing time is greater than a constant $c$, the job is entitled to GoS level 1 and otherwise to GoS level 2.

3. **Two Machines with GoS Levels to Minimize Total Stretch.** Let $g(M_i) = i$ and $G_j = \{J_k | g(J_k) = j, k = 1, 2, \cdots, n\}$, for $i$, $j = 1, 2$. Then, jobs in $G_1$ should be processed only on machine $M_1$, but jobs in $G_2$ either on machine $M_1$ or $M_2$.

**Lemma 3.1.** *There exist no unforced idle times on each machine in an optimal schedule.*

**Lemma 3.2.** *An optimal job sequence on each machine is SPT.*

Processing times of jobs in $G_1$ should be larger than in $G_2$. Lemma 3.2 implies that jobs in $G_2$ should be processed before jobs in $G_1$ on machine $M_1$. Partition $G_2$ into three subsets $X$, $Y$ and $B$ such that the starting time of $G_1$ is less than or equal to that of $B$. Locate $X$ before $G_1$ and $Y$ before $B$. Let $a_B$ be a starting time of the first job in $B$. Define $\langle B\rangle$ as the SPT sequence for job set $B$.

**Lemma 3.3.** *In an optimal schedule $\sigma = (\sigma_1, \sigma_2)$, where $\sigma_1 = \langle X \cup G_1\rangle$, $\sigma_2 = \langle Y \cup B\rangle$ and $a_{G_1} \leq a_B$, if $a_{G_1} < a_B$, $\sum_{t\in G_1} \frac{1}{p_t} \geq \sum_{t\in B} \frac{1}{p_t}$.*

**Proof:** By contradiction. Suppose that $\sum_{t\in G_1} \frac{1}{p_t} < \sum_{t\in B} \frac{1}{p_t}$. If we interchange $\langle X\rangle$ and $\langle Y\rangle$, there is no change of stretch for jobs in $X \cup Y$. Then the change of total stretch is calculated as follows.

$$\Delta S = \sum\nolimits_{t\in G_1} \Delta s_t + \sum\nolimits_{t\in B} \Delta s_t = (a_B - a_{G_1}) \left( \sum\nolimits_{t\in G_1} \frac{1}{p_t} - \sum\nolimits_{t\in B} \frac{1}{p_t} \right) < 0.$$

Thus, the schedule is not optimal. This completes the proof.

**Lemma 3.4.** *In an optimal schedule $\sigma = (\sigma_1, \sigma_2)$, where $\sigma_1 = \langle X \cup G_1\rangle$, $\sigma_2 = \langle Y \cup B\rangle$ and $a_{G_1} \leq a_B$, the smallest processing time of jobs in $B$ is greater than or equal to the largest processing time of job in $X \cup Y$.*

**Proof:** Since $\sigma_2 = \langle Y \cup B\rangle$ is SPT, the processing times of jobs in $B$ should be greater than or equal to those of jobs in $Y$.

Let $J_u$ be the last job in $\langle X\rangle$ and $J_v$ be the first job in $\langle B\rangle$. By contradiction. Suppose that $p_u > p_v$. Interchanging jobs $J_u$ and $J_v$ reduced to the stretch change as follows:

$$\Delta S = \sum\nolimits_{t\in X-\{J_u\}} \Delta s_t + \sum\nolimits_{t\in Y} \Delta s_t + \sum\nolimits_{t\in G_1} \Delta s_t + \sum\nolimits_{t\in B-\{J_v\}} \Delta s_t + \Delta s_u + \Delta s_v.$$

Since the stretches of $X - \{J_u\}$ and $Y$ are not changed,

$$\begin{aligned}\Delta S &= (p_v - p_u) \sum\nolimits_{t\in G_1} \frac{1}{p_t} + (p_u - p_v) \sum\nolimits_{t\in B-\{J_v\}} \frac{1}{p_t} + \frac{a_B - a_u}{p_u} + \frac{a_u - a_B}{p_v} \\ &= (p_v - p_u) \left[ \sum\nolimits_{t\in G_1} \frac{1}{p_t} - \sum\nolimits_{t\in B-\{J_v\}} \frac{1}{p_t} \right] + (a_B - a_u) \left[ \frac{1}{p_u} - \frac{1}{p_v} \right].\end{aligned}$$

Since $\sum_{t\in B-\{J_v\}} \frac{1}{p_t} < \sum_{t\in B} \frac{1}{p_t} \le \sum_{t\in G_1} \frac{1}{p_t}$, $\Delta S < 0$.

This completes the proof.

**Lemma 3.5.** *In an optimal schedule $\sigma = (\sigma_1, \sigma_2)$, where $\sigma_1 = \langle X \cup G_1\rangle$, $\sigma_2 = \langle Y \cup B\rangle$ and $a_{G_1} \le a_B$, let the last job in the sequences $\langle X\rangle$ and $\langle Y\rangle$ be $J_s$ and $J_t$. Then, $0 \le \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} \le \min\left\{\frac{1}{p_s}, \frac{1}{p_t}\right\}$.*

**Proof:** By contradiction. Assume that $\sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} > \min\left\{\frac{1}{p_s}, \frac{1}{p_t}\right\}$. If $p_s > p_t$, the total stretch can be reduced by interchanging $J_s$ and $J_t$, and thus the schedule is not optimum. We only consider $p_s \le p_t$.

Case 1. $a_s \ge a_t$.

Move $J_s$ before $J_t$ to construct a new schedule $\sigma' = (\sigma'_1, \sigma'_2)$, where $\sigma'_1 = \langle X - \{J_s\}, G_1\rangle$ and $\sigma'_2 = \langle Y \cup \{J_s\} \cup B\rangle$. Then the change of the total stretch is calculated as below:

$$\begin{aligned}\Delta S &= \sum\nolimits_{k\in G_1} \Delta s_k + \Delta s_s + \Delta s_t + \sum\nolimits_{k\in B} \Delta s_k \\ &= -p_s \sum\nolimits_{k\in G_1} \frac{1}{p_k} + (a_t - a_s)\frac{1}{p_s} + p_s\frac{1}{p_t} + p_s \sum\nolimits_{k\in B} \frac{1}{p_k} \\ &= p_s\left(\frac{1}{p_t} - \sum\nolimits_{k\in G_1} \frac{1}{p_k} + \sum\nolimits_{k\in B} \frac{1}{p_k}\right) + (a_t - a_s)\frac{1}{p_s} < 0.\end{aligned}$$

Case 2. $a_s < a_t$.

Interchange $\langle X\rangle$ with $\langle Y - \{J_t\}\rangle$ to construct a new schedule $\sigma'' = (\sigma''_1, \sigma''_2)$, where $\sigma''_1 = \langle Y - \{J_t\}, G_1\rangle$ and $\sigma''_2 = \langle X \cup \{J_t\} \cup B\rangle$. Then the change of the total stretch is calculated as below:

$$\begin{aligned}\Delta S &= \sum\nolimits_{k\in G_1} \Delta s_k + \Delta s_t + \sum\nolimits_{k\in B} \Delta s_k \\ &= (a_t - a_{G_1}) \sum\nolimits_{k\in G_1} \frac{1}{p_k} + (a_{G_1} - a_t)\frac{1}{p_t} + (a_{G_1} - a_t) \sum\nolimits_{k\in B} \frac{1}{p_k} \\ &= (a_{G_1} - a_t)\left(\frac{1}{p_t} - \sum\nolimits_{k\in G_1} \frac{1}{p_k} + \sum\nolimits_{k\in B} \frac{1}{p_k}\right) < 0.\end{aligned}$$

Thus, the schedule is not optimum. This completes the proof.

**Lemma 3.6.** *Suppose that jobs in $G_1 \cup G_2$ are ordered by the SPT-rule and let $n_1$ be the number of jobs in $G_2$, i.e., $J_{[j]} \in G_2$, $j = 1, 2, \cdots, n_1$ and $J_{[j]} \in G_1$, $j = n_1 + 1, n_2 + 2, \cdots, n$. For an optimal schedule $\sigma = (\sigma_1, \sigma_2)$, where $\sigma_1 = \langle X \cup G_1\rangle$, $\sigma_2 = \langle Y \cup B\rangle$ and $a_{G_1} \le a_B$, a job sequence $B = \left\{J_{[k]}, J_{[k+1]}, \cdots, J_{[n_1]}\right\}$, where $k = \arg\min_t \left\{\sum_{i=t}^{n_1} \frac{1}{p_{[i]}} \le \sum_{j\in G_1}^{n} \frac{1}{p_j}\right\}$ satisfies $0 \le \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} < \min\left\{\frac{1}{p_s}, \frac{1}{p_t}\right\}$, where $p_s$ and $p_t$ are the processing times of the last job in sequences $\langle X\rangle$ and $\langle Y\rangle$, respectively.*

**Algorithm *SPT-LS***

Step 1. Construct a list of jobs by the SPT-rule.

Step 2. Assign a job to the least loaded machine according to the list.

**Algorithm *GoS[c]***

On parallel identical machines with two GoS levels,

Step 1. Initialization

Sort jobs with GoS level 1 by the SPT-rule and label the set of jobs by $G_1$. Likewise, sort jobs with GoS level 2 by the SPT-rule and label the set of jobs by $G_2$.

Let $n$ be the number of jobs and $n_1$ be the number of jobs in $G_2$.

Step 2. Select jobs in $G_2$ for $B$.

(a) Calculate $\sum_{j\in G_1} \frac{1}{p_j}$.

(b) For jobs in $G_2$, let $k = \arg\min_t \left\{\sum_{i=t}^{n_1} \frac{1}{p_{[i]}} \le \sum_{j\in G_1} \frac{1}{p_j}\right\}$.

(c) Set $B = \left\{J_{[k]}, J_{[k+1]}, \cdots, J_{[n_1]}\right\}$.
Step 3. Order jobs for Machines $M_1$ and $M_2$.
(a) If $G_2 - B$ is empty, go to Step 4.
(b) If the number of jobs in $G_2 - B$ is odd, assign the first job in $G_2 - B$ to machine $M_2$ and apply Algorithm SPT-LS for the remaining jobs in $G_2 - B$ until there is no job in $G_2 - B$.
(c) If the number of jobs in $G_2 - B$ is even, assign the first job in $G_2 - B$ to machine $M_1$ and apply Algorithm SPT-LS for the remaining jobs in $G_2 - B$ until there is no job in $G_2 - B$.
(d) Label jobs for machine $M_1$ by $X$ and jobs for machine $M_2$ by $Y$.
Step 4. Sequence jobs for machines $M_1$ and $M_2$. Attach sequence $\langle G_1\rangle$ after $\langle X\rangle$ and sequence $\langle B\rangle$ after $\langle Y\rangle$.

**Numerical Example.** Consider two identical machines with two GoS levels. Assume that if a job processing time is greater than 3, the job is entitled to GoS level 1 and otherwise to GoS level 2. The processing times and GoS levels of jobs are provided in the following table.

TABLE 1. The processing times and GoS levels of jobs

| **Job** | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ | $J_7$ | $J_8$ | $J_9$ | $J_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $p_j$ | 4 | 1 | 3 | 1 | 5 | 3 | 2 | 2 | 4 | 1 |
| $g(J_j)$ | 1 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 1 | 2 |

Step 1. Set $G_1 = \{J_1, J_9, J_5\}$, $G_2 = \{J_2, J_4, J_{10}, J_7, J_8, J_3, J_6\}$, $n = 10$, and $n_1 = 7$.
Step 2. Since $\sum_{j\in G_1} \frac{1}{p_j} = 0.7$, $k = \min_t \left\{\sum_{i=t}^{7} \frac{1}{p_{[i]}} \leq 0.7\right\} = 6$.
Therefore, $B = \left\{J_{[6]}, J_{[7]}\right\} = \{J_3, J_6\}$.
Step 3. Since the number of jobs in $G_2 - B$ is odd, we assign first job in $G_2$, i.e., $J_{[1]} = J_2$, to machine $M_2$ and remaining jobs in $G_2 - B$ to two machines by Algorithm SPT-LS.
Then $X = \{J_4, J_7\}$ and $Y = \{J_2, J_{10}, J_8\}$.
Step 4. Attach sequence $\langle G_1\rangle$ after $\langle X\rangle = \langle J_4, J_7\rangle$ and attach sequence $\langle B\rangle$ after $\langle Y\rangle = \langle J_2, J_{10}, J_8\rangle$. The optimal sequence is obtained in Figure 1.

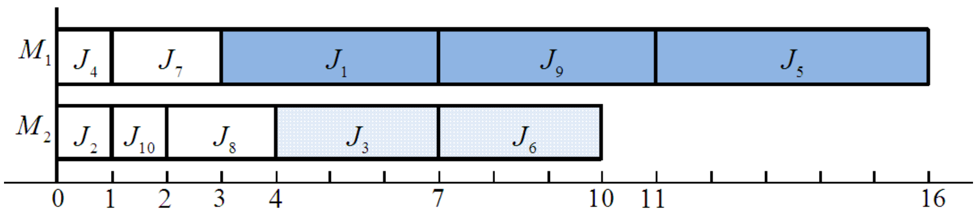


FIGURE 1. Optimal sequence

**Theorem 3.1.** *Algorithm GoS[c] is optimal for* $P2|GoS[c]|\sum_{j=1}^{n} s_j$.

**Proof:** Let $\sigma = (\sigma_1, \sigma_2)$ be a schedule constructed by Algorithm $GoS[c]$, where $\sigma_1 = \langle X \cup G_1\rangle$, $\sigma_2 = \langle Y \cup B\rangle$ and $a_{G_1} \leq a_B$. Suppose $\sigma = (\sigma_1, \sigma_2)$ is not an optimal schedule. Since the jobs in each machine are ordered by SPT, the total stretch cannot be improved by interchanging two jobs in the same machine. Thus, we consider only interchanging two jobs in the different machines. Let $J_{i,[k]}$ be the $k$th job in the sequence $\sigma_i$, $i = 1, 2$. Let $p_{i,[k]}$ and $a_{i,[k]}$ be the processing time and starting time of job $J_{i,[k]}$, respectively.

Interchange $J_{1,[u]}$ and $J_{2,[v]}$, where $1 \leq u \leq |X|$, $1 \leq v \leq |Y \cup B|$. If $p_{1,[u]} = p_{2,[v]}$, the total stretch is not changed. Thus, we assume that $p_{1,[u]} \neq p_{2,[v]}$.

Then, the change of total stretch is calculated as below:

$$\Delta S = \Delta s_{1,[u]} + \Delta s_{2,[v]} + \sum_{k=u+1}^{|X \cup G_1|} \Delta s_{1,[k]} + \sum_{k=v+1}^{|Y \cup B|} \Delta s_{2,[k]}$$

Case 1. $J_{2,[v]} \in B$. Note that $u < v$, $p_{1,[u]} < p_{2,[v]}$, $a_{1,[u]} < a_{2,[v]}$.

$$\begin{aligned}\Delta S &= \left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) + \left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k=u+1}^{|X \cup G_1|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y \cup B|} \frac{1}{p_{2,[k]}}\right) \\ &> \left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) + \left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k}\right)\end{aligned}$$

Since $\left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) > 0$ and $\left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k}\right) \geq 0$, $\Delta S > 0$.

Case 2. $J_{2,[v]} \in Y$.

$$\begin{aligned}\Delta S &= \Delta s_{1,[u]} + \Delta s_{2,[v]} + \sum_{k=u+1}^{|X|} \Delta s_{1,[k]} + \sum_{k=v+1}^{|Y|} \Delta s_{2,[k]} + \sum_{k \in G_1} \Delta s_k + \sum_{k \in B} \Delta s_k \\ &= \left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) \\ &\quad + \left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k}\right)\end{aligned}$$

In Algorithm *GoS[c]*, if the number of remaining jobs for $X \cup Y$ is even, jobs are assigned first to $X$, and otherwise first to $Y$.

Case 2.1. $|X| = |Y|$. Note that $p_{1,[k-1]} \leq p_{2,[k-1]} \leq p_{1,[k]}$, $k = 2, 3, \cdots, |X|$.

Case 2.1.1. $u > v$. Note that $a_{1,[u]} > a_{2,[v]}$, $p_{1,[u]} > p_{2,[v]}$.

$$\begin{aligned}&\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k} \\ &= \sum_{k=u+1}^{|X|} \left(\frac{1}{p_{1,[k]}} - \frac{1}{p_{2,[k-1]}}\right) - \sum_{k=v+1}^{u-1} \frac{1}{p_{2,[k]}} - \frac{1}{p_{2,|Y|}} + \sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k} < 0,\end{aligned}$$

since $\sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k} < \frac{1}{p_{2,|Y|}}$ by Lemma 3.6.

Thus, $\Delta S > 0$, since $\left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k}\right) > 0$ and $(a_{2,[v]} - a_{1,[u]})\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) > 0$.

Case 2.1.2. $u = v$. Note that $a_{1,[u]} \leq a_{2,[v]}$, $p_{1,[u]} < p_{2,[v]}$.

$$\begin{aligned}&\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k} \\ &= \sum_{k=u+1}^{|X|} \left(\frac{1}{p_{1,[k]}} - \frac{1}{p_{2,[k]}}\right) + \sum_{k \in G_1} \frac{1}{p_k} - \sum_{k \in B} \frac{1}{p_k} \geq 0\end{aligned}$$

Thus, $\Delta S \geq 0$, since $\left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\right) \geq 0$ and $\left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) \geq 0$.

Case 2.1.3. $u < v$. Note that $a_{1,[u]} < a_{2,[v]}$, $p_{1,[u]} < p_{2,[v]}$.

$$\begin{aligned}&\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\\ &= \sum_{k=u+1}^{v} \frac{1}{p_{1,[k]}} + \sum_{k=v+1}^{|X|}\left(\frac{1}{p_{1,[k]}} - \frac{1}{p_{2,[k]}}\right) + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} > 0.\end{aligned}$$

Thus, $\Delta S > 0$, since $\left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\right) > 0$ and $\left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) > 0$.

Case 2.2. $|X| = |Y| - 1$. Note that $p_{1,[k]} \leq p_{2,[k+1]} \leq p_{1,[k+1]}$, $k = 1, 2, \cdots, |X| - 1$.

Case 2.2.1. $u > v$ Note that $a_{1,[u]} > a_{2,[v]}$, $p_{1,[u]} > p_{2,[v]}$.

$$\begin{aligned}&\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\\ &= \sum_{k=u+1}^{|X|}\left(\frac{1}{p_{1,[k]}} - \frac{1}{p_{2,[k]}}\right) - \sum_{k=v+1}^{u} \frac{1}{p_{2,[k]}} - \frac{1}{p_{2,|Y|}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} < 0,\end{aligned}$$

since $\sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} < \frac{1}{p_{2,|Y|}}$ by Lemma 3.6.

Thus, $\Delta S > 0$, since $\left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\right) > 0$ and $\left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) > 0$.

Case 2.2.2. $u = v$. Note that $a_{1,[u]} \geq a_{2,[v]}$, $p_{1,[u]} > p_{2,[v]}$.

$$\begin{aligned}&\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\\ &= \sum_{k=u+1}^{|X|}\left(\frac{1}{p_{1,[k]}} - \frac{1}{p_{2,[k]}}\right) - \frac{1}{p_{2,|Y|}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} < 0,\end{aligned}$$

since $\sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} < \frac{1}{p_{2,|Y|}}$ by Lemma 3.6.

Thus, $\Delta S > 0$, since $\left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\right) > 0$ and $\left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) \geq 0$.

Case 2.2.3. $u < v$. Note that $a_{1,[u]} < a_{2,[v]}$, $p_{1,[u]} < p_{2,[v]}$.

$$\begin{aligned}&\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\\ &= \sum_{k=u+1}^{v-1} \frac{1}{p_{1,[k]}} + \sum_{k=v}^{|X|}\left(\frac{1}{p_{1,[k]}} - \frac{1}{p_{2,[k+1]}}\right) + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k} \geq 0.\end{aligned}$$

Thus, $\Delta S > 0$, since $\left(p_{2,[v]} - p_{1,[u]}\right)\left(\sum_{k=u+1}^{|X|} \frac{1}{p_{1,[k]}} - \sum_{k=v+1}^{|Y|} \frac{1}{p_{2,[k]}} + \sum_{k\in G_1} \frac{1}{p_k} - \sum_{k\in B} \frac{1}{p_k}\right) \geq 0$ and $\left(a_{2,[v]} - a_{1,[u]}\right)\left(\frac{1}{p_{1,[u]}} - \frac{1}{p_{2,[v]}}\right) > 0$.

For all cases, the total stretch cannot be decreased. This completes the proof.

4. **Conclusions.** We consider a minimization problem of the total stretch in parallel machines, in which release times of jobs are assumed zeros. This is a special case of the NP-hard total weighted completion times when the weights are the inverse of job processing times. For the case of two machines with two GoS levels where the levels are determined by job size, we propose a polynomial time algorithm.

## REFERENCES

[1] S. Muthukrishnan, R. Rajaraman, A. Shaheen and J. Gehrke, Online scheduling to minimize average stretch, *SIAM Journal on Scientific Computing*, vol.34, no.2, pp.433-452, 2005.

[2] H. C. Hwang, S. Y. Chang and K. Lee, Parallel machine scheduling under a grade of service provision, *Computers and Operations Research*, vol.31, pp.2055-2061, 2004.

[3] T. Luo, Y. Xu, L. Luo and C. He, Semi-online scheduling with two GoS levels and unit processing time, *Theoretical Computer Science*, vol.521, pp.62-72, 2014.

[4] K. Lee, H. C. Hwang and K. Lim, Semi-online scheduling with GoS eligibility constraints, *International Journal of Production Economics*, vol.153, pp.204-214, 2014.

[5] Y. W. Jiang, Online scheduling on parallel machines with two GoS levels, *Journal of Combinatorial Optimization*, vol.16, pp.28-38, 2008.

[6] J. Park, S. Y. Chang and K. Lee, Online and semi-online scheduling of two machines under a grade of service provision, *Oper. Res. Lett.*, vol.34, no.6, pp.692-696, 2006.

[7] Y. W. Jiang, Y. He and C. M. Tang, Optimal online algorithms for scheduling on two identical machines under a grade of service, *Journal of Zhejiang University Science A*, vol.7, no.3, pp.309-314, 2006.

[8] X. Lu and Z. Liu, Semi-online scheduling problems on two uniform machines under a grade of service provision, *Theoretical Computer Science*, vol.489-490, pp.58-66, 2013.

[9] M. A. Bender, S. Chakrabarti and S. Muthukrishnan, Flow and stretch metrics for scheduling continuous job streams, *Proc. of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp.270-279, 1998.

[10] M. A. Bender, S. Muthukrishnan and R. Rajaraman, Improved algorithms for stretch scheduling, *Proc. of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp.762-771, 2002.

[11] A. Legrand, A. Su and F. Vivien, Minimizing the stretch when scheduling flows of divisible requests, *Journal of Scheduling*, vol.11, pp.381-404, 2008.