Journal of Scheduling **8:** 49–74, 2005.

SCHEDULING PARALLEL MACHINES FOR THE CUSTOMER ORDER PROBLEM

JAEHWAN YANG[1] AND MARC E. POSNER[2]

[1] *Department of Business Administration, University of Incheon, 177 Dohwa-dong, Nam-gu, Incheon, 402-749, Republic of Korea*
[2] *The Ohio State University, Industrial Engineering, 1971 Neil Avenue, Columbus, Ohio 43210*

ABSTRACT

This paper considers scheduling problems where jobs are dispatched in batches. The objective is to minimize the sum of the completion times of the batches. While a machine can process only one job at a time, multiple machines can simultaneously process jobs in a batch. This simple environment has a variety of real world applications such as part kitting and customer order scheduling.

A heuristic is presented for the parallel machine version of the problem. Also, a tight worst case bound on the relative error is found. For the case of two parallel machines, we examine two heuristics, which are based on simple scheduling rules. We find tight worst case bounds of 6/5 and 9/7 on the relative error and show that neither procedure is superior for all instances. Finally, we empirically evaluate these two heuristics. For large problems, the methods find solutions that are close to optimal.

KEY WORDS: batch scheduling, kitting, customer order scheduling, heuristic analysis

1. INTRODUCTION

This work considers a scheduling problem where each job is part of some batch (customer order). The composition of the jobs in the batch is prespecified. While a machine can process only one job at a time, multiple machines can simultaneously process jobs in the batch. The completion time of the batch is the latest completion time of any job in the batch. The objective is to minimize the sum of the batch completion times. In a production shop where jobs are dispatched by batches, this objective minimizes total Work in Process (WIP).

Various practical problems motivate our research. One example is scheduling customer orders. Consider a manufacturing facility which produces different types of products. A customer can request a variety of products in an order. After the entire order is produced, the products are shipped to the customer. Each order is a batch and a product is a job. The composition of the batch is specified by the order.

Another application of our research is in the production of components for subsequent assembly (part kitting). Consider a production process with two stages, say fabrication and assembly. Final products are assembled from the various components. The components are produced by machines in the fabrication stage. The components required to produce one unit of a final product form a batch. Assembly of a product can begin only when all required components complete the fabrication stage.

Correspondence to: Marc E. Posner. E-mail: posner.l@osu.edu

A final example is crane scheduling at a port. The importance of the problem is noted by Peterkofsky and Daganzo (1990). They report that a typical cargo ship spends 60% of its time in port, costing about $1000 per hour. Each ship has several holds, and each hold requires either loading or unloading. A crane represents a machine, and a ship and a hold can be considered as a batch and a job, respectively. At most one crane can load or unload one hold at a time. Since there exist multiple holds on a ship, more than one crane can simultaneously work on a ship. A ship stays at berth until all work on the ship completes. The objective is to minimize the total time that the ships spend at port.

For a comprehensive survey of batch scheduling problems, see Jordan (1996). Also, reviews of batching and lot-sizing decision problems are given by Potts and Van Wassenhove (1992) and Webster and Baker (1995).

In our more restricted problem, the completion time is based on the latest completion time of a job in the batch. Julien and Magazine (1990) examine a single machine problem where the objective is to minimize the total completion time of the batches. A job-dependent setup time is incurred between two different types of jobs. They develop a polynomial time dynamic programming (DP) algorithm for the problem when there are two types of jobs and when the batch processing order is fixed. Coffman, Nozari, and Yannakakis (1989) examines a similar problem where the batch processing order is not fixed. Baker (1988) considers a problem similar to Coffman, Nozari, and Yannakakis (1989). However, for one type of job, those jobs processed during the same production run (setup) are not available until the completion of the production run. This restriction is called *batch availability* (see Santos and Magazine, 1985). Gupta, Ho, and van der Veen (1997) consider the single machine problem where each order must have one job from each of several job classes. Also, there is a setup time whenever the job class changes. Gerodimos, Glass, and Potts (2000) study single machine problems where each batch has one common job and one distinct job.

Daganzo (1989) and Peterkofsky and Daganzo (1990) consider a crane scheduling problem. Both papers consider an open shop problem with identical machines, where jobs are preemptable. The objective function is to minimize the sum of weighted batch tardiness. Daganzo (1989) develops a heuristic procedure and finds an optimal solution for some special cases. Peterkofsky and Daganzo (1990) develop a branch and bound solution procedure.

The batch scheduling problems that we discuss are simpler than most types of batch problems because no setup times exist between different jobs or different batches. Blocher and Chhajed (1996) examine the problem considered in this work, minimizing the sum of batch completion times in a parallel machine environment. They show that the problem is NP-hard, develop several heuristic methods, and two lower bounds. Blocher, Chhajed, and Leung (1998) generalize this work to consider dispatching rules in a job shop environment.

We first introduce some notation. Next, we study the problem $P\|\sum C_{B_i}$, where C_{B_i} is the completion time of batch B_i. We develop a tight worst case bound for a heuristic procedure. Next, for problem $P2\|\sum C_{B_i}$ we present a new heuristic and find a tight worst case bound of $9/7$. We find a tight worse case bound of $6/5$ for one of the heuristics of Blocher and Chhajed (1996). We show that neither heuristic is always superior. Then, we empirically evaluate these two heuristics. For problems with approximately 2500 jobs, the average relative error is less than 0.001. Also, we find classes of problems where the $9/7$ heuristic may be preferable. Finally, some open problems and future research are discussed.

2. NOTATION

The decision variables in our models are

σ_k	=	schedule of all jobs on machine k for $k \in M$
σ	=	schedule of all jobs $= (\sigma_1, \sigma_2, \ldots, \sigma_m)$.

Other notation that is used in this work include

n	=	number of jobs
N	=	set of jobs $= \{1, 2, \ldots, n\}$
b	=	number of batches
B	=	set of batches $= \{1, 2, \ldots, b\}$
n_i	=	number of jobs in batch i for $i \in B$
B_i	=	set of jobs in batch i for $i \in B = \{\sum_{j=1}^{i-1} n_j + 1, \sum_{j=1}^{i-1} n_j + 2, \ldots, \sum_{j=1}^{i} n_j\}$
$\beta_i(\sigma)$	=	first job selected for processing in B_i for $i \in B$
m	=	number of machines
M	=	set of machines $= \{1, 2, \ldots, m\}$
p_j	=	processing time of job j for $j \in N$
P_i	=	$\sum_{j\in B_i} p_j =$ total processing time of batch $i \in B$
$C_i(\sigma_k)$	=	completion time of batch i on machine k for $i \in B$ and $k \in M$
$C_i(\sigma)$	=	completion time of batch i in schedule σ for $i \in B = \max_{k\in M} C_i(\sigma_k)$
z^*	=	value of optimal schedule.

We represent $\beta_i(\sigma)$ as β_i and $C_i(\sigma)$ as C_i when there is no ambiguity.

The standard classification scheme for scheduling problems (Graham et al., 1979) is $\alpha_1|\alpha_2|\alpha_3$, where α_1 describes the machine structure, α_2 gives the job characteristics or restrictive requirements, and α_3 defines the objective function to be minimized. We extend this scheme to provide for batch completion times by using C_{B_i} in the α_3 field. This notation is used to eliminate the confusion between our problem and the classical scheduling problem.

All of the heuristic procedures we develop use the following rule to determine the order in which the batches are processed.

SB (Shortest Batch): When a machine becomes available, an unscheduled job in the batch with a shortest total processing time is selected first for processing.

3. PROBLEM $P\|\sum C_{B_i}$

In this section, we construct a heuristic that has a worst case bound on the relative error of $2 - 1/m$. First we establish some properties of an optimal schedule.

Since there are no restrictions that delay jobs, we have the following result.

Remark 1. For problem $P\|\sum C_{B_i}$, there exists an optimal schedule without inserted idle time.

We say that batch $i \in B$ is *separated* if on some machine $k \in M$, jobs in batch i are not processed consecutively.

Lemma 1 (Blocher and Chhajed, 1996). *For problem $P\|\sum C_{B_i}$, there exists an optimal schedule where no batch is separated.*

As a result of Remark 1 and Lemma 1, we only consider those schedules where there is no inserted idle time and where batches are not separated.

Heuristic H1 uses the SB rule to determine the batch sequence. To determine the job-machine assignment, an LS (List Scheduling) rule is used. For this rule, when a machine becomes available, an unscheduled job with the smallest index is selected for processing. We assume that the order of the jobs is arbitrary.

Heuristic H1

0. Reindex the batches so that $P_i \leq P_{i+1}$ for $i = 1, 2, \ldots, b-1$.
   Set $i = j = 1$ and $F_k = 0$ for $k = 1, 2, \ldots, m$.
1. Select the first available machine $u = \text{argmin}_{k=1,2,\ldots,m}\{F_k\}$. Assign job j in batch i to machine u. Set $F_u = F_u + p_j$ and $j = j + 1$.
   Repeat Step 1 until all jobs in batch i are scheduled.
2. Set $C_i = \max_{k=1,2,\ldots,m}\{F_k\}$.
   If $i < b$, then set $i = i + 1$ and go to Step 1.
   Otherwise, output $\Sigma_{i=1}^{b} C_i$ and stop.

In Step 0, reindexing the batches requires O($b \log b$) time. By storing $F_1, F_2, \ldots, F_m$ in a heap in Step 1, selecting F_u requires O($n_i \log m$) time for $i = 1, 2, \ldots, b$. Since all other operations require O(n) time, the time requirement of H1 is O($n \log m + b \log b$).

Before we establish the worst case relative error of H1, we provide a lower bound for the optimal cost based on linear programming. Assume that the batches are indexed so that $P_1 \leq P_2 \leq \cdots \leq P_b$.

Lemma 2 (Blocher and Chhajed, 1996). *For problem $P\|\sum C_{B_i}$,*

$$\frac{bP_1}{m} + \frac{(b-1)P_2}{m} + \cdots + \frac{P_b}{m} \leq z^*.$$

Consider the Linear Programming (LP) relaxation of problem $P\|\sum C_{B_i}$ where a job can be split into pieces of any size and processed, simultaneously if desired, on multiple machines. Observe that an optimal LP solution, z^{LP}, divides each batch into m equal processing times and then processes them in index order. Consequently,

$$z^{LP} = \frac{bP_1}{m} + \frac{(b-1)P_2}{m} + \cdots + \frac{P_b}{m}.$$

The next theorem establishes a worst case bound for the solution value found by Heuristic H1, z^{H1}.

Theorem 1. For problem $P\|\sum C_{B_i}$, $z^{H1}/z^* \leq 2 - 1/m$ and this bound is tight.

Proof. Let σ^* and σ^{H1} be an optimal schedule and the schedule produced by H1, respectively. Suppose that for batch $v \in B$, job $\ell \in B_v$ finishes last at time $C_v(\sigma^{H1})$. Then, $C_v(\sigma^{H1}) \leq C_v(\sigma_k^{H1}) +$

p_ℓ for all $k = 1, 2, \ldots, m$, because job ℓ is assigned to the first available machine. Thus,

$$
\begin{aligned}
C_v(\sigma^{H1}) &\leq \frac{\sum_{i=1}^{v} P_i - \max_{j \in B_v}\{p_j\}}{m} + \max_{j \in B_v}\{p_j\} \\
&= \frac{\sum_{i=1}^{v} P_i}{m} + \left(\frac{m-1}{m}\right) \max_{j \in B_v}\{p_j\}. \qquad (1)
\end{aligned}
$$

Then,

$$
\begin{aligned}
\left(2 - \frac{1}{m}\right) z^* &= z^* + \left(\frac{m-1}{m}\right) \sum_{i=1}^{b} C_i(\sigma^*) \\
&\geq \frac{P_1}{m} + \frac{P_1 + P_2}{m} + \cdots + \frac{\sum_{i=1}^{b} P_i}{m} \\
&\quad + \left(\frac{m-1}{m}\right)\left(\max_{j \in B_1}\{p_j\} + \max_{j \in B_2}\{p_j\} + \cdots + \max_{j \in B_b}\{p_j\}\right) \\
&= \left[\frac{P_1}{m} + \left(\frac{m-1}{m}\right) \max_{j \in B_1}\{p_j\}\right] + \left[\frac{P_1 + P_2}{m} + \left(\frac{m-1}{m}\right) \max_{j \in B_2}\{p_j\}\right] \\
&\quad + \cdots + \left[\frac{\sum_{i=1}^{b} P_i}{m} + \left(\frac{m-1}{m}\right) \max_{j \in B_b}\{p_j\}\right] \\
&\geq C_1(\sigma^{H1}) + C_2(\sigma^{H1}) + \cdots + C_b(\sigma^{H1}) \\
&= z^{H1}.
\end{aligned}
$$

The first inequality follows from Lemma 2, and the second inequality follows from (1).

We now show that this bound is tight by using an example from Graham (1966). Consider an instance of the problem where $n = 2m - 1$ and $b = 1$. Also, $p_j = m - 1$ for $j = 1, 2, \ldots, m - 1$, $p_j = 1$ for $j = m, m+1, \ldots, 2m-2$, and $p_{2m-1} = m$. Now, $\sigma_1^{H1} = (1, 2m-1)$, $\sigma_k^{H1} = (k)$ for $k = 2, 3, \ldots, m-1$, and $\sigma_m^{H1} = (m, m+1, \ldots, 2m-2)$. The solution value $z^{H1} = 2m - 1$. However, $\sigma_k^* = (m + k - 1, k)$ for $k = 1, 2, \ldots, m-1$ and $\sigma_m^* = (2m-1)$. The optimal solution value is m. Thus, $z^{H1}/z^* = (2m-1)/m = 2 - 1/m$, and the bound is tight. ■

Remark 2 (Graham, 1966). *The bound* $2 - 1/m$ *is also the worst case bound using the LS rule for problem* $P\|C_{\max}$.

4. THE SET LPT HEURISTIC

In this section, we introduce a heuristic procedure to find a schedule for problem $P2\|\sum C_{B_i}$ that has a worst case relative error of 9/7. The SB rule determines the batch sequence. Each batch is partitioned into two sets using the LPT (Longest Processing Time) rule. For this rule, when a machine becomes available, an unscheduled job with the longest processing time is selected for processing. The partitioning process assumes that processing of a batch can start on both machines at time zero. The set which has larger total processing time is assigned to the machine that becomes available first. The set which has smaller processing time is assigned to the other machine.

Heuristic H2

0. Reindex the batches so that $P_i \leq P_{i+1}$ for $i = 1, 2, \ldots, b-1$.
   For $i = 1, 2, \ldots, b$, reindex the jobs so that $p_j \geq p_{j+1}$ if $j, j+1 \in B_i$.
   Set $i = j = 1$ and $F_1' = F_2' = F_1 = F_2 = 0$.
1. Assign job j in batch i to the first available machine $k = \text{argmin}\,\{F_1', F_2'\}$. Set $F_k' = F_k' + p_j$ and $j = j + 1$. Repeat Step 1 until all jobs in batch i are scheduled.
2. Find ℓ and u such that $\ell = \text{argmin}\,\{F_1, F_2\}$ and $u = \text{argmin}\,\{F_1', F_2'\}$. Set $F_\ell = F_\ell + F_{3-u}'$ and $F_{3-\ell} = F_{3-\ell} + F_u'$.
3. Set $C_i = \max\{F_1, F_2\}$.
   If $i < b$, then set $i = i + 1$, $F_1' = F_2' = 0$, and go to Step 1.
   Otherwise, output $\sum_{i=1}^{b} C_i$ and stop.

In Step 0, reindexing the batches requires $O(b \log b)$ time and reindexing the jobs in each batch $i = 1, 2, \ldots, b$ requires $O(\sum_{i=1}^{b} n_i \log n_i)$ time. Since all other operations require $O(n)$ time, the time requirement of H2 is $O(n \log n)$.

We motivate the use of the LPT rule by observing that for the classical scheduling problem $P\|C_{\max}$, this rule provides a good heuristic solution (Graham, 1969). Since minimizing the completion time of a given batch is a makespan problem, scheduling each batch by this rule is likely to provide good schedules.

We assume that the jobs are indexed so that

$$P_1 \leq P_2 \leq \cdots \leq P_b \tag{2}$$

and

$$p_j \geq p_{j+1} \quad \text{if } j, j+1 \in B_i, \tag{3}$$

i.e., the batches are in SB order, and the jobs within a batch are in LPT order, respectively.

Let σ^{H2} be the schedule found by Heuristic H2, and let z^{H2} be sum of batch completion times of this schedule. In the remainder of this section, we establish that $z^{H2}/z^* \leq 9/7$ and show that this bound is tight.

For $k, i \in B$, suppose k is the last batch to complete before i in schedule σ. Let

$$\delta_i(\sigma) = \begin{cases} C_i(\sigma) - C_k(\sigma) & \text{if only one machine processes batch } i \\ |C_i(\sigma_1) - C_i(\sigma_2)| & \text{if both machines process batch } i. \end{cases}$$

For each $i \in B$, $\delta_i(\sigma)$ is the absolute difference between completion time of batch i on machine 1 and machine 2 in σ. If batch i is processed on only one machine, then $\delta_i(\sigma)$ is the difference between the completion time of batch i and the completion time of the last batch to complete before i (see Figure 1). If batch i is the first batch to complete, then we assume $C_k(\sigma) = 0$. We use $\delta_i(\sigma)$ to provide a description of the completion time of batch i. If $G \subseteq B$ is the set of batches that complete no later than batch i, then

$$C_i(\sigma) = \frac{\sum_{\ell \in G} P_\ell + \delta_i(\sigma)}{2}. \tag{4}$$

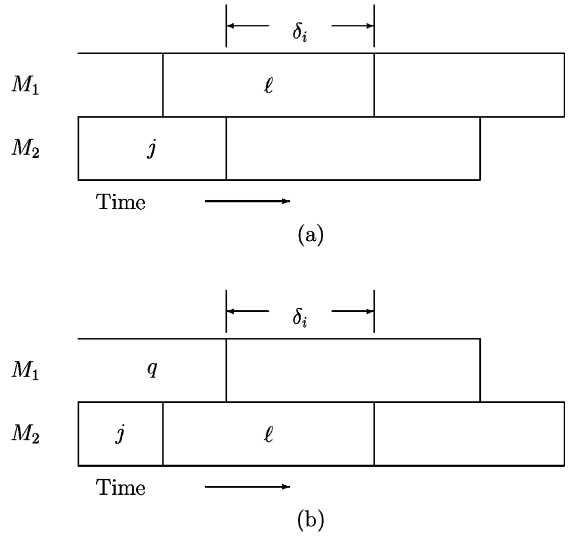

Figure 1. Examples of δ_i where the last two jobs processed in batch i are j and ℓ : (a) batch i is processed by both machines; (b) batch i is processed only by M_2 and q is the last job to complete in batch k.

Notice that since P_ℓ is known for $\ell = 1, 2, \ldots, i$, $C_i(\sigma)$ only depends on the size of $\delta_i(\sigma)$.

For $i \in B$, let $\bar{C_i}$ be the minimal makespan of just the jobs in batch i and let $\bar{\gamma}^i$ be the corresponding schedule. Similarly, let C_i^L be the makespan of the schedule generated by the LPT rule for batch i, and let γ^i be the corresponding schedule. Let $\bar{\delta}_i = 2\bar{C_i} - P_i$ and $\delta_i^L = 2C_i^L - P_i$ for $i \in B$. For $i \in B$, $\bar{\delta}_i$ is the difference between completion time of batch i on the two machines in $\bar{\gamma}^i$. Similarly, δ_i^L is the difference between completion time of batch i on the two machines in γ^i.

For notational convenience, let $\delta_i^* = \delta_i(\sigma^*)$ and $\delta_i^{H2} = \delta_i(\sigma^{H2})$ for $i \in B$, and $\delta_0^{H2} = 0$. From the steps of H2,

$$\delta_i^{H2} = \left|\delta_i^L - \delta_{i-1}^{H2}\right|, \quad i = 1, 2, \ldots, b. \tag{5}$$

The next remark establishes a lower bound for δ_i^{H2}.

Remark 3. *For $i \in B$, $\delta_i^{H2} \le P_i$.*

Proof. Follows from the fact that $\delta_\ell^L \le P_\ell$ for $\ell \in B$ and from (5). ■

We now present a preliminary result when job $\beta_i + 3$ completes last in γ^i for $i \in B$.

Lemma 3. If in γ^i, batch i completes when job $j \ge \beta_i + 3$ finishes processing for $i \in B$, then $\delta_i^L \le P_i/5$.

Proof. We first consider the case when job $j = \beta_i + 3$. Suppose $p_{\beta_i} \ge p_{\beta_i+1} + p_{\beta_i+2}$. From (3), $p_{\beta_i+3} \le P_i/5$. The result now follows.

Alternatively, suppose that $p_{\beta_i} < p_{\beta_i+1} + p_{\beta_i+2}$. Then,

$$\delta_i^L = p_{\beta_i} + p_{\beta_i+3} - p_{\beta_i+1} - p_{\beta_i+2} - \sum_{j=4}^{n_i-1} p_{\beta_i+j}.$$

Since $p_{\beta_i+3} \le p_{\beta_i+2}$, $\delta_i^L \le p_{\beta_i} - p_{\beta_i+1}$. If, $p_{\beta_i} - p_{\beta_i+1} < P_i/5$, then the result follows. If $p_{\beta_i} - p_{\beta_i+1} \ge P_i/5$, then $2p_{\beta_i+1} + p_{\beta_i+2} + p_{\beta_i+3} \le 4P_i/5$ and $p_{\beta_i+3} \le P_i/5$.

When job $j \ge \beta_i + 4$ is the last job to complete, the result follows from the definition of δ and from (3). ■

Next, we present some results regarding the bound for δ_i^L.

Lemma 4. If $C_i^L > \bar{C}_i$ for $i \in B$, then $\delta_i^L \le P_i/5$.

Proof. Since $C_i^L > \bar{C}_i$, jobs β_i and $\beta_i + 2$ can not be the last jobs to complete in γ^i. Job $\beta_i + 1$ can never be the last job to complete in γ^i. If job $j \ge \beta_i + 3$ is the last job to complete in γ^i, then the result follows from Lemma 3. ■

Lemma 5. Suppose $C_i^L = \bar{C}_i$ for $i \in B$. In σ^*, if batches $v_1, v_2, \ldots, v_\ell$ complete while batch i is processing, then $\delta_i^* + \sum_{j=1}^{\ell} \delta_{v_j}^* \ge \delta_i^L$ for integer $\ell \ge 0$.

Proof. Consider the partial schedule of σ^* from the start to the completion of batch i. Let the start time of batch i in σ^* be $S_i(\sigma^*)$. In σ^*, the amount of time that the jobs of batch v_j are processed in the interval $[S_i(\sigma^*), C_i(\sigma^*)]$ is at most $\delta_{v_j}^*$, for $j \in \{1, 2, \ldots, \ell\}$. Consequently, $\bar{C}_i \le C_i(\sigma^*) - S_i(\sigma^*) \le (P_i + \sum_{j=1}^{\ell} \delta_{v_j}^* + \delta_i^*)/2$. Thus,

$$\frac{P_i + \sum_{j=1}^{\ell} \delta_{v_j}^* + \delta_i^*}{2} \ge \bar{C}_i = \frac{P_i + \bar{\delta}_i}{2} = \frac{P_i + \delta_i^L}{2}.$$

The last equality holds because $C_i^L = \bar{C}_i$ by assumption. ■

Without loss of generality, we assume that $b \ge 4$. If $b < 4$, then we add dummy batches which have zero processing time. In σ^*, let the batches complete in the order of $v_1, v_2, \ldots, v_b$. From (2) and (4),

$$\begin{aligned}
\frac{z^{H2}}{z^*} &= \frac{\sum_{i=1}^{k} C_i(\sigma^{H2})}{\sum_{i=1}^{k} C_i(\sigma^*)} \\
&= \frac{bP_1 + \delta_1^{H2} + (b-1)P_2 + \delta_2^{H2} + \cdots + 2P_{b-1} + \delta_{b-1}^{H2} + P_b + \delta_b^{H2}}{bP_{v_1} + \delta_{v_1}^* + (b-1)P_{v_2} + \delta_{v_2}^* + \cdots + P_{v_b} + \delta_{v_b}^*} \\
&\le \frac{bP_1 + (b-1)P_2 + \cdots + P_b + \sum_{i=1}^{b} \delta_i^{H2}}{bP_1 + (b-1)P_2 + \cdots + P_b + \sum_{i=1}^{b} \delta_i^*}.
\end{aligned}$$

From Remark 3, since $\delta_i^{H2} \leq P_i$, for $i \in B$, we have, for $\ell = 0, 1, \ldots, b-4$,

$$\begin{aligned}
&\frac{bP_1 + (b-1)P_2 + \cdots + (b-\ell)P_{\ell+1} + \sum_{i=1}^{\ell+1} \delta_i^{H2}}{bP_1 + (b-1)P_2 + \cdots + (b-\ell)P_{\ell+1}} \\
&\quad = 1 + \frac{\sum_{i=1}^{\ell+1} \delta_i^{H2}}{bP_1 + (b-1)P_2 + \cdots + (b-\ell)P_{\ell+1}} \\
&\quad = 1 + \frac{\delta_1^{H2} + \delta_2^{H2} + \cdots + \delta_{\ell+1}^{H2}}{bP_1 + (b-1)P_2 + \cdots + 4P_{\ell+1}} \\
&\quad \leq 1 + \frac{1}{4} < \frac{9}{7}.
\end{aligned}$$

To establish that $z^{H2}/z^* \leq 9/7$, it is sufficient to show that there is an integer $q \geq 0$ such that $q \leq b-3$, and

$$\begin{aligned}
&\frac{(b-q)P_{q+1} + (b-q-1)P_{q+2} + \cdots + P_b + \sum_{i=q+1}^{b} \delta_i^{H2}}{(b-q)P_{q+1} + (b-q-1)P_{q+2} + \cdots + P_b + \sum_{i=1}^{b} \delta_i^*} \\
&\quad \leq 1 + \frac{\sum_{i=q+1}^{b} \delta_i^{H2}}{(b-q)P_{q+1} + (b-q-1)P_{q+2} + \cdots + P_b + \sum_{i=1}^{b} \delta_i^*} \\
&\quad \leq \frac{9}{7}.
\end{aligned} \tag{6}$$

This follows because

$$\frac{z^{H2}}{z^*} \leq \frac{\left[bP_1 + \cdots + (b-q+1)P_q + \sum_{i=1}^{q} \delta_i^{H2}\right] + \left[(b-q)P_{q+1} + \cdots + P_b + \sum_{i=q+1}^{b} \delta_i^{H2}\right]}{\left[bP_1 + \cdots + (b-q+1)P_q + \sum_{i=1}^{q} \delta_i^{*}\right] + \left[(b-q)P_{q+1} + \cdots + P_b + \sum_{i=q+1}^{b} \delta_i^{*}\right]}.$$

The next theorem establishes the worst case bound for H2.

Theorem 2. For problem $P2\|\sum C_{B_i}$, $z^{H2}/z^* \leq 9/7$ and this bound is tight.

Proof. We consider five cases depending on the sign of $\delta_\ell^L - \delta_{\ell-1}^{H2}$ for $\ell = b-2, b-1, b$.

Case 1. $\delta_{b-3}^{H2} \leq \delta_{b-2}^L$, $\delta_{b-2}^{H2} > \delta_{b-1}^L$, and $\delta_{b-1}^{H2} > \delta_b^L$.

From (5),

$$\begin{aligned}
\delta_{b-2}^{H2} + \delta_{b-1}^{H2} + \delta_b^{H2} &= \left(\delta_{b-2}^L - \delta_{b-3}^{H2}\right) + \left(\delta_{b-2}^{H2} - \delta_{b-1}^L\right) + \left(\delta_{b-1}^{H2} - \delta_b^L\right) \\
&= \left(\delta_{b-2}^L - \delta_{b-3}^{H2}\right) + \left(\delta_{b-2}^L - \delta_{b-3}^{H2} - \delta_{b-1}^L\right) + \left(\delta_{b-2}^{H2} - \delta_{b-1}^L - \delta_b^L\right) \\
&= 2\delta_{b-2}^L - 2\delta_{b-3}^{H2} - \delta_{b-1}^L + \left(\delta_{b-2}^L - \delta_{b-3}^{H2} - \delta_{b-1}^L - \delta_b^L\right) \\
&\leq 3\delta_{b-2}^L.
\end{aligned}$$

Suppose $C^L_{b-2} > \bar{C}_{b-2}$. By Lemma 4, $\delta^L_{b-2}/P_{b-2} \leq 1/5$. Thus,

$$\begin{aligned}\frac{3P_{b-2}+2P_{b-1}+P_b+\delta^{H2}_{b-2}+\delta^{H2}_{b-1}+\delta^{H2}_b}{3P_{b-2}+2P_{b-1}+P_b+\sum_{i=1}^b \delta^*_i} &\leq 1+\frac{3\delta^L_{b-2}}{3P_{b-2}+2P_{b-1}+P_b}\\ &\leq 1+\frac{3\delta^L_{b-2}}{3P_{b-2}}\\ &< \frac{9}{7}.\end{aligned}$$

Hence, the result is established from (6).

Alternatively, suppose $C^L_{b-2} = \bar{C}^L_{b-2}$. From Lemma 5, $\delta^L_{b-2} \leq \sum_{i=1}^b \delta^*_i$. By Remark 3, $2\delta^L_{b-2}/(3P_{b-2}+2P_{b-1}+P_b+\sum_{i=1}^b \delta^*_i) \leq 2/7$. Hence,

$$\begin{aligned}\frac{3P_{b-2}+2P_{b-1}+P_b+\delta^{H2}_{b-2}+\delta^{H2}_{b-1}+\delta^{H2}_b}{3P_{b-2}+2P_{b-1}+P_b+\sum_{i=1}^b \delta^*_i} &\leq 1+\frac{\left(3P_{b-2}+2P_{b-1}+P_b+\delta^L_{b-2}\right)+2\delta^L_{b-2}}{3P_{b-2}+2P_{b-1}+P_b+\sum_{i=1}^b \delta^*_i}\\ &\leq 1+\frac{2\delta^L_{b-2}}{3P_{b-2}+2P_{b-1}+P_b+\sum_{i=1}^b \delta^*_i}\\ &\leq \frac{9}{7}.\end{aligned}$$

Hence, from (6), the result is established.

For the remaining cases, the proof is similar, except that

Case 2. $\delta^{H2}_{b-3} > \delta^L_{b-2}$, $\delta^{H2}_{b-2} > \delta^L_{b-1}$, and $\delta^{H2}_{b-1} > \delta^L_b$ implies $\delta^{H2}_{b-2}+\delta^{H2}_{b-1}+\delta^{H2}_b \leq 3\delta^{H2}_{b-3}$. Also, the upper bound is $14/11$.

Case 3. $\delta^{H2}_{b-2} \leq \delta^L_{b-1}$ and $\delta^{H2}_{b-1} > \delta^L_b$ implies $\delta^{H2}_{b-2}+\delta^{H2}_{b-1}+\delta^{H2}_b \leq 2\delta^L_{b-1}$. Also, the upper bound is $5/4$.

Case 4. $\delta^{H2}_{b-3} > \delta^L_{b-2}$ and $\delta^{H2}_{b-1} \leq \delta^L_b$ implies $\delta^{H2}_{b-2}+\delta^{H2}_{b-1}+\delta^{H2}_b \leq \delta^{H2}_{b-3}+\delta^L_b$. Also, the upper bound is $6/5$.

Case 5. $\delta^{H2}_{b-3} \leq \delta^L_{b-2}$ and $\delta^{H2}_{b-1} \leq \delta^L_b$ implies $\delta^{H2}_{b-2}+\delta^{H2}_{b-1}+\delta^{H2}_b \leq \delta^L_{b-2}+\delta^L_b$. Also, the upper bound is $6/5$.

Now, we show that the bound is tight. Consider the instance where $b=3, n_1=1, n_2=2, n_3=2$, $p_1=2$ and $p_2=p_3=p_4=p_5=1$ (see Figure 2). An optimal schedule is $\sigma^*=(\sigma^*_1,\sigma^*_2)$ where $\sigma^*_1=(2,3,5)$ and $\sigma^*_2=(1,4)$, and $z^*=2+2+3=7$. However, $\sigma^{H2}=(\sigma^{H2}_1,\sigma^{H2}_2)$ where $\sigma^{H2}_1=(1,3,5)$ and $\sigma^{H2}_2=(2,4)$. The corresponding solution value is $z^{H2}=2+3+4=9$. Hence, $z^{H2}/z^*=9/7$ for the given example. ■

5. JOB LPT HEURISTIC

As mentioned in the prior section, the LPT rule finds a good schedule for problem $P\|C_{\max}$. To find a schedule for problem $P2\|\sum C_{B_i}$, Heuristic H3 tries to obtain the maximum benefit of the LPT rule. Based on the current partial schedule, this rule is used to decide both the job processing

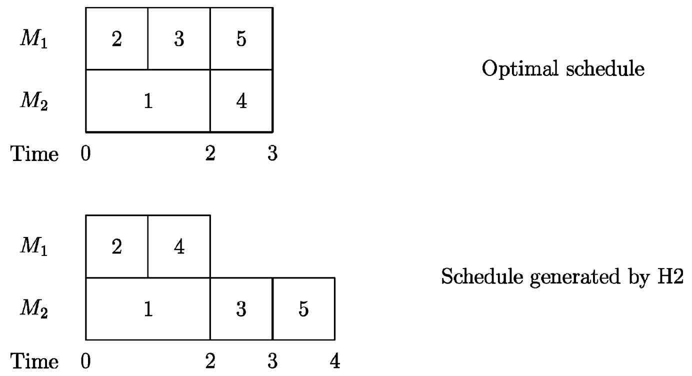

Figure 2. A worst case example of H2.

order and the machine-job assignment. Similar to the other procedures, the SB rule determines the batch sequence.

Heuristic H3 (Blocher and Chhajed, 1996)

0. Reindex the batches so that $P_i \leq P_{i+1}$ for $i = 1, 2, \ldots, b-1$.
   For $i = 1, 2, \ldots, b$, reindex the jobs so that $p_j \geq p_{j+1}$ if $j, j+1 \in B_i$.
   Set $i = j = 1$ and $F_1 = F_2 = 0$.
1. Select the first available machine $k = \arg\min\{F_1, F_2\}$. Choose job j in batch i and assign it to machine k. Set $F_k = F_k + p_j$ and $j = j + 1$.
   Repeat Step 1 until all jobs in batch i are scheduled.
2. Set $C_i = \max\{F_1, F_2\}$.
   If $i = b$, then output $\sum_{i=1}^{b} C_i$ and stop.
   Otherwise, set $i = i + 1$ and go to Step 1.

In Step 0, reindexing the batches requires O($b \log b$) time and reindexing the jobs in each batch $i = 1, 2, \ldots, b$ requires O($\sum_{i=1}^{b} n_i \log n_i$) time. Since all other operations require O(n) time, the time requirement of H3 is O($n \log n$).

As described, the job-machine assignment rule of H3 is the LPT rule. Also, from Lemma 1, there exists an optimal schedule where no batch is separated.

We define some notation. Let σ^{H3} be the schedule found by Heuristic H3 and z^{H3} be total completion time of this schedule. Also, let $\delta_i^{H3} = \delta_i(\sigma^{H3})$ for $i \in B$. To establish that the worst case relative error bound for Heuristic H3 is 6/5, we present two preliminary results concerning δ_i^{H3}.

Lemma 6. For $i = 2, 3, \ldots, b$, $\delta_{i-1}^{H3} + \delta_i^{H3} \leq P_i$.

Proof. Because jobs are assigned to the first available machine, $\delta_{i-1}^{H3} \leq P_{i-1}$. Since batches start processing in SB order in H3, $\delta_{i-1}^{H3} \leq P_i$. Now, by definition of δ, δ_i^{H3} can be no larger than $P_i - \delta_{i-1}^{H3}$. ■

Lemma 7. If in σ^{H3}, $\beta_i + j$ *is the last job to complete in batch* $i \in B$, *then* $\delta_i^{H3} \leq p_{\beta_{i+j}} \leq P_i/(j+1)$.

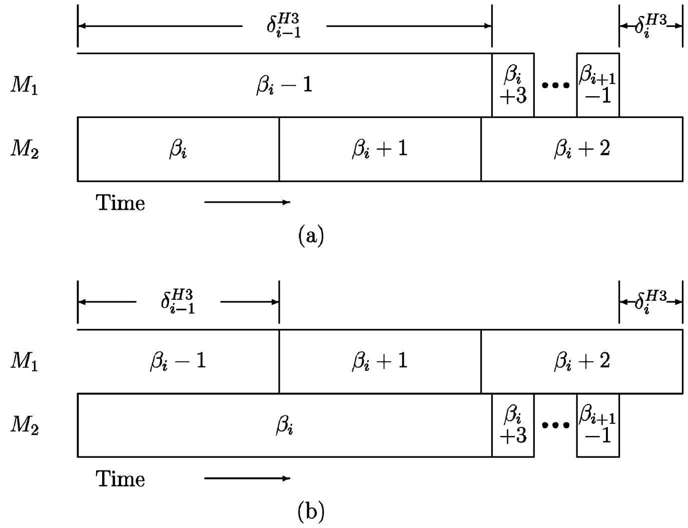

Figure 3. Examples for Lemma 8: (a) $p_{\beta_i} \leq \delta_{i-1}^{H3}$ and (b) $p_{\beta_i} > \delta_{i-1}^{H3}$.

Proof. The first inequality follows from the definition of δ and from the fact that jobs are assigned to the first available machine. The second inequality follows from (3). ■

Much of our analysis is based on which job is last to complete in a given batch in σ^{H3}. We now present some preliminary results for various possible final jobs.

Lemma 8. Suppose that in σ^{H3}, batch i completes when job $\beta_i + 2$ finishes processing for $i = 2, 3, \ldots, b$. If $p_{\beta_i} \leq \delta_{i-1}^{H3}$, then $\delta_i^{H3} \leq (P_i + \delta_{i-1}^{H3})/5$. If $p_{\beta_i} \geq \delta_{i-1}^{H3}$, then $\delta_i^{H3} \leq (P_i - \delta_{i-1}^{H3})/3$.

Proof. Suppose $p_{\beta_i} \leq \delta_{i-1}^{H3}$ (see Figure 3(a) for an example). Assign a dummy job of size δ_{i-1}^{H3} to batch i. This dummy job is now the largest job in batch i. Since both the dummy job and β_i start at the same time, the schedule of this modified batch under H3 is the same as in γ^i. By Lemma 3, the result follows.

Alternatively, suppose $p_{\beta_i} \geq \delta_{i-1}^{H3}$. If jobs $\beta_i + 1$ and $\beta_i + 2$ are processed on the same machine in σ^{H3}, then $p_{\beta_i} - \delta_{i-1}^{H3} \geq p_{\beta_i+1} \geq p_{\beta_i+2} \geq \delta_i^{H3}$ (see Figure 3(b) for an example). If jobs $\beta_i + 1$ and $\beta_i + 2$ are not processed on the same machine in σ^{H3}, then

$$\delta_i^{H3} = p_{\beta_i+2} + \left(p_{\beta_i} - \delta_{i-1}^{H3}\right) - p_{\beta_i+1} - \sum_{j=3}^{n_i-1} p_{\beta_i+j}.$$

Since $p_{\beta_i+1} \geq p_{\beta_i+2}$, $\delta_i^{H3} \leq p_{\beta_i} - \delta_{i-1}^{H3}$. From Lemma 7, $\delta_i^{H3} \leq p_{\beta_i+2} \leq p_{\beta_i+1}$. Thus,

$$\frac{\delta_i^{H3}}{P_i - \delta_{i-1}^{H3}} \leq \frac{\delta_i^{H3}}{p_{\beta_i} + p_{\beta_i+1} + p_{\beta_i+2} - \delta_{i-1}^{H3}}$$
$$\leq \frac{1}{3}.$$
■

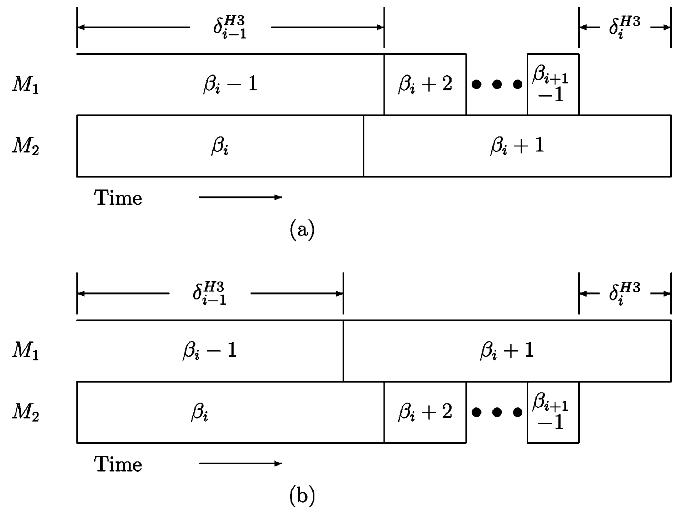

Figure 4. Examples for Lemma 9: (a) $p_{\beta_i} \leq \delta_{i-1}^{H3}$ and (b) $p_{\beta_i} > \delta_{i-1}^{H3}$

Lemma 9. If batch i completes in σ^{H3} when job $\beta_i + 1$ finishes processing, then $\delta_i^{H3} \leq \min\{\delta_{i-1}^{H3}, P_i/2\}$ *for* $i = 2, 3, \ldots, b$.

Proof. From Lemma 7, $\delta_i^{H3} \leq p_{\beta_i+1} \leq P_i/2$.

Suppose $p_{\beta_i} \leq \delta_{i-1}^{H3}$ (see Figure 4(a)). Then by Lemma 7, $\delta_i^{H3} \leq p_{\beta_i+1}$, and the result follows. Alternatively, suppose $p_{\beta_i} \geq \delta_{i-1}^{H3}$. Assume without loss of generality that in σ^{H3}, job β_i is processed on machine 2. Then, job $\beta_i + 1$ is processed on machine 1. Because job $\beta_i + 1$ completes last in batch i, jobs $\beta_i + 2, \beta_i + 3, \ldots, \beta_i + n_i - 1$ are processed on machine 2 (see Figure 4(b)). Hence,

$$\delta_i^{H3} = \delta_{i-1}^{H3} + p_{\beta_i+1} - p_{\beta_i} - \sum_{j=2}^{n_i-1} p_{\beta_i+j}.$$

Since $p_{\beta_i} \geq p_{\beta_i+1}$, $\delta_i^{H3} \leq \delta_{i-1}^{H3}$. ■

We now present two results which provide lower bounds for $\sum_{l=1}^{b} \delta_l^*$.

Lemma 10. If in σ^{H3}, β_i is the last job to complete in batch i, then $\delta_{i-1}^{H3} + \delta_i^{H3} = p_{\beta_i} - \sum_{j \in B_i \setminus \{\beta_i\}} p_j \leq \sum_{l=1}^{b} \delta_l^*$.

Proof. If β_i is the last job to complete in σ^{H3}, then the last job to complete in batch $i - 1$ and the remaining jobs in batch i are processed on a different machine than β_i. This establishes the first equality.

Without loss of generality, suppose that β_i is processed on machine 2 in σ^*. For integer $r \geq 0$, let $v_1, v_2, \ldots, v_r$ be the batches that complete in σ^* while job β_i is processing. By assumption, the last job from each of these batches are processed on machine 1. If there are other jobs from batch

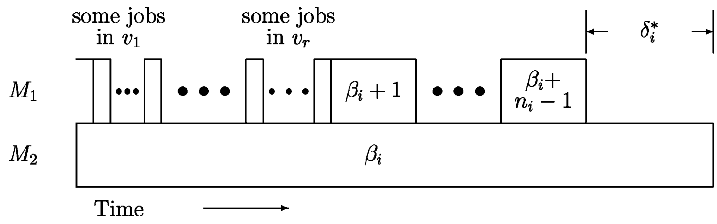

Figure 5. An example for Lemma 10

i that are processed on machine 1, then the optimality of σ^* requires that they are processed after the jobs in $v_1, v_2, \ldots, v_r$ (see Figure 5). Thus,

$$p_{\beta_i} - \sum_{j \in B_i \setminus \{\beta_i\}} p_j \leq \sum_{j=1}^{r} \delta_{v_j}^* + \delta_i^*$$
$$\leq \sum_{l=1}^{b} \delta_l^*.$$

■

The next result considers the case when and γ^i and $\bar{\gamma}^i$ complete at the same time. Recall from Section 4 that for $i \in B$, $\bar{C}_i$ is the minimal makespan of just the jobs in batch i, and C_i^L is the makespan of the schedule generated by the LPT rule for batch i.

Lemma 11. Suppose $C_i^L = \bar{C}_i$ for $i \in B$. Then, $\delta_i^{H3} - \delta_{i-1}^{H3} \leq \sum_{l=1}^{b} \delta_l^*$.

Proof. From Lemma 5, $\sum_{l=1}^{b} \delta_l^* \geq \delta_i^L$ for $i \in B$. We prove that $\delta_i^{H3} - \delta_{i-1}^{H3} \leq \delta_i^L$ for $i \in B$. Suppose that when batch i starts in σ^{H3}, both machines are free. Then, $\delta_i^{H3} = \delta_i^L$.

If both machines are not free, then processing is delayed on one machine by δ_{i-1}^{H3}. Since the starting time order does not change, the delay of a job is at most δ_{i-1}^{H3} larger than in the schedule where both machines are free. This result holds for the last job processed in batch i. Thus, for $i \in B$, $\delta_i^{H3} \leq \delta_{i-1}^{H3} + \delta_i^L$. ■

Next, we present a result when γ^i completes after $\bar{\gamma}^i$.

Lemma 12. If $C_i^L > \bar{C}_i$, then $\delta_i^{H3} \leq \max\{\delta_{i-1}^{H3}, P_i/5\}$ for $i = 2, 3, \ldots, b$.

Proof. From the proof of Lemma 4, the condition $C_i^L > \bar{C}_i$ implies that batch i has at least four jobs.

Without loss of generality, assume that for $i \in B$, job β_i is processed on machine 2 in σ^{H3}. Suppose that $p_{\beta_i} \leq \delta_{i-1}^{H3}$. Then by Lemma 7, $\delta_i^{H3} \leq p_{\beta_i} \leq \delta_{i-1}^{H3}$ and the result is established. Hence, we consider the case where $p_{\beta_i} > \delta_{i-1}^{H3}$. This implies that job $\beta_i + 1$ is processed on machine 1.

If $p_{\beta_i} \geq \delta_{i-1}^{H3} + p_{\beta_i+1}$, then assign a dummy job of size $p_{\beta i} - \delta_{i-1}^{H3}$ to batch i and remove job β_i from batch i. This dummy job is now the largest job in batch i. Since both the dummy job and $\beta_i + 1$ start at the same time, the schedule of this modified batch under H3 is the same as in γ^i. By Lemma 4, $\delta_i^{H3} \leq (P_i - \delta_{i-1}^{H3})/5 \leq P_i/5$.

The only case left to consider is when $\delta_{i-1}^{H3} < p_{\beta_i} < \delta_{i-1}^{H3} + p_{\beta_i+1}$. This condition implies that $\beta_i + 2$ is processed on machine 2. Also, batch i does not complete when job β_i finishes processing.

Suppose batch i completes when job $\beta_i + 1$ finishes processing in σ^{H3}. Then, $\delta_i^{H3} \leq \delta_{i-1}^{H3}$ follows from Lemma 9.

Suppose batch i completes when job $\beta_i + 2$ finishes processing. Then, $\sum_{j=3}^{n_i-1} p_{\beta+j} + \delta_i^{H3} \leq p_{\beta_i+2}$ Thus,

$$\delta_i^{H3} \leq p_{\beta_i+2} - \sum_{j=3}^{n_i-1} p_{\beta+j}.$$

Because $C_i^L > \bar{C}_i$, we have that $p_{\beta_i+1} + p_{\beta_i+2} < p_{\beta_i} + \sum_{j=3}^{n_i-1} p_{\beta+j}$. As a result,

$$p_{\beta_i+2} - \sum_{j=3}^{n_i-1} p_{\beta+j} < p_{\beta_i} - p_{\beta_i+1}.$$

Since $p_{\beta_i} - p_{\beta_i+1} < \delta_{i-1}^{H3}$, it follows that $\delta_i^{H3} \leq \delta_{i-1}^{H3}$.

Suppose batch i completes when job $\beta_i + 3$ finishes processing. If job $\beta_i + 3$ is processed on machine 2 in σ^{H3}, then $p_{\beta_i+2} + p_{\beta_i} \leq \delta_{i-1}^{H3} + p_{\beta_i+1}$. By Lemma 7, $\delta_i^{H3} \leq p_{\beta_i+3} \leq p_{\beta_i+2} \leq p_{\beta_i+2} \leq \delta_{i-1}^{H3}$.

If job $\beta_i + 3$ is processed on machine 1, then

$$\delta_i^{H3} = \delta_{i-1}^{H3} + p_{\beta_i+1} + p_{\beta_i+3} - p_{\beta_i} - p_{\beta_i+2} - \sum_{j=4}^{n_i-1} p_{\beta+j}.$$

Since $p_{\beta_i} \geq p_{\beta_i+1}$ and $p_{\beta_i+2} \geq p_{\beta_i+3}$, we have that $\delta_i^{H3} \leq \delta_{i-1}^{H3}$.

Suppose batch i completes when job β_j finishes processing for $j \geq \beta_i + 4$ in σ^{H3}. Then by Lemma 7, $\delta_i^{H3} \leq P_i/5$. ■

Without loss of generality, we assume that there are at least four batches. If $b < 4$, then we add dummy batches that have zero processing time. In σ^*, let the batches complete in the order $v_1, v_2, \ldots, v_b$. From (2),

$$\begin{aligned}\frac{z^{H3}}{z^*} &= \frac{\sum_{i=1}^{b} C_i(\sigma^{H3})}{\sum_{i=1}^{b} C_i(\sigma^*)}\\ &= \frac{bP_1 + \delta_1^{H3} + (b-1)P_2 + \delta_2^{H3} + \cdots + P_b + \delta_b^{H3}}{bP_{v_1} + \delta_{v_1}^* + (b-1)P_{v_2} + \delta_{v_2}^* + \cdots + P_{v_b} + \delta_{v_b}^*}\\ &\leq \frac{bP_1 + (b-1)P_2 + \cdots + P_b + \sum_{i=1}^{b}\delta_i^{H3}}{bP_1 + (b-1)P_2 + \cdots + P_b + \sum_{i=1}^{b}\delta_i^*}.\end{aligned}$$

Since $\delta_i^{H3} \leq P_i$ for $i \in B$,

$$\frac{bP_1 + \delta_1^{H3} + (b-1)P_2 + \delta_2^{H3} + \cdots + 5P_{b-4} + \delta_{b-4}^{H3}}{bP_1 + (b-1)P_2 + \cdots + 5P_{b-4}} = 1 + \frac{\sum_{i=1}^{b-4}\delta_i^{H3}}{bP_1 + (b-1)P_2 + \cdots + 5P_{b-4}} \leq \frac{6}{5}.$$

Hence, to prove that $z^{H3}/z^* < 6/5$, it is sufficient to show that

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=b-3}^{b} \delta_i^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} \leq \frac{6}{5}. \tag{7}$$

The proof of the bound for H3 is established by the following series of results based on the job of batch b that completes last in σ^{H3}: Propositions 1, 3, 4, 5, and Theorem 3.

Proposition 1. If batch b completes in σ^{H3} when job β_b finishes processing, then $z^{H3}/z^* \leq 6/5$.

Proof. From Lemma 10, $\delta_{b-1}^{H3} + \delta_b^{H3} \leq \sum_{i=1}^{b} \delta_i^*$. Also, (2) and Lemma 7 imply that $5\delta_{b-2}^{H3} \leq 2P_{b-2} + 2P_{b-1} + P_b$ and $5\delta_{b-3}^{H3} \leq 4P_{b-3} + P_{b-2}$. Thus,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3}}{(4P_{b-3} + P_{b-2}) + (2P_{b-2} + 2P_{b-1} + P_b)} \leq 1 + \frac{1}{5}.$$

The proof now follows from (7). ■

We present three lemmas and a proposition that establish the bound for H3 for the case when $\delta_b^{H3} \leq \min\{\delta_{b-1}^{H3}, P_b/2\}$. Each result considers a different final job for batch $b-1$. Several situations satisfy these conditions including the one established by Proposition 3.

Lemma 13. If $\delta_b^{H3} \leq \min\{\delta_{b-1}^{H3}, P_b/2\}$ and job β_{b-1} finishes last in batch $b-1$ for schedule σ^{H3}, then $z^{H3}/z^* < 6/5$.

Proof. From Lemma 10, $\delta_{b-2}^{H3} + \delta_{b-1}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$. From Lemma 7 and (2), $\delta_{b-3}^{H3} \leq P_{b-3} \leq P_{b-2}$. Since $\delta_b^{H3} \leq \delta_{b-1}^{H3} \leq P_{b-1}$ and $\delta_b^{H3} \leq \delta_{b-1}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \delta_{b-2}^{H3} + \delta_{b-1}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-3}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*}$$
$$\leq 1 + \frac{\delta_{b-3}^{H3} + \delta_b^{H3}}{4\delta_{b-3}^{H3} + 3\delta_{b-3}^{H3} + 2\delta_b^{H3} + 2\delta_b^{H3} + \delta_b^{H3}} \leq \frac{6}{5}.$$

The proof now follows from (7). ■

Lemma 14. If $\delta_b^{H3} \leq \min\{\delta_{b-1}^{H3}, P_b/2\}$ and job $\beta_{b-1} + 1$ finishes last in batch $b-1$ for shcedule σ^{H3}, then $z^{H3}/z^* \leq 6/5$.

Proof. From Lemma 9, $\delta_{b-1}^{H3} \leq \min\{\delta_{b-2}^{H3}, P_{b-1}/2\}$. We consider three cases based on which job completes last in batch $b-2$.

Case 1. In σ^{H3}, batch $b-2$ completes when job β_{b-2} finishes processing.

From Lemma 10, $\delta_{b-3}^{H3} + \delta_{b-2}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$. Since $\delta_{b-1}^{H3} \leq \delta_{b-2}^{H3} \leq P_{b-2}$, $\delta_{b-1}^{H3} \leq P_{b-2}$ and $\delta_{b-1}^{H3} \leq \delta_{b-2}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \delta_{b-3}^{H3} + \delta_{b-2}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-1}^{H3} + \delta_b^{H3}}{3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*}$$
$$\leq 1 + \frac{2\delta_{b-1}^{H3}}{3\delta_{b-1}^{H3} + 4\delta_{b-1}^{H3} + 2\delta_{b-1}^{H3} + \delta_{b-1}^{H3}} = \frac{6}{5}.$$

Case 2. In σ^{H3}, batch $b-2$ completes when job $\beta_{b-2}+1$ finishes processing.

From Lemma 9, $\delta_{b-2}^{H3} \leq \min\{\delta_{b-3}^{H3}, P_{b-2}/2\}$.

Suppose that in σ^{H3}, batch $b-3$ completes when job β_{b-3} finishes processing. Then, $\delta_{b-3}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$ from Lemma 10. Since $\delta_{b-2}^{H3} \leq \delta_{b-3}^{H3} \leq P_{b-3}$, from (2),

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \delta_{b-3}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-3}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b}$$
$$\leq 1 + \frac{3\delta_{b-2}^{H3}}{4\delta_{b-2}^{H3} + 6\delta_{b-2}^{H3} + 4\delta_{b-2}^{H3} + 2\delta_{b-2}^{H3}} < \frac{6}{5}.$$

Alternatively, suppose that in σ^{H3}, batch $b-3$ completes when job $\beta_{b-3}+1$ or larger finishes processing. Then, from Lemma 7, $\delta_{b-3}^{H3} \leq P_{b-3}/2$. From (2),

$$\frac{4P_{b-3} + 3P_{b-2} + P_{b-1} + P_b}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b} + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b}$$
$$\leq 1 + \frac{4\delta_{b-3}^{H3}}{8\delta_{b-3}^{H3} + 6\delta_{b-3}^{H3} + 4\delta_{b-3}^{H3} + 2\delta_{b-3}^{H3}} = \frac{6}{5}.$$

Hence, the result is established from (7).

Case 3. In σ^{H3}, batch $b-2$ completes when job $\beta_{b-2}+2$ or larger finishes processing.

From Lemma 7, $\delta_{b-2}^{H3} \leq P_{b-2}/3$. Also, $\delta_{b-3}^{H3}/(4P_{b-3} + P_{b-2}) \leq 1/5$. Hence, from (2),

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b} + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{(4P_{b-3} + P_{b-2}) + (2P_{b-2} + 2P_{b-1} + P_b)}$$
$$\leq 1 + \frac{\delta_{b-3}^{H3} + 3\delta_{b-2}^{H3}}{\left(4\delta_{b-3}^{H3} + \delta_{b-3}^{H3}\right) + 15\delta_{b-2}^{H3}} = \frac{6}{5}.$$

The proof now follows from (7). ■

Lemma 15. If $\delta_b^{H3} \leq \min\{\delta_{b-1}^{H3}, P_b/2\}$ and job $\beta_{b-1}+2$ finishes last in batch $b-1$ for schedule σ^{H3}, then $z^{H3}/z^* \leq 6/5$.

Proof. We consider two cases: $\delta_{b-2}^{H3} \geq \delta_{b-1}^{H3}$ and $\delta_{b-2}^{H3} < \delta_{b-1}^{H3}$.

Case 1. $\delta_{b-2}^{H3} \geq \delta_{b-1}^{H3}$.

From Lemma 7, $\delta_{b-1}^{H3} \leq P_{b-1}/3$. Since the proof of Lemma 14 uses the conditions $\delta_{b-1}^{H3} \leq \min\{\delta_{b-2}^{H3}, P_{b-1}/2\}$, the proof of this case follows in a similar manner.

Case 2. $\delta_{b-2}^{H3} < \delta_{b-1}^{H3}$.

Since $\delta_{b-2}^{H3} < \delta_{b-1}^{H3} \leq p_{\beta_{b-1}}$, $\delta_b^{H3} \leq \delta_{b-1}^{H3} \leq (P_{b-1} - \delta_{b-2}^{H3})/3$ follows from Lemma 8. Because $P_{b-1} \leq P_b$, we have that $3\delta_{b-2}^{H3} + 4\delta_{b-1}^{H3} + 5\delta_b^{H3} \leq 2P_{b-1} + P_b$.

Suppose $C_{b-1}^L = \bar{C}_{b-1}$. Then, $\delta_{b-1}^{H3} - \delta_{b-2}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$ from Lemma 11. Since $3\delta_{b-3}^{H3} + 3\delta_{b-2}^{H3} \leq 3P_{b-2}$ from Lemma 6,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b} + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*}$$
$$\leq 1 + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{4\delta_{b-3}^{H3} + \left(3\delta_{b-3}^{H3} + 3\delta_{b-2}^{H3}\right) + \left(3\delta_{b-2}^{H3} + 4\delta_{b-1}^{H3} + 5\delta_b^{H3}\right) + \left(\delta_{b-1}^{H3} - \delta_{b-2}^{H3}\right)} \leq \frac{6}{5}.$$

The result now follows from (7).

Alternatively, suppose $C_{b-1}^L > \bar{C}_{b-1}$. From Lemma 12, $\delta_{b-1}^{H3} \leq P_{b-1}/5$. Since $\delta_b^{H3} \leq \delta_{b-1}^{H3}$ and $P_{b-2} \leq P_b$, $5\delta_{b-2}^{H3} + 5\delta_{b-1}^{H3} + 5\delta_b^{H3} \leq 2P_{b-1} + P_b$. The proof is now similar to the prior analysis where $C_{b-1}^L = \bar{C}_{b-1}$. ■

Proposition 2. If $\delta_b^{H3} \leq \min\{\delta_{b-1}^{H3}, P_b/2\}$, then $z^{H3}/z^* \leq 6/5$.

Proof. From Lemmas 13, 14, and 15, the only case left to consider is when job $\beta_{b-1} + 3$ or larger finishes last in batch $b-1$ for schedule σ^{H3}. From Lemma 7, $\delta_{b-1}^{H3} \leq P_{b-1}/4$. In Case 2 of Lemma 15, we use the conditions that $\delta_b^{H3} \leq \min\{\delta_{b-1}^{H3}, P_b/2\}$, and $\delta_{b-1}^{H3} \leq (P_{b-1} - \delta_{b-2}^{H3})/3$. The proof of the proposition follows in a similar manner. ■

Proposition 3. If batch b completes in σ^{H3} when job $\beta_b + 1$ finishes processing, then $z^{H3}/z^* \leq 6/5$.

Proof. Lemma 9 establishes that $\delta_b^{H3} \leq \min\{\delta_{b-1}^{H3}, P_b/2\}$. The result now follows from Proposition 2. ■

To establish (7) when in σ^{H3}, job $\beta_b + 2$ finishes last in batch b, we separate the analysis into two cases: $\delta_{b-1}^{H3} \leq P_b/7$ and $\delta_{b-1}^{H3} > P_b/7$. These cases are considered in Propositions 4 and 5.

Proposition 4. If batch b completes in σ^{H3} when job $\beta_b + 2$ finishes processing and $\delta_{b-1}^{H3} \leq P_b/7$, then $z^{H3}/z^* \leq 6/5$.

Proof. From Lemma 7, $\delta_b^{H3} \leq P_b/3$. If $\delta_b^{H3} \leq \delta_{b-1}^{H3}$, then both conditions of Proposition 2 are satisfied and the result is established. Consequently, we assume that $\delta_b^{H3} > \delta_{b-1}^{H3}$. Since $p_{\beta_b} \geq \delta_b^{H3}$ by Lemma 7, $\delta_b^{H3} \leq (P_b - \delta_{b-1}^{H3})/3$ follows from Lemma 8.

Suppose that $C_b^L > \bar{C}_b$. Because $\delta_b^{H3} > \delta_{b-1}^{H3}$, Lemma 12 establishes that $\delta_b^{H3} \leq P_b/5$. When the last job in batch $b-1$ to complete is β_{b-1}, $\beta_{b-1} + 1$, $\beta_{b-1} + 2$, and $\beta_{b-1} + 3$ or larger, we use proofs similar to those in Lemmas 13, 14, 15, and Proposition 2, respectively.

Alternatively, suppose that $C_b^L = \bar{C}_b$. Then, Lemma 11 implies that $\delta_b^{H3} - \delta_{b-1}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$. We consider two cases based on which job completes last in batch $b-1$.

Case 1. In σ^{H3}, batch $b-1$ completes when job β_{b-1} finishes processing.

From Lemma 6, $\delta_{b-2}^{H3} + \delta_{b-1}^{H3} \leq P_{b-1}$ and $\delta_{b-3}^{H3} + \delta_{b-2}^{H3} \leq P_{b-2}$. From Lemma 10, $\delta_{b-2}^{H3} + \delta_{b-1}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$. Thus,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b - \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + 2\delta_{b-1}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*}$$
$$\leq 1 + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + 2\delta_{b-1}^{H3}}{4\delta_{b-3}^{H3} + \left(3\delta_{b-3}^{H3} + 3\delta_{b-2}^{H3}\right) + \left(2\delta_{b-2}^{H3} + 2\delta_{b-1}^{H3}\right) + 7\delta_{b-1}^{H3} + \left(\delta_{b-2}^{H3} + \delta_{b-1}^{H3}\right)} \leq \frac{6}{5}.$$

Hence, the result is established from (7).

Case 2. In σ^{H3}, batch $b-1$ completes when job $\beta_{b-1}+1$ or larger finishes processing.

From Lemma 7, $\delta_{b-1}^{H3} \leq P_{b-1}/2$. Suppose that in σ^{H3}, batch $b-2$ completes when job β_{b-2} finishes processing. Lemma 10 implies that $\delta_{b-3}^{H3} + \delta_{b-2}^{H3} \leq \sum_{i=1}^{b} \delta_i^*$, Thus,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b - \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + 2\delta_{b-1}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*}$$
$$\leq 1 + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + 2\delta_{b-1}^{H3}}{4\delta_{b-3}^{H3} + \left(3\delta_{b-3}^{H3} + 3\delta_{b-2}^{H3}\right) + \left(\delta_{b-2}^{H3} + \delta_{b-1}^{H3} + 2\delta_{b-1}^{H3}\right) + 7\delta_{b-1}^{H3} + \left(\delta_{b-2}^{H3} + \delta_{b-1}^{H3}\right)} \leq \frac{6}{5}.$$

Hence, the result is established from (7).

Alternatively, suppose that in σ^{H3}, batch $b-2$ completes when job $\beta_{b-2}+1$ or larger finishes processing. Lemma 6 implies that $\delta_{b-2}^{H3} \leq P_{b-2}/2$. Thus,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b - \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + 2\delta_{b-1}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b}$$
$$\leq 1 + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + 2\delta_{b-1}^{H3}}{4\delta_{b-3}^{H3} + \left(\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + 4\delta_{b-2}^{H3}\right) + \left(\delta_{b-2}^{H3} + \delta_{b-1}^{H3} + 2\delta_{b-1}^{H3}\right) + 7\delta_{b-1}^{H3}} \leq \frac{6}{5}.$$

Hence, the result is established from (7). ■

Proposition 5. If batch b completes in σ^{H3} when job $\beta_b + 2$ finishes processing and $\delta_{b-1}^{H3} > P_b/7$, then $z^{H3}/z^* \leq 6/5$.

Proof. As in the proof of Proposition 4, we assume that $\delta_b^{H3} > \delta_{b-1}^{H3}$. This implies that $\delta_b^{H3} \leq (P_b - \delta_{b-1}^{H3})/3 < 2P_b/7$. We consider four cases, depending on which job completes last in batch $b-1$.

Case 1. In σ^{H3}, batch $b-1$ completes when job β_{b-1} finishes processing.

From Lemma 10, $\delta_{b-2}^{H3} + \delta_{b-1}^{H3} \le \sum_{i=1}^{b} \delta_i^*$. Lemma 7 implies that $\delta_b^{H3} < 2P_b/7 < 2\delta_{b-1}^{H3} \le 2P_{b-1}$. Also, $\delta_{b-2}^{H3} + \delta_{b-1}^{H3} \le \sum_{i=1}^{b} \delta_i^*$ implies that $\delta_b^{H3}/2 \le \sum_{i=1}^{b} \delta_i^*$. From Lemma 7 and (2), $\delta_{b-3}^{H3} \le P_{b-3} \le P_{b-2}$. Thus,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \delta_{b-2}^{H3} + \delta_{b-1}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-3}^{H3} + \delta_b^{H3}}{(4P_{b-3} + 3P_{b-2}) + \left(2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*\right)}$$
$$\le 1 + \frac{\delta_{b-3}^{H3} + \delta_b^{H3}}{4\delta_{b-3}^{H3} + 3\delta_{b-3}^{H3} + \delta_b^{H3} + 7\delta_b^{H3}/2 + \delta_b^{H3}/2} \le \frac{6}{5}.$$

Hence, the result is established from (7).

Case 2. In σ^{H3}, batch $b-1$ completes when job $\beta_{b-1}+1$ finishes processing.

From Lemma 7, $\delta_b^{H3} < 2P_b/7 < 2\delta_{b-1}^{H3} \le P_{b-1}$. Also, from Lemma 9 and (2), $\delta_{b-1}^{H3} \le \delta_{b-2}^{H3} \le P_{b-2}$. Suppose that in σ^{H3}, batch $b-2$ completes when job β_{b-2} finishes processing. Then from Lemma 10, $\delta_{b-3}^{H3} + \delta_{b-2}^{H3} \le \sum_{i=1}^{b} \delta_i^*$. Thus,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \delta_{b-3}^{H3} + \delta_{b-2}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-1}^{H3} + \delta_b^{H3}}{3P_{b-3} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*}$$
$$\le 1 + \frac{\delta_{b-1}^{H3} + \delta_b^{H3}}{3\delta_{b-1}^{H3} + \left(\delta_{b-1}^{H3} + 3\delta_b^{H3}/2\right) + 7\delta_b^{H3}/2 + \left(\delta_{b-3}^{H3} + \delta_{b-1}^{H3}\right)} \le \frac{6}{5}.$$

Hence, the result is established from (7).

Now, suppose that in σ^{H3}, batch $b-2$ completes when job $\beta_{b-2}+1$ finishes processing. From Lemma 7, $\delta_{b-3}^{H3} \le P_{b-3}$ and $2\delta_{b-2}^{H3} \le P_{b-2}$. Also, Lemma 9 implies that $\delta_{b-2}^{H3} \le \delta_{b-3}^{H3}$. If in σ^{H3}, batch $b-3$ completes when job β_{b-3} finishes processing, then Lemma 10 implies that $\delta_{b-3}^{H3} \le \sum_{i=1}^{b} \delta_i^*$. Thus,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \delta_{b-3}^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b + \sum_{i=1}^{b} \delta_i^*} + \frac{\delta_{b-2}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b}$$
$$\le 1 + \frac{\delta_{b-2}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{4\delta_{b-2}^{H3} + \left(2\delta_{b-2}^{H3} + 4\delta_{b-1}^{H3}\right) + \left(\delta_{b-1}^{H3} + 3\delta_b^{H3}/2\right) + 7\delta_b^{H3}/2} \le \frac{6}{5}.$$

Hence, the result is established from (7).

Alternatively, if in σ^{H3}, batch $b-3$ completes when job $\beta_{b-3}+1$ or larger finishes processing, then Lemma 7 implies that $\delta_{b-3}^{H3} \le P_{b-3}/2$. Thus,

$$\frac{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b} + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{4P_{b-3} + 3P_{b-2} + 2P_{b-1} + P_b}$$
$$\le 1 + \frac{\delta_{b-3}^{H3} + \delta_{b-2}^{H3} + \delta_{b-1}^{H3} + \delta_b^{H3}}{\left(5\delta_{b-3}^{H3} + 3\delta_{b-2}^{H3}\right) + \left(2\delta_{b-2}^{H3} + 4\delta_{b-1}^{H3}\right) + \left(\delta_{b-1}^{H3} + 3\delta_b^{H3}/2\right) + 7\delta_b^{H3}/2} \le \frac{6}{5}.$$

Hence, the result is established from (7).

Finally, suppose that in σ^{H3}, batch $b-2$ completes when job $\beta_{b-2}+2$ or larger finishes processing. Then, $\delta_{b-2}^{H3} \leq P_{b-2}/3$. Since $\delta_{b-1}^{H3} \leq \delta_{b-2}^{H3} \leq P_{b-2}/3 \leq P_{b-1}/3$ and $\delta_b^{H3} \leq 2P_{b-1}/3$,

$$\frac{4P_{b-3}+3P_{b-2}+2P_{b-1}+P_b}{4P_{b-3}+3P_{b-2}+2P_{b-1}+P_b} + \frac{\delta_{b-3}^{H3}+\delta_{b-2}^{H3}+\delta_{b-1}^{H3}+\delta_b^{H3}}{4P_{b-3}+3P_{b-2}+2P_{b-1}+P_b}$$
$$\leq 1 + \frac{\delta_{b-3}^{H3}+\delta_{b-2}^{H3}+\delta_{b-1}^{H3}+\delta_b^{H3}}{4\delta_{b-3}^{H3}+\left(\delta_{b-3}^{H3}+\delta_{b-2}^{H3}+4\delta_{b-2}^{H3}+2\delta_{b-1}^{H3}\right)+\left(3\delta_{b-1}^{H3}+3\delta_b^{H3}/2\right)+7\delta_b^{H3}/2} \leq \frac{6}{5}.$$

Hence, the result is established from (7).

Case 3. In σ^{H3}, batch $b-1$ completes when job $\beta_{b-1}+2$ finishes processing.

From Lemma 7, $\delta_b^{H3} < 2P_b/7 < 2\delta_{b-1}^{H3} \leq 2P_{b-1}/3$. Hence, $5\delta_b^{H3} < P_{b-1}+P_b$. If $\delta_{b-2}^{H3} \geq \delta_{b-1}^{H3}$, then the proof is similar to Case 2. As a result, we assume that $\delta_{b-2}^{H3} < \delta_{b-1}^{H3}$.

When $C_b^L > \bar{C}_b$, the proof is similar to the corresponding situation in Proposition 4. When $C_b^L = \bar{C}_b$, Lemma 11 implies that $\delta_b^{H3} - \delta_{b-1}^{H3} \leq \sum_{i=1}^{b}\delta_i^*$.

Since $p_{\beta_{b-1}} \geq \delta_{b-1}^{H3} > \delta_{b-2}^{H3}$, $\delta_{b-1}^{H3} \leq (P_{b-1}-\delta_{b-2}^{H3})/3$ follows from Lemma 8. As previously noted, $\delta_b^{H3} \leq (P_b - \delta_{b-1}^{H3})/3$. Also, from Lemma 6, $3\delta_{b-3}^{H3}+3\delta_{b-2}^{H3} \leq 3P_{b-2}$. Thus,

$$\frac{4P_{b-3}+3P_{b-2}+2P_{b-1}+P_b-\delta_{b-1}^{H3}+\delta_b^{H3}}{4P_{b-3}+3P_{b-2}+2P_{b-1}+P_b+\sum_{i=1}^{b}\delta_i^*} + \frac{\delta_{b-3}^{H3}+\delta_{b-2}^{H3}+2\delta_{b-1}^{H3}}{4P_{b-3}+3P_{b-2}+2P_{b-1}+P_b}$$
$$\leq 1 + \frac{\delta_{b-3}^{H3}+\delta_{b-2}^{H3}+2\delta_{b-1}^{H3}}{4\delta_{b-3}^{H3}+\left(3\delta_{b-3}^{H3}+3\delta_{b-2}^{H3}\right)+\left(2\delta_{b-2}^{H3}+6\delta_{b-1}^{H3}\right)+\left(\delta_{b-1}^{H3}+3\delta_b^{H3}\right)} \leq \frac{6}{5}.$$

Hence, the result is established from (7).

Case 4. In σ^{H3}, batch $b-1$ completes when job $\beta_{b-1}+3$ or larger finishes processing.

When job $\beta_{b-1}+3$ or larger completes last in batch $b-1$, Lemma 7 implies that $\delta_{b-1}^{H3} \leq P_{b-1}/4$. As a result, the proof is similar to Case 3. ■

We now prove the main result of this section.

Theorem 3. For problem $P2\|\sum C_{B_i}$, $z^{H3}/z^* \leq 6/5$ *and this bound is tight.*

Proof. From Propositions 1, 3, 4, and 5, $z^{H3}/z^* \leq 6/5$ when the last job in batch b to complete in σ^{H3} is β_b, β_{b+1}, and β_{b+2}, respectively. We now show that (7) is satisfied when the last job in batch b to complete in σ^{H3} is β_{b+3} or larger. From Lemma 7, $4\delta_b^{H3} \leq P_b$. Since $\delta_{b-1}^{H3}+3\delta_b^{H3} \leq 4\delta_b^{H3}$ follows from Proposition 2, the proof is now similar to Propositions 4 and 5.

Now, we show that the bound is tight. Consider the instance where $b=2$, $n_1=1$, $n_2=3$, $p_1=1$, $p_2=3$, and $p_3=p_4=2$ (see Figure 6). The optimal schedule $\sigma^*=(\sigma_1^*,\sigma_2^*)$ where $\sigma^*=(1,2)$ and $\sigma_2^*=(3,4)$, and $z^*=1+4=5$. However, $\sigma^{H3}=(\sigma_1^{H3},\sigma_2^{H3})$ where $\sigma_1^{H3}=(1,3)$ and $\sigma_2^{H3}=(2,4)$. The solution value is $z^{H3}=1+5=6$. Hence, $z^{H3}/z^*=6/5$, and the bound is tight. ■

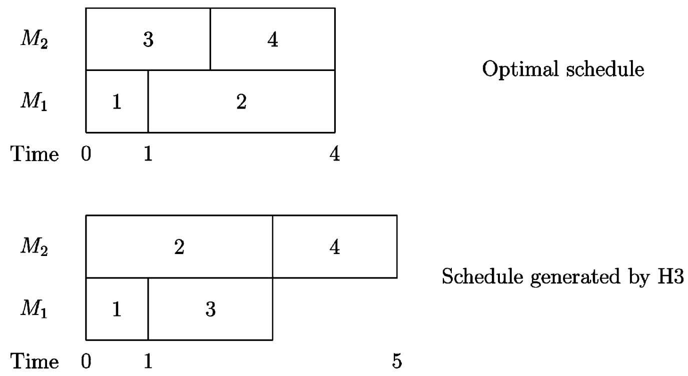

Figure 6. A worst case example of H3

6. COMPUTATIONAL STUDY OF HEURISTICS H2 AND H3

We empirically evaluate heuristics H2 and H3 by comparing solution values generated by the heuristics to an optimal LP solution value z^{LP}. For notational convenience, we let H* be the heuristic that selects the best schedule from those produced by H2 and H3. As performance indicators of H2, H3, and H*, we use upper bounds on relative errors z^{H2}/z^{LP}, z^{H3}/z^{LP}, and z^{H^*}/z^{LP}, respectively. Another performance indicator that we use is the number of test problems for which $z^{H2} \leq z^{H3}$ and $z^{H2} \geq z^{H3}$.

In this computational study, we compare the performances of H2 and H3 under various conditions. We also observe the impact of different factors such as b, n_i, p_j, and $E(n)$ on the performances of H3 and H2, where $E(\cdot)$ is the expectation operator.

For each problem instance, $n_i, \sim DU[1, \bar{n}]$, and $p_j \sim DU[p^{LB}, p^{UB}]$, where $\bar{n}$, p^{LB}, and p^{UB} are parameters and where $DU[\ell, u]$ represents a discrete random variable uniformly distributed between ℓ and u. For a given set of test problems, b is fixed. It follows that $E(n_i) = (1 + \bar{n})/2$ and $E(n) = bE(n_i) = b(1 + \bar{n})/2$.

We generate 750 test problems under 25 conditions. To test the effects of varying $E(n)$, we consider three different values of $E(n)$: 16, 100, and 2500. To determine whether different combinations of b and n_i, have an impact on the performance of the heuristics, we consider five different combinations of b and n_i, for a given value of $E(n)$. It is also possible that the standard deviation of the p_js may affect the performance of the heuristics. Consequently, when $E(n) = 100$, we consider three different distributions of p_j: $p_j \sim DU[l, 99]$, $p_j \sim DU[25, 75]$, and $p_j \sim DU[40, 60]$. For each combination of the different factors, we solve 30 problems. Table 1 presents a summary of the design for the computational study.

The results for the cases where $p_j \sim DU[1, 99]$ are presented in Table 2. The average relative error bound is the average ratio of the solution value of a heuristic to an optimal LP solution. Since each design point has 30 replications, the average relative error is calculated over 30 test problems. When $n = 1$, all average relative error bounds are equal to z^*/z^{LP} because the heuristics produce an optimal schedule (Blocher and Chhajed, 1996). Also, when $b = 1$, both heuristics use the LPT rule to determine machine-job assignment for each batch. Therefore, $z^{H2} = z^{H3}$.

We now summarize the results of our study. For both heuristics, the average relative error bound decreases as $E(n)$ increases. While there may be errors due to the use of z^{LP} instead of the optimal value, the results indicate that both heuristics perform better as the number of jobs increases. For

Table 1. Design for the computational study

$p_j \sim DU[1, 99]$						$p_j \sim DU[25, 75]$		$p_j \sim DU[40, 60]$	
$E(n) = 16$		$E(n) = 100$		$E(n) = 2500$		$E(n) = 100$		$E(n) = 100$	
b	$\bar{n}$	b	$\bar{n}$	b	$\bar{n}$	b	$\bar{n}$	b	$\bar{n}$
1	31	1	199	1	4999	1	199	1	199
2	15	4	49	10	499	4	49	4	49
4	7	10	19	50	99	10	19	10	19
8	3	25	7	250	19	25	7	25	7
16	1	100	1	2500	1	100	1	100	1

Table 2. Performance of the heuristics

$p_j \sim DU[1, 99]$			Average relative error bound			Number of problems	
$E(n)$	b	$\bar{n}$	H2	H3	H*	$z^{H2} \leq z^{H3}$	$z^{H2} \geq z^{H3}$
16	1	31	1.0352	1.0352	1.0352	30	30
	2	15	1.0558	1.0393	1.0370	18	19
	4	7	1.0690	1.0566	1.0541	15	17
	8	3	1.0823	1.0772	1.0751	16	17
	16	1	1.0910	1.0910	1.0910	30	30
100	1	199	1.0338	1.0338	1.0338	30	30
	4	49	1.0098	1.0048	1.0046	12	23
	10	19	1.0123	1.0079	1.0075	11	19
	25	7	1.0111	1.0103	1.0100	9	21
	100	1	1.0150	1.0150	1.0150	30	30
2500	1	4999	1.0000	1.0000	1.0000	30	30
	10	499	1.0000	1.0000	1.0000	20	25
	50	99	1.0001	1.0001	1.0001	8	23
	250	19	1.0003	1.0003	1.0003	9	21
	2500	1	1.0006	1.0006	1.0006	30	30

large problems, these methods routinely find solutions that are close to optimal. Also, for a given $E(n)$, the average relative error bound increases as b increases (as $\bar{n}$ decreases). Hence, for a given number of jobs, the heuristics perform better when there are fewer batches.

For each design point, the average relative error bound for H3 is less than or equal to that for H2. This suggests that H3 performs better than H2 when $p_j \sim DU[1, 99]$. This result is supported by the performance measure which gives the number of test problems for which $z^{H2} \leq z^{H3}$.

We compare the average relative error bounds of the two heuristics when the standard deviation (s.d.) of p_j changes in Table 3. For each design point in Table 3, the average relative error bound for H3 increases slightly as the standard deviation of p_j decreases. Also for each design point in Table 3, the average relative error bound for H3 is about the same as that of H2 when $p_j \sim DU[25, 75]$

Table 3. Sensitivity of the error bounds to processing time variance

$E(n) = 100$		$p_j \sim DU[I, 99]$ (s.d. = 28.28)		$p_j \sim DU[25, 75]$ (s.d. = 14.43)		$p_j \sim DU[40, 60]$ (s.d. = 5.77)	
b	$\bar{n}$	H2	H3	H2	H3	H2	H3
1	199	1.0338	1.0338	1.0365	1.0365	1.0378	1.0378
4	49	1.0098	1.0048	1.0087	1.0075	1.0108	1.0104
10	19	1.0123	1.0079	1.0108	1.0104	1.0126	1.0129
25	7	1.0111	1.0103	1.0091	1.0104	1.0110	1.0112
100	1	1.0150	1.0150	1.0120	1.0120	1.0106	1.0106

Table 4. Sensitivity of the number of problems where H2 and H3 are best to processing time variance

$E(n) = 100$		$p_j \sim DU[I, 99]$ (s.d. = 28.28)		$p_j \sim DU[25, 75]$ (s.d. = 14.43)		$p_j \sim DU/[40, 60]$ (s.d. = 5.77)	
b	$\bar{n}$	$z^{H2} \leq z^{H3}$	$z^{H2} \geq z^{H3}$	$z^{H2} \leq z^{H3}$	$z^{H2} \geq z^{H3}$	$z^{H2} \leq z^{H3}$	$z^{H2} \geq z^{H3}$
1	199	30	30	30	30	30	30
4	49	12	23	19	11	23	11
10	19	11	19	18	14	21	10
25	7	9	21	27	3	18	13
100	1	30	30	30	30	30	30

and $p_j \sim DU[40, 60]$. In Table 4, we compare the number of test problems where $z^{H3} \leq z^{H2}$ and $z^{H3} \geq z^{H2}$. Observe when $p_j \sim DU[25, 75]$ and $p_j \sim DU[40, 60]$, the number of test problems with $z^{H2} \leq z^{H3}$ is greater than the number of test problems with $z^{H2} \geq z^{H3}$. This suggests that H2 may be a better choice than H3 when the standard deviation of p_j is small.

7. DISCUSSION AND FURTHER RESEARCH

We propose two conjectures regarding H2 and H3. The following conjecture suggests that a tighter bound on the relative error can be found if we use H2 and H3 together.

Conjecture 1. Given instance of problem $P2\|\sum C_{B_i}$, *min*$\{z^{H2}/z^*, z^{H3}/z^*\} \leq 7/6$ *and this bound is tight.*

In our experiments, each heuristic eliminates the other heuristic's worst case. As a result, we are not able to find a case where both heuristics simultaneously perform poorly. (It would be interesting to prove this fact.) Also, note that $7/6$ is the worst case bound of the LPT rule for problem $P2\|C_{\max}$ (Graham, 1969). As a result, this is a lower bound for our combined heuristic.

The next conjecture suggests an extension of H3 to problem $P\|\sum C_{B_i}$.

Conjecture 2. For problem $P\|\sum C_{B_i}$, $z^{H3}/z^* \leq (4m-2)/(3m-1)$ *and this bound is tight.*

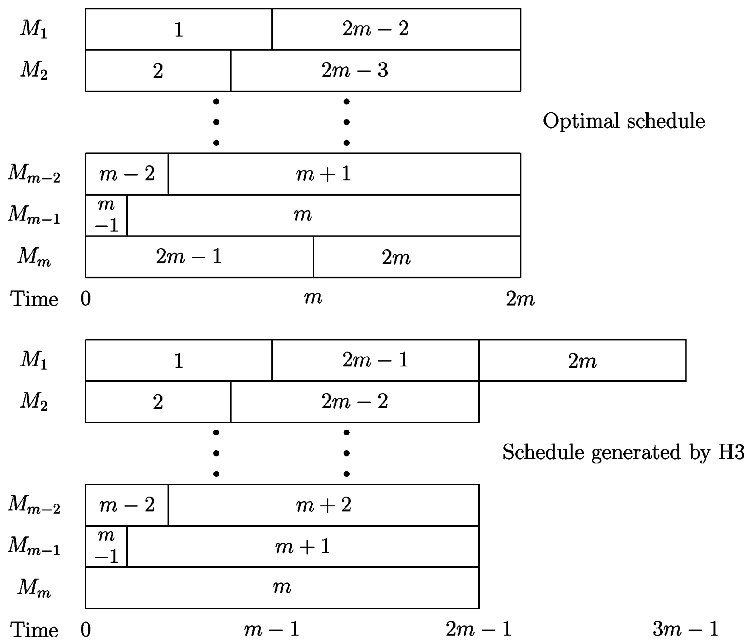

Figure 7. An example of H3 for problem $P\|\sum C_{B_i}$

The following example of H3, which we believe to be the worst case, suggests the conjecture. Let $B_1 = \{1, 2, \ldots, m-1\}$, $B_2 = \{m, m+1, \ldots, 2m\}$, $p_j = m - j$ for $j = 1, 2, \ldots, m-1$, $p_j = 3m - j - 1$ for $j = m, m+1, \ldots, 2m-2$, and $p_{2m-1} = p_{2m} = m$. An optimal schedule is $\sigma^* = (\sigma_1^*, \sigma_2^*, \ldots, \sigma_m^*)$ where $\sigma_j^* = (j, 2m-1-j)$ for $j = 1, 2, \ldots, m-1$ and $\sigma_m^* = (2m-1, 2m)$, and $z^* = (m-1) + 2m = 3m-1$ (see Figure 7). However, $\sigma^{H3} = (\sigma_1^{H3}, \sigma_2^{H3}, \ldots, \sigma_m^{H3})$ where $\sigma_1^{H3} = (1, 2m-1, 2m)$, $\sigma_j^{H3} = (j, 2m-j)$ for $j = 2, 3, \ldots, m-1$, and $\sigma_m^{H3} = (m)$. The corresponding solution value is $z^{H3} = (m-1) + (3m-1) = 4m-2$ (see Figure 7). Hence, $z^{H3}/z^* = (4m-2)/(3m-1)$ for the given example. Notice that this example is a direct extension of the worst case example of H3 for problem $P2\|\sum C_{B_i}$.

Observe that $(4m-1)/(3m)$ is the worst case bound of the LPT rule for problem $P2\|C_{\max}$ (Graham, 1969).

In addition to establishing Conjectures 1 and 2, it still has to be determined whether the complexity of problem $P2\|\sum C_{B_i}$ is unary NP-complete or pseudo-polynomial. We leave these issues to future research.

We have explored three heuristics for the parallel machine customer order scheduling problem. Tight worst case bounds on the relative error are established. These problems are simpler than many batch scheduling problems because the composition of the batches is prespecified. Also, the objective is concerned with batch completion times instead of job completion times. While the structure is simple, the problem has various real world applications such as scheduling customer orders, scheduling the production of components for subsequent assembly into final products, crane scheduling at a port, and automotive repair shop scheduling. We hope that our results can be used to develop solution procedures for more complex and realistic applications.

REFERENCES

Baker, K. R., "Scheduling the production of components at a common facility," *IIE Trans.*, **20**, 32–35 (1988).

Blocher, J. D. and D. Chhajed, "The customer order lead time problem on parallel machines," *Naval Res. Logist.*, **43**, 629–654 (1996).

Blocher, J. D., D. Chhajed, and M. Leung, "Customer order scheduling in a general job shop environment," *Dec. Sci.*, **29**, 951–981 (1998).

Coffman, E. G., A. Nozari, and M. Yannakakis, "Optimal scheduling of products with two subassemblies on a single machine," *Opns. Res.*, **37**, 426–436 (1989).

Daganzo, C. F., "The crane scheduling problem," *Transpn. Res.-B*, **23B**, 159–175 (1989).

Gerodimos, A. E., C. A. Glass, and C. N. Potts, "Scheduling the production of two-component jobs on a single machine," *Euro. J. Opnl. Res.*, **120**, 250–259 (2000).

Graham, R. L., "Bounds for certain multiprocess anomalies," *Bell Sys. Tech. J.*, **45**, 1563–1581 (1966).

Graham, R. L., "Bounds on multiprocessing timing anomalies,' *SIAM J. Appl. Math.*, **17**, 416–429 (1969).

Graham, R. L., E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan, "Optimization and approximation in deterministic machine scheduling: A survey," *Ann. Disc. Math.*, **5**, 287–326 (1979).

Gupta, J. N. D., J. C. Ho, and A. A. van der Veen, "Single machine hierarchical scheduling with customer orders and multiple job classes," *Ann. Oper. Res.*, **70**, 127–143 (1997).

Jordan, C., *Batching and Scheduling: Models and Methods for Several Problem Classes*, Springer, New York, 1996.

Julien, F. M. and M. J. Magazine, "Scheduling customer orders: An alternative production scheduling approach," *J. Mfg. Oper. Mgt.*, **3**, 177–199 (1990).

Peterkofsky, R. L. and C. F. Daganzo, "A branch and bound solution method for the crane scheduling problem," *Transpn. Res.-B*, **24B**, 159–172 (1990).

Potts, C. N. and L. N. Van Wassenhove, "Integrating scheduling with batching and lot-sizing: A review of algorithm and complexity," *J. Opnl. Res. Soc.*, **43**, 395–406 (1992).

Santos, C. and M. Magazine, "Batching in single operation manufacturing systems," *OR Lett.*, **4**, 99–103 (1985).

Vickson, R. G., M. J. Magazine, and C. A. Santos, "Batching and sequencing of components at a single facility," *IIE Trans.*, **25**, 65–70 (1993).

Webster, C. and K. R. Baker, "Scheduling groups of jobs on a single machine," *Opns. Res.*, **43**, 692–703 (1995).