POSTERIOR WORST-CASE BOUNDS FOR LPT SCHEDULES

Johnny C. Ho, Turner College of Business, Columbus State University, Columbus, GA 31907, USA
Ivar Massabò, Dipartimento di Economia Statistica e Finanza, Università della Calabria, 87036 Arcavacata di Rende (CS), Italy
Giuseppe Paletta, Dipartimento di Economia Statistica e Finanza, Università della Calabria, 87036 Arcavacata di Rende (CS), Italy
Alex J. Ruiz-Torres, Departamento de Gerencia, Facultad de Administración de Empresas, Universidad de Puerto Rico – Rio Piedras, San Juan PR, 00931-3332, USA

ABSTRACT

This paper proposes and analyzes a posterior tight worst-case bound for the Longest Processing Time (LPT) heuristic for scheduling independent jobs on identical parallel machines to minimize the makespan, and makes remarks that improve the well-known posterior worst-case bounds previously proposed in the literature when the makespan is realized on multiple machines. Our theoretical and computational comparative analysis shows that the proposed bound can complement the well-known posterior bounds to synergistically achieve a better posterior worst-case bound for the LPT heuristic. Moreover, the proposed bound can be used to further show that LPT schedules are asymptotically optimal.

1. INTRODUCTION

The paper considers the classic problem of identical parallel machine scheduling with the objective of minimizing the makespan, $P || C_{\text{max}}$ for short, see Graham et al. (1979). In $P || C_{\text{max}}$, a set $J = \{1, \ldots, j, \ldots, n\}$ of $n$ simultaneously available independent jobs, with processing times $p_j > 0$, $j \in J$, are scheduled on a set $M = \{1, \ldots, i, \ldots, m\}$ of $m$ identical machines. Each machine can process at most one job at a time, and each job must be processed without interruption by exactly one of the $m$ machines. The goal is to allocate every job to exactly one machine, so that the makespan, i.e., maximum time required by any of the machines to complete all the jobs, is minimized.

$P || C_{\text{max}}$ is strongly NP-Hard for an arbitrary $m \geq 2$, see Garey and Johnson (1979), and it is one of the most studied problems from the viewpoint of approximation algorithms. Many polynomial time algorithms, having a known worst-case performance ratio, have been developed for $P || C_{\text{max}}$, see Cheng and Sin (1990), Lawler et al. (1993), Chen et al. (1998), and Mokotoff (2001) for an overview. The first important approach is List Scheduling, see Graham (1966, 1969), that includes Largest Processing Time (LPT). Other known constructive methods are MultiFit (MF), see Coffman et al. (1978), Largest Differencing Method (LDM), see Karmarkar and Karp (1982), and Multi-Subset, see Dell’Amico and Martello (1995). A more recent algorithm, that is based on the idea of combining iteratively partial solutions, is described in Paletta and Pietramala (2007) and Paletta and Vocaturo (2010).

This paper deals with the worst-case approximation ratio of the LPT heuristic that considers the jobs sorted in non-increasing order with respect to their processing times, and iteratively assigns each job to the least loaded machine. Let $C^{\text{LPT}}_{\text{max}}$ be the value of the makespan given by LPT heuristic, and $C_{\text{max}}$ be the optimal value of the makespan. Analysis of the worst-case performance of the LPT rule was done by Graham (1966, 1969), Coffman and Sethi (1976), Chen (1993), Blocher and Chand (1991), and Blocher and Sevastyanov (2015).

Graham (1966, 1969) computes the performance guarantee ratio of the LPT heuristic as
Coffman and Sethi (1976) establish the first posterior worst-case ratio:

\[
\frac{c_{LPT}^{\text{max}}}{c_{\text{max}}^*} \leq 1 + \frac{m-1}{3m},
\]

where \(k\) is the number of jobs assigned to the makespan machine (i.e., the machine where the makespan takes place).

Chen (1993) perfects this posterior worst-case ratio by proposing:

\[
\frac{c_{LPT}^{\text{max}}}{c_{\text{max}}^*} = \begin{cases} 
\frac{4}{3} - \frac{1}{3(m-1)} & \text{if } k = 1 \\
1 + \frac{m-1}{mk} & \text{if } k \geq 3.
\end{cases}
\]

Blocher and Sevastyanov (2015) propose the following posterior worst-case ratio:

\[
\frac{c_{LPT}^{\text{max}}}{c_{\text{max}}^*} = \begin{cases} 
\frac{4}{3} - \frac{1}{3(m-1)} & \text{if } K = 1 \\
1 + \frac{m-1}{mk} & \text{if } K \geq 3,
\end{cases}
\]

where \(K (K \geq k)\) is the maximum number of jobs on a machine in the truncated \(LPT\) schedule, i.e., in the \(LPT\) schedule obtained by assigning the first \(l\) jobs and neglecting the remaining jobs, where \(l\) is the index of the last job inserted in the makespan machine of the \(LPT\) schedule.

Observing that in three particular cases

i. \(k = 1\)
ii. \(m = 2\) and \(k = 2\)
iii. \(m \geq 3, k = 2,\) and \(K \leq 3\)

the Chen bound is better than the Blocher and Sevastyanov bound, thus the latter bound can be perfected by considering:

- for \(m = 2\)

\[
\frac{c_{LPT}^{\text{max}}}{c_{\text{max}}^*} = \begin{cases} 
1 & \text{if } k \leq 2 \\
1 + \frac{m-1}{mk} & \text{if } K \geq k \geq 3,
\end{cases}
\]

- for \(m \geq 3\)

\[
\frac{c_{LPT}^{\text{max}}}{c_{\text{max}}^*} = \begin{cases} 
\frac{4}{3} - \frac{1}{3(m-1)} & \text{if } k = 1 \\
1 + \frac{m-1}{mk} & \text{if } K = 2 \text{ and } K \leq 3 \\
1 + \frac{m-1}{mk} & \text{otherwise}.
\end{cases}
\]
Blocher and Chand (1991) propose the posterior worst-case ratio:

\[ \frac{C_{\text{max}}^{LPT}}{C_{\text{max}}} \leq \frac{C_{\text{max}}^{LPT}}{p_l \alpha^T}, \]

where

\[ \alpha^T = \left\lfloor \frac{\sum_{i=1}^t p_i}{m} \right\rfloor \]

is the number of jobs assigned to a makespan machine of the optimal schedule for the relaxed problem obtained by considering \( m \) machines and \( \sum_{j=1}^i \left\lfloor \frac{p_j}{p_l} \right\rfloor \) jobs each of them having identical processing time equal to \( p_l \). Here \( \lfloor x \rfloor \) denotes the smallest integer not less than \( x \), and \( \lceil x \rceil \) denotes the largest integer not greater than \( x \).

This paper provides the posterior tight worst-case performance ratio

\[ \frac{C_{\text{max}}^{LPT}}{C_{\text{max}}} \leq 1 + \frac{m - 1}{\rho}, \]

where

\[ \rho = \frac{1}{p_l} \sum_{j=1}^l p_j \]

represents the ratio between the total processing time of the first \( l \) jobs and the processing time \( p_l \) of job \( l \), and then it compares this bound with the existing posterior tight worst-case bounds on \( LPT \) schedule. This paper also provides remarks to improve the well-known posterior worst-case bounds when the makespan is realized on multiple machines. Moreover, by using the new bound, this paper provides a further proof that \( LPT \) schedule is asymptotically optimal for \( n \rightarrow \infty \). Excellent results regarding the asymptotical optimality of \( LPT \) schedules are provided in Ibarra and Kim (1977) and, by using the probabilistic analysis (see Coffman et al. 1988) in Frenk and Rinnooy Kan (1986, 1987).

### 2. A POSTERIOR TIGHT WORST-CASE PERFORMANCE RATIO

Let \( r \) be the machine that determines the makespan on an \( LPT \) schedule, and \( l \) be the index of the last job inserted in \( r \). Let \( C_i \) be the time employed by the machine \( i, i = 1, ..., m, \) to perform the set of jobs assigned to it by the \( LPT \) heuristic before assigning job \( l \), i.e., by excluding the last \( n - l + 1 \) jobs. It is well known that the \( LPT \) schedule satisfies:

\[ C_{\text{max}}^{LPT} = C_r + p_l \]

and

\[ C_r \leq \frac{1}{m} \sum_{j \in N \setminus \{l\}} p_j = \frac{1}{m} \sum_{j \in N} p_j - \frac{1}{m} p_l \leq C_{\text{max}}^* - \frac{1}{m} p_l. \tag{1} \]

Since the jobs are indexed so that \( p_1 \geq ... \geq p_n \), and \( mC_{\text{max}}^* \geq \sum_{j=1}^n p_j \geq lp_l \), it follows that

\[ p_l \leq \frac{m}{l} C_{\text{max}}^*. \]
Moreover, by using the parameter $\rho$ that has been introduced by Massabó et al. (2015) in a uniform machine environment as the ratio between the total processing time of the first $l$ jobs and the processing time of job $l$

$$\rho = \frac{1}{l} \sum_{j=1}^{l} p_j$$

we have that $l \leq \rho$ because $p_1 \geq p_2 \geq \ldots \geq p_l$, and then

$$p_l = \frac{1}{\rho} \sum_{j=1}^{l} p_j \leq \frac{m}{\rho} c^*.$$  \hspace{1cm} (2)

**Theorem 1.** The makespan obtained by the $LPT$ heuristic satisfies

$$\frac{C^{LPT}_{max}}{C^*_{max}} \leq 1 + \frac{m-1}{\rho}.$$  

**Proof.** Since

$$C^{LPT}_{max} = C_r + p_l,$$

by (1) and (2) it follows that

$$C^{LPT}_{max} \leq C^*_{max} + \frac{m-1}{m} p_l \leq C^*_{max} + \frac{m-1}{m} \frac{m}{\rho} C^*_{max}$$

and then

$$\frac{C^{LPT}_{max}}{C^*_{max}} \leq 1 + \frac{m-1}{\rho},$$

which is the bound stated in the theorem.

**Theorem 2.** For each instance that satisfies the following properties:

1) $C^*_{max} = \frac{1}{m} \sum_{j=1}^{n} p_j$, and
2) $C^{LPT}_{max} = \frac{1}{m} \sum_{j=1}^{n} p_j + p_n$

$LPT$ makes the bound $1 + \frac{m-1}{\rho}$ tight.

**Proof.** For each instance that satisfies the properties 1) and 2), the $LPT$ heuristic gives $l = n$ and $C^{LPT}_{max} - C^*_{max} = p_n - \frac{p_n}{m}$, thus

$$\frac{C^{LPT}_{max}}{C^*_{max}} = 1 + \frac{p_n - \frac{p_n}{m}}{\frac{1}{m} \sum_{j=1}^{n} p_j} = 1 + \frac{p_n m - p_n}{\sum_{j=1}^{n} p_j} = 1 + \frac{m-1}{\frac{1}{p_n} \sum_{j=1}^{n} p_j} = 1 + \frac{m-1}{\rho}.$$  

This proves the theorem.

The following two classes of instances that satisfy the properties of Theorem 2 show that the posterior worst-case bound $1 + \frac{m-1}{\rho}$ is tight.
**Class 1** (from Graham, 1969). Consider the set of \( n = 2m + 1 \) jobs with processing times:

- \( p_i = 2m - \left\lfloor \frac{i}{2} \right\rfloor \), for \( i = 1, \ldots, 2m \), and
- \( p_{2m+1} = m \).

The \( LPT \) schedule yields \( C_{\text{max}}^{LPT} = 4m - 1 \) whereas \( C^{*}_{\text{max}} = 3m \), therefore

\[
\frac{C_{\text{max}}^{LPT}}{C^{*}_{\text{max}}} = \frac{4m - 1}{3m}.
\]

Now, the \( LPT \) schedule has \( l = n, p_l = m, \sum_{j=1}^{l} p_j = 3m^2 \), hence \( \rho = 3m \), therefore

\[
\frac{C_{\text{max}}^{LPT}}{C^{*}_{\text{max}}} = \frac{4m - 1}{3m} = 1 + \frac{m - 1}{\rho}.
\]

**Class 2.** Consider the set of \( n = 2\left\lfloor \frac{m}{2} \right\rfloor m + 1 \) jobs with processing times:

- \( p_1 = m + 2\left\lfloor \frac{m}{2} \right\rfloor \) and for \( k = 1, \ldots, \left\lfloor \frac{m}{2} \right\rfloor \),
- \( p_j = p_k - (2k - 1) \) for \( j = 2m(k - 1) + 2, \ldots, 2mk - 1 \), and
- \( p_{2mk} = p_{2mk+1} = p_1 - 2k \).

The \( LPT \) schedule yields \( C_{\text{max}}^{LPT} = 2\left\lfloor \frac{m}{2} \right\rfloor (m + \left\lfloor \frac{m}{2} \right\rfloor) + m \) whereas \( C^{*}_{\text{max}} = 2\left\lfloor \frac{m}{2} \right\rfloor (m + \left\lfloor \frac{m}{2} \right\rfloor) + 1 \), thus

\[
\frac{C_{\text{max}}^{LPT}}{C^{*}_{\text{max}}} = \frac{2\left\lfloor \frac{m}{2} \right\rfloor (m + \left\lfloor \frac{m}{2} \right\rfloor) + m}{2\left\lfloor \frac{m}{2} \right\rfloor (m + \left\lfloor \frac{m}{2} \right\rfloor) + 1} = 1 + \frac{m - 1}{2\left\lfloor \frac{m}{2} \right\rfloor (m + \left\lfloor \frac{m}{2} \right\rfloor) + 1}.
\]

Observing that the \( LPT \) schedule has \( l = n, p_l = m, \) and \( \sum_{j=1}^{l} p_j = m\left[2\left\lfloor \frac{m}{2} \right\rfloor (m + \left\lfloor \frac{m}{2} \right\rfloor) + 1 \right] \), so \( \rho = 2\left\lfloor \frac{m}{2} \right\rfloor (m + \left\lfloor \frac{m}{2} \right\rfloor) + 1 \), therefore

\[
\frac{C_{\text{max}}^{LPT}}{C^{*}_{\text{max}}} = 1 + \frac{m - 1}{2\left\lfloor \frac{m}{2} \right\rfloor (m + \left\lfloor \frac{m}{2} \right\rfloor) + 1} = 1 + \frac{m - 1}{\rho}.
\]

By using our bound it is easy to prove that the \( LPT \) schedules are asymptotically optimal for \( n \to \infty \).

**Theorem 3.** Given \( n \) jobs, if it exists an appropriate integer value \( s \) such that \( p_h < \sum_{j=h+1}^{n} p_j \) for each \( h = 1, \ldots, n-s \), then

\[
\frac{C_{\text{max}}^{LPT}}{C^{*}_{\text{max}}} \to 1
\]

when \( n \to \infty \).

**Proof.** If \( p_h < \sum_{j=h+1}^{n} p_j \) for each \( h = 1, \ldots, n-s \), when \( n \to \infty \) also \( l \) and \( \rho \to \infty \), therefore

\[
\frac{C_{\text{max}}^{LPT}}{C^{*}_{\text{max}}} \leq 1 + \frac{m-1}{\rho} \to 1.
\]

A particular class of instances that satisfies the condition \( p_h < \sum_{j=h+1}^{n} p_j \) for each \( h = 1, \ldots, n-s \), it is obtained when \( p_n \geq \frac{1}{n^y} \) with \( 0 < y \leq 1 \).
2.1 Theoretical Comparison Between the Bounds

It is important to compare theoretically our bound with the existing posterior bounds. By comparing our bound with the Blocher and Sevastyanov bound, we obtain

\[ 1 + \frac{m - 1}{mK} \leq 1 + \frac{m - 1}{\rho} \quad \text{when} \quad K \geq \frac{\rho}{m}, \]

on the contrary

\[ 1 + \frac{m - 1}{mK} \geq 1 + \frac{m - 1}{\rho} \quad \text{when} \quad K \leq \frac{\rho}{m}. \]

By comparing the Blocher and Chand bound with our bound, we obtain that

\[ \frac{C_{\text{LPT}}^{\text{max}}}{p_l \alpha^l} \leq 1 + \frac{m - 1}{\rho} \quad \text{when} \quad \alpha^l \geq \frac{\rho C_{\text{LPT}}^{\text{max}}}{p_l (\rho + m - 1)}, \]

on the contrary

\[ \frac{C_{\text{LPT}}^{\text{max}}}{p_l \alpha^l} \geq 1 + \frac{m - 1}{\rho} \quad \text{when} \quad \alpha^l \leq \frac{\rho C_{\text{LPT}}^{\text{max}}}{p_l (\rho + m - 1)}. \]

These comparisons show that our worst-case bound can complement the existing worst-case bounds to synergistically achieve a better bound.

**Remark.** Coffman and Sethi, Chen, Blocher and Chand, and Blocher and Sevastyanov papers do not provide any guideline in case there are multiple machines that yield the maximum completion time. Observing that when the makespan is realized on multiple machines, and if \( L \) is the set of indexes of the last jobs inserted in the makespan machines, we conclude that:

- our bound, and the Blocher and Sevastyanov bound can be improved by using the largest index \( l \) belonging to \( L \);
- the Blocher and Chand bound can be improved by using the \( l \) that makes maximum \( p_l \cdot \alpha^l \), i.e., \( l = \arg \max_{j \in L} (p_j \cdot \alpha^j) \); and
- the Coffman and Sethi, and Chen bounds do not depend on the index \( l \), but they can be improved by using the largest number of jobs assigned to the makespan machines.

2.2 Computational Study

This section presents a series of computational experiments conducted to evaluate the performance of our posterior worst case bound (HMPT) relative to the bounds proposed by Blocher and Chand (BC), and by Blocher and Sevastyanov (BS), and to evaluate the considerations made when the makespan is realized on multiple machines. The Chen bound that perfects Coffman and Sethi bound was not considered because it is dominated by the Blocher and Sevastyanov bound. The computational experiments used the UNIFORM family of instances that have been presented in França et al. (1994).

The instances are characterized by three parameters used to randomly generate the integer processing times: the number of machines \( m \in \{5, 10, 25\} \), the number of jobs \( n \in \{50, 100, 500, 1000\} \) and the processing time interval \( D \in \{[1, 100], [1, 1000], [1, 10000]\} \). For each of 36 combinations of \( m, n, \) and \( D \), there exist 10 instances for a total of 360 instances.
According to our computational results, HMPT gives the lowest (best) average bound equal to 1.042, and the largest (best) average number of best bounds equal to 0.881, corresponding to 317 instances out of 360; BC gives an average bound equal to 1.075, and an average number of best bounds equal to 0.131 corresponding to 47 instances; while BS gives an average bound equal to 1.085, and an average number of best bounds equal to 0.017 corresponding to 6 instances.

3. CONCLUSION

This paper provides the posterior worst-case bound $1 + \frac{m-1}{\rho}$, where $\rho = \frac{1}{p_l} \sum_{j=1}^{l} p_j$, and $l$ is the largest index of the latest jobs inserted in the makespan machine, and shows that this bound is tight. It is shown, both theoretically and computationally, that our worst-case performance ratio can complement the existing bounds to synergistically achieve a better bound. This paper also provides some remarks that improve the well-known posterior worst-case bounds. Moreover, our bound can be used to further show that LPT schedules are asymptotically optimal for $n \to \infty$.

REFERENCES


