A GENERALIZED BOUND ON LPT SEQUENCING*

E. G. Coffman, Jr. and Ravi Sethi
Computer Science Department
The Pennsylvania State University
University Park, PA. 16802

INTRODUCTION

For a number of NP-complete sequencing problems [C], the worst-case performance of heuristics has been bounded relative to optimal performance. The bounds are usually shown to be best in the sense that they are achievable. However, when these bounds are based on a single, pathological example, they are not as informative as might be desired. Such is largely the case with Graham's bound [G] of $4/3 - 1/3m$ on the performance of largest-processing-time-first (LPT) sequencing for the classical problem of minimizing schedule lengths, assuming independent tasks on $m \geq 2$ identical processors.

In particular, one is led to expect much better performance from the LPT rule than is indicated by the general bound, especially as the number of tasks becomes large. In this paper we shall generalize Graham's result so as to include a parameter characterizing the number of tasks assigned to processors by the LPT rule. The new result will show that the worst-case performance bound for LPT sequencing approaches unity approximately as $1 + 1/k$, where k is the least number of tasks on any processor, or where k is the number of tasks on a processor whose last task terminates the schedule. Thus, we shall have a result very similar to the parameterized bounds for bin-packing heuristics [JDUGG]. We shall also obtain out of the analysis an alternate proof of Graham's result.

MODEL AND RESULTS

We use the classical model [C] in which the tasks of $\mathcal{J} = \{T_1, T_2, \ldots, T_n\}$ are independent and assumed to be simultaneously available for (non-preemptive) execution beginning at time $t = 0$. There are $m \geq 2$ identical processors $P_1, P_2, \ldots, P_m$ available for executing the tasks; the execution time of T_i, $1 \leq i \leq n$, will be denoted by τ_i. A processor cannot remain idle if there is a task to be executed. Schedules will be represented by timing diagrams as in Figure 1.

We let $s(T)$ and $f(T)$ denote respectively the starting and finishing time of task T in a given schedule (understood by context). We shall regard a processor P as an ordered set of the tasks assigned to it; thus $|P|$ denotes the number of tasks assigned to P in a given schedule.

If in a schedule S, $s(T_i) \leq s(T_j)$ implies $\tau_i \geq \tau_j$ for all $1 \leq i,j \leq n$, then S is called an LPT schedule. Figure 1 shows an example. Note that there may be many LPT schedules depending on how ties are resolved; however, all such schedules have the same length. We shall let S_L and S_0 denote LPT and minimum-length (optimal) schedules, respectively, and ω_L and ω_0 the respective schedule lengths.

There is more than one way to state the main result that follows. Initially, we shall use a simpler, more readily interpreted statement, although it is somewhat weaker than necessary. In a subsequent corollary a stronger statement will be seen to be subject to essentially the same proof.

Theorem: Let S_L be an LPT schedule for $\mathcal{J}$ on m processors such that $|P_i| \geq k \geq 1$ for all $1 \leq i \leq m$. Then

$$\frac{\omega_L}{\omega_0} \leq \frac{k+1}{k} - \frac{1}{mk}. \qquad (1)$$

---

*This research was supported in part by NSF Grant #28290.

τ_i : 8,6½,6,4,3,2½,2½

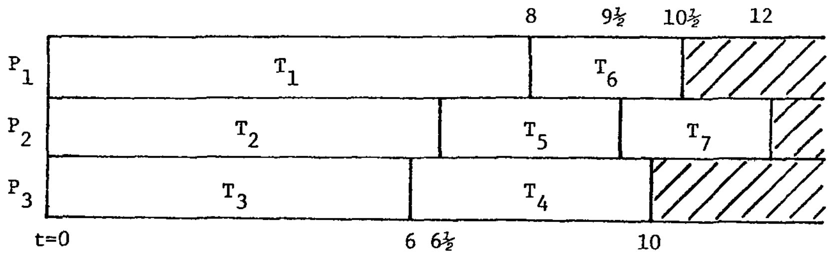


Figure 1. Example Timing Diagram (m=3)

Moreover, this bound can be approached arbitrarily closely for all $m \ge 1$ and $k \ge 3$.

Proof: The result is obviously true for $m = 1$ and Graham's result verifies that the theorem is true for $k \le 3$. Thus, in the sequel we restrict ourselves to $m > 1$ and $k > 3$.

The proof will be by contradiction; i.e., let us suppose that $\mathcal{T}$ violates (1), assuming S_L is such that $|P_i| \ge k$ for all i. Let T^* denote a smallest task in $\mathcal{T}$ such that $f(T^*) = \omega_L$, and let P^* in S_L denote the processor on which T^* is assigned. We first show that violation of (1) implies that exactly k tasks are scheduled on P^* in S_L.

By definition we can write

$$\omega_L = s(T^*) + \tau^*.$$

From the properties of the LPT rule we know that all processors are busy up to time $s(T^*)$, and hence

$$s(T^*) \le \frac{1}{m} \sum_{T \ne T^*} \tau .$$

Thus,

$$\omega_L \le \frac{1}{m} \sum_{T \ne T^*} \tau + \tau^* = \frac{1}{m} \sum_{i=1}^{n} \tau_i + \frac{m-1}{m} \tau^*$$

and since $\omega_0 \ge \frac{1}{m} \sum_{i=1}^{n} \tau_i$ we have

$$\omega_L \le \omega_0 + \frac{m-1}{m} \tau^*. \qquad (2)$$

From (2) we can write

$$\frac{\omega_L}{\omega_0} \le \frac{\omega_L}{\omega_L - \frac{m-1}{m} \tau^*} .$$

Hence, violation of (1) implies

$$\frac{\omega_L}{\omega_L - \frac{m-1}{m} \tau^*} > \frac{k+1}{k} - \frac{1}{mk} .$$

On manipulation this yields

$$\tau^* > \frac{1}{k+1-1/m} \omega_L > \frac{1}{k+1} \omega_L$$

from which we conclude that P^* can have at most k tasks. Since by hypothesis P^* must have at least k, we obtain $|P^*| = k$.

Now suppose

$$\omega_0 \ge k\tau^*. \qquad (3)$$

From (2) we have

$$\frac{\omega_L}{\omega_0} \le \frac{\omega_0 + \frac{m-1}{m}\tau^*}{\omega_0} = 1 + \frac{m-1}{m} \frac{\tau^*}{\omega_0} .$$

Substituting (3) into this inequality gives us (1), and with this contradiction we are done. Thus, it remains to show that (3) holds, assuming (1) is violated.

Define $\mathcal{T}' = \mathcal{T} - \{T \ne T^* | s(T) \ge s(T^*) \text{ in } S_L\}$; i.e., $\mathcal{T}'$ is the subset of $\mathcal{T}$ which remains after discarding all tasks that start no earlier than T^*, and hence are no larger than T^*. Let S_L' and S_0' be LPT and optimal schedules for $\mathcal{T}'$ and let ω_L' and ω_0' be the corresponding schedule lengths. Note that P^* is unchanged in S_L', except possibly for a renaming of equal-length tasks. We shall prove that $\omega_0' \ge k\tau^*$; since $\omega_0 \ge \omega_0'$, (3) will follow immediately. We proceed with a proof by contradiction and begin with the following result.

Claim 1: In S'_L let P be an arbitrary processor other than P^*, and let $T(i)$, $1 \le i \le \ell = |P|$, denote the i^{th} task scheduled on P in S'_L. Let $\tau(i)$ denote the execution time of $T(i)$. Define $T^*(i)$ and $\tau^*(i)$, $1 \le i \le k$, similarly for P^* and suppose $\ell \le k-2$. (Note that $T^* \equiv T^*(k)$.) If $\omega'_0 < k\tau^*$, then

$$\tau^*(i) \ge \tau(i+1) \qquad 1 \le i \le \ell-1. \qquad (4)$$

Proof: From the properties of the LPT rule we can establish the claim by showing alternatively that

$$s(T^*(i)) \le s(T(i+1)) \qquad 1 \le i \le \ell-1$$

or

$$\sum_{j=1}^{i-1} \tau^*(j) \le \sum_{j=1}^{i} \tau(j) \qquad 1 \le i \le \ell-1$$

where the sum on the left is defined to be 0 for $i = 1$. But suppose for some i, $1 \le i \le \ell-1$,

$$\sum_{j=1}^{i-1} \tau^*(j) > \sum_{j=1}^{i} \tau(j).$$

Since $\Sigma_{j=1}^{i}\tau(j) \ge i\tau^*$, we have $\Sigma_{j=1}^{i-1}\tau^*(j) \ge i\tau^*$, and hence

$$\sum_{j=1}^{k} \tau^*(j) = \sum_{j=1}^{i-1} \tau^*(j) + \sum_{j=1}^{k} \tau^*(j) \ge i\tau^* + (k-i+1)\tau^*$$

or

$$\sum_{j=1}^{k} \tau^*(j) \ge (k+1)\tau^*.$$

Since all processors in S'_L are busy up to $s(T^*)$, we have $\omega'_0 > s(T^*) \ge k\tau^*$, contradicting the hypothesis $\omega'_0 < k\tau^*$. The claim must therefore be true. □

After the following result, which follows readily from Claim 1, we shall be nearly done.

Claim 2: With the same assumptions and notation of Claim 1 we have that $\omega'_0 < k\tau^*$ implies

$$\tau(1) \ge (k-\ell)\tau^*$$

and hence

$$\sum_{i=1}^{\ell} \tau(i) = \tau(1) + \sum_{i=2}^{\ell} \tau(i) \ge (k-\ell)\tau^* + (\ell-1)\tau^*$$

or

$$\sum_{i=1}^{\ell} \tau(i) \ge (k-1)\tau^*.$$

Proof: To verify this claim we use the definition of P^* to write

$$\sum_{i=1}^{k-1} \tau^*(i) \le \sum_{i=1}^{\ell} \tau(i)$$

and therefore

$$\tau(1) \ge \sum_{i=1}^{k-1} \tau^*(i) - \sum_{i=2}^{\ell} \tau(i)$$

or

$$\tau(1) \ge \sum_{i=1}^{\ell-1} [\tau^*(i) - \tau(i+1)] + \sum_{i=\ell}^{k-1} \tau^*(i).$$

Using (4) in the above we have from Claim 1

$$\tau(1) \ge \sum_{i=\ell}^{k-1} \tau^*(i) \ge (k-\ell)\tau^*$$

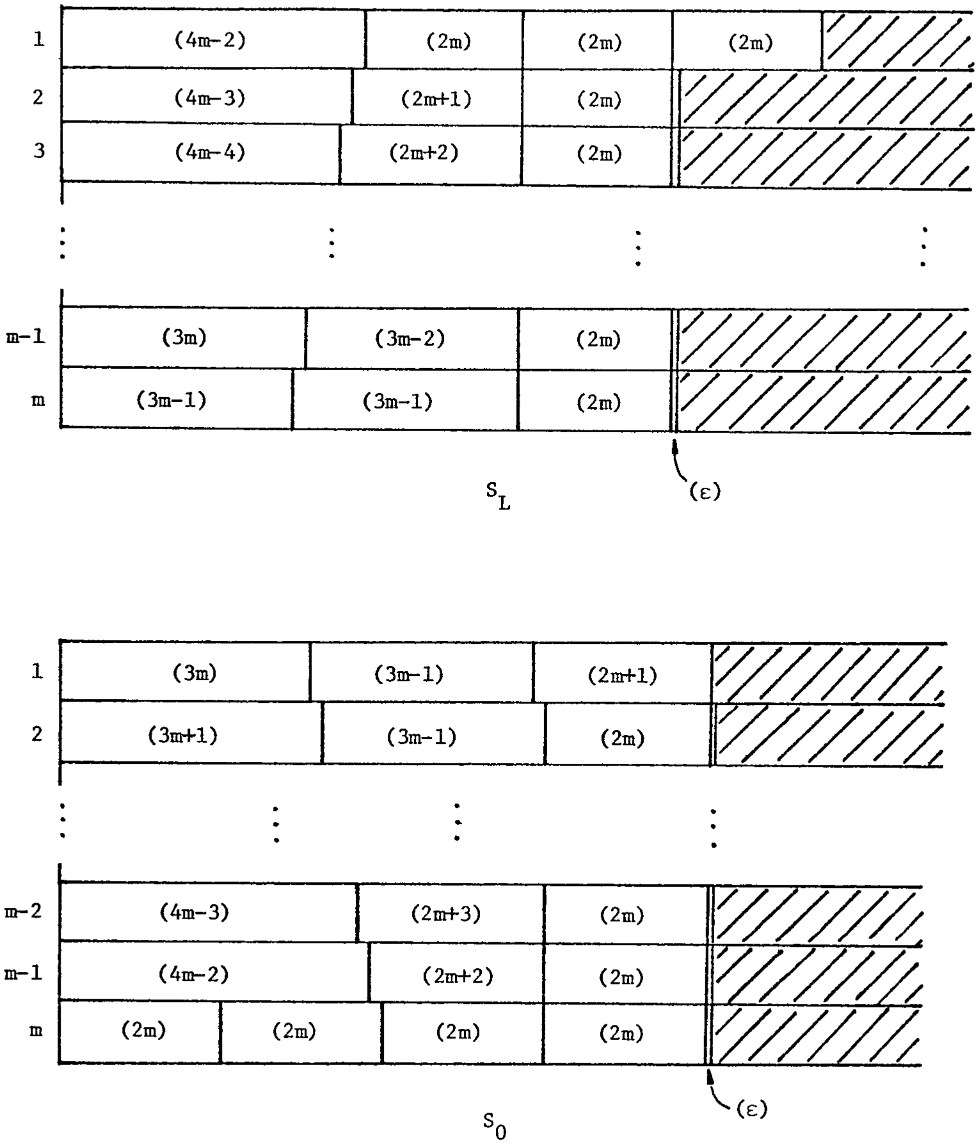
thus establishing Claim 2. □

To complete the proof of the theorem we consider the set $\mathcal{J}''$ of tasks formed as follows from $\mathcal{J}'$. We proceed in sequence through the processors $P_1, P_2, \ldots, P_m$ of S'_L. If $|P_i| \ge k-1$ all tasks on P_i are placed in $\mathcal{J}''$. If $|P_i| = \ell < k-1$ then place all non-initial tasks of P_i into $\mathcal{J}''$ plus $k-\ell$ new tasks of length τ^*. From Claim 2 we make the key observation that the sum of the lengths of these last $k-\ell$ tasks can not exceed the length of the initial (largest) task of P_i.

Let S''_0 be an optimal schedule of length ω''_0 for $\mathcal{J}''$. It is readily seen that $\omega''_0 \le \omega'_0$. Since P^* has k tasks, and since Claim 2 assures us that at least $k-1$ were placed into $\mathcal{J}''$ for each $P \ne P^*$, we have $|\mathcal{J}''| \ge m(k-1)+1$. Thus, S''_0 has at least one processor with k or more tasks and it follows that $\omega_0 \ge \omega'_0 \ge \omega''_0 \ge k\tau^*$. This contradiction completes the proof of the bound in (1).

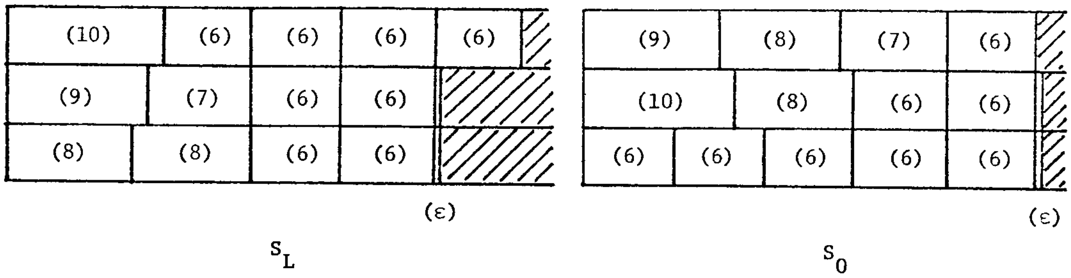
It remains to exhibit for each $m \ge 1$ and $k > 3$ examples for which ω_L/ω_0 is given by (1). For this consider the set $\mathcal{J}$ such that $n = km$ and

$$\tau_i = \begin{cases} 3m+i-2 & 1 \le i \le m \\ m+i-1 & m+1 \le i \le 2m \\ 2m & 2m+1 \le i \le (k-1)m+1 \\ \varepsilon & (k-1)m+2 \le i \le km \end{cases} .$$



$$\frac{\omega_L}{\omega_0} = \frac{5m-1}{4m+\varepsilon/2}$$

Figure 2. Illustrating Worst-Case Bound (k=4)



$m = 3$, $k = 5$

(Execution Times in Parentheses)

Figure 3. Example for $m = 3$, $k = 5$

Figure 2 shows the general example for $k=4$. The general example for arbitrary $k>4$ is obtained by adding the appropriate number of ranks of tasks having execution times $2m$. Figure 3 shows the specific example for $m=3$ and $k=5$. It is readily verified for these examples that

$$\lim_{\varepsilon\to 0} \frac{\omega_L}{\omega_0} = \frac{k+1}{k} - \frac{1}{km}$$

and hence the theorem is proved. □

We are now in a position to prove the slightly stronger result given in the following corollary.

<u>Corollary</u>: In an LPT schedule S_L suppose $f(T)=\omega_L$, T is assigned to P, and $|P|=k$. Then

$$\frac{\omega_L}{\omega_0} = 1, \qquad k = 1,2 \tag{5}$$

$$\frac{\omega_L}{\omega_0} \le \frac{k+1}{k} - \frac{1}{km}, \qquad k \ge 3 \tag{6}$$

That is, if there is a task terminating the schedule on a processor assigned k tasks, then (5) and (6) hold.

<u>Proof</u>: First, it is readily verified that the proof of the previous theorem carries through under the weaker condition of the corollary for $k \ge 3$. (One identifies T of the corollary with T^* in the proof of the theorem.)

All schedules must have a length at least as large as the largest task scheduled. Hence, (5) must be true for $k=1$. For $k=2$, let $T_{i_1},\ldots,T_{i_m}$ be those (largest) tasks assigned first on $P_1,\ldots,P_m$ in S_L. In any schedule at least two of $T, T_{i_1},\ldots,T_{i_m}$ must appear on the same processor. Of all the possibilities, scheduling T and T_{i_m} (the smallest of the T_{i_j}) together clearly provides the shortest schedule; hence, the length of S_L must be minimal.

Graham's worst-case example corresponds to $k=3$ and is given by [G],

$$\tau_i = \begin{cases} 2m = \lfloor (i+1)/2 \rfloor, & 1 \le i \le 2m \\ m, & i = 2m+1 \end{cases}$$

This example in conjunction with those of Fig. 2, <u>without</u> the ε-length tasks, shows that (6) is best possible. □

FINAL REMARKS

Note that the proof of the main theorem is valid for $k=3$. Hence, the corollary also provides an alternate proof of Graham's $4/3-1/3m$ result.

Informally, our result states if an LPT schedule has at least k tasks scheduled on each processor, then the LPT sequence can be no more than $1/k$ worse than an optimal sequence. In this precise sense we have clear support for the intuitive expectation that LPT comes close to optimal when many tasks are scheduled, relative to the number of processors.

REFERENCES

[C] Coffman, E. G., Jr., <u>Computer and Job-Shop Scheduling Theory</u>, John Wiley and Sons, 1975.

[G] Graham, R. L., "Bounds on Multiprocessor Timing Anomalies," <u>SIAM Journal on Applied Math</u>, <u>17</u> (1969), 416-429.

[JDUGG] Johnson, D. S., A. Demers, J. D. Ullman, M. R. Garey and R. L. Graham, "Worst-Case Performance Bounds for Simple One-Dimensional Packing Algorithms," <u>SIAM Journal on Computing</u>, <u>3</u> (1974), 299-326.

ACKNOWLEDGEMENTS

We wish to thank the anonymous referee for his/her comments. Unfortunately, they arrived after the paper had been typed.