SIAM J. COMPUT.
Vol. 4, No. 2, June 1975

BOUNDS FOR MULTIPROCESSOR SCHEDULING WITH RESOURCE CONSTRAINTS*

M. R. GAREY AND R. L. GRAHAM†

Abstract. One well-studied model of a multiprocessing system involves a fixed number n of identical abstract processors, a finite set of tasks to be executed, each requiring a specified amount of computation time, and a partial ordering on the tasks which requires certain tasks to be completed before certain others can be initiated. The nonpreemptive operation of the system is guided by an ordered list L of the tasks, according to the rule that whenever a processor becomes idle, it selects for processing the first unexecuted task on L which may validly be executed. We introduce an additional element of realism into this model by postulating the existence of a set of "resources" with the property that for each resource, the total usage of that resource at any instant of time may not exceed its total availability. For this augmented model, we determine upper bounds on the ratio of finishing times achieved using two different lists, L and L', and exhibit constructions to show that the bounds are best possible.

Key words. scheduling models, graph theory, worst-case analysis, performance bounds

1. Introduction. A number of authors (cf. [12], [16], [7], [3], [11], [4], [5], [9]) have recently been concerned with scheduling problems associated with a certain model of an abstract multiprocessing system (to be described in the next section) and, in particular, with bounds on the worst-case behavior of this system as a function of the way in which the inputs are allowed to vary. In this paper, we introduce an additional element of realism into the model by postulating the existence of a set of "resources" with the property that at no time may the system use more than some predetermined amount of each resource. With this extra constraint taken into consideration, we derive a number of rather close bounds on the behavior of this augmented system. It will be seen that this investigation also leads to several interesting results in graph theory and analysis.

2. The standard model. We consider a system composed of (usually n) abstract identical processors. The function of the system is to execute some given set $\mathscr{T} = \{T_1, \cdots, T_r\}$ of tasks. However, $\mathscr{T}$ is partially ordered by some relation[1] $\prec$ which must be respected in the execution of $\mathscr{T}$ as follows: if $T_i \prec T_j$, then the execution of T_i must be completed before the execution of T_j can begin. To each task T_i is associated a positive real number τ_i which represents the amount of time T_i requires for its execution. The operation of the system is assumed to be *non-preemptive*, which means that once a processor begins to execute a task T_i, it must continue to execute it to completion, τ_i time units later. Finally, the order in which the tasks are chosen is determined as follows: a permutation (or *list*) $L = \{T_{i_1}, \cdots, T_{i_r}\}$ of $\mathscr{T}$ is given initially. At any time a processor is idle, it instantaneously scans L from the beginning and selects the first task T_k (if any) which may validly be executed (i.e., all $T_i \prec T_k$ have been completed) and which is not currently being executed by another processor. Ties by two or more processors for the same task may be broken arbitrarily (since the processors are assumed to be identical).

* Received by the editors June 14, 1974, and in revised form August 9, 1974.

† Bell Laboratories, Murray Hill, New Jersey 07974.

[1] Thus, $\prec$ is transitive, antisymmetric and irreflexive.

The system begins at time $t = 0$ and starts executing $\mathscr{T}$. The *finishing time* ω is defined to be the least time at which all tasks have been completed. Of course, ω is a function of $L, \prec$, n and the τ_i. It is known [7] that if $\mathscr{T}' = \{T'_1, \cdots, T'_r\}$ with $T'_i \prec T'_j \Rightarrow T_i \prec T_j$ and $\tau_i \leqq \tau'_i$ for all i and j, and $\mathscr{T}'$ is executed by the system using a list L', then the corresponding finishing time ω' satisfies

$$\omega'/\omega \leqq 2 - 1/n. \tag{1}$$

Furthermore, this bound is best possible. Efficient procedures are known [3], [4], [9] for generating optimal lists when all the τ_i are 1 and *either* $\prec$ (viewed as a directed graph in the obvious way) is a tree *or* $n = 2$. However, Ullman [12] has recently shown that even the case of $n = 2$ and $\tau_i \in \{1, 2\}$ for all i is polynomial complete[2] and therefore probably has no efficient solution in general.

3. The augmented model. Before proceeding to a description of the new model we first introduce some notation which will make the ensuing discussion mathematically more convenient.

For a given list L, let $F: \mathscr{T} \to 2^{[0,\omega)}$ be defined by $F(T_i) = [\sigma_i, \sigma_i + \tau_i)$, where σ_i is the time at which the execution of T_i was started. Let $f: [0, \omega) \to 2^{\mathscr{T}}$ be defined by $f(t) = \{T_i \in \mathscr{T} : t \in F(T_i)\}$. Thus $f(t)$ is just the set of tasks which are being executed at time t. The restriction that we have at most n processors can be expressed by requiring $|f(t)| \leqq n$ for all $t \in [0, \omega)$.

Assume now that we are also given a set of resources $\mathscr{R} = \{\mathscr{R}_1, \cdots, \mathscr{R}_s\}$ and that these resources have the following properties. The total amount of resource $\mathscr{R}_i$ available at any time is (normalized without loss of generality to) 1. For each j, the task T_j requires the use of $\mathscr{R}_i(T_j)$ units of resource $\mathscr{R}_i$ at all times during its execution, where $0 \leqq \mathscr{R}_i(T_j) \leqq 1$. For each $t \in [0, \omega)$, let $r_i(t)$ denote the total amount of resource $\mathscr{R}_i$ which is being used at time t. Thus

$$r_i(t) = \sum_{T_j \in f(t)} \mathscr{R}_i(T_j).$$

In this augmented model, the fundamental constraint is simply this:

$$r_i(t) \leqq 1 \quad \text{for all } t \in [0, \omega).$$

In other words, at no time can we use more of any resource than is currently available.

The basic problem we shall consider is to what extent the use of different lists for this model can affect the finishing time ω.

4. Summary of results. There are essentially three results which will be proved in this paper. They all are derived from the following situation. We assume we are given a set of tasks $\mathscr{T} = \{T_1, \cdots, T_r\}$, execution times τ_i, a partial order $\prec$ on $\mathscr{T}$, a set of resources $\mathscr{R} = \{\mathscr{R}_1, \cdots, \mathscr{R}_s\}$, task resource requirements[3] $\mathscr{R}_i(T_j)$ and a positive integer n. For an arbitrary list L, let $\omega = \omega(L)$ be the finishing time for the (augmented) system of n processors executing $\mathscr{T}$ according to list L. Let $\omega^* = \omega(L^*)$ denote the minimum of $\omega(L)$ over all lists L. (Note that the use of $n \geqq r$ processors is equivalent to having an unlimited number of processors

[2] See [10] for a definition of this term.

[3] These are as described in the preceding section.

available, since clearly there can never be more than r processors active at any time.)

THEOREM 1. *For* $\mathscr{R} = \{\mathscr{R}_1\}$,

$$\omega/\omega^* \leqq n. \tag{2}$$

THEOREM 2. *For* $\mathscr{R} = \{\mathscr{R}_1, \mathscr{R}_2, \cdots, \mathscr{R}_s\}, \prec$ *empty, and* $n \geqq r$,

$$\omega/\omega^* \leqq s + 1. \tag{3}$$

THEOREM 3. *For* $\mathscr{R} = \{\mathscr{R}_1, \mathscr{R}_2, \cdots, \mathscr{R}_s\}, \prec$ *empty, and* $n \geqq 2$,

$$\frac{\omega}{\omega^*} \leqq \min\left\{\frac{n+1}{2}, s + 2 - \frac{2s+1}{n}\right\}. \tag{4}$$

By way of comparison, the following result (now a special case of Theorem 3) is proved in [7].

THEOREM 0. *For* $\mathscr{R} = \varnothing$,

$$\omega/\omega^* \leqq 2 - 1/n.$$

Furthermore, as in the case of Theorem 0, examples will be given to show that each of these results is essentially best possible.

Thus the addition of limited resources into the standard model causes an increase in the worst-case behavior bounds, as might be expected. What is somewhat surprising, however, is the significant effect the partial order $\prec$ can have on these bounds. This is in contrast to the previous case of $\mathscr{R} = \varnothing$ in which the upper bound $\omega/\omega^* \leqq 2 - 1/n$ which holds for arbitrary $\prec$ could, in fact, be achieved by examples with $\prec$ empty. Also significant is the apparent need for somewhat more sophisticated mathematical techniques than were required previously.

Proof of Theorem 1. The proof of (2) is immediate. We merely need to observe that

$$\omega \leqq \sum_{i=1}^{r} \tau_i \leqq n\omega^*,$$

since at no time before time ω are all processors idle when using list L, and the number of processors busy at any time never exceeds n.

More interesting is the following example, which shows that (2) is best possible.

Example 1.

$$T = \{T_1, \cdots, T_n, \hat{T}_1, \cdots, \hat{T}_n\}, \qquad \mathscr{R} = \{\mathscr{R}_1\}$$

$$\tau_i = 1, \qquad \hat{\tau}_i = \varepsilon > 0,$$

$$\mathscr{R}_1(T_i) = \frac{1}{n}, \qquad \mathscr{R}_1(\hat{T}_i) = 1, 1 \leqq i \leqq n.$$

$\prec$ is defined by

$$\hat{T}_i \prec T_i \quad \text{for } 1 \leqq i \leqq n,$$

$$L = (T_1, \cdots, T_n, \hat{T}_1, \cdots, \hat{T}_n), \qquad L' = (\hat{T}_1, \cdots, \hat{T}_n, T_1, \cdots, T_n).$$

A simple calculation[4] shows that

$$\omega = n + n\varepsilon, \qquad \omega^* = \omega' = 1 + n\varepsilon.$$

Thus

$$\frac{\omega}{\omega^*} = \frac{n + n\varepsilon}{1 + n\varepsilon} \to n \quad \text{as } \varepsilon \to 0.$$

Proof of Theorem 2. In this case, we assume $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, \cdots, \mathcal{R}_s\}$, $\prec$ is empty and $n \geqq r$. The proof will require several preliminary results. The meaning of undefined terminology in graph theory may be found in [8].

Let G denote a graph with vertex set $V = V(G)$ and edge set $E = E(G)$. By a *valid labeling* L of G we mean a function $L: V \to [0, \infty)$ which satisfies

$$\text{(5)} \qquad \text{for all } e = \{a, b\} \in E, \quad L(a) + L(b) \geqq 1.$$

Define the *score of* G, denoted by $S(G)$, by

$$S(G) = \inf_L \left\{ \sum_{v \in V} L(v) \right\},$$

where the inf is taken over all valid labelings L of G.

LEMMA 1. *For any graph G, there exists a valid labeling $L: V \to \{0, \frac{1}{2}, 1\}$ such that*

$$S(G) = \sum_{v \in V} L(v).$$

Proof. For the case of a bipartite graph, König's theorem [8] states that the number of edges in a maximum matching equals the point covering number.[5] Thus for any bipartite graph G, there exists a valid labeling $L: V \to \{0, 1\}$ such that $S(G) = \sum_{v \in V} L(v)$.

For an arbitrary graph G, we construct a bipartite graph G_B as follows: for each vertex $v \in V(G)$ we have two vertices $v_1, v_2 \in V(G_B)$; for each edge $\{u, v\} \in E(G)$ we have two edges $\{u_1, v_2\}, \{u_2, v_1\} \in E(G_B)$. It is not difficult to verify that $S(G_B) = 2S(G)$ and furthermore, if $L_B: V(G_B) \to \{0, 1\}$ is a valid labeling of G_B, then $L: V(G) \to \{0, \frac{1}{2}, 1\}$ by $L(v) = \frac{1}{2}(L(v_1) + L(v_2))$ is a valid labeling of G. □

For positive integers m and s, let $G(m, s)$ denote the graph with vertex set $\{0, 1, \cdots, (s + 1)m - 1\}$ and edge set consisting of all pairs $\{a, b\}$ for which $|a - b| \geqq m$.

LEMMA 2. *Suppose $G(m, s)$ is partitioned into s spanning subgraphs H_i, $1 \leqq i \leqq s$. Then*

$$\text{(6)} \qquad \max_{1 \leqq i \leqq s} \{S(H_i)\} \geqq m.$$

Proof. Assume the lemma is false, i.e., there exists a partition of $G(m, s)$ into H_i, $1 \leqq i \leqq s$, such that $S(H_i) < m$ for $1 \leqq i \leqq s$. Thus, by Lemma 1, for each i there exists a valid labeling $L_i: V(H_i) \to \{0, \frac{1}{2}, 1\}$ such that

$$\text{(7)} \qquad \sum_{v \in V(H_i)} L_i(v) = S(H_i) < m.$$

[4] The reader will probably find it helpful to construct a timing diagram to understand the behavior of this (and succeeding) examples.

[5] That is, the cardinality of the smallest set of vertices of G incident to every edge of G.

Let $A = \{a_1 < \cdots < a_p : L_i(a_j) \leqq \frac{1}{2}$ for all i, $1 \leqq i \leqq s\}$, and let S^* denote $\sum_{i=1}^{s} S(H_i)$. There are three cases.

(i) $p \leqq m$. In this case we have $S^* \geqq m(s+1) - p \geqq m(s+1) - m = ms$, which contradicts (7).

(ii) $m < p \leqq 2m + 1$. For each edge $\{a_j, a_{m+j}\}$, $1 \leqq j \leqq p - m$, there must exist an i such that $L_i(a_j) + L_i(a_{m+j}) \geqq 1$. Thus $S^* \geqq m(s+1) - p + (p - m) = ms$, again contradicting (7).

(iii) $p > 2m + 1$. We first note that for each vertex $v \in V(G(m, s))$, there exists an i such that $L_i(v) \geqq \frac{1}{2}$. For suppose $L_i(v) = 0$ for $1 \leqq i \leqq s$. There must be some a_j such that $|a_j - v| \geqq m$. But since $L_i(a_j) \leqq \frac{1}{2}$ for all i, then $L_i(a_j) + L_i(v) \leqq \frac{1}{2}$ for all i, which is a contradiction.

For each i, let n_i denote the number of vertices v such that $L_i(v) = 1$. Then

$$|\{v : L_i(v) > 0\}| \leqq 2m - 1 - n_i,$$

since otherwise

$$\sum_{v \in V(H_i)} L_i(v) \geqq n_i \cdot 1 + (2m - 2n_i) \cdot \tfrac{1}{2} = m,$$

which contradicts (7). Therefore

$$\sum_{i=1}^{s} |\{v : L_i(v) > 0\}| \leqq (2m - 1)s - \sum_{i=1}^{s} n_i. \tag{8}$$

Let q denote the number of vertices v such that there is exactly one i for which $L_i(v) > 0$. Then

$$\sum_{i=1}^{s} |\{v : L_i(v) > 0\}| \geqq 2(m(s+1) - q) + q. \tag{9}$$

Combining (8) and (9), we have

$$q \geqq 2m + s + \sum_{i=1}^{s} n_i. \tag{10}$$

Of course, we may assume without loss of generality that if $L_i(v) = 1$, then $L_j(v) = 0$ for all $j \neq i$. Hence, by the definition of n_i, there must be at least $2m + s$ vertices, say $b_1 < \cdots < b_{2m+s}$, such that $\sum_{i=1}^{s} L_i(b_j) = \frac{1}{2}$, i.e., for each b_j there is a unique L_i such that $L_i(b_j) = \frac{1}{2}$ and $L_k(b_j) = 0$ for all $k \neq i$. Thus, if $|b_j - b_k| \geqq m$, then for some i, $L_i(b_j) = L_i(b_k) = \frac{1}{2}$. Since $|b_1 - b_{2m+s}| \geqq m$, let i_0 be such that $L_{i_0}(b_1) = L_{i_0}(b_{2m+s}) = \frac{1}{2}$. But, by the same reasoning we must also have $L_{i_0}(b_{m+j}) = L_{i_0}(b_1) = \frac{1}{2}$ and $L_{i_0}(b_{2m+s}) = L_{i_0}(b_j) = \frac{1}{2}$ for $1 \leqq j \leqq m + s$. Therefore

$$S(H_{i_0}) = \sum_{v \in V(H_{i_0})} L_{i_0}(v) \geqq (2m + s) \cdot \tfrac{1}{2} \geqq m,$$

which is a contradiction. This completes the proof of Lemma 2. □

Recall that when $\mathcal{T}$ is executed using the list L, $F(T_i)$ is defined to be the interval $[\sigma_i, \sigma_i + \tau_i)$, where σ_i is the time at which T_i starts to be executed and

$\sigma_i + \tau_i$ is the time at which T_i is finished. Note that because of the way in which the operation of the system is defined, each σ_i is a sum of a subset of the τ_j's.

We may assume without loss of generality that $\omega^* = 1$. Assume now that $\omega > s + 1$. Furthermore, suppose each τ_i can be written as $\tau_i = k_i/m$, where k_i is a positive integer. Thus $k_i \leqq m$, since $\tau_i \leqq \omega^* = 1$. Also, for $1 \leqq i \leqq s$, each $r_i(t)$ is constant on each interval $[k/m, (k+1)/m)$, this value being $r_i(k/m)$. An important fact to note is that since $\prec$ is empty and $n \geqq r$, then, for $t_1, t_2 \in [0, \omega)$ with $t_2 - t_1 \geqq 1$, we must have

$$\max_{1 \leqq i \leqq s} \{r_i(t_1) + r_i(t_2)\} > 1.$$

For otherwise, any task being executed at time t_2 should have been executed at time t_1 or sooner. Thus, for each i, $1 \leqq i \leqq s$, we can construct a graph H_i as follows:

$$V(H_i) = \{0, 1, \cdots, (s+1)m - 1\};$$
$$\{a, b\} \text{ is an edge of } H_i \quad \text{iff} \quad r_i\left(\frac{a}{m}\right) + r_i\left(\frac{b}{m}\right) > 1. \tag{11}$$

Note that if $|a - b| \geqq m$, then $\{a, b\}$ is an edge of at least one H_i, $1 \leqq i \leqq s$. Hence it is not difficult to see that $G(m, s) \subseteq \bigcup_i H_i$. Note that by (11), the mapping $L_i : V(H_i) \to [0, \infty)$ defined by $L_i(a) = r_i(a/m)$ is a valid labeling of H_i. Since $G \subseteq G'$ implies $S(G) \leqq S(G')$ and the condition on the r_i in (11) is a strict inequality, then by Lemma 2 it follows that

$$\max_i \left\{ \sum_{k=0}^{(s+1)m-1} r_i\left(\frac{k}{m}\right) \right\} = \max_i \left\{ \sum_{v \in V(H_i)} L_i(v) \right\} > \max_i \{S(H_i)\} \geqq m. \tag{12}$$

But we must have

$$\frac{1}{m} \sum_{k=0}^{(s+1)m-1} r_i\left(\frac{k}{m}\right) \leqq \int_0^\infty r_i(t)\, dt \leqq 1, \qquad 1 \leqq i \leqq s, \tag{13}$$

i.e.,

$$\sum_{k=0}^{(s+1)m-1} r_i\left(\frac{k}{m}\right) \leqq m, \qquad 1 \leqq i \leqq s.$$

This is a *contradiction*, and Theorem 2 is proved in the case that $\tau_i = k_i/m$, where k_i is a positive integer for $1 \leqq i \leqq r$. Of course, it follows immediately that Theorem 2 holds when all the τ_i are *rational*. The proof of Theorem 2 will be completed by establishing the following lemma.

LEMMA 3. *Let* $\tau = (\tau_1, \cdots, \tau_r)$ *be a sequence of positive real numbers. Then for any* $\varepsilon > 0$, *there exists* $\tau' = (\tau'_1, \cdots, \tau'_r)$ *such that*

(i) $|\tau'_i - \tau_i| < \varepsilon$ *for* $1 \leqq i \leqq r$;

(ii) *for all* $S, T \subseteq \{1, \cdots, r\}$,

$$\sum_{s \in S} \tau_s \leqq \sum_{t \in T} \tau_t \quad \textit{iff} \quad \sum_{s \in S} \tau'_s \leqq \sum_{t \in T} \tau'_t;$$

(iii) *all* τ'_i *are positive rational numbers.*

Remark. The importance of (ii) is that it guarantees that the *order* of execution of the T_i using the list L is the same for τ and τ'. Thus if L is used to execute $\mathscr{T}$, once using execution times τ_i and once using execution times τ_i', then the corresponding finishing times ω and ω' satisfy $|\omega - \omega'| \leqq r\varepsilon$. Hence if there were an example $\mathscr{T}$ with $\omega/\omega^* > s + 1$ and some of the τ_i irrational, then we could construct another example $\mathscr{T}'$ by slightly changing the τ_i to rational τ_i' so that the corresponding new finishing times ω' and ω^* satisfy $|\omega - \omega'| \leqq r\varepsilon$ and $|\omega^* - \omega^{*\prime}| \leqq r\varepsilon$, and, therefore if ε is sufficiently small, we still have $\omega'/\omega^{*\prime} > s + 1$. However, this would contradict what has already been proved. Lemma 3 is implied by the following slightly more general result. The proof we give here is due to V. Chvátal (personal communication).

LEMMA 3'. *Let S denote a finite system of inequalities of the form*

$$\sum_{i=1}^{r} a_i x_i \geqq a_0 \quad \text{or} \quad \sum_{i=1}^{r} a_i x_i > a_0,$$

where the a_i are rational. Then, for any $\varepsilon > 0$, if S has a real solution $(x_1, \cdots, x_r)$, then S has a rational *solution $(x', \cdots, x_r')$ with $|x_i - x_i'| < \varepsilon$ for all i.*

Proof. We proceed by induction on r. For $r = 1$ the result is immediate. Now, let S be a system of inequalities in $r > 1$ variables which is solvable in reals. S splits into two classes: S_0, the subset of inequalities not involving x_r, and $S_1 = S - S_0$. Each inequality in S_1 can be written in one of the following four ways:

(a) $$\alpha_0 + \sum_{i=1}^{r-1} \alpha_i x_i \leqq x_r,$$

(b) $$\alpha_0 + \sum_{i=1}^{r-1} \alpha_i x_i < x_r,$$

(c) $$\beta_0 + \sum_{i=1}^{r-1} \beta_i x_i \geqq x_r,$$

(d) $$\beta_0 + \sum_{i=1}^{r-1} \beta_i x_i > x_r.$$

For each pair of inequalities, one of type (a) and one of type (c), we shall consider the inequality

(e) $$\alpha_0 + \sum_{i=1}^{r-1} \alpha_i x_i \leqq \beta_0 + \sum_{i=1}^{r-1} \beta_i x_i.$$

Similarly, the pairs of types {(a), (d)}, {(b), (c)} and {(b), (d)} give rise to inequalities

(f) $$\alpha_0 + \sum_{i=1}^{r-1} \alpha_i x_i < \beta_0 + \sum_{i=1}^{r-1} \beta_i x_i.$$

Let S^* be the set of all inequalities of type (e) and (f) that we obtain from S_1. Since by hypothesis, $S = S_0 \cup S_1$ has a real solution $(x_1, \cdots, x_r)$, then $S_0 \cup S^*$ has the real solution $(x_1, \cdots, x_{r-1})$. But $S_0 \cup S^*$ only involves $r - 1$ variables, so that, by the induction hypothesis, $S_0 \cup S^*$ has a rational solution $(x_1', \cdots, x_{r-1}')$ with $|x_i - x_i'| < \varepsilon'$ for all i and any preassigned $\varepsilon' > 0$. Substituting the x_i' into (a), (b), (c) and (d), we obtain a set of inequalities

$$\text{(g)} \qquad a' \leqq x_r, \quad b' < x_r, \quad c' \geqq x_r, \quad d' > x_r,$$

where the a', b', c' and d' are rational. Since the x_i satisfy (e) and (f), we have $a' \leqq c'$, $b' < c'$, $a' < d'$, $b' < d'$. Thus for any $\varepsilon > 0$, if ε' is chosen to be suitably small, then there is a rational x_r' satisfying (g) and with $|x_r - x_r'| < \varepsilon$, completing the proof of Lemma 3'. This proves Lemma 3, and hence, Theorem 2. □

The following example shows that the bound in Theorem 2 cannot be improved.

Example 2.

$$\mathscr{T} = \{T_1, T_2, \cdots, T_{s+1}, T_1', T_2', \cdots, T_{sN}'\};$$

$$\prec \; = \varnothing; \qquad n \geqq s(N+1) + 1 = r;$$

$$\tau_i = 1 \quad \text{for} \quad 1 \leqq i \leqq s+1; \qquad \tau_i' = \frac{1}{N} \quad \text{for} \quad 1 \leqq i \leqq sN;$$

$$\mathscr{R}_i(T_i) = 1 - \frac{1}{N}, \qquad \mathscr{R}_i(T_j) = \frac{1}{sN}, \qquad j \neq i, \quad i \leqq i \leqq s;$$

$$\mathscr{R}_i(T_j') = \frac{1}{N}, \qquad 1 \leqq j \leqq sN, \qquad 1 \leqq i \leqq s;$$

$$L = (T_1, T_1', \cdots, T_N', T_2, T_{N+1}', \cdots, T_{k+1}, T_{kN+1}',$$
$$T_{kN+2}', \cdots, T_{(k+1)N}', T_{k+2}, \cdots, T_{sN}', T_{s+1});$$
$$L' = (T_1', T_2', \cdots, T_{sN}', T_1, T_2, \cdots, T_{s+1}).$$

It is easily checked that for this case, $\omega = s + 1$ and $\omega' = 1 + s/N$, so that ω/ω' (and hence ω/ω^*) is arbitrarily close to $s + 1$ for N sufficiently large.

Proof of Theorem 3. The proof of Theorem 3 consists primarily of two main lemmas, each of which gives a bound on ω/ω^* which is best possible for certain values of s and n. We let λ denote ordinary Lebesgue measure[6] on the real line.

LEMMA 4. *For* $\mathscr{R} = \{\mathscr{R}_1, \mathscr{R}_2, \cdots, \mathscr{R}_s\}$ *and* $\prec$ *empty,*

$$\omega = \frac{n+1}{2}, \qquad \omega^* = \omega' = 1,$$

Proof. Let $I = \{t : |f(t)| = 1\}$. We first show

$$\text{(14)} \qquad \lambda(I) \leqq \omega^*.$$

Consider the set T of tasks defined by $T = \bigcup_{t \in I} f(t)$. For any pair of tasks T_i, T_j belonging to T, there must exist some k, $1 \leqq k \leqq s$, such that $\mathscr{R}_k(T_i) + \mathscr{R}_k(T_j) > 1$, for otherwise, one of those tasks should have been started earlier (unless $n = 1$, in which case the lemma is trivial). But this implies that in the optimal schedule

[6] Since in all of our applications, the subsets X of $[0, \omega)$ under consideration are finite unions of disjoint half-open intervals, then $\lambda(X)$ is just the sum of the lengths of these intervals.

no two members of T can be executed simultaneously. Therefore we have

$$\omega^* \geqq \sum_{T_i \in T} \tau_i \geqq \lambda(I),$$

which proves (14).

To complete the proof of Lemma 4, observe that at least two processors must be active at each time $t \in \bar{I} = [0, \omega) - I$.
Thus

$$n\omega^* \geqq \sum_{i=1}^{r} \tau_i \geqq 2\lambda(\bar{I}) + \lambda(I) = 2\omega - \lambda(I) \geqq 2\omega - \omega^*,$$

and therefore $(n + 1)\omega^* \geqq 2\omega$. □

The bound given by Lemma 4 is best possible whenever $n \leqq s + 1$, as shown by the following examples.

Example 3.

$$\mathscr{T} = \{T_0, T_1, T'_1, T_2, T'_2, \cdots, T_{n-1}, T'_{n-1}\};$$

$$\mathscr{R} = \{\mathscr{R}_1, \mathscr{R}_2, \cdots, \mathscr{R}_s\}, \qquad 2 \leqq n \leqq s + 1; \qquad \prec = \varnothing;$$

$$\tau_0 = 1; \qquad \tau_j = \tau'_j = \tfrac{1}{2}, \qquad 1 \leqq j \leqq n - 1;$$

$$\mathscr{R}_i(T_0) = \frac{1}{2n}, \qquad 1 \leqq i \leqq s;$$

$$\mathscr{R}_i(T_i) = \mathscr{R}_i(T'_i) = \tfrac{1}{2}, \qquad 1 \leqq i \leqq n - 1;$$

$$\mathscr{R}_i(T_j) = \mathscr{R}_i(T'_j) = \frac{1}{2n}, \qquad i \neq j, \quad 1 \leqq i \leqq s, \quad 1 \leqq j \leqq n - 1;$$

$$L = (T_1, T'_1, T_2, T'_2, \cdots, T_{n-1}, T'_{n-1}, T_0);$$

$$L = (T_0, T_1, T_2, \cdots, T_{n-1}, T'_1, T'_2, \cdots, T'_{n-1}).$$

It is easily checked that for this case,

$$\omega = \frac{n + 1}{2}, \qquad \omega^* = \omega' = 1,$$

showing that the bound of Lemma 4 is best possible whenever $n \leqq s + 1$. □

The following example, for the case $s + 1 < n \leqq 2s + 1$, is somewhat more complicated.

Example 4. For suitably small $\varepsilon > 0$ and a positive integer k, define

$$\varepsilon_i = \varepsilon(n - 1)^{i - 2k}, \qquad 1 \leqq i \leqq 2k;$$

$$\mathscr{T} = \{T_0\} \cup \{T_{ij}: 1 \leqq i \leqq n - 1, 1 \leqq j \leqq k\} \cup \{T'_{ij}: 1 \leqq i \leqq n - 1, 1 \leqq j \leqq k\};$$

$$\mathscr{R} = \{\mathscr{R}_1, \mathscr{R}_2, \cdots, \mathscr{R}_s\}; \quad s + 1 < n \leqq 2s + 1; \quad \prec = \varnothing;$$

$$\tau_0 = 2k; \quad \tau_{ij} = \tau'_{ij} = 1, \qquad 1 \leqq i \leqq n - 1, \quad 1 \leqq j \leqq k;$$

$$\mathscr{R}_i(T_0) = \varepsilon_1, \qquad 1 \leqq i \leqq s;$$

$$\mathscr{R}_i(T_{ij}) = 1 - (n-1)\varepsilon_{2j-1}, \quad 1 \leqq i \leqq s, \quad 1 \leqq j \leqq k;$$

$$\mathscr{R}_l(T_{ij}) = \varepsilon_{2j-1}, \quad l \neq i, \quad 1 \leqq l \leqq s, \quad 1 \leqq i \leqq n-1, \quad 1 \leqq j \leqq k;$$

$$\mathscr{R}_i(T'_{s+i,j}) = 1 - (n-1)\varepsilon_{2j}, \quad 1 \leqq i \leqq n-s-1, \quad 1 \leqq j \leqq k;$$

$$\mathscr{R}_l(T'_{ij}) = \varepsilon_{2j}, \quad l \neq i-s, \quad 1 \leqq l \leqq s, \quad 1 \leqq i \leqq n-1, \quad 1 \leqq j \leqq k.$$

To illuminate the structure of the two lists, L and L', we describe them in block form.

$$L = (A_1, A_2, \cdots, A_k, A'_1, A'_2, \cdots, A'_{k-1}, A_0),$$

where

$$A_i = (B_{1i}, B_{2i}, \cdots, B_{si}), \quad 1 \leqq i \leqq k;$$

$$B_{ji} = (T_{ji}, T'_{ji}), \quad 1 \leqq i \leqq k, \quad 1 \leqq j \leqq s;$$

$$A'_i = (B'_{1i}, B'_{2i}, \cdots, B'_{n-1,i}), \quad 1 \leqq i \leqq k-1;$$

$$B'_{ji} = (T'_{s+j,i}, T_{s+j,i+1}), \quad 1 \leqq i \leqq k-1, \quad 1 \leqq j \leqq n-1;$$

$$A_0 = (T_0, T_{s+1,1}, T_{s+2,1}, \cdots, T_{n-1,1}, T'_{s+1,k}, T'_{s+2,k}, \cdots, T'_{n-1,k}).$$

Also

$$L' = (C_0, C_1, C'_1, C_2, C'_2, \cdots, C_k, C'_k),$$

where

$$C_0 = (T_0);$$

$$C_i = (T_{1i}, T_{2i}, \cdots, T_{n-1,i}), \quad 1 \leqq i \leqq k;$$

$$C'_i = (T'_{1i}, T'_{2i}, \cdots, T'_{n-1,i}), \quad 1 \leqq i \leqq k.$$

It is not difficult to check that when the list L is used, each of the pairs of tasks given in the sublists B_{ji} and B'_{ji} will be executed simultaneously on the first two processors, with the other $n-2$ processors remaining inactive during that time. After all such pairs have been executed, the tasks on sublist A_0 will be started. This results in

$$\omega = k(n-1) - (n-s-1) + 2k = k(n+1) - (n-s-1).$$

When the list L' is used, each of the sets of $n-1$ tasks given in the sublists C_i and C'_i, will be executed simultaneously on processors 2 through n, with processor 1 executing T_0. Thus $\omega^* = \omega' = 2k$. We then have

$$\frac{\omega}{\omega^*} = \frac{n+1}{2} - \frac{(n-s-1)}{2k},$$

which is arbitrarily close to $(n+1)/2$ for k sufficiently large. □

We now prove an upper bound for ω/ω^* which is best possible whenever $n > 2s+1$.

LEMMA 5. *For* $\mathscr{R} = \{\mathscr{R}_1, \mathscr{R}_2, \cdots, \mathscr{R}_s\}$, $\prec = \varnothing$, *and* $n \geqq 3$,

$$\frac{\omega}{\omega^*} \leqq s + 2 - \frac{2s+1}{n}.$$

Proof. Suppose that we have a counterexample to the lemma. By Lemma 3, we may assume all the τ_i are rational, i.e., there exists a positive integer m such that for each i, $1 \leqq i \leqq r$, there exists an integer k_i satisfying $\tau_i = k_i/m$. Without loss of generality, we may also assume that $\omega^* = 1$. Thus each k_i satisfies $1 \leqq k_i \leqq m$ and $\omega = \omega(L) > s + 2 - (2s + 1)/n$.

Consider the operation of the system using the list L. Let $I = \{t \in [0, \omega) : |f(t)| = 1\}$, $I' = \{t \in [0, \omega) : |f(t)| = n\}$ and let $\bar{I} = [0, \omega) - I'$. By the proof of Lemma 4, $\lambda(I) \leqq 1$. Since at least two processors are active at each time $t \in \bar{I}$,

$$n \geqq \sum_{i=1}^{r} \tau_i \geqq n \cdot \lambda(I') + \lambda(I) + 2(\omega - \lambda(I) - \lambda(I'))$$

$$\geqq (n - 2)\lambda(I') + 2\omega - 1,$$

or

$$\lambda(I') \leqq \frac{n + 1 - 2\omega}{n - 2}. \tag{15}$$

Since $\omega > s + 2 - (2s + 1)/n$, we then have

$$\lambda(\bar{I}) = \omega - \lambda(I') \geqq \omega - \frac{n + 1 - 2\omega}{n - 2}$$

$$> s + 2 - \frac{2s + 1}{n} - \frac{n + 1 - 2\left(s + 2 - \frac{2s + 1}{n}\right)}{n - 2} \tag{16}$$

$$= s + 1.$$

Now observe that for any $t_1, t_2 \in \bar{I}$ satisfying $t_2 - t_1 \geqq 1$, there must exist an i, $1 \leqq i \leqq s$, such that

$$r_i(t_1) + r_i(t_2) > 1, \tag{17}$$

for otherwise, some task being executed at time t_2 should have been started at time t_1 or earlier. Recalling that $\bar{I}$ is a collection of intervals, each having the form $[k/m, (k + 1)/m)$ for some integer k, let $a_0 < a_1 < \cdots < a_p$ be integers such that

$$\bar{I} = \left\{\left[\frac{a_i}{m}, \frac{a_i + 1}{m}\right) : 0 \leqq i \leqq p\right\}.$$

Notice that (16) implies that $p \geqq (s + 1)m$. For each i, $1 \leqq i \leqq s$, we construct a graph H_i as follows:

$$V(H_i) = \{0, 1, 2, \cdots, (s + 1)m - 1\};$$

$$\{u, v\} \text{ is an edge of } H_i \quad \text{iff} \quad r_i\left(\frac{a_u}{m}\right) + r_i\left(\frac{a_v}{m}\right) > 1.$$

Note that $|u - v| \geqq m$ implies $|a_u - a_v| \geqq m$, which, by (17), implies that $\{u, v\}$ is an edge of at least one H_i, $1 \leqq i \leqq s$. Hence it is not difficult to see that $G(m, s) \subseteq \bigcup_i H_i$. The same reasoning used in the proof of Theorem 2 can be used now to show that, for some i, $1 \leqq i \leqq s$, $\int_0^\infty r_i(t)\,dt > 1$, which contradicts the assumption that $\omega^* = 1$. This completes the proof of Lemma 5. □

Combining Lemmas 4 and 5, we obtain Theorem 3. It remains to be shown that Lemma 5 is best possible whenever $n > 2s + 1$. This is done by the following example.

Example 5. For suitably small $\varepsilon > 0$ and a positive integer k', let $k = k'n$, and define

$$\varepsilon_i = \varepsilon(n-1)^{i-k}, \qquad 1 \leqq i \leqq k;$$

$$\mathscr{T} = \{T_0\} \cup \{T_{ij}: 1 \leqq i \leqq n-1, 1 \leqq j \leqq k\};$$

$$\mathscr{R} = \{\mathscr{R}_1, \mathscr{R}_2, \cdots, \mathscr{R}_s\}; \quad n > 2s+1; \quad \prec\, = \varnothing;$$

$$\tau_0 = k; \qquad \tau_{ij} = 1, \qquad 1 \leqq i \leqq n-1, \quad 1 \leqq j \leqq k;$$

$$\mathscr{R}_i(T_0) = \varepsilon_1, \qquad 1 \leqq i \leqq s;$$

$$\mathscr{R}_i(T_{ij}) = 1 - (n-1)\varepsilon_j, \qquad 1 \leqq i \leqq s, \quad 1 \leqq j \leqq k;$$

$$\mathscr{R}_l(T_{ij}) = \varepsilon_j, \qquad l \neq i, \quad 1 \leqq l \leqq s, \quad 1 \leqq i \leqq n-1, \quad 1 \leqq j \leqq k.$$

As in Example 4, we again describe the lists L and L' in block form.

$$L = (A_1, A_2, \cdots, A_{n-2s-1}, B_1, B_2, \cdots, B_s, C),$$

where

$$A_i = (T_{2s+i,1}, T_{2s+i,2}, \cdots, T_{2s+i,k}), \qquad 1 \leqq i \leqq n-2s-1;$$

$$B_i = (T_{i1}, T_{s+i,2}, T_{i2}, T_{s+i,3}, \cdots, T_{i,k-1}, T_{s+i,k}), \qquad 1 < i < s;$$

$$C = (T_0, T_{s+1,1}, T_{s+2,1}, \cdots, T_{2s,1}, T_{1k}, T_{2k}, \cdots, T_{sk}).$$

Also

$$L' = (T_0, D_1, D_2, \cdots, D_k),$$

where

$$D_i = (T_{1i}, T_{2i}, \cdots, T_{n-1,i}), \qquad 1 \leqq i \leqq k.$$

It is not difficult to check that

$$\omega = k'(n-2s-1) + (k-1)s + k = (s+2)k'n - (2s+1)k' - s$$

and $\omega^* = \omega' = k = k'n$. Thus

$$\frac{\omega}{\omega^*} = s + 2 - \frac{2s+1}{n} - \frac{s}{k'n},$$

which is arbitrarily close to the bound of Lemma 5 for k' sufficiently large.

5. Concluding remarks. The results which have been discussed in this paper lead naturally to a number of possible extensions, several of which we mention here.

We first note that for the case $\mathscr{R} = \{\mathscr{R}_1\}$, $n \geqq r$, and general $\prec$, Example 1 may be used to show that ω/ω^* can be arbitrarily large.

Regarding Lemma 1, an algorithm can be given which determines $S(G)$ (and a corresponding valid labeling as well) in at most

$$O(|E|\sqrt{|V|})$$

operations. A similar algorithm may be used for the following dual problem: given a graph G, determine

$$\max_{L^*} \sum_{e \in E} L^*(e),$$

where the max ranges over all functions $L^*: E \to [0, \infty)$ such that for all $v \in V$,

$$\sum_{e' \in E(v)} L^*(e') \leqq 1,$$

where $E(v)$ is the set of all edges incident to v. It would be interesting to investigate the analogous questions for hypergraphs.

The following result follows more or less directly from Lemma 2.

COROLLARY. *For a positive integer n, let $f_i: [0, n+1) \to [0, \infty)$, $1 \leqq i \leqq n$, be (Lebesgue) measurable functions satisfying the following condition:*

If $t_1, t_2 \in [0, n+1)$ with $|t_1 - t_2| \geqq 1$, then

$$\max_{1 \leqq i \leqq n} \{f_i(t_1) + f_i(t_2)\} \geqq 1.$$

Then

$$\max_{1 \leqq i \leqq n} \int_{[0,n+1]} f_i \, d\lambda \geqq 1.$$

It is interesting to note that, at present, no purely analytical proof of the Corollary is known.

The techniques of Lemma 2 may also be used to derive several new results in graph theory. In particular, it follows that if m is a positive integer and G_m denotes the graph with vertex set

$$V_m = \{0, 1, \cdots, 3m - 1\}$$

and edge set

$$E_m = \{\{a, b\} \subseteq V_m : \min\{a - b, 3m - a + b\} \geqq m\},$$

then *any* 2-coloring of E_m contains m disjoint edges having the same color.

The corresponding general conjecture is that for a fixed $s \geqq 1$, if we take

$$V_m = \{0, 1, \cdots, (s+1)m - 1\}$$

and

$$E_m = \{\{a, b\} \subseteq V_m : \min\{a - b, (s+1)m - a + b\} \geqq m\},$$

then any s-coloring of E_m contains m disjoint edges having the same color. At present, this conjecture is still open. If true, it is close to being best possible, since there exist m-colorings of the edges of the *complete* graph on $(s+1)m - s$ vertices which have no set of m disjoint edges having a single color (cf. [1], [2]).

Finally, it is natural to inquire under what restrictions do there exist efficient algorithms for determining optimal schedules for problems of the type considered herein (cf. e.g., [6], [12]).

Acknowledgments. The authors take pleasure in acknowledging the resourceful suggestions of S. A. Burr, J. C. Cheng, V. Chvátal, D. S. Johnson, Shen Lin, and H. S. Witsenhausen. The introduction of limited resources into scheduling problems of this type was first suggested to the authors by E. Arthurs.

REFERENCES

[1] S. A. BURR, *Generalized Ramsey theory for graphs—A survey of recent results*, Graphs and Combinatorics, R. Bari and F. Harary, eds., Lecture Notes in Mathematics No. 406, Springer-Verlag, Berlin, 1974, pp. 52–75.

[2] E. J. COCKAYNE AND P. J. LORIMER, *The Ramsey graph numbers for stripes*, Univ. of Auckland Math. Dept. Rep. Ser. No. 37, Auckland, New Zealand 1973.

[3] E. G. COFFMAN AND R. L. GRAHAM, *Optimal scheduling for two-processor systems*, Acta Informatica 1 (1972), pp. 200–213.

[4] M. FUJII, T. KASAMI AND K. NINOMIYA, *Optimal sequencing of two equivalent processors*, SIAM J. Appl. Math., 17 (1969), pp. 784–789.

[5] ———, *Erratum: Optimal sequencing of two equivalent processors*, Ibid., 20 (1971), p. 141.

[6] R. L. GRAHAM, *Bounds on multiprocessing anomalies and related packing algorithms*, AFIPS Conf. Proc., 40 (1972), pp. 205–217.

[7] ———, *Bounds on multiprocessing timing anomalies*, SIAM J., Appl. Math., 17 (1969), pp. 263–269.

[8] F. HARARY, *Graph Theory*, Addison-Wesley, Reading, Mass., 1969.

[9] T. C. HU, *Parallel scheduling and assembly line problems*, Operations Res., 9 (1961), pp. 841–848.

[10] R. M. KARP, *Reducibility among combinatorial problems*, Complexity of Computer Computations, R. E. Miller and J. W. Thatcher, eds., Plenum Press, New York, 1972.

[11] M. T. KAUFMAN, *Anomalies in scheduling unit-time tasks*, Stanford Electronics Lab. Tech. Rep. 34, 1972.

[12] J. D. ULLMAN, *Polynomial complete scheduling problems*, Computer Science Tech. Rep. 9, Univ. of Calif., Berkeley, 1973.