

AD-A222 726

UMIACS-TR-90-58

April 1990

CS-TR -2462

**Evaluation of a Decomposition Approach
for Real-Time Scheduling
Using a Stochastic Model***

Xiaoping (George) Yuan

Department of Computer Science

University of Maryland

College Park, MD 20742

Ashok K. Agrawala

Institute for Advanced Computer Studies and

Department of Computer Science

University of Maryland

College Park, MD 20742

Abstract

In this paper, a stochastic model is developed to evaluate the decomposition scheduling approach. [Yuan89a] Since the scheduling complexity of the approach directly depends on the number of tasks in every subset, we calculate the probability of the event that "there are n tasks in a subset," for any n , and then the expected number of tasks in a subset. The results indicate that the decomposition scheduling technique not only assures the generation of a feasible schedule if one exists, but also is computationally efficient.

Keywords: probability distributions. (1CR)

DISTRIBUTION STATEMENT A

Approved for public release
Distribution Unlimited

*This work is supported in part by contract DSAG60-87-0066 from the U.S. Army Strategic Defense Command to the Department of Computer Science, University of Maryland at College Park. The views, opinions and/or findings contained in this report are those of the authors and should not be construed as an official Department of Defense position, policy, or decision, unless so designated by other official documentation.

90 05 31 049

Contents

I Introduction	1
II Background and Assumption	2
A. Brief Review	2
B. Assumption of the Evaluation	3
III The Probability Distribution for a Special Case	4
IV Probability and Expectation	6
V Conclusion	12



Accession For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

I Introduction

Let us consider the problem of scheduling a set of N tasks on a processor such that the scheduling time constraints of each task are satisfied. In a hard real-time system, each task T_i has a ready time r_i , a deadline d_i , and a computation time c_i . In a feasible schedule the execution of the tasks must not begin before r_i and must complete by d_i . It is well known that this is an NP-complete problem [Gare79], which may require an examination of $O(N!)$ schedules to assume finding a feasible schedule.

In [Yuan89a, Yuan89b, Yuan89c], we developed a *decomposition scheduling* technique for real-time scheduling. In this approach, the scheduling is carried out in two steps. In the first step, the tasks are divided into subsets, and a sequence of the subsets is determined. In the second step, the tasks are scheduled according to the sequence. It has been shown that when a subset of tasks (τ^i), is before a subset (τ^j) in this sequence, and if a non empty set of feasible schedules exist, then there is a feasible schedule where all tasks of τ^i are scheduled before all the tasks of τ^j . The overall complexity of the scheduling problem reduces to that of finding a feasible schedule of each subset, as the complexity of decomposing the tasks into subsets has been shown to be $O(N)$ [Yuan89a], if tasks are ordered according to their ready times¹.

In this paper, we evaluate the decomposition scheduling approach introduced. The number of schedules to be examined directly depends on the number of tasks in each subset which is also called the *size* of the subset. It is important to know the average size of each subset, and the probability that there are k tasks in one subset for assessing the performance of this scheme. If there is only one task in each subset, the scheduling becomes $O(N)$ computation, and the smaller the size of the subsets, the less is the complexity of scheduling.

In order to evaluate the performance of the decomposition scheduling technique, we use a stochastic model of task requests. For this model, we proceed to calculate the probability that k tasks will be in a subset, for an arbitrary k . The expected value of the number of tasks in a subset is derived next. A similar model has also been used in [Zhao89, Leho89].

In the next section, we briefly introduce the decomposition scheduling approach, and basic terminologies. In section III, we calculate the probability for a special case where the last task is contained in the first, when the tasks are ordered by their ready times. This result is used in section IV to calculate the probability and expectation of the subset size in general cases. The conclusion is in the last section.

¹If we count the sorting cost, the complexity of the task decomposition is $O(N \log N)$.

II Background and Assumption

A. Brief Review

The request of a real-time task, T_i , is represented by a triple $\langle c_i, r_i, d_i \rangle$, where c_i is the *computation time*, r_i is the *ready time*, and d_i is the *deadline*. We also define the task *interarrival time* to be the difference of the ready times of two consecutive tasks, assuming tasks are ordered according to their ready times.

Task T_i is also identified as task i in the rest of this paper, whenever there is no possibility of confusion.

We define w_i ($w_i = [r_i, d_i]$) as the window for task i . The window length $|w_i|$ is $d_i - r_i$.

For a scheduled task i , s_i is its start time, and its finish time f_i is $s_i + c_i$. Clearly, for a feasible schedule, the following equation should hold for all tasks,

$$r_i \leq s_i \leq d_i - c_i.$$

If we consider two tasks i and j , they must have one of these three relations (or we can switch i and j around to have these relations):

1. *leading* - $i \prec j$, if $r_i \leq r_j$, $d_i \leq d_j$ and $w_i \neq w_j$.
2. *matching* - $i \parallel j$, if $r_i = r_j$ and $d_i = d_j$.
3. *containing* - $i \sqcup j$, if $r_i < r_j$ and $d_j < d_i$.

These three relations are shown in Fig. 1. When $i \sqcup j$, we also say that i and j are concurrent.

For an arbitrary set of tasks, any form of task window combination is possible. In our previous report [Yuan89a], we establish the *decomposed leading schedule sequence* by using the leading relation to decompose a set of tasks into a sequence of single schedule subsets, $\tau^1, \tau^2, \dots, \tau^m$. An order among these subsets is therefore defined. The decomposed leading schedule sequence (*DLSS*) is represented by,

$$DLSS = \tau^1 \circ \tau^2 \circ \dots \circ \tau^m$$

such that $\forall k^i \in \tau^i \forall k^j \in \tau^j \ k^i \prec k^j$, for $1 \leq i < j \leq m$, and τ^i cannot be further decomposed for $i = 1, 2, \dots, m$.

In [Yuan89b], we generalize our approach and define the strongly-leading relation to further decompose tasks by taking into account the computation time for the tasks also. In this paper, we focus our attention on the leading relation decomposition.

In [Yuan89a], we show that if a set of feasible schedules exists for the given set of tasks, then there must exist a feasible schedule in which all tasks of a subset, τ^i are scheduled before all tasks of another subset τ^j , if τ^i appears before τ^j in DLSS. Therefore, in the

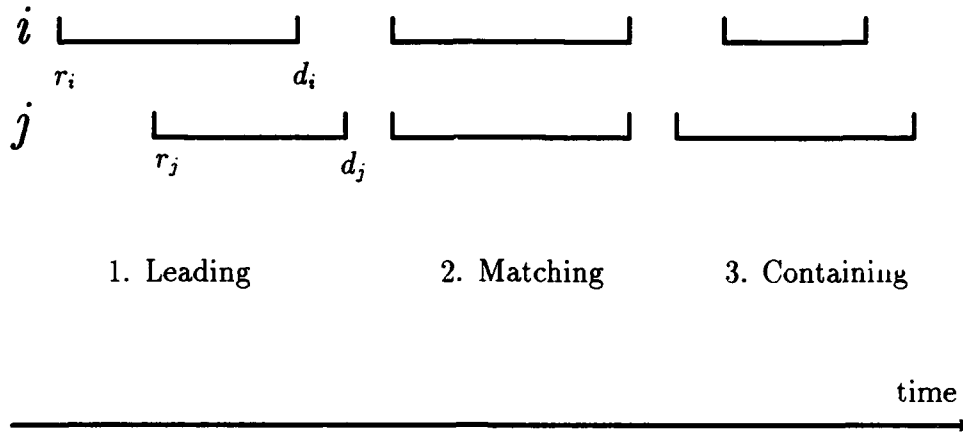


Figure 1: Task window relations

decomposition scheduling technique, first we construct DLSS for the given tasks and then schedule tasks in the subsets, attempting to create a minimum length schedule for each subset. Our approach to finding a minimum length schedule for a subset has been presented in [Yuan89c].

B. Assumption of the Evaluation

Assumptions made in this paper are as follows,

1. only the leading relation is used for the decomposition;
2. the task arrival is a Poisson process with parameter λ_1 ;
3. the task time window length is exponentially distributed with parameter λ_2 .

The Poisson process and the exponential distribution are the most common assumptions in the computer system evaluation to characterize task arrival and workload [Triv82, Lazo84].

III The Probability Distribution for a Special Case

Before presenting the probability density function of the number of tasks in a subset, we first look into a special case.

What we are interested in is the probability for the last task to be contained in the first one, when there are n tasks sorted in a list according to their ready times. The configuration is showed in Fig. 2. We say that these tasks form a *cluster*.

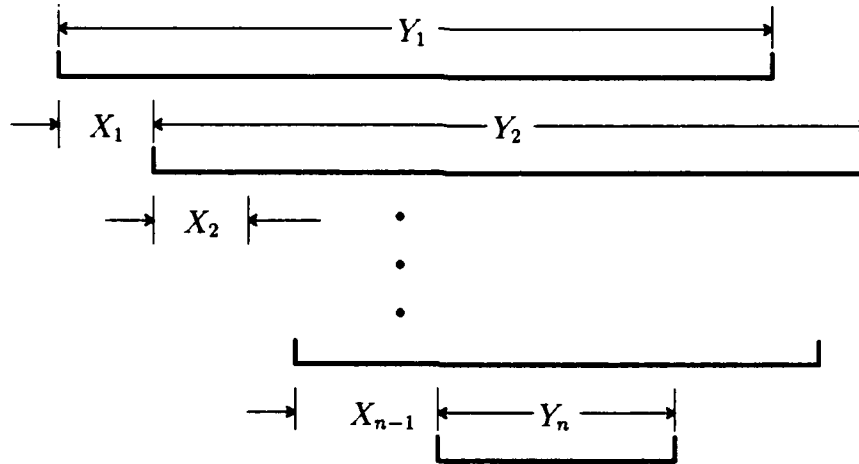


Figure 2: A cluster of n tasks

First, let us examine a two-task case, as showed in Fig. 3. Let random variable X_1 denote the interarrival time between the first task and the second one, and let random variable Y_1 , Y_2 denote the window lengths of the two tasks. X_1 , Y_1 , and Y_2 are independent random variables.

If the first task contains² the second one, the following condition holds:

$$X_1 + Y_2 \leq Y_1$$

Therefore,

²The two task will match if $X_1 = 0$ and $Y_1 = Y_2$. In our model, the probability of two successive task matching is zero and we will not consider that case any further.

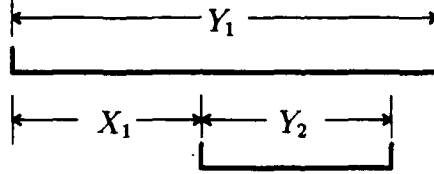


Figure 3: A two task case

$$\begin{aligned}
 P(X_1 + Y_2 \leq Y_1) &= \int_0^\infty \int_0^\infty \int_{x_1+y_2}^\infty f_{X_1, Y_1, Y_2}(x_1, y_1, y_2) dy_1 dx_1 dy_2 \\
 &= \int_0^\infty \int_0^\infty \int_{x_1+y_2}^\infty \lambda_1 \lambda_2^2 e^{-\lambda_1 x_1} e^{-\lambda_2 y_1} e^{-\lambda_2 y_2} dy_1 dx_1 dy_2 \\
 &= \frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})}
 \end{aligned}$$

The calculation of the two-task case can be generalized to the n task case of Fig. 2 ($n \geq 2$). Assume that X_i ($i = 1, \dots, n-1$) is a random variable representing the interarrival time between task i and task $(i+1)$ and random variable Y_i ($i = 1, \dots, n$) is a random variable representing the window length of task i .

The condition for the first task contains the n th task is that

$$X_1 + X_2 + \dots + X_{n-1} + Y_n \leq Y_1$$

In this case, all these n tasks are in the same subset, and cannot be further decomposed by the leading relation.

Therefore,

$$\begin{aligned}
 &P(X_1 + X_2 + \dots + X_{n-1} + Y_n \leq Y_1) \\
 &= \int_0^\infty \dots \int_0^\infty \int_{x_1+x_2+\dots+x_{n-1}+y_n}^\infty f_{X_1, X_2, \dots, X_{n-1}, Y_1, Y_n}(x_1, x_2, \dots, x_{n-1}, y_1, y_n) \\
 &\quad dy_1 dy_n dx_1 \dots dx_{n-1} \\
 &= \int_0^\infty \dots \int_0^\infty \int_{x_1+x_2+\dots+x_{n-1}+y_n}^\infty \lambda_1^{n-1} \lambda_2^2 e^{-\lambda_1(x_1+x_2+\dots+x_{n-1})} e^{-\lambda_2(y_1+y_n)}
 \end{aligned}$$

$$\frac{dy_1 dy_n dx_1 \cdots dx_{n-1}}{2(1 + \frac{\lambda_2}{\lambda_1})^{n-1}}$$

The probability of the n -task cluster is used in the next section to calculate the probability of arbitrary n -task subsets in general.

IV Probability and Expectation

In the last section, we examined the probability of an n -task cluster where the first task contains the last one. In this section, we develop a general approach to calculate the probability of an n -task subset where tasks may have arbitrary relationships, but the subset cannot be partitioned further by the leading relation.

For an arbitrary undecomposable subset, we can always count the first task in a subset as task 1 and the last task as task n .

First let us examine tasks in the order of their ready times. We find that tasks in one subset can be further decomposed into a sequence of groups, as shown in Fig. 4. The definition of the *group* is as follows:

1. the first task in a subset forms a singleton group;
2. the last task of a group is contained in one task in the previous group, and no other task with later ready time has the same property.
3. the tasks between the last task of the current group and the last one of the previous group are the members of the current group. The group excludes the last task of the previous group.

Assuming that there are k groups, we name the last task in the p th group to be task i_p ($1 < p \leq k$). As a special case of two tasks in Fig. 5, $i_1 = 1$, and $i_2 = 2$. A more complicated case with $i_1 = 1$, $i_2 = 4$, and $i_3 = 7$ is presented in Fig. 6.

The next question is how many groups exist in an n -task subset. We observed that for a subset of n tasks. ($n > 1$), the number of groups (k) is less than or equal to $n/2 + 1$, but greater than or equal to two. The proof that the number of groups is greater than or equal to two can be directly inherited from the group definition. The proof that the number of groups is less than $n/2 + 1$ is shown later in this section.

We also name the task as j_p , which is in the p th group, and contains task i_{p+1} of the $(p+1)$ th group. There are $k-1$ such tasks in a k -group subset. In Fig. 5, $j_1 = 1$. In Fig. 6, $j_1 = 1$, $j_2 = 3$.

It is always the case that $i_1 = j_1 = 1$, and $i_k = n$ for an n -task subset with k groups. Tasks between j_p to i_{p+1} form a cluster.

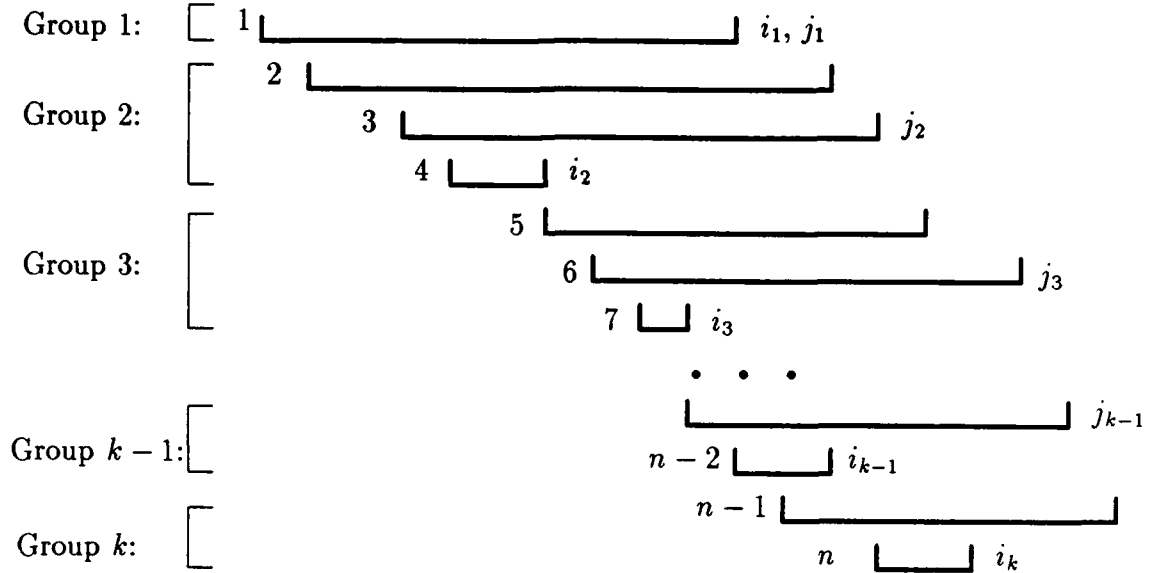


Figure 4: A subset with n tasks

For a k -group subset, j_p is different from i_p for $p = 2, \dots, k-1$. That is, $j_p < i_p$. Suppose that there is a group p ($2 \leq p \leq k-1$), such that $j_p = i_p$ (note that it is impossible that $j_p > i_p$ by the definition). In other words, task i_{p+1} is contained in i_p . But we know that i_p is contained in j_{p-1} . Thus, i_{p+1} is also contained or matched by i_{p-1} . The conclusion contradicts the fact i_p and i_{p+1} are in two different groups.

Next, we calculate the probability that n tasks form a subset which cannot be decomposed by the leading relation.

Let random variable A count the number of tasks in a subset. For $n \geq 2$,

$$\begin{aligned}
 p_A(n) &= p(\exists k \exists i_1, i_2, \dots, i_k, j_1, j_2, \dots, j_{k-1} \\
 &\quad (j_1 = i_1 = 1 < j_2 < i_2 < j_3 < i_3 < \dots < j_{k-1} < i_{k-1} < i_k = n) \\
 &\quad \wedge (j_1 | i_2 \wedge j_2 | i_3 \wedge \dots \wedge j_{k-1} | i_k)) \\
 &= \sum_{k=2}^{n/2+1} \frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})^{i_2-j_1}} \cdot \frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})^{i_3-j_2}} \cdots \frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})^{i_k-j_{k-1}}} \\
 &= \sum_{k=2}^{n/2+1} \frac{1}{2^{k-1}(1 + \frac{\lambda_2}{\lambda_1})^{\sum_{p=2}^{k-1} (i_p-j_p)+n-1}} \tag{1}
 \end{aligned}$$

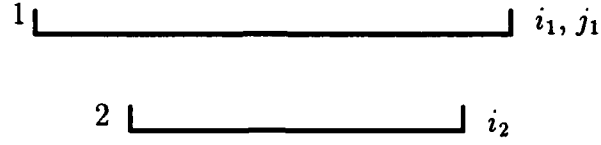


Figure 5: A two-task case

The summation $\sum_{p=2}^{k-1}(i_p - j_p)$ is undecidable, since j_p can be any number between (i_p, i_{p+1}) by the group definition. But we can find the upper and lower bounds for the summation.

By $i_p < j_{p+1} < i_{p+1}$ for $p = 1, \dots, k-2$, we can get $i_p + 1 \leq j_{p+1} \leq i_{p+1} - 1$. Thus,

$$\begin{aligned} \sum_{p=2}^{k-1}(i_p - j_p) &\leq \sum_{p=2}^{k-1}(i_p - (i_{p-1} + 1)) \\ &= i_{k-1} - i_1 - (k-2) \\ &\leq (n-1) - 1 - (k-2) \\ &= n - k \end{aligned}$$

On the other hand,

$$\begin{aligned} \sum_{p=2}^{k-1}(i_p - j_p) &\geq \sum_{p=2}^{k-1}(i_p - (i_p - 1)) \\ &= k - 2 \end{aligned}$$

Note that,

$$k - 2 \leq \sum_{p=2}^{k-1}(i_p - j_p) \leq n - k$$

We have

$$2k \leq n + 2$$

Or,

$$k \leq n/2 + 1$$

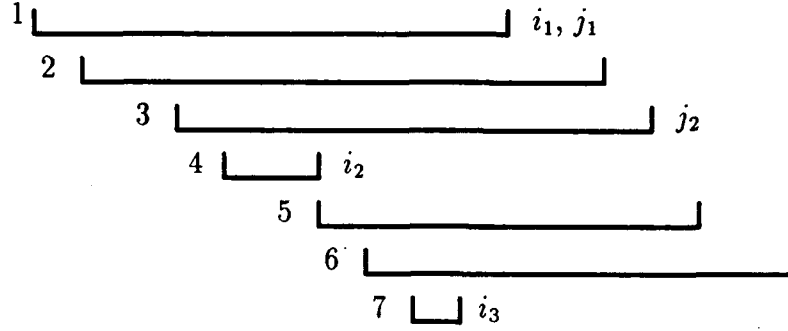


Figure 6: A more complicated example

which shows that the number of groups is less than $n/2 + 1$.

Replacing the upper and lower bounds of $\sum_{p=2}^{k-1}(i_p - j_p)$ in $p_A(n)$, we get the upper and lower bounds for $p_A(n)$ ($n \geq 2$).

First, the upper bound of $p_A(n)$:

$$\begin{aligned}
 p_A(n) &\leq \sum_{k=2}^{n/2+1} \frac{1}{2^{k-1} \left(1 + \frac{\lambda_2}{\lambda_1}\right)^{(k-2)+n-1}} \\
 &= \frac{1 - \left(\frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})}\right)^{n/2}}{\left(1 + \frac{\lambda_2}{\lambda_1}\right)^{n-2} \left(1 + \frac{2\lambda_2}{\lambda_1}\right)}
 \end{aligned}$$

Then the lower bound of $p_A(n)$:

$$\begin{aligned}
 p_A(n) &\geq \sum_{k=2}^{n/2+1} \frac{1}{2^{k-1} \left(1 + \frac{\lambda_2}{\lambda_1}\right)^{(n-k)+n-1}} \\
 &= \begin{cases} \frac{1 - \left(\frac{1}{2} \left(1 + \frac{\lambda_2}{\lambda_1}\right)\right)^{n/2}}{\left(1 + \frac{\lambda_2}{\lambda_1}\right)^{2n-3} \left(1 - \frac{\lambda_2}{\lambda_1}\right)} & \text{when } \lambda_1 \neq \lambda_2 \\ \frac{n}{2^{2n-1}} & \text{otherwise} \end{cases}
 \end{aligned}$$

From the probability of $p_A(n)$ ($n \geq 2$), we can derive the probability that only one task forms a subset. The probability is $p_A(1)$.

$$p_A(1) = 1 - \sum_{n=2}^{\infty} p_A(n)$$

$$= 1 - \sum_{n=2}^{\infty} \sum_{k=2}^{n/2+1} \frac{1}{2^{k-1} \left(1 + \frac{\lambda_2}{\lambda_1}\right)^{\sum_{p=2}^{k-1} (i_p - j_p) + n - 1}} \quad (2)$$

The upper bound of $p_A(1)$:

$$p_A(1) \leq \begin{cases} 1 - \sum_{n=2}^{\infty} \frac{1 - (\frac{1}{2}(1 + \frac{\lambda_2}{\lambda_1}))^{n/2}}{(1 + \frac{\lambda_2}{\lambda_1})^{2n-3} (1 - \frac{\lambda_2}{\lambda_1})} & \text{when } \lambda_1 \neq \lambda_2 \\ 1 - \sum_{n=2}^{\infty} \frac{n}{2^{2n-1}} & \text{otherwise} \end{cases}$$

And the lower bound of $p_A(1)$:

$$p_A(1) \geq 1 - \sum_{n=2}^{\infty} \frac{1 - (\frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})})^{n/2}}{(1 + \frac{\lambda_2}{\lambda_1})^{n-2} (1 + \frac{2\lambda_2}{\lambda_1})}$$

The bounds of $p_A(1)$ can be further simplified. For the upper bound of $p_A(1)$, and $\lambda_1 \neq \lambda_2$:

$$\begin{aligned} p_A(1) &\leq 1 - \sum_{n=0}^{\infty} \frac{1 - (\frac{1}{2}(1 + \frac{\lambda_2}{\lambda_1}))^{(n+2)/2}}{(1 + \frac{\lambda_2}{\lambda_1})^{2n+1} (1 - \frac{\lambda_2}{\lambda_1})} \\ &= 1 - \frac{1 + \frac{\lambda_2}{\lambda_1}}{1 - \frac{\lambda_2}{\lambda_1}} \left(\frac{1}{(1 + \frac{\lambda_2}{\lambda_1})^2 - 1} - \frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2} (2^{1/2} (1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)} \right) \end{aligned}$$

For the upper bound of $p_A(1)$, and $\lambda_1 = \lambda_2$:

$$\begin{aligned} p_A(1) &\leq 1 - \sum_{n=2}^{\infty} \frac{n}{2^{2n-1}} \\ &= \frac{11}{18} \end{aligned}$$

The lower bound of $p_A(1)$ can be simplified too.

$$\begin{aligned} p_A(1) &\geq 1 - \sum_{n=0}^{\infty} \frac{1 - (\frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})})^{(n+2)/2}}{(1 + \frac{\lambda_2}{\lambda_1})^n (1 + \frac{2\lambda_2}{\lambda_1})} \\ &= 1 - \frac{1}{1 + \frac{2\lambda_2}{\lambda_1}} \left(1 + \frac{\lambda_1}{\lambda_2} - \frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2} (2^{1/2} (1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)} \right) \end{aligned}$$

When $\lambda_1 = \lambda_2$,

$$p_A(1) \geq \frac{4}{9}$$

The expectation of the number of tasks in a subset,

$$E(A) = \sum_{n=1}^{\infty} n p_A(n)$$

$$\begin{aligned}
&= \left(1 - \sum_{n=2}^{\infty} \sum_{k=2}^{n/2+1} \frac{1}{2^{k-1}(1 + \frac{\lambda_2}{\lambda_1})^{\sum_{p=2}^{k-1} (i_p - j_p) + n - 1}}\right) \\
&\quad + \sum_{n=2}^{\infty} \left(n \sum_{k=2}^{n/2+1} \frac{1}{2^{k-1}(1 + \frac{\lambda_2}{\lambda_1})^{\sum_{p=2}^{k-1} (i_p - j_p) + n - 1}}\right) \quad (3)
\end{aligned}$$

The boundary condition can be also achieved according to the upper and lower bounds of $p_A(n)$ for $n = 1, \dots, \infty$.

The upper bound of $E(A)$, when $\lambda_1 \neq \lambda_2$:

$$\begin{aligned}
E(A) &\leq 1 - \frac{1 + \frac{\lambda_2}{\lambda_1}}{1 - \frac{\lambda_2}{\lambda_1}} \left(\frac{1}{(1 + \frac{\lambda_2}{\lambda_1})^2 - 1} - \frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)} \right) \\
&\quad + \sum_{n=2}^{\infty} \left(n \cdot \frac{1 - (\frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})})^{n/2}}{(1 + \frac{\lambda_2}{\lambda_1})^{n-2}(1 + \frac{2\lambda_2}{\lambda_1})} \right) \\
&= 1 - \frac{1 + \frac{\lambda_2}{\lambda_1}}{1 - \frac{\lambda_2}{\lambda_1}} \left(\frac{1}{(1 + \frac{\lambda_2}{\lambda_1})^2 - 1} - \frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)} \right) \\
&\quad + \frac{1}{1 + \frac{2\lambda_2}{\lambda_1}} \left(\left(\frac{\lambda_1}{\lambda_2} \right)^2 \left(1 + \frac{\lambda_2}{\lambda_1} \right) + \frac{2\lambda_1}{\lambda_2} \left(1 + \frac{\lambda_2}{\lambda_1} \right) \right. \\
&\quad \left. - \frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)^2} - \frac{2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1} \right)
\end{aligned}$$

If $\lambda_1 = \lambda_2$:

$$\begin{aligned}
E(A) &\leq \frac{11}{18} + \sum_{n=2}^{\infty} \left(n \cdot \frac{1 - (\frac{1}{2(1 + \frac{\lambda_2}{\lambda_1})})^{n/2}}{(1 + \frac{\lambda_2}{\lambda_1})^{n-2}(1 + \frac{2\lambda_2}{\lambda_1})} \right) \\
&= \frac{127}{54} \approx 2.352
\end{aligned}$$

The expectation of the number of tasks in a subset is upper bounded by 2.352.

The lower bound of $E(A)$, when $\lambda_1 \neq \lambda_2$:

$$\begin{aligned}
E(A) &\geq 1 - \frac{1}{1 + \frac{2\lambda_2}{\lambda_1}} \left(1 + \frac{\lambda_1}{\lambda_2} - \frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)} \right) \\
&\quad + \sum_{n=2}^{\infty} \left(n \frac{1 - (\frac{1}{2}(1 + \frac{\lambda_2}{\lambda_1}))^{n/2}}{(1 + \frac{\lambda_2}{\lambda_1})^{2n-3}(1 - \frac{\lambda_2}{\lambda_1})} \right) \\
&= 1 - \frac{1}{1 + \frac{2\lambda_2}{\lambda_1}} \left(1 + \frac{\lambda_1}{\lambda_2} - \frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)} \right) \\
&\quad + \frac{1 + \frac{\lambda_2}{\lambda_1}}{1 - \frac{\lambda_2}{\lambda_1}} \left(\frac{1}{((1 + \frac{\lambda_2}{\lambda_1})^2 - 1)^2} + \frac{2}{(1 + \frac{\lambda_2}{\lambda_1})^2 - 1} \right)
\end{aligned}$$

$$-\frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)^2} - \frac{2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1})$$

If $\lambda_1 = \lambda_2$:

$$\begin{aligned} E(A) &\geq 1 - \frac{1}{1 + \frac{2\lambda_2}{\lambda_1}} \left(1 + \frac{\lambda_1}{\lambda_2} - \frac{(1 + \frac{\lambda_2}{\lambda_1})^{1/2}}{2^{1/2}(2^{1/2}(1 + \frac{\lambda_2}{\lambda_1})^{3/2} - 1)}\right) + \sum_{n=2}^{\infty} \left(n \frac{1}{2^{2n-1}}\right) \\ &= \frac{4}{9} + \sum_{n=2}^{\infty} \frac{n^2}{2^{2n-1}} \\ &= \frac{77}{54} \approx 1.426 \end{aligned}$$

We note that when $\lambda_1 = \lambda_2$, $1.426 \leq E(A) \leq 2.352$.

From the upper and lower bounds of $E(A)$, we find that the boundary value only depend on the ratio of λ_1 and λ_2 (i.e. $\frac{\lambda_1}{\lambda_2}$). The boundaries are also discontinuous at the point of $\lambda_1 = \lambda_2$. In Fig. 7, ratio $\frac{\lambda_1}{\lambda_2}$ changes from 0.2 to 0.9. In Fig. 8, the ratio changes from 1.11 to 10. We can see that when $\frac{\lambda_1}{\lambda_2} > 0.4$, the expectation of task number in a subset is upper-bounded by 7.8. The expectation becomes smaller, as the ratio becomes bigger.

Intuitively, only when the average window length ($1/\lambda_2$) is much greater than the average interarrival time ($1/\lambda_1$), the upper bound can be quite large in comparison with the one when λ_1 is close to λ_2 . For example, when $\frac{1}{\lambda_2} = 5 \cdot \frac{1}{\lambda_1}$, the upper bound can be as large as 27. But on the other hand, the lower bound is close to 8.

From the expectation bounds, we observe that:

- The number of tasks in a subset only depends on the ratio $\frac{\lambda_1}{\lambda_2}$. Changes in the bounds is insignificant when $\frac{\lambda_1}{\lambda_2} > 0.2$.
- In the case where λ_1 is close to λ_2 , the decomposition scheduling is close to the linear computation.

V Conclusion

In this paper, we examine the performance of the decomposition scheduling technique using a stochastic model for tasks, and compute bounds on the expected number of tasks in a subset. We note that the number of tasks in a subset is a function of λ_1/λ_2 of our model, and that over wide range of this ratio, relatively tight bounds exists for the expected number of tasks in a subset.

The computation complexity of the decomposition scheduling depends one the algorithm of decomposition and the scheduling of tasks in a subset. It has been shown that the complexity of decomposition is $O(N)$ [Yuan89a]. The scheduling of tasks in a subset requires computations which grow rapidly. But if the size of subset is limited, the complexity of

this step is limited also. In practical situations where task interarrival times and window lengths have exponential distributions, the complexity of using decomposition scheduling is $O(N)$ with multiplicative constants being a function of λ_1/λ_2 .

References

- [Gare79] Garey, M. R. and Johnson, D. S., *Computers and Intractability, a Guide to the Theory of NP-Completeness*, W. H. Freeman Company, San Francisco, 1979.
- [Lazo84] Lazowska, E. D., Zahorjan, J., Graham, G. S., and Sevcik, K. C., *Quantitative System Performance: Computer System Analysis Using Queueing Network Models*, Prentice-Hall, Inc., New Jersey, 1984.
- [Leho89] Lehoczky, J. P., Sha, L., and Ding, Y., "The Rate Monotonic Scheduling Algorithm: Exact Characterization and Average Case Behavior", In *Proc. IEEE Real-Time Syst. Symp.*, pages 166–171, Dec. 1989.
- [Triv82] Trivedi, K. S., *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*, Prentice-Hall, NJ, 1982.
- [Yuan89a] Yuan, X. and Agrawala, A. K., "A Decomposition Approach to Nonpreemptive Scheduling in Hard Real-Time Systems", In *Proc. IEEE Real-Time Syst. Symp.*, Dec. 1989.
- [Yuan89b] Yuan, X. and Agrawala, A. K., *Decomposition with the Strongly-Leading Relation for Hard Real-Time Scheduling*, Technical Report CS-TR-2346, Dept. of Computer Science, Univ. of Maryland, Coll. Pk., MD 20742, Nov. 1989.
- [Yuan89c] Yuan, X. and Agrawala, A. K., *Scheduling Real-Time Task in Single Schedule Subsets*, Technical Report CS-TR-2347, Dept. of Computer Science, Univ. of Maryland, Coll. Pk., MD 20742, Nov. 1989.
- [Zhao89] Zhao, W. and Stankovic, J. A., "Performance Analysis of FCFS and Improved FCFS Scheduling Algorithms", In *Proc. IEEE Real-Time Syst. Symp.*, pages 156–165, Dec. 1989.

$E(A)$

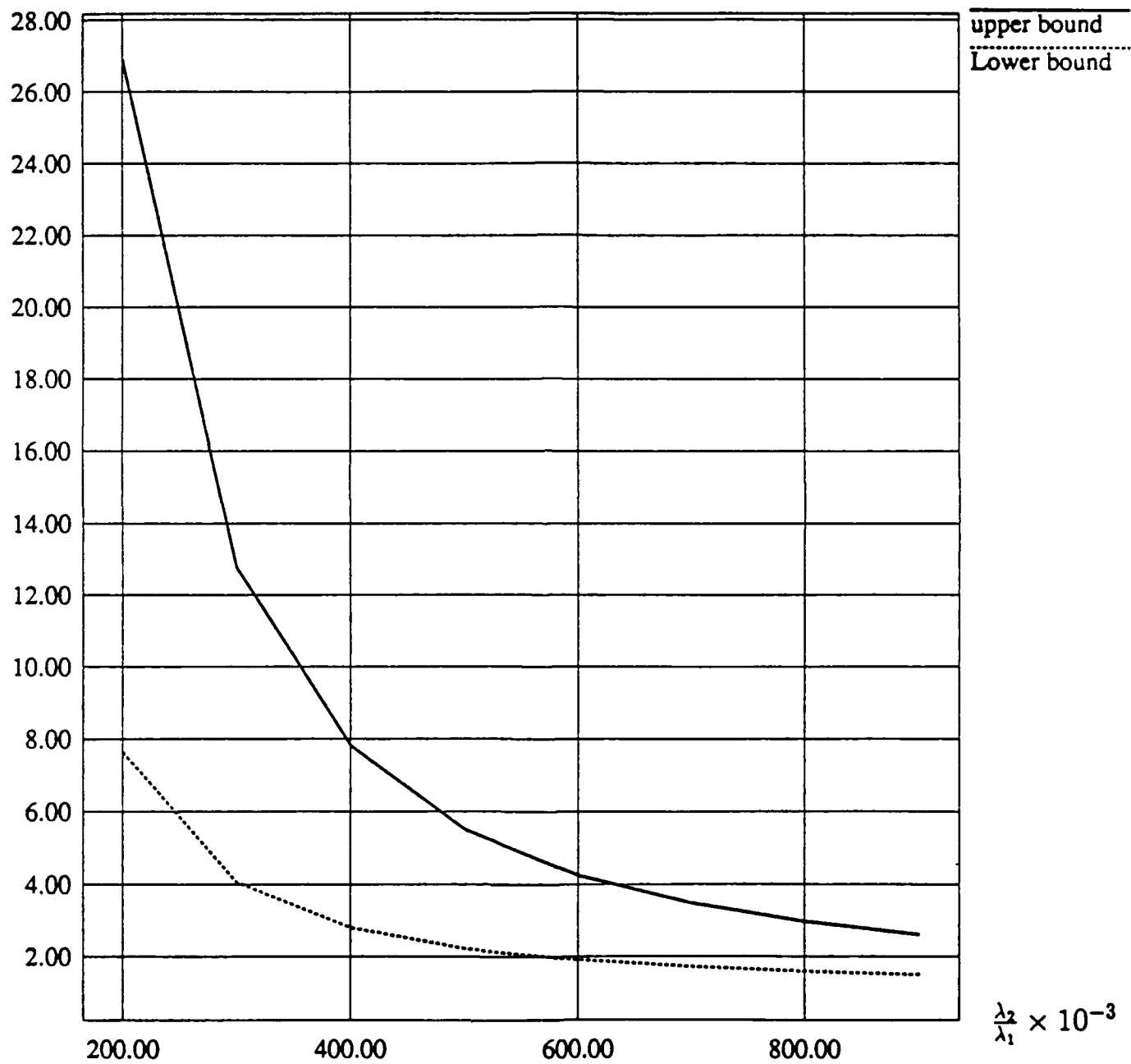


Figure 7

$E(A)$

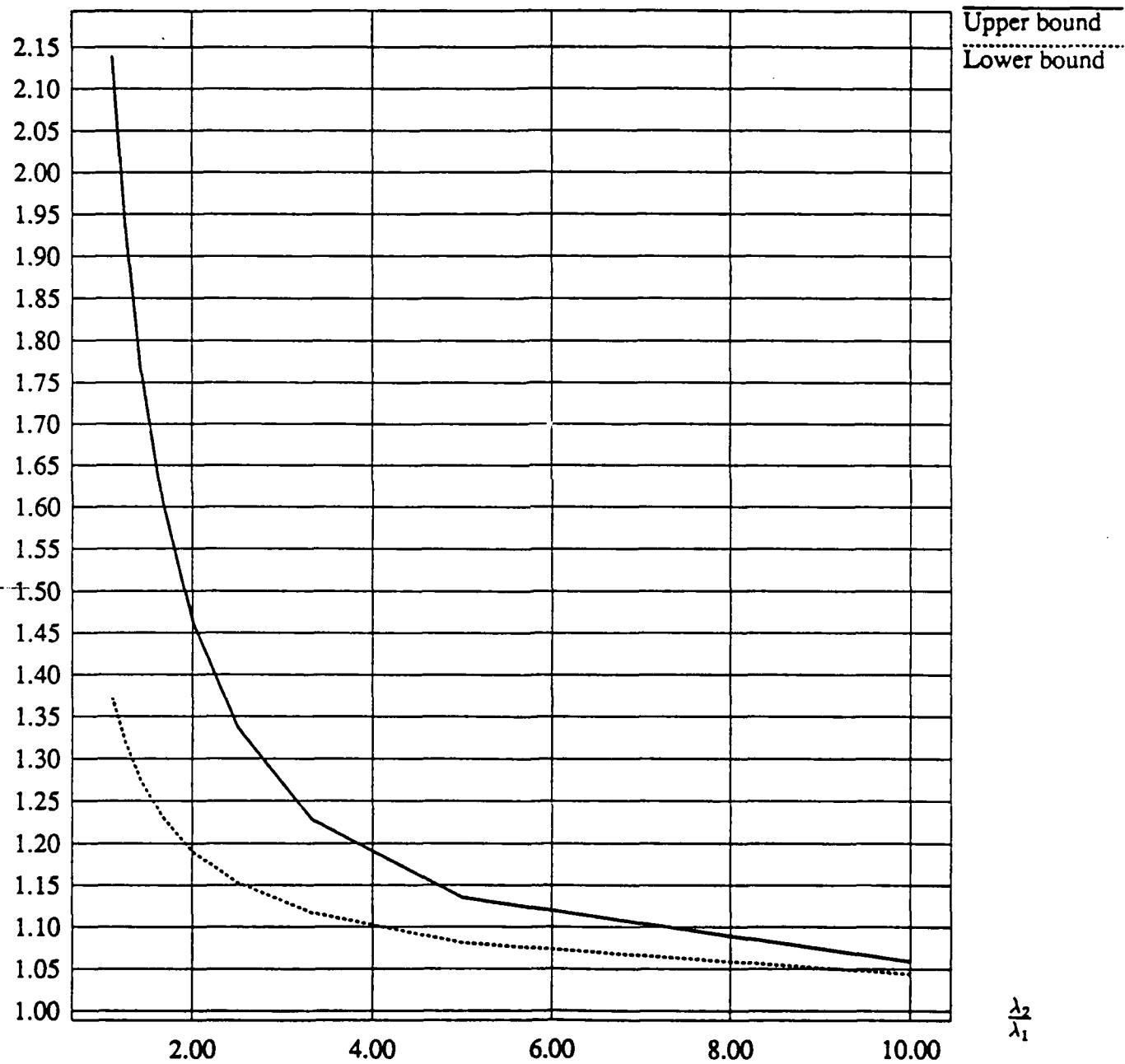


Figure 8

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS N/A	
2a. SECURITY CLASSIFICATION AUTHORITY N/A			3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.	
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE N/A				
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UMIACS-TR- 90-58 CS-TR- 2462			5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION University of Maryland		6b. OFFICE SYMBOL (if applicable)	7a. NAME OF MONITORING ORGANIZATION U.S. Army Strategic Defense Command	
6c. ADDRESS (City, State, and ZIP Code) Department of Computer Science University of Maryland College Park, MD 20742			7b. ADDRESS (City, State, and ZIP Code) Contr & Acq Mgt Ofc. CSSD-H-CRS, P.O. Box 1500 Huntsville, AL 35807-3801	
8a. NAME OF FUNDING / SPONSORING ORGANIZATION		8b. OFFICE SYMBOL (if applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DASG60-87-C-0066	
8c. ADDRESS (City, State, and ZIP Code)			10. SOURCE OF FUNDING NUMBERS	
			PROGRAM ELEMENT NO.	PROJECT NO.
11. TITLE (Include Security Classification) Evaluation of a Decomposition Approach for Real-Time Scheduling Using a Stochastic Model.				
12. PERSONAL AUTHOR(S) X. Yuan, A. Agrawala				
13a. TYPE OF REPORT Technical Report		13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Year, Month, Day) April 30, 1990	15. PAGE COUNT 15
16. SUPPLEMENTARY NOTATION				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP		
19. ABSTRACT (Continue on reverse if necessary and identify by block number)				
Abstract In this paper, a stochastic model is developed to evaluate the decomposition scheduling approach [Yuan89a]. Since the scheduling complexity of the approach directly depends on the number of tasks in every subset, we calculate the probability of the event that "there are n tasks in a subset", for any n, and then the expected number of tasks in a subset. The results indicate that the decomposition scheduling technique not only assures the generation of a feasible schedule if one exists, but also is computationally efficient.				
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL Ashok K. Agrawala			22b. TELEPHONE (Include Area Code) (301) 454-4968	22c. OFFICE SYMBOL