Periodic Resource Model for Compositional Real-Time Guarantees *

Insik Shin and Insup Lee
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104 USA
{ishin,lee}@cis.upenn.edu

Abstract

We address the problem of providing compositional hard real-time guarantees in a hierarchy of schedulers. We first propose a resource model to characterize a periodic resource allocation and present exact schedulability conditions for our proposed resource model under the EDF and RM algorithms. Using the exact schedulability conditions, we then provide methods to abstract the timing requirements that a set of periodic tasks demands under the EDF and RM algorithms as a single periodic task. With these abstraction methods, for a hierarchy of schedulers, we introduce a composition method that derives the timing requirements of a parent scheduler from the timing requirements of its child schedulers in a compositional manner such that the timing requirement of the parent scheduler is satisfied, if and only if, the timing requirements of its child schedulers are satisfied.

1. Introduction

Scheduling is to assign resources according to scheduling policies in order to service workloads. The scheduling can be accurately characterized by a scheduling model that consists of three elements: a resource model, a scheduling algorithm, and a workload model. In real-time scheduling, there has been a growing attention to a hierarchical scheduling framework [4, 8, 10, 12, 5] that supports hierarchical resource sharing under different scheduling algorithms for different scheduling services. A hierarchical scheduling framework can be generally represented as a tree, or a hierarchy, of nodes, where each node represents a scheduling model and a resource is allocated from a parent node to its children nodes, as illustrated in Figure 1. To characterize such a resource allocation between a parent node and a child node, we consider a scheduling interface model $I(G_S, G_D)$, where G_S represents the real-time guarantee that the parent node supplies to the child node and G_D represents the real-time guarantee that the child node demands to the parent node. It is desirable that such a hierarchical scheduling framework satisfies the following properties: (1) *independence:* the schedulability of a scheduling model is analyzed independent of other scheduling models, (2) *separation:* a parent scheduling model and each child scheduling model are separated such that they interact with each other only through a scheduling interface model, (3) *universality:* any scheduling algorithm can be employed in a scheduling model, and (4) *compositionality:* a parent scheduling model is computed from its child scheduling models such that the timing guarantee of the parent scheduling model is satisfied, if and only if, the timing guarantees of its child scheduling models are satisfied together in the framework. In this paper, we introduce a scheduling interface model for constructing a hierarchical scheduling framework that meets these desirable properties.

Deng and Liu [4] and Lipari and Baruah [10] introduced hierarchical scheduling frameworks where a scheduling interface model $I(G_S, G_D)$ is implicitely specified in terms of a uniformly slow resource, or a *fractional* resource $R_F(U_F)$ that is always available only at a fractional capacity U_F. A parent scheduling model provides a fractional resource $R_F(G_S)$ to a child scheduling model, and the child model demands a fractional resource $R_F(G_D)$ to the parent model. The schedulability of the child scheduling model is analyzed with G_S according to the traditional scheduling theories, and G_D can be easily derived from this schedulability analysis. However, G_D does not capture any task-level timing requirements of the child model. Thus, the parent model's scheduler was limited to the EDF scheduler that needs to interact with the child model's scheduler for the knowledge of the task-level deadline information.

Feng and Mok [5] proposed the bounded-delay resource partition model $R_B(U_B, D_B)$ for a hierarchical scheduling framework. This resource partition model describes a behavior of a *partitioned* resource that is available at its full capacity at some times but not available at all at the other

*This research was supported in part by NSF CCR-9988409, NSF CCR-0086147, NSF CCR-0209024, Boeing, and ARO DAAD19-01-1-0473.

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

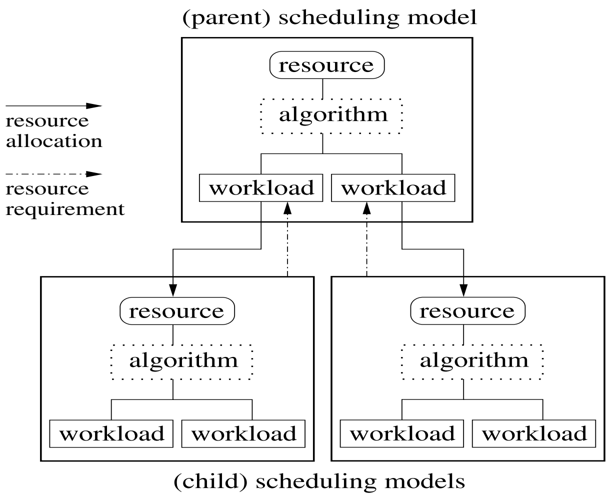


Figure 1. Hierarchcial scheduling framework: parent and children scheduling models.

times, with reference to a fractional resource $R_F(U_B)$. The following property holds between $R_B(U_B, D_B)$ and $R_F(U_B)$: when an event e happens t time after another event e' over R_F, the time distance between e and e' over R_B is between $t - D_B$ and $t + D_B$. This property yields the following sufficient schedulability condition: a scheduling model is schedulable over R_B if all the tasks in the scheduling model complete their execution D_B time earlier than their deadlines over R_F. This bounded-delay resource partition model $R_B(U_B, D_B)$ can be used for specifying the real-time guarantees supplied from a parent model to a child model. The schedulability of the child model is then sufficiently analyzed with $R_B(U_B, D_B)$ accordingly. Even though the child model runs over a partitioned resource, its schedulability is analyzed as if it runs over a fractional resource. Thus, the scheduling algorithms in all child models are required to handle this difference by employing the notion of virtual time scheduling.

Regehr and Stankovic [12] introduced another hierarchical scheduling framework that considers various kinds of real-time guarantees. An implicit scheduling interface model $I(G_S, G_D)$ is specified such that G_S and G_D can be of different kinds of real-time guarantees. They focused on converting one kind of guarantee to another kind of guarantee such that whenever the former is satisfied, the latter is satisifed. With their conversion rules, the schedulability of the child model is sufficiently analyzed such that it is schedulable if G_S is converted to G_D. They assumed that G_D is given for any child model and did not consider the problem of deriving G_D from a child model, which we address in this paper.

In this paper, we propose a periodic resource model $R_P(\Pi, \Theta)$ for a scheduling interface model in a hierarchical scheduling framework. The periodic resource model can characterize a resource allocation of Θ time units every Π time units. When this periodic resource is given as the real-time guarantees supplied from a parent model to a child model, we introduce the necessary and sufficient schedulability conditions for the child model with the EDF and RM scheduling algorithms. Using this exact schedulability analysis, the real-time guarantees demanded by a child model to a parent model can be derived as a traditional periodic task model [11]. With a scheduling interface model that is specified in terms of a periodic resource model and a periodic task model, we introduce a composition method to develop a parent scheduling model from its child scheduling models in a compositional manner. In addition, we derive the utilization bounds of a periodic resource and the capacity bounds of a periodic resource for a set of peridic tasks under the EDF and RM algorithms, respectively.

The rest of this paper is organized as follows: Section 2 presents our system models and problem statements. Section 3 proposes a periodic resource model. For a scheduling model that contains our proposed resource model, Section 4 presents its schedulability analysis and Section 5 provides its schedulability bounds for the RM scheduling algorithm and the EDF scheduling algorithm, respectively. Section 6 shows a composition method for a hierarchical scheduling framework that supports compositional real-time guarantees. Finally, we conclude in Section 7 with discussion on future research.

2. System Model and Problem Statement

A scheduling model M is defined as (W, R, A), where W is a workload model that describes the workloads (applications) supported in the scheduling model, R is a resource model that describes the resources available to the scheduling model, and A is a scheduling algorithm that defines how the workloads share the resources at all times. For the workload model, we consider the Liu and Layland periodic task model [11] that defines a task T as (p, e), where p is the period of T and e is the execution time requirement of T. In this paper, we assume that each task is independent and preemptive. For the scheduling algorithm, we use the rate monotonic (RM) algorithm, which is an optimal fixed-priority algorithm, or the earliest deadline first (EDF) algorithm, which is an optimal dynamic scheduling algorithm. For the resource model, we consider a partitioned resource model. For instance, the bounded-delay resource partition model $R_B(U_B, D_B)$ is a good example of a partitioned resource model, where U_B is the overall capacity (utilization) of a partitioned resource and D_B is the bounded delay between the partitioned resource and a fractional resource with a capacity U_B [5]. A scheduling model

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

$M(W, R, A)$ is said to be *schedulable* if a set of periodic workloads W is schedulable under a scheduling algorithm A with a partitioned resource R. Example 2.1 shows how to model a partitioned resource with a bounded-delay resource partition model $R_B(U_B, D_B)$ and then shows how to analyze the schedulability of a scheduling model containing $R_B(U_B, D_B)$. This example is a motivating example to show the difficulty of a schedulability analysis with a partitioned resource.

Example 2.1 *Consider two periodic tasks, $T_1(7,3)$ and $T_2(21,1)$, that are to execute under the EDF scheduling algorithm with a partitioned resource R that guarantees the resource allocations of 3 time units every 5 time units. In modeling this partitioned resource R with a bounded-delay resource partition model $R_B(U_B, D_B)$, U_B and D_B are determined as follows:*

$$U_B = 3/5 \text{ and } D_B = 4, \text{ by Definitions 4 and 7 in [5].}$$

Then, we can construct a scheduling model M as $M(\{T_1, T_2\}, R_B(0.6, 4), EDF)$. Over the fractional resource with a fractional capacity $U_B = 0.6$, T_1 and T_2 finish their execution at least D time units earlier than their deadlines, where $D = 2$ in this example. According to Theorem 1 in [5], M is schedulable if $D \geq D_B$. In this example, since $D = 2$ and $D_B = 4$, it turns out $D < D_B$. Hence, the schedulability of M is inconclusive[1].

The bounded-delay resource partition model is introduced to characterize a delay between a partitioned resource and its corresponding fractional resource, not necessarily to characterize a periodic behavior of a partitioned resource. In this paper, we propose a periodic resource model $\Gamma(\Pi, \Theta)$ that describes a partitioned resource guaranteeing an allocation of Θ time units every Π time unit period. With our proposed periodic resource model, it is possible to consider the following problems.

1. *Exact schedulability analysis:* given W, Γ, and A, determine whether or not $M(W, \Gamma, A)$ is schedulable in the necessary and sufficient way.

2. *Periodic capacity bound:* given W,A, and Π, find the smallest possible periodic capacity bound (Θ^*/Π) such that $M(W, \Gamma(\Pi, \Theta), A)$ is schedulable if $\Theta \geq \Theta^*$. This problem can be viewed as modeling a workload task set W under algorithm A as a single periodic task $T(p, e)$ by abstracting its timing requirements such that $p = \Pi$ and $e = \Theta^*$.

3. *Utilization bound:* given Γ and A, find the largest possible utilization bound UB such that $M(W, \Gamma, A)$ is schedulable if
$$\sum_{T_i \in W} \frac{e_i}{p_i} \leq UB.$$

4. *Algorithm set:* given W and Γ, find a set of algorithms $\mathcal{A}$ such that $M(W, \Gamma, A)$ is schedulable if $A \in \mathcal{A}$.

5. *Compositional guarantee:* given n scheduling models, derive a new scheduling model from the n scheduling models such that we call the new scheduling model a parent scheduling model of the n models and that the parent scheduling model is schedulable, if and only if, the n child models are schedulable.

In this paper, we address the problems #1, #2, #3, and #5, but not the problem #4.

3. Periodic Resource Model

For real-time systems, the Liu and Layland periodic task model [11] and its various extensions have been accepted as a workload model that accurately characterizes many traditional hard real-time applications, such as digital control and constant bit-rate voice/video transmission. Many scheduling algorithms based on this workload model have been shown to have good performance and well-understood behaviors. We define a periodic application as a real-time application that consists of periodic tasks and thus exhibits a periodic behavior. In abstracting a periodic application with a workload model, we naturally consider an approach to abstract it as a single periodic task[2]. We can then directly use the traditional real-time scheduling theories based on the periodic task model. When a resource is allocated to a workload such that the workload's periodic timing requirement is satisfied, then the resource allocation to the workload clearly has a periodic behavior. Thus, there needs to be a resource model that characterizes accurately a periodic behavior of a resource allocation. We propose a periodic resource model $\Gamma(\Pi, \Theta)$ in order to characterize a partitioned resource that guarantees allocations of Θ time units every Π time units, where a resource period Π is a positive integer and a resource allocation time Θ is a real number in $(0, \Pi]$. For example, $\Gamma(5, 3)$ describes a partitioned resource that guarantees 3 time units every 5 time units, and $\Gamma(k, k)$ represents a dedicated resource that is available all the time, for any integer k.

[1] It is shown in Example 4.1 that the schedulability of M is conclusive, when the partitioned resource R is modeled with our proposed periodic resource model.

[2] In this paper, we do not address the issue of modeling a non-periodic application as a single periodic task. This issue has been addressed well in the literature [9, 14, 15, 4].

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

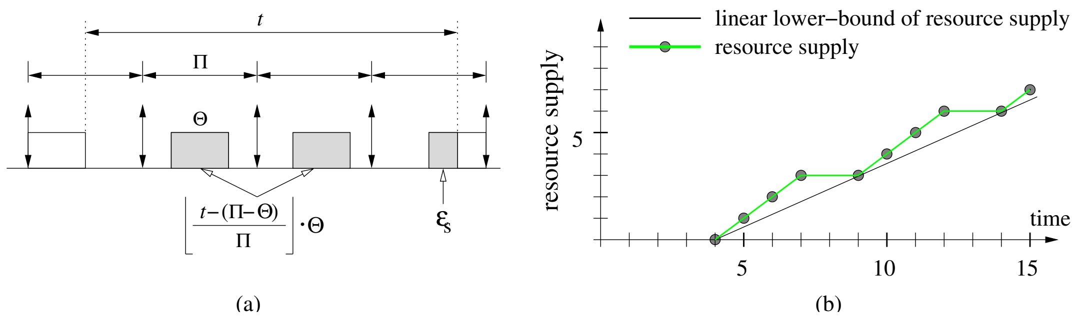


Figure 2. Resource supply function: (a) how to calculate the minimum resource supply of Γ during t and (b) the minimum resource supply and its linear lower-bound for $\Gamma(5,3)$.

We define the *resource supply* of a resource as the amount of resource allocations that the resource provides. During a time interval, a dedicated resource can clearly provide a resource supply equal to the interval length, however, a partitioned resource is to provide a resource supply that is smaller than or equal to the interval length. For a periodic resource $\Gamma(\Pi, \Theta)$, we define a resource supply bound function $\mathbf{sbf}_\Gamma(t)$ of a time interval length t that calculates the minimum resource supply of Γ during t time units as follows:

$$\mathbf{sbf}_\Gamma(t) = \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor \cdot \Theta + \epsilon_s, \tag{1}$$

where

$$\begin{aligned} \epsilon_s &= \max\left(t - (\Pi - \Theta) - \Pi \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor - (\Pi - \Theta), 0\right) \\ &= \max\left(t - 2(\Pi - \Theta) - \Pi \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor, 0\right). \end{aligned}$$

Figure 2 (a) illustrates how Eq. (1) calculates the minimum resource supply of Γ during t. The supply bound function $\mathbf{sbf}_\Gamma$ is a non-decreasing step function. Here, the following lemma introduces a linear function that lower-bounds $\mathbf{sbf}_\Gamma(t)$.

Lemma 1 *A linear supply bound function* $\mathbf{lsbf}_\Gamma(t)$ *lower-bounds* $\mathbf{sbf}_\Gamma(t)$ *as follows:*

$$\mathbf{lsbf}_\Gamma(t) = \frac{\Theta}{\Pi}(t - 2 \cdot (\Pi - \Theta)) \leq \mathbf{sbf}_\Gamma(t).$$

Proof. We consider two cases depending on the value of ϵ_s in $\mathbf{sbf}_\Gamma(t)$: (1) $\epsilon_s = 0$ and (2) $\epsilon_s > 0$.

For the first case where $\epsilon_s = 0$,

$$t - 2(\Pi - \Theta) - \Pi \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor \leq 0. \tag{2}$$

In this case,

$$\mathbf{sbf}_\Gamma(t) = \Theta \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor.$$

From Eq. (2), we have

$$\frac{\Theta}{\Pi}(t - 2(\Pi - \Theta)) \leq \Theta \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor. \tag{3}$$

Eq. (3) shows $\mathbf{lsbf}_\Gamma(t) \leq \mathbf{sbf}_\Gamma(t)$.

For the second case where $\epsilon_s > 0$,

$$t - 2(\Pi - \Theta) - \Pi \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor > 0. \tag{4}$$

In this case,

$$\mathbf{sbf}_\Gamma(t) = t - 2(\Pi - \Theta) - (\Pi - \Theta) \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor.$$

From Eq. (4), we have

$$\frac{t - 2(\Pi - \Theta)}{\Pi} - \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor > 0. \tag{5}$$

With Eq. (5) and the definition of Θ ($0 < \Theta \leq \Pi$), we have

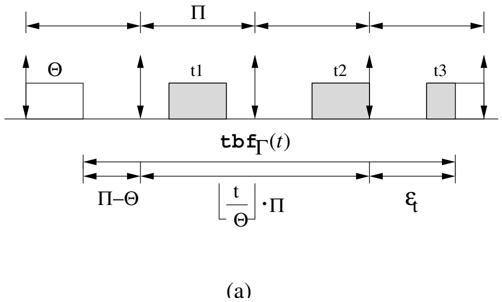
$$\begin{aligned} & \mathbf{sbf}_\Gamma(t) - \mathbf{lsbf}_\Gamma(t) \\ = \; & (\Pi - \Theta)\left(\frac{t - 2(\Pi - \Theta)}{\Pi} - \left\lfloor \frac{t - (\Pi - \Theta)}{\Pi} \right\rfloor\right) \\ > \; & 0. \end{aligned}$$

□

Example 3.1 *Consider a periodic resource* $\Gamma(5,3)$. *Figure 2 (b) plots its minimum supply* $\mathbf{sbf}_\Gamma(t)$ *and its linear supply lower bound* $\mathbf{lsbf}_\Gamma(t)$. *For instance, during a time interval of 10 time units, the periodic resource* $\Gamma(5,3)$ *supplies at least a resource allocation of 4 time units.*

We define the *service time* of a resource as the duration that it takes for the resource to provide a resource supply. It is obvious that it takes a service time of t time units for a dedicated resource to provide a resource supply of t time units. It is also clear that it takes a service time longer than

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)



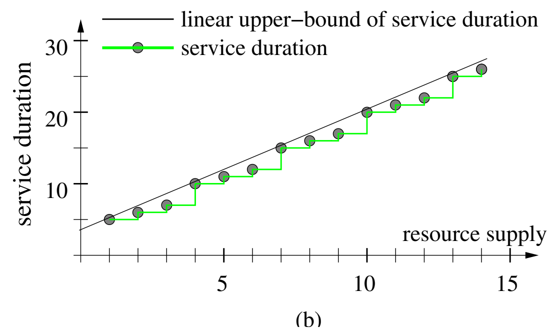


Figure 3. Service time function: (a) how to calculate the maximum service service of Γ for a supply of $t = t1 + t2 + t3$ and (b) the maximum service time and its linear upper-bound for $\Gamma(5, 3)$.

or equal to t time units for a partitioned resource to provide a resource supply of t time units. For a periodic resource $\Gamma(\Pi, \Theta)$, we define a service time bound function $\mathbf{tbf}_\Gamma(t)$ of a resource supply of t that calculates the maximum service time of Γ for a t-time-unit resource supply as follows:

$$\mathbf{tbf}_\Gamma(t) = (\Pi - \Theta) + \Pi \cdot \left\lfloor \frac{t}{\Theta} \right\rfloor + \epsilon_t, \tag{6}$$

where

$$\epsilon_t = \begin{cases} \Pi - \Theta + t - \Theta \left\lfloor \frac{t}{\Theta} \right\rfloor & \text{if } \left(t - \Theta \left\lfloor \frac{t}{\Theta} \right\rfloor > 0 \right) \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Figure 3 (a) illustrates how Eq. (6) calculates the maximum service time of Γ for a resource supply of t. The service time bound function $\mathbf{tbf}_\Gamma(t)$ is a non-decreasing step function. Here, the following lemma shows a linear function that upper-bounds $\mathbf{tbf}_\Gamma(t)$.

Lemma 2 *A linear service time bound function* $\mathbf{ltbf}_\Gamma(t)$ *upper-bounds* $\mathbf{tbf}_\Gamma(t)$ *as follows:*

$$\mathbf{ltbf}_\Gamma(t) = \frac{\Pi}{\Theta} \cdot t + 2(\Pi - \Theta) \geq \mathbf{tbf}_\Gamma(t).$$

Proof. The idea of proving this lemma is similar to that for Lemma 1. Due to the space limit, we refer [13] for a full proof. □

Example 3.2 *Consider a periodic resource* $\Gamma(5, 3)$. *Figure 3 (b) plots its maximum service time* $\mathbf{tbf}_\Gamma(t)$ *and its linear service time upper bound* $\mathbf{ltbf}_\Gamma(t)$. *For instance, it takes up to 7 time units to receive a resource supply of 3 time units.*

4. Schedulability Analysis

For a scheduling model $M(W, \Gamma, A)$ that characterizes all its three elements, we address the problem of analyzing the schedulability of M. This section presents sufficient and necessary schedulability conditions for a set of periodic workloads under the EDF algorithm and a fixed-priority scheduling algorithm with a periodic resource.

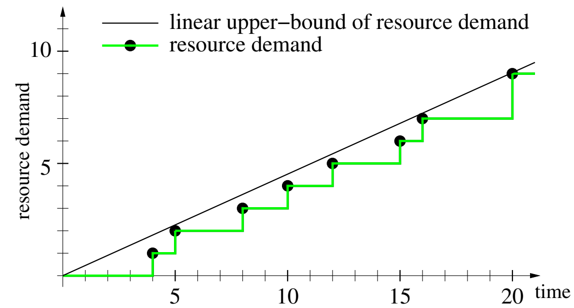


Figure 4. An example of a maximum demand bound and its linear upper-bound.

4.1. Schedulability Analysis under EDF Scheduling

We define the *resource demand* of a workload set as the amount of resource allocation that the workload set requests. For a periodic workload set W, we define a resource demand bound function $\mathbf{dbf}_W(t)$ of a time interval length t that calculates the maximum resource demand of W under EDF scheduling during t time units as follows:

$$\mathbf{dbf}_W(t) = \sum_{T_i \in W} \left\lfloor \frac{t}{p_i} \right\rfloor \cdot e_i.$$

Figure 4 shows an example of the maximum resource demand of a periodic workload set W. As shown in Figure 4, the resource demand function $\mathbf{dbf}_W(t)$ is a discrete step function. Here, the following lemma shows a linear function that upper-bounds $\mathbf{dbf}_W(t)$.

Lemma 3 *A linear demand bound function* $\mathbf{ldbf}_W(t)$ *upper-bounds* $\mathbf{dbf}_W(t)$ *as follows:*

$$\mathbf{ldbf}_W(t) = U_W \cdot t \geq \mathbf{dbf}_W(t),$$

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

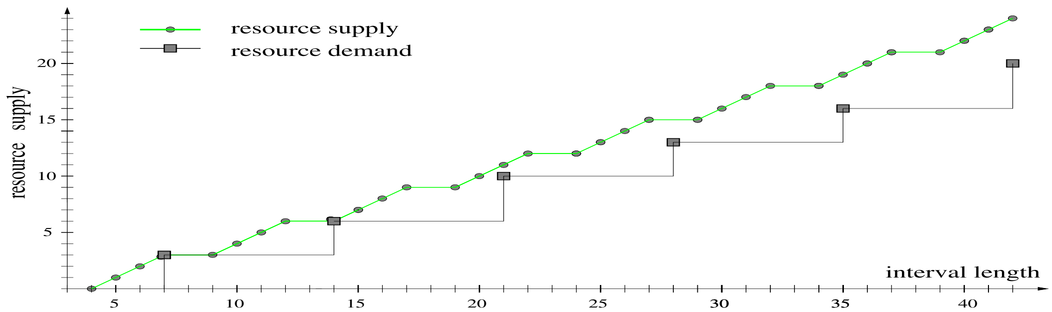


Figure 5. An example of EDF schedulability analysis.

where U_W is the utilization of the workload set W.

Proof. According to the definition of $\mathbf{dbf}_W(t)$ and U_W, we have the followings:

$$\begin{aligned}\mathbf{dbf}_W(t) &= \sum_{T_i \in W} \left\lfloor \frac{t}{p_i} \right\rfloor \cdot e_i \\ &\leq \sum_{T_i \in W} \frac{t}{p_i} \cdot e_i = U_W \cdot t = \mathbf{ldbf}_W(t).\end{aligned}$$

□

With a dedicated resource, a workload set W is schedulable with the EDF scheduling algorithm if and only if the resource demand during a time interval is no greater than the length of the time interval for all time intervals during a hyperperiod [2], i.e.,

$$\mathbf{dbf}_W(t) \leq t, \quad \text{for all } 0 < t \leq 2 \cdot LCM_W, \tag{8}$$

where LCM_W is the least common multiplier of the periods of all the workloads in the workload set W.

Now, we consider a sufficient and necessary schedulability condition for a workload set with a partitioned resource. The traditional schedulability condition of Eq. (8) basically means that for any time interval, the resource demand of a workload set during the time interval should be no greater than the resource supply of a resource during the same interval. Since the resource demand of a workload set is independent of a resource, the left-hand side of Eq. (8) is not affected by a partitioned resource. However, the right-hand side of Eq. (8) that represents the resource supply should change depending on a partitioned resource. For a periodic partitioned resource Γ, since the resource supply bound function $\mathbf{sbf}_\Gamma(t)$ defines the minimum resource supply of Γ for a time interval length t, the right-hand side of Eq. (8) is replaced by $\mathbf{sbf}_\Gamma(t)$.

Theorem 1 (EDF Schedulability Analysis) *For a given scheduling model $M(W, \Gamma, EDF)$, M is schedulable if and only if the resource demand of W during a time interval is no greater than the resource supply of Γ during the same time interval for all time intervals during a hyperperiod, i.e.,*

$$\forall 0 < t \leq 2 \cdot LCM_W : \mathbf{dbf}_W(t) \leq \mathbf{sbf}_\Gamma(t). \tag{9}$$

Proof. To show the necessity, we prove the contrapositive, i.e., if Eq. (9) is false, all workload members of W are not schedulable by EDF. If the total resource demand of W under EDF scheduling during t exceeds the total resource supply provided by Γ during t, there is clearly no feasible schedule.

To show the sufficiency, we prove the contrapositive, i.e., if all workload members of W are not schedulable by EDF, then Eq. (9) is false. Let t_2 be the first instant at which a job of some workload member T_i of W that misses its deadline. Let t_1 be the latest instant at which the resource supplied to W was idle or was executing a job whose deadline is after t_2. By the definition of t_1, there is a job whose deadline is before t_2 was released at t_1. Without loss of generality, we can assume that $t = t_2 - t_1$. Since T_i misses its deadline at t_2, the total demand placed on W in the time interval $[t_1, t_2)$ is greater than the total supply provided by Γ in the same time interval length t. □

Example 4.1 *Consider a scheduling model $M(W, \Gamma(5, 3), EDF)$, where $W = \{T_1(7, 3), T_2(21, 1)\}$. Figure 5 plots the minimum resource supply of Γ and the maximum resource demand of W. According to Theorem 1, M is schedulable if and only if the resource supply of Γ is no less than the resource demand of W for a time interval of length t, for $0 < t \leq 2 \cdot LCM_\Gamma$. It is shown in Figure 5 that $\mathbf{dbf}_W(t) \leq \mathbf{sbf}_\Gamma(t)$, for $0 < t \leq 42$. Thus, M is schedulable.*

4.2. Schedulability Analysis under Fixed-Priority Scheduling

For a given scheduling model $M'(W, \Gamma(1, 1), FP)$, where $\Gamma(1, 1)$ represents a dedicated resource and FP is

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

a fixed-priority scheduling algorithm, M' is schedulable if and only if the worst-case response time of each workload in W is no greater than its relative deadline [7]. The worst-case response time r_i of a workload T_i occurs when T_i experiences the worst-case interference from its higher-priority workloads. T_i is maximally interfered by its higher-priority workloads when it is released together with all of its higher-priority workloads at the same time, which is called a *critical instant*. Using the iterative response time analysis method introduced in [1], r_i can be computed as follows:

$$r_i^{(k)} = e_i + \sum_{T_k \in HP(W,T_i)} \left\lceil \frac{r_i^{(k-1)}}{p_k} \right\rceil \cdot e_k, \text{ where } T_k = (p_k, e_k), \tag{10}$$

where $HP(W,T_i)$ denotes a subset of W that consists of the higher-priority workloads of T_i. The iteration continues until $r_i^{(k)} = r_i^{(k-1)}$, where $r_i^{(0)} = e_i$.

Now, we consider a periodic partitioned resource $\Gamma(\Pi, \Theta)$ such that Π is not necessarily equal to Θ and a scheduling model $M(W, \Gamma(\Pi, \Theta), FP)$. For the schedulability analysis of M, we first consider the worst-case response time r_i of a workload T_i under fixed-priority scheduling with a periodic partitioned resource $\Gamma(\Pi, \Theta)$. The response time analysis method of Eq. (10) has been developed under the traditional assumption of a dedicated resource and therefore under the assumption that the service duration of a resource for a resource supply of t time is t time. The service duration of a partitioned resource for a resource supply of t time can be longer than t time. Considering this, we extend the traditional response time analysis method of Eq. (10) for a periodic partitioned resource. For a workload T_i with a periodic partitioned resource $\Gamma(\Pi, \Theta)$, its maximum response time r_i can be computed using the following iterative method:

$$r_i^{(k)}(\Gamma) = \mathbf{tbf}_\Gamma(I_i^{(k)}), \tag{11}$$

where

$$I_i^{(k)} = e_i + \sum_{T_k \in HP(W,T_i)} \left\lceil \frac{r_i^{(k-1)}(\Gamma)}{p_k} \right\rceil \cdot e_k. \tag{12}$$

I_i captures the worst-case interference to a workload T_i from its higher-priority workloads, and $r_i(\Gamma)$ represents the maximum service duration of a resource supply of I_i. The iteration continues until $r_i^{(k)} = r_i^{(k-1)}$, where $r_i^{(0)} = e_i$.

Theorem 2 (Fixed-Priority Schedulability Analysis)
For a given scheduling model $M(W, \Gamma, FP)$, where FP is a fixed-priority scheduling algorithm, M is schedulable if and only if

$$\forall T_i \in W : r_i(\Gamma) \le p_i, \quad \textit{where } T_i = (p_i, e_i). \tag{13}$$

Proof. An individual workload is schedulable with Γ if and only if the maximum service duration of Γ for the execution time of the workload is no greater than the workload's relative deadline. The maximum response time of a workload T_i occurs when T_i experiences the worst-case interference from its higher-priority workloads and Γ provides the worst-case resource supply. For a workload T_i, the worst-case interference from its higher-priority workloads is given by I_i and the maximum service duration of Γ for I_i is given by $\mathbf{tbf}_\Gamma(I_i)$, which is the maximum response time r_i of T_i with Γ. Consequently, a necessary and sufficient condition for T_i to meet its deadline with Γ is $r_i(\Gamma) \le p_i$. The entire workload set W is schedulable with Γ if and only if each of the workloads is schedulable with Γ. This means

$$\forall T_i \in W : r_i(\Gamma) \le p_i. \tag{14}$$

Thus, Eq. (14) is necessary and sufficient for the workload set to be schedulable with Γ. $\square$

Example 4.2
Consider a scheduling model $M(W, \Gamma(5,3), RM)$, where $W = \{T_1(7,3), T_2(21,1)\}$. In this example, we first show how to calculate the maximum response time of T_1 in M. According to Eq. (12), $I_1^{(1)} = 3 + \lceil 0/3 \rceil \cdot 3 = 3$. According to Eq. (11), $r_1^{(1)}(\Gamma) = \mathbf{tbf}_\Gamma(3) = (5-3) + 3 \cdot \lfloor 3/3 \rfloor + \epsilon_t = 5$, where $\epsilon_t = 0$. Subsequently, $I_1^{(2)} = 3$ and $r_1^{(2)}(\Gamma) = 5$. Since $r_1^{(2)}(\Gamma) = r_1^{(1)}(\Gamma)$, the iteration stops here, and $r_1(\Gamma) = 5$. We then show how to calculate $r_2(\Gamma)$. Initially, $I_2^{(1)} = 1 + \lceil 1/3 \rceil \cdot 3 = 4$ and $r_2^{(1)}(\Gamma) = \mathbf{tbf}_\Gamma(4) = (5-3) + 3 \cdot \lfloor 4/3 \rfloor + \epsilon_t = 10$, where $\epsilon_t = (3-2) + 4 - 3 \cdot \lfloor 4/3 \rfloor = 2$. Then, $I_2^{(2)} = 7$ and $r_2^{(2)}(\Gamma) = 15$. Subsequently, $I_2^{(3)} = 10$ and $r_2^{(3)}(\Gamma) = 20$. Eventually, $I_2^{(4)} = 10$, $r_2^{(4)}(\Gamma) = 20$. Since $r_2^{(3)}(\Gamma) = r_2^{(3)}(\Gamma)$, the iteration stops here, and $r_2(\Gamma) = 20$. According to Theorem 2, since $r_1(\Gamma) \le p_1$ and $r_2(\Gamma) \le p_2$, M is schedulable.

5. Schedulability Bounds

For a scheduling model M that characterizes its two elements but does not characterize the other element, we address the problems of deriving a schedulability bound for the missing element of M. When M characterizes its workload W and scheduling algorithm A, we find a periodic capacity bound for its resource Γ that guarantees the schedulability of $M(W, \Gamma, A)$. Similarly, when M characterizes its resource Γ and scheduling algorithm A, we find a utilization bound for its workload W that guarantees the schedulability of $M(W, \Gamma, A)$. We derive the periodic capacity bounds and the utilization bounds for the EDF algorithm and the RM algorithm, respectively.

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

5.1. Periodic Capacity Bounds

We define the *periodic capacity* C_Γ of a periodic resource $\Gamma(\Pi, \Theta)$ as Θ/Π. In this section, given a set of periodic workloads W under a scheduling algorithm A, we address the problem of characterizing a set of periodic resources that satisfy the timing requirements of W under A. A reasonable approach is to classify such a set of periodic resources by their periodic capacities subject to their resource periods. For such a classification, we define the *periodic capacity bound* $PCB_W(\Pi, A)$ of a resource period Π as a number such that a scheduling model $M(W, \Gamma(\Pi, \Theta), A)$ is schedulable if

$$PCB_W(\Pi, A) \leq \frac{\Theta}{\Pi}.$$

With this $PCB_W(\Pi, A)$, we can easily determine whether or not a given periodic resource $\Gamma(\Pi, \Theta)$ can satisfies the timing requirements of W under A. Moreover, we can easily abstract the timing requirements of W under A as a single periodic workload $T(p, e)$ such that $p = \Pi$ and $e = \Pi \cdot PCB_W(\Pi, A)$. In this section, we derive the periodic capacity bounds for the EDF algorithm and the RM algorithm.

5.1.1 Periodic Capacity Bound for EDF scheduling

Given W under the EDF scheduling algorithm, we first address the problem of finding the optimal (minimum) periodic capacity bound of a resource period Π. The following theorem derives the optimal bound using the exact schedulability condition in Theorem 1.

Theorem 3 (Optimal Periodic Capacity Bound for EDF) *For a given periodic workload set W under the EDF scheduling algorithm, the optimal (minimum) periodic capacity bound $PCB^*_W(\Pi, EDF)$ of a period Π is*

$$PCB^*_W(\Pi, EDF) = \frac{\Theta^*}{\Pi},$$

where Θ^* is the smallest possible Θ satisfying

$$\forall 0 < t \leq 2LCM_W : \mathbf{dbf}_W(t) \leq \mathbf{sbf}_\Gamma(t). \qquad (15)$$

A scheduling model $M(W, \Gamma(\Pi, \Theta), EDF)$ is schedulable if and only if $PCB^*_W(\Pi, EDF) \leq C_\Gamma$.

Proof. According to Theorem 1, $M(W, \Gamma(\Pi, \Theta), EDF)$ is schedulable if and only if Eq. (15) holds with Θ. Since Θ^* is the smallest possible Θ satisfying Eq. (15), the schedulability of M is guaranteed if and only if $(\Theta^*/\Pi) \leq C_\Gamma$. □

Due to the `max` operation in Eq. (15), Theorem 3 inherently presents an algorithm to find the optimal periodic capacity bound rather than a function to derive it. Here, the following theorem presents a function to derive a periodic capacity bound.

Theorem 4 (Periodic Capacity Bound for EDF) *For a given periodic workload set W under the EDF scheduling algorithm, a periodic capacity bound $PCB_W(\Pi, EDF)$ of a resource period Π is*

$$PCB_W(\Pi, EDF) = \frac{\Theta^+}{\Pi}, \qquad \textit{where}$$

$$\Theta^+ = \max_{0<t\leq 2LCM_W} \left(\frac{\sqrt{(t-2\Pi)^2 + 8\Pi\mathbf{dbf}_W(t)} - (t-2\Pi)}{4} \right). \qquad (16)$$

Proof. Since $\mathbf{lsbf}_\Gamma(t) \leq \mathbf{sbf}_\Gamma(t)$, we can have the following from Theorem 1:

$$\mathbf{dbf}_W(t) \leq \mathbf{lsbf}_\Gamma(t) = \frac{\Theta}{\Pi}(t - 2\Pi + 2\Theta) \leq \mathbf{sbf}_\Gamma(t). \qquad (17)$$

From Eq. (17), we have

$$\Theta \geq \frac{\sqrt{(t-2\Pi)^2 + 8\Pi\mathbf{dbf}_W(t)} - (t-2\Pi)}{4}. \qquad (18)$$

Hence, when we find Θ^+ such that Θ^+ is the smallest possible Θ satisfying Eq. (18), we can guarantee that $M(W, \Gamma(\Pi, \Theta), EDF)$ is schedulable if $(\Theta^+/\Pi) \leq C_\Gamma$. □

Example 5.1 *For a given $W = \{T_1(7, 3), T_2(12, 3)\}$ under the EDF algorithm, this example considers the problem of deriving a periodic capacity bound. We systematically find the optimal periodic capacity bound of resource period 5 according to the algorithm in Theorem 3, as 0.75 with $\Theta^* = 3.75$. That is, we can model W under the EDF algorithm as a single period workload $T(5, 3.75)$ preserving its timing requirement. Hence, for a scheduling model $M(W, \Gamma, EDF)$ where Γ does not yet characterize its resource, we define Γ as $\Gamma(5, 3.75)$ and make M schedulable. According to Theorem 4, we can numerically find a periodic capacity bound of resource period 5 as 0.77, with $\Theta^+ = 3.85$. We can also model W under EDF as $T(5, 3.85)$.*

5.1.2 Periodic Capacity Bound for RM Algorithm

In this section, we address the issues of deriving periodic capacity bounds for the RM scheduling algorithm. Given W under the RM scheduling algorithm, the following theorem shows how to find the optimal (minimum) periodic capacity bound of a resource period Π using the exact schedulability condition in Theorem 2.

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

Theorem 5 (Optimal Periodic Capacity Bound for RM) *For a given periodic workload set W under the RM scheduling algorithm, the optimal (minimum) periodic capacity bound $PCB^*_W(\Pi, RM)$ of a resource period Π for a periodic partition resource Γ is*

$$PCB^*_W(\Pi, RM) = \frac{\Theta^*}{\Pi},$$

where Θ^* is the smallest possible Θ satisfying the following necessary and sufficient schedulability condition in Theorem 2:

$$\forall T_i \in W : r_i(\Gamma) \le p_i, \quad \textit{where } T_i = (p_i, e_i). \tag{19}$$

A scheduling model $M(W, \Gamma(\Pi, \Theta), RM)$ is schedulable if and only if $PCB^*_W(\Pi, RM) \le C_\Gamma$.

Proof. According to Theorem 2, $M(W, \Gamma(\Pi, \Theta), RM)$ is schedulable if and only if Eq. (19) is true with Θ. Since Θ^* is the smallest possible Θ satisfying Eq. (19), the schedulability of M is guaranteed if and only if $(\Theta^*/\Pi) \le C_\Gamma$. □

The supply bound function $\mathbf{tbf}_\Gamma(t)$ that is used to calculate the maximum response time $r_i^{(k)}(\Gamma)$ has a discrete operation as shown in Eq. (7). Like the optimal periodic capacity bound for the EDF algorithm, due to this discrete operation, Theorem 5 inherently presents an algorithm to find the optimal periodic capacity bound rather than a function to derive it. Here, we present an integrative method to derive a periodic capacity bound using $\mathbf{ltbf}_\Gamma(t)$ that linearly upper-bounds $\mathbf{tbf}_\Gamma(t)$.

Recall that the maximum response time $r_i^{(k)}(\Gamma)$ is computed with the following iterative method:

$$r_i^{(k)}(\Gamma) = \mathbf{tbf}_\Gamma(I_i^{(k)}), \tag{20}$$

where

$$I_i^{(k)} = e_i + \sum_{T_k \in HP(W,T)} \left\lceil \frac{r_i^{(k-1)}(\Gamma)}{p_k} \right\rceil \cdot e_k. \tag{21}$$

Let $\hat{r}_i^{(k)}(\Gamma)$ denote the upper-bound of the maximum response time that is computed as follows:

$$\hat{r}_i^{(k)}(\Gamma) = \mathbf{ltbf}_\Gamma(I_i^{(k)}), \tag{22}$$

Lemma 4 *A scheduling model $M(W, \Gamma, RM)$ is schedulable if $\forall T_i \in W : \hat{r}_i(\Gamma) \le p_i$.*

Proof. Since $\mathbf{tbf}_\Gamma(t) \le \mathbf{ltbf}_\Gamma(t)$, clearly, $r_i^{(k)}(\Gamma) \le \hat{r}_i^{(k)}(\Gamma)$. Then, it is obvious that for all $T_i \in W$, if $\hat{r}_i(\Gamma) \le p_i$, then $r_i(\Gamma) \le p_i$. □

Theorem 6 (Periodic Capacity Bound for RM) *For a given periodic workload set W under the RM scheduling algorithm, a periodic capacity bound $PCB_W(\Pi, RM)$ of a period Π for a periodic partition resource Γ is*

$$PCB_W(\Pi, RM) = \frac{\Theta^+}{\Pi}, \qquad \textit{where}$$

$$\Theta^+ = \max_{\forall T_i \in W} \left(\frac{-(p_i - 2\Pi) + \sqrt{(p_i - 2\Pi)^2 + 8\Pi I_i}}{4} \right), \tag{23}$$

where

$$I_i = e_i + \sum_{T_k \in HP(W,T)} \left\lceil \frac{p_i}{p_k} \right\rceil \cdot e_k. \tag{24}$$

Proof. According to Theorem 2, $M(W, \Gamma, RM)$ is schedulable even though for all $T_i \in W$, $r_i = p_i$. I_i captures the worst-case interference to a workload T_i from its higher-priority workloads. According to Lemma 4, then $M(W, \Gamma(\Pi, \Theta), RM)$ is schedulable, if $\mathbf{ltbf}_\Gamma(I_i) \le p_i$ for all $T_i \in W$, that is,

$$\forall T_i \in W : \mathbf{ltbf}_\Gamma(I_i) = \frac{\Pi}{\Theta} \cdot I_i + 2(\Pi - \Theta) \le p_i, \tag{25}$$

Θ^+ captures the smallest possible Θ satisfying Eq. (25). Thus, it is guaranteed that $M(W, \Gamma(\Pi, \Theta), RM)$ is schedulable if $(\Theta^+/\Pi) \le C_\Gamma$. □

Example 5.2 *Given $W = \{T_1(7, 3), T_2(12, 3)\}$ under the RM scheduling algorithm, this example shows how to derive periodic capacity bounds of resource period 5. According to Theorem 5, we can systematically find the optimal periodic capacity bound $PCB^*_W(5, RM)$ as 0.85, with $\Theta^* = 4.25$. Thus, we can model W under RM as a single periodic workload $T(5, 4.25)$. According to Theorem 6, we can also numerically find a periodic capacity bound $PCB_W(5, RM)$. According to Eq. (24), $I_1 = 3$ and $I_2 = 9$. According to Eq. (23), $\Theta^+ = 4.27$ since Eq. (25) is true for T_1 with $\Theta = 3.59$ and true for T_2 with $\Theta = 4.27$. Thus, $PCB_W(5, RM) = 0.85$ with $\Theta^+ = 4.27$, and we can also model W under RM as $T(5, 4.27)$.*

5.2. Utilization Bounds

Given a periodic resource Γ, we define the *utilization bound* $UB_\Gamma(A)$ of a scheduling algorithm A as a number such that a scheduling model $M(W, \Gamma, A)$ is schedulable if

$$\sum_{T_i \in W} \frac{e_i}{p_i} \le UB_\Gamma(A).$$

These utilization bounds are useful in performing an admission test of a periodic workload set W over a periodic resource Γ with a scheduling algorithm A. In this section, we

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

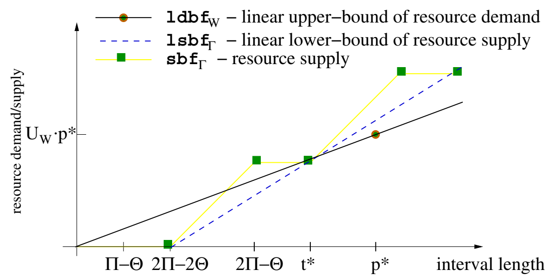


Figure 6. An example of linear upper-bound of demand and linear lower-bound of supply.

derive the utilization bounds for the EDF algorithm and for the RM algorithm.

5.2.1 Utilization Bound for EDF Algorithm

When a scheduling model $M(W, \Gamma(\Pi, \Theta), EDF)$ is schedulable, it is clear that the utilization of W is no greater than the periodic capacity of Γ. That is,

$$U_W \leq C_\Gamma = \frac{\Theta}{\Pi}. \tag{26}$$

Recall the definitions of two linear functions, $\mathbf{ldbf}_W(t)$ and $\mathbf{lsbf}_\Gamma(t)$, as follows:

$$\mathbf{ldbf}_W(t) = U_W \cdot t \text{ and } \mathbf{lsbf}_\Gamma(t) = \frac{\Theta}{\Pi}(t - 2 \cdot (\Pi - \Theta)).$$

When $M(W, \Gamma(\Pi, \Theta), EDF)$ is schedulable, we can easily observe that the slope of $\mathbf{ldbf}_W(t)$ is no greater than the slope of $\mathbf{lsbf}_\Gamma(t)$, since $U_W \leq \frac{\Theta}{\Pi}$. As shown in Figure 6, it is obvious that if $\mathbf{ldbf}_W(t^*) \leq \mathbf{lsbf}_\Gamma(t^*)$, then $\mathbf{ldbf}_W(t) \leq \mathbf{lsbf}_\Gamma(t)$ for all $t > t^*$. Let p^* denotes the smallest period in a periodic workload set W. The following lemma shows that if $\mathbf{ldbf}_W(p^*) \leq \mathbf{lsbf}_\Gamma(p^*)$, then $M(W, \Gamma, EDF)$ is schedulable.

Lemma 5 *When $\mathbf{lsbf}_\Gamma(p^*) \geq \mathbf{ldbf}_W(p^*)$, a scheduling model $M(W, \Gamma, EDF)$ is schedulable, where p^* is the smallest period in W.*

Proof. Due to the space limit, we refer [13] for a full proof. □

Based on Lemma 5, the following theorem presents a utilization bound for the EDF algorithm over a periodic resource.

Theorem 7 (Utilization Bound for EDF Algorithm) *Given a periodic resource $\Gamma(\Pi, \Theta)$, a utilization bound $UB_\Gamma(EDF)$ of the EDF algorithm for a periodic workload set W is*

$$UB_\Gamma(EDF) = \frac{\Theta}{\Pi}\Big(1 - \frac{2(\Pi - \Theta)}{p^*}\Big), \tag{27}$$

where p^* is the smallest period in the workload set W.

Proof. Lemma 5 says that if $\mathbf{ldbf}_W(p^*) \leq \mathbf{lsbf}_\Gamma(p^*)$, $M(W, \Gamma(\Pi, \Theta), EDF)$ is schedulable. When $\mathbf{ldbf}_W(p^*) \leq \mathbf{lsbf}_\Gamma(p^*)$, we can get

$$\mathbf{ldbf}_W(p^*) = p^* \cdot U_W \leq \mathbf{lsbf}_\Gamma(p^*) = \frac{\Theta}{\Pi} \cdot (p^* - 2(\Pi - \Theta)).$$

With the above equation, we can get

$$\begin{aligned} U_W \leq \frac{\mathbf{lsbf}_\Gamma(t)}{p^*} &= \frac{\Theta}{\Pi}\Big(\frac{p^* - 2(\Pi - \Theta)}{p^*}\Big) \\ &= \frac{\Theta}{\Pi}\Big(1 - \frac{2(\Pi - \Theta)}{p^*}\Big). \end{aligned}$$

□

Example 5.3 *Given a periodic resource $\Gamma(5, 3)$ under the EDF scheduling, this example shows how to derive a utilization bound. Let p^* denotes the shortest period of a periodic workload set W. According to Theorem 7, when $p^* = 10$, $UB_\Gamma(EDF) = (3/5) \cdot (1 - (2(5 - 3)/10) = 0.36$. When $p^* = 100$, $UB_\Gamma(EDF) = (3/5) \cdot (1 - (2(5 - 3)/100) = 0.58$.*

5.2.2 Utilization Bound for RM Algorithm

In this subsection, we derive a utilization bound for the RM scheduling algorithm. Given a periodic resource Γ, the following theorem derives a utilization bound of the RM algorithm.

Theorem 8 (Utilization Bound for RM Algorithm) *Given a periodic resource $\Gamma(\Pi, \Theta)$, a utilization bound $UB_\Gamma(RM)$ of the RM scheduling algorithm for a set of m periodic workloads is*

$$UB_\Gamma(RM) = \frac{\Theta}{\Pi}\Big(m(\sqrt[m]{2} - 1) - \frac{\sqrt[m]{2}(\Pi - \Theta)}{p^*}\Big), \tag{28}$$

where p^* is the shortest period of W.

Proof. Due to the space limit of this paper, we present a sketch of our proof. We refer [13] for a full proof that is based on Theorem 3, Theorem 4 and Theorem 5 in [11]. Consider a periodic workload set $W = \{T_1, \cdots, T_m\}$, where T_i is defined as (p_i, e_i), for $1 \leq i \leq m$. Without loss of generality, we assume that $p_m > p_{m-1} > \cdots > p_2 > p_1$. The main idea of the proof is to show that the

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

achievable utilization factor of W is minimized when the execution time requirement e_i of each workload T_i is set as follows:

$$e_i = \frac{\Theta}{\Pi}(p_{i+1} - p_i), \quad \text{for } 1 \le i < m$$
$$e_m = \mathbf{sbf}_\Gamma(p_m) - 2(e_1 + e_2 + \cdots + e_{m-1}),$$

and the period p_i of each workload T_i is set as follows:

$$p_i = 2^{(i-1)/m} \cdot p_1, \quad \text{for } 1 \le i \le m.$$

□

For large m, Eq. (28) becomes

$$UB_\Gamma(RM) = \frac{\Theta}{\Pi}\Big(\ln 2 - \frac{\sqrt[m]{2}(\Pi - \Theta)}{p^*}\Big) \simeq \frac{\Theta}{\Pi}\Big(\ln 2 - \frac{\Pi - \Theta}{p^*}\Big).$$

Example 5.4 *Given a periodic resource $\Gamma(5,3)$ under the RM scheduling, this example shows how to derive a utilization bound. Let p^* denote the shortest period of a periodic workload set W. Assume W has a large number of workloads. According to Theorem 8, when $p^* = 10$,* $UB_\Gamma(RM) = (3/5) \cdot (\ln 2 - (5-3)/10) = 0.29$. *When* $p^* = 100$, $UB_\Gamma(RM) = (3/5) \cdot (\ln 2 - (5-3)/100) = 0.40$.

6. Compositional Real-Time Guarantees

A hierarchical scheduling framework is said to support *compositional real-time guarantee* if each parent scheduling model is computed from its child scheduling models such that the real-time guarantee of the parent scheduling model is satisfied, if and only if, the real-time guarantees of its child scheduling models are satisfied in the framework. In this section, we address the problem of developing a parent scheduling model from its child scheduling model in order to construct a hierarchical scheduling framework that supports compositional real-time guarantees. The following theorem introduces a composition method that derives a parent scheduling model from its child scheduling models and shows how to construct a hierarchical scheduling framework supporting compositional real-time guarantees.

Definition 6.1 (Composition Method)
Given multiple scheduling models $M_1, \cdots, M_n$, we derive a scheduling model $M_P(W_P, \Gamma_P, A_P)$ from $M_1, \cdots, M_n$ as follows:

- *we assume that A_P and Π_P are given;*
- *we derive W_P by simply mapping the resource model of a child scheduling model $\Gamma_i(\Pi_i, \Theta_i)$ to its periodic task $T_i(p_i, e_i)$ such that $W_P = \{T_1(\Pi_1, \Theta_1), \cdots, T_n(\Pi_n, \Theta_n)\}$;*
- *we first derive $PCB^*_{W_P}(\Pi_P, A_P)$ according to Theorem 3 or Theorem 5 depending on A_P. If $PCB^*_{W_P}(\Pi_P, A_P)$ is derived, we then compute Θ_P such that $\Theta_P = \Pi_P \cdot PCB^*_{W_P}(\Pi_P, A_P)$.*

Theorem 9 (Compositional Real-Time Guarantees)
Given multiple scheduling models $M_1, \cdots, M_n$ that are individually schedulable, we derive a scheduling model $M_P(W_P, \Gamma_P, A_P)$ from $M_1, \cdots, M_n$ according to the composition method in Definition 6.1. Then, we construct a hierarchical scheduling framwork H such that M_P is a parent scheduling model of $M_1, \cdots, M_n$. H supports the compositional real-time guarantees such that M_P is schedulable, if and only if, $M_1, \cdots, M_n$ are schedulable in the framework.

Proof. To show its sufficiency, we consider $M_1, \cdots, M_n$ are schedulable together in the framework. That is, the combined timing requirements of $M_1, \cdots, M_n$ can be satisfied. According to the composition method, for all $1 \le i \le n$, T_i in W_P has the same timing requirements as Γ_i in M_i has. Thus, the combined timing requirements of $T_1, \cdots, T_n$ can be also satisfied. Then, $PCB^*_{W_P}(\Pi_P, A_P)$ is derived as Θ^*_P/Π_P such that $0 < \Theta^*_P \le \Pi_P$, according to Theorem 3 and Theorem 5. Since the composition method derives Θ_P as Θ^*_P, M_P is derived to be schedulable.

To show its necessity, we consider M_P is schedulable. Then, for all $1 \le i \le n$, T_i and its corresponding Γ_i are guaranteed to receive e_i time units every p_i time units. That is, M_i receives from M_P a resource allocation of Θ_i time units every Π_i time units. Thus, $M_1, \cdots, M_n$ are schedulable together in the framework. □

Example 6.1 *Consider two schedulable scheduling models $M_1(W_1, \Gamma_1(7,3), A_1)$ and $M_2(W_2, \Gamma_2(12,3), A_2)$. This example shows how to derive a parent scheduling model M_P from M_1 and M_2 preserving the real-time guarantees of M_1 and M_2. For $M_P(W_P, \Gamma_P(\Pi_P, \Theta_P), A_P)$, we assume that A_P is given as EDF and Π_P is given as 5. Then, we derive W_P and Θ_P according to the composition method in Definition 6.1. We construct W_P as $W_P = \{T_1(7,3), T_2(12,3)\}$ and compute $PCB^*_{W_P}(5, EDF)$ to derive Θ_P. As shown in Example 5.1, $PCB^*_{W_P}(5, EDF)$ is 0.75 according to Theorem 3. Then, Θ_P is set as $5 \cdot 0.75 = 3.75$. Now, we create M_P as $M_P(\{T_1(7,3), T_2(12,3)\}, \Gamma(5, 3.75), EDF)$. According to Theorem 3, M_P is schedulable.*

7. Conclusion

We proposed a resource model that can describe a periodic behavior of a partitioned resource and provided the exact schedulability condition for a scheduling model with our

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE

proposed model. For a hierarchical scheduling framework, we introduced a scheduling interface model that bridges two independently developed scheduling models by modeling the temporal guarantees of a parent scheduling model as a periodic resource model and abstracting the temporal requirement of a child scheduling model as a periodic workload model. With this scheduling interface model, a scheduling model can use any scheduling algorithm and its schedulability is independently analyzed without any interaction with another scheduling model. Furthermore, we provided a composition method to derive a parent scheduling model from its child scheduling model in a compositional manner such that if the parent scheduling model is schedulable, if and only, its child scheduling models are schedulable.

In this paper, we derive a parent scheduling model from its child scheduling models. To preserve the timing requirements of the child scheduling models, the parent scheduling model may demand more timing requirements than a simple sum of the timing requirements of all individual scheduling models. We are evaluating the overhead to support the compositional timing guarantees. We are also studying the properties that our compositional framework has, i.e., an associativity. In this paper, we consider only a periodic task workload model for characterizing hard real-time applications. Our future work is to extend our resource model and its scheduling theory to different task workload models for soft real-time applications such as the (m, k)-firm deadline model [6] and the weakly hard task model [3]. In this paper, we assume that each task is independent. However, tasks may interact with each other through communications and synchronizations. The study of this issue remains as a topic of future research.

Acknowledgements

We thank to Xiang "Alex" Feng for helpful discussions on their bounded-delay resource partition model and John Regehr for clarifying their hierarchical scheduling framework. We also thank to referees for several useful comments.

References

[1] N. Audsley, A. Burns, M. Richardson, and A. Wellings. Applying new scheduling theory to static priority pre-emptive scheudling. *Software Engineering Journal*, 8(5):284–292, 1993.

[2] S. Baruah, R. Howell, and L. Rosier. Algorithms and complexity concerning the preemptive scheduling of periodic, real-time tasks on one processor. *Journal of Real-Time Systems*, 2:301–324, 1990.

[3] G. Bernat, A. Burns, and A Llamosi. Weakly hard real-time systems. *IEEE Transactions on Computers*, 50(4):308–321, 2001.

[4] Z. Deng and J. W.-S. Liu. Scheduling real-time applications in an open environment. In *Proc. of IEEE Real-Time Systems Symposium*, pages 308–319, December 1997.

[5] M. Feng and A. Mok. A model of hierarchical real-time virtual resources. In *Proc. of IEEE Real-Time Systems Symposium*, pages 26–35, December 2002.

[6] M. Hamdaoui and P. Ramanathan. A dynamic priority assignment technique for streams with (m, k)-firm deadlines. *IEEE Transactions on Computers*, 44(12):1443–1451, 1995.

[7] M. Joseph and P. Pandya. Finding response times in a real-time system. *The Computer Journal*, 29(5):390 – 395, 1986.

[8] T.-W. Kuo and C.H. Li. A fixed-priority-driven open environment for real-time applications. In *Proc. of IEEE Real-Time Systems Symposium*, pages 256–267, December 1999.

[9] J. Lehoczky, L. Sha, and J.K. Strosnider. Enhanced aperiodic responsiveness in hard-real-time environments. In *Proc. of IEEE Real-Time Systems Symposium*, pages 261–270, 1987.

[10] G. Lipari and S. Baruah. A hierarchical extension to the constant bandwidth server framework. In *Proc. of IEEE Real-Time Technology and Applications Symposium*, pages 26–35, May 2001.

[11] C.L. Liu and J.W. Layland. Scheduling algorithms for multi-programming in a hard-real-time environment. *Journal of the ACM*, 20(1):46 – 61, 1973.

[12] J. Regehr and J. Stankovic. HLS: A framework for composing soft real-time schedulers. In *Proc. of IEEE Real-Time Systems Symposium*, pages 3–14, December 2001.

[13] I. Shin and I. Lee. Periodic resource model for compositional real-time guarantees. Technical report, MS-CIS-03-28, University of Pennsylvania, 2003.

[14] B. Sprunt, L. Sha, and J. Lehoczky. Aperiodic task scheduling for hard real-time systems. *Journal of Real-Time Systems*, 1(1):27–60, 1989.

[15] M. Spuri and G. C. Buttazzo. Scheduling aperiodic tasks in dynamic priority systems. *Journal of Real-Time Systems*, 10(2):179–210, 1996.

Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03)
0-7695-2044-8/03 $ 17.00 © 2003 IEEE