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

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

antees 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