Towards Exploiting the Preservation Strategy of Deferrable Servers

Reinder J. Bril and Pieter J.L. Cuijpers

Technische Universiteit Eindhoven (TU/e), Department of Mathematics and Computer Science, Den Dolech 2, 5600 AZ Eindhoven, The Netherlands
r.j.bril@tue.nl, p.j.l.cuijpers@tue.nl

Abstract

Worst-case response time analysis of hard real-time tasks under hierarchical fixed priority pre-emptive scheduling (H-FPPS) has been addressed in a number of papers. Based on an exact schedulability condition, we showed in [4] that the existing analysis can be improved for H-FPPS when deferrable servers are used. In this paper, we reconsider response time analysis and show that improvements are not straightforward, because the worst-case response time of a task is not necessarily assumed for the first job when released at a critical instant. The paper includes a brief investigation of best-case response times and response jitter.

1. Introduction

Today, fixed-priority pre-emptive scheduling (FPPS) is a de-facto standard in industry for scheduling systems with real-time constraints. A major shortcoming of FPPS, however, is that temporary or permanent faults occurring in one application can hamper the execution of other applications. To resolve this shortcoming, the notion of *resource reservation* [8] has been proposed. Resource reservation provides *isolation* between applications, effectively protecting an application against other, malfunctioning applications.

In a basic setting of a real-time system, we consider a set of independent applications, where each application consists of a set of periodically released, hard real-time tasks that are executed on a shared resource. We assume two-level hierarchical scheduling, where a *global* scheduler determines which application should be provided the resource and a *local* scheduler determines which of the chosen application's tasks should execute. Although each application could have a dedicated scheduler, we assume FPPS for every application. For temporal protection, each application is associated a dedicated reservation. We assume a *periodic resource model* [11] for reservations. Conceivable implementations include FPPS for global scheduling using a specific type of server, such as the *periodic server* [5] or the *deferrable server* [12].

Worst-case response time analysis of real-time tasks under hierarchical FPPS (H-FPPS) using deferrable servers has been addressed in [1, 5, 6, 10], where the analysis presented in [5] improves on the earlier work. Based on an exact schedulability condition, we showed in [4] that the analysis in [5] can be improved for a deferrable server at highest priority when that server is exclusively used for hard real-time tasks. In this paper, we reconsider worst-case response time analysis. We show that improving the existing analysis is not straightforward, because the worst-case response time of a task is not necessarily assumed for the first job when released at a critical instant. For illustration purposes, we consider a specific class of subsystems $\mathcal{S}$ and an example subsystem $S \in \mathcal{S}$. The paper includes a brief investigation of best-case response times and response jitter.

This paper is organized as follows. In Section 2, we briefly recapitulate existing results for our class of subsystems $\mathcal{S}$ and introduce our example subsystem $S \in \mathcal{S}$. This example clearly illustrated the potential for improvement. We investigate response times and response jitter for our example in Section 3, and conclude the paper in Section 4.

2. A recapitulation of existing analysis

In this section, we briefly recapitulate existing analysis. We start with a description of a scheduling model for our class $\mathcal{S}$ and present our example $S \in \mathcal{S}$. Next, we recapitulate the analysis for a periodic resource model [11], a periodic server [5], and a deferrable server [4], which we illustrate by means of S. We conclude with an overview.

2.1. A scheduling model

We assume FPPS for global scheduling, and consider a class of subsystems $\mathcal{S}$ consisting of an application with a single, periodic hard real-time task τ and an associated server σ at highest priority. The server σ is characterized by a *replenishment period* T^σ and a *capacity* C^σ, where $0 < C^\sigma \leq T^\sigma$. Without loss of generality, we assume that σ is replenished for the first time at time $\varphi^\sigma = 0$. The task τ is characterized by a *period* T^τ, a *computation time* C^τ, and a *relative deadline* D^τ, where $0 < C^\tau \leq D^\tau \leq T^\tau$. We assume that τ is released for the first time at time $\varphi^\tau \geq \varphi^\sigma$, i.e. *at* or *after* the first replenishment of σ. The *worst-case response time* WR^τ of the task τ is the longest possible time from its arrival to its completion. The utilization U^τ of τ is given by $\frac{C^\tau}{T^\tau}$ and the utilization U^σ of σ by $\frac{C^\sigma}{T^\sigma}$. A *necessary* schedulability condition for $\mathcal{S}$ is given by [4]

$$U^\tau \leq U^\sigma \leq 1. \tag{1}$$

2.2. An example subsystem

For illustration purposes, we use an example subsystem $S \in \mathcal{S}$ with characteristics as described in Table 1. Note

	$T = D$	C
σ	3	C^σ
τ	5	2

Table 1. Characteristics of subsystem S.

that τ is an *unbound* task [5], because its period T^τ is not an integral multiple of the period T^σ of the server. In this section, we are interested in the minimum capacity $C^\sigma_{\min}$ for the various approaches, where $C^\sigma_{\min} = \min\{C^\sigma | WR^\tau \leq D^\tau\}$. Given (1), $C^\sigma_{\min} \geq U^\sigma \cdot T^\tau = 1.2$.

2.3. Analysis for periodic resource model

Based on [11], we merely postulate the following lemma. Without further elaboration, we mention that we can postulate similar lemmas for the analysis of $\mathcal{S}$ based on the abstract server model in [6] and deferrable servers in [10].

Lemma 1 *Assuming a periodic resource model for $\mathcal{S}$, the worst-case response time WR^τ of task τ is given by*

$$WR^\tau = C^\tau + \left(\left\lceil \frac{C^\tau}{C^\sigma} \right\rceil + 1 \right) (T^\sigma - C^\sigma). \tag{2}$$

Given (2), we derive for our example S that the minimum capacity for a periodic resource model is given by $C^\sigma_{\min} = 2$. For this capacity, we find $WR^\tau = 4$.

2.4. Analysis for a periodic server

Strictly spoken, our class of subsystems $\mathcal{S}$ does not satisfy the model described in [5], because that article assumes that every set of tasks associated with a server contains at least one soft real-time task. Fortunately, a periodic server provides its resources irrespective of demand. As a result, the soft real-time tasks of a task set do not hamper the execution of the hard real-time tasks with which they share a periodic server. The analysis presented in [5] therefore equally well applies to $\mathcal{S}$ in general and S in particular. For an unbound task, we derive from [5] that WR^τ is given by

$$WR^\tau = C^\tau + \left\lceil \frac{C^\tau}{C^\sigma} \right\rceil (T^\sigma - C^\sigma). \tag{3}$$

Without further elaboration, we mention that (3) also holds for the analysis of $\mathcal{S}$ based on a deferrable server in [1]. Given (3), we derive that $C^\sigma_{\min} = 1.5$, giving rise to $WR^\tau = 5$.

2.5. Analysis for a deferrable server

The following theorem for $\mathcal{S}$ has been formulated in [4] as a corollary of a central theorem.

Theorem 1 *Consider a highest-priority deferrable server σ with period T^σ and capacity C^σ. Furthermore, assume that the server is associated with a periodic task τ with period T^τ, worst-case computation time C^τ, and deadline $D^\tau = T^\tau$, where the first release of τ takes place at or after the first replenishment of σ. The deadline D^τ is met when the respective utilizations satisfy the following inequality*

$$U^\tau \leq U^\sigma \leq 1. \tag{4}$$

Note that (4) is a necessary and sufficient (i.e. exact) schedulability condition for both the task and the server. Further note that (1) and (4) are identical, implying that a deferrable server is optimal for $\mathcal{S}$ when $D^\tau = T^\tau$.

According to Theorem 1, S is schedulable using a deferrable server with $C^\sigma_{\min} = U^\tau \cdot T^\sigma = 1.2$. The worst-case response time WR^τ of task τ is a topic of Section 3.

2.6. Overview

Table 2 gives an overview of the minimum capacities $C^\sigma_{\min}$ and minimum server utilities $U^\sigma_{\min}$ for the various approaches for S that guarantee schedulability of task τ. The table includes the worst-case response time WR^τ of τ as determined by the various approaches.

	$C^\sigma_{\min}$	$U^\sigma_{\min}$	WR^τ
periodic resource model [11]	2.0	5/6	4.0
abstract server model [6]	2.0	5/6	4.0
deferrable server [10]	2.0	5/6	4.0
periodic server [5]	1.5	1/2	5.0
deferrable server [1]	1.5	1/2	5.0
deferrable server (this paper)	1.2	2/5	4.4

Table 2. A comparison of approaches for S.

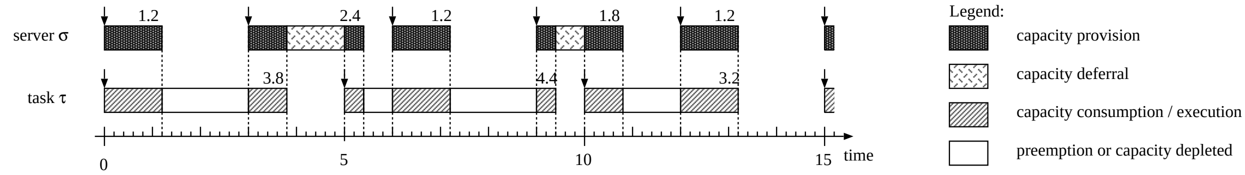

Figure 1. Timeline for S with a release of task τ at the start of the period of the deferrable server σ. The numbers at the top right corner of the boxes denote the response times of the respective releases.

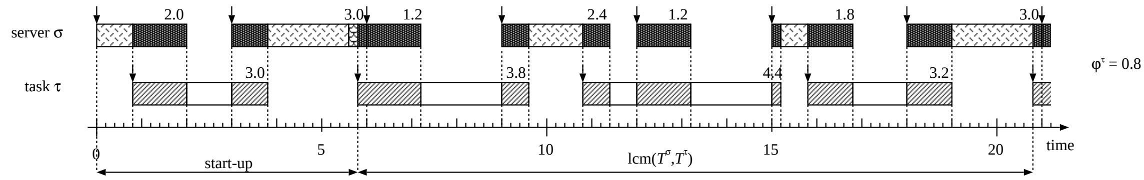

Figure 2. Timelines for S with a first release of task τ at $\varphi^\tau = 0.8$ using a deferrable server σ.

3. On response times and response jitter

We will now explore the example in more detail by considering the worst-case response time, best-case response time, and response jitter of task τ of S as a function of φ^τ for a deferrable server with a capacity $C^\sigma = 1.2$.

3.1. Worst-case response times

Because the greatest common divisor of T^τ and T^σ is equal to 1, we can restrict φ^τ to values in the interval $[0,1)$. As illustrated in Figure 3, WR^τ is equal to 4.4 and assumed for $\varphi^\tau = 0$, i.e. when τ is released at the start of the period of the deferrable server σ. Hence, a *critical instant* [7] occurs for $\varphi^\tau = 0$. Figure 1 shows a timeline with the executions of the server and the task for $\varphi^\tau = 0$ in an interval of length 15, i.e. equal to the *hyperperiod H* of the server and the task, which is equal to the least common multiple (lcm) of their periods, i.e. $H = \text{lcm}(T^\sigma, T^\tau)$. The schedule in $[0,15)$ is repeated in the intervals $[hH,(h+1)H)$, with $h \in \mathbb{N}$, i.e. the schedule is periodic with period H. From this figure, we conclude that capacity deferral of σ is a prerequisite for schedulability of S with a capacity $C^\sigma = 1.2$, and S is therefore not schedulable with a periodic server with that capacity. We observe that the worst-case response time of the task is assumed for the 2^{nd} rather than the 1^{st} job. Hence, we need to revisit the notion of *active period* [2] in the context of H-FPPS to take account of this fact.

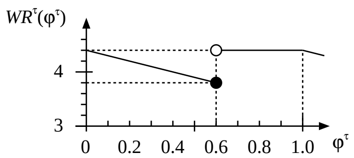

Figure 3. Worst-case response times of task τ as a function of the first release time φ^τ.

3.2. Investigating best-case response times

Unlike worst-case response times, we cannot restrict φ^τ to values in the interval $[0, \gcd(T^\tau, T^\sigma))$, but have to consider values in the interval $[0, T^\sigma)$ instead. This is caused by the fact that the response time of τ in the *start-up phase* can be smaller than the response time in the *stable phase*, as illustrated for $\varphi^\tau = 0.8$ in Figure 2. Although the relative phasing of the 1^{st} job of τ at time $t = 0.8$ compared to the 1^{st} replenishment of σ is identical to that of the 4^{th} job of τ at time $t = 15.8$ compared to the 6^{th} replenishment of σ, the response time of the 1^{st} job $R_1^\tau = 3.0$ and of the 4^{th} job $R_4^\tau = 3.2$. These differences in response times are caused by the fact that the execution of the 1^{st} job is *not* influenced by earlier jobs, whereas the execution of the 4^{th} job is.

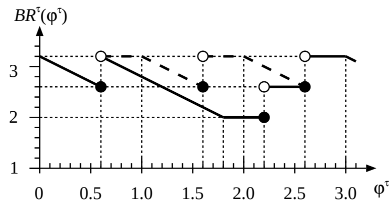

Figure 4. Best-case response time of task τ during its lifetime as a function of φ^τ. The dashed line shows the shortest response time in the stable phase.

The best-case response time $BR^\tau(\varphi^\tau)$ of τ is shown in

Figure 4. The dashed line in this figure shows for which values of φ^τ the shortest response time in the stable phase is larger than the shortest response time in the start-up phase. From this figure, we draw the following conclusions. Firstly, the best-case response time under arbitrary phasing is 2.0, which is equal to the computation time C^τ of τ. Secondly, if we only consider response times of τ in the stable phase, the shortest response time becomes 2.6. Finally, $BR^\tau(\varphi^\tau)$ is determined by the start-up phase for phasings $\varphi^\tau \in (0.6, 2.6)$.

3.3. Investigating response jitter

The response jitter of task τ as function of φ^τ is defined as

$$RJ^\tau(\varphi^\tau) = WR^\tau(\varphi^\tau) - BR^\tau(\varphi^\tau). \qquad (5)$$

The response jitter $RJ^\tau(\varphi^\tau)$ is illustrated in Figure 5. Notably, $RJ^\tau(\varphi^\tau)$ is constant in the stable phase.

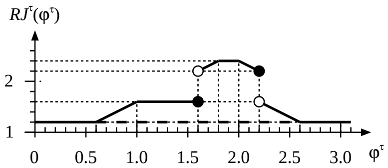

Figure 5. Response jitter of task τ during its lifetime as a function of φ^τ. The dashed line shows the response jitter in the stable phase.

4. Conclusion

Based on an exact schedulability condition, we showed in [4] that existing worst-case response time analysis of hard real-time tasks under H-FPPS can be improved when deferrable servers are used. In this paper, we investigated that identified opportunity to exploit the preservation strategy of deferrable servers. To that end, we considered a specific example subsystem with (i) a server used at highest priority and (ii) a period of its task that is not an integral multiple of the period of its server. For our example, the utilization of the server can be significantly reduced when using a deferrable server rather than a periodic server or assuming a periodic resource model. Given these initial results, application of a deferrable server can be an attractive alternative for resource-constrained systems with stringent timing requirements for a specific application when no appropriate period can be selected for its associated server. Unfortunately, improving the existing analysis turns out to be non-trivial, because the worst-case response time of a task is not necessarily assumed for the first job when released at a critical instant.

Using the same example, we briefly investigated best-case response times and response jitter. Unlike existing best-case response times of tasks under FPPS [3, 9], we did not assume infinite repetitions towards both ends of the time axis. As a result, the best-case response time of a task is determined by a start-up phase for specific phasings of the task relative to the server. When the start-up phase can be ignored, the best-case response time becomes larger and, correspondingly, the response jitter becomes smaller.

Improved response time analysis of H-FPPS using deferrable servers is a topic of future work, and we are currently re-investigating the notions of critical instant and active period in this context.

References

[1] L. Almeida and P. Peidreiras. Scheduling with temporal partitions: response-time analysis and server design. In *Proc. 4th ACM EMSOFT*, pp. 95 – 103, September 2004.

[2] R. Bril, J. Lukkien, and W. Verhaegh. Worst-case response time analysis of real-time tasks under fixed-priority scheduling with deferred preemption revisited. In *Proc. 19th ECRTS*, pp. 269–279, July 2007.

[3] R. Bril, E. Steffens, and W. Verhaegh. Best-case response times and jitter analysis of real-time tasks. *Journal of Scheduling*, 7(2):133–147, March 2004.

[4] P. Cuijpers and R. Bril. Towards budgetting in real-time calculus: deferrable servers. In *Proc. 5th International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS), LNCS-4763*, pp. 98 – 113, October 2007.

[5] R. Davis and A. Burns. Hierarchical fixed priority pre-emptive scheduling. In *Proc. 26th IEEE Real-Time Systems Symposium (RTSS)*, pp. 389–398, December 2005.

[6] G. Lipari and E. Bini. Resource partitioning among real-time applications. In *Proc. 15th ECRTS*, pp. 151–158, July 2003.

[7] C. Liu and J. Layland. Scheduling algorithms for multiprogramming in a real-time environment. *Journal of the ACM*, 20(1):46–61, January 1973.

[8] R. Rajkumar, K. Juvva, A. Molano, and S. Oikawa. Resource kernels: A resource-centric approach to real-time and multimedia systems. In *Proc. SPIE, Vol. 3310, Conference on Multimedia Computing and Networking (CMCN)*, pp. 150–164, January 1998.

[9] O. Redell and M. Sanfridson. Exact best-case response time analysis of fixed priority scheduled tasks. In *Proc. 14th ECRTS*, pp. 165–172, June 2002.

[10] S. Saewong, R. Rajkumar, J. Lehoczky, and M. Klein. Analysis of hierarchical fixed-priority scheduling. In *Proc. 14th ECRTS*, pp. 152–160, June 2002.

[11] I. Shin and I. Lee. Periodic resource model for compositional real-time guarantees. In *Proc. 24th IEEE Real-Time Systems Symposium (RTSS)*, pp. 2–13, December 2003.

[12] J. Strosnider, J. Lehoczky, and L. Sha. The deferrable server algorithm for enhanced aperiodic responsiveness in hard real-time environments. *IEEE Transactions on Computers*, 44(1):73–91, January 1995.