Towards Periodic Budgeting in Real-Time Calculus

P.J.L. Cuijpers and R.J. Bril

Abstract

In this report, we extend the real-time calculus model with a periodic budgeting mechanism known as deferrable service. Based on this extended model, we prove a schedulability theorem for a real-time task under a deferrable server. Using this theorem, we show by means of an example that existing schedulability analysis for real-time tasks under a deferrable server is pessimistic rather than exact.

1 Introduction

One of the main scheduling approaches in real-time computing systems is given by fixed-priority preemptive scheduling (FPPS) [5, 6]. The approach is founded on a fixed-priority scheduling theory, supported by a suite of open software standards, commercially available schedulability analysis tools and real-time operating systems, and adopted by leading companies and institutions world-wide.

Scheduling a set of real-time tasks sharing a resource gives rise to the so-called temporal interference problem, i.e. a malfunctioning task may cause other tasks to fail to meet their time constraints. An often used solution for this problem is to introduce resource budgets for tasks, which provide temporal protection between tasks by guaranteeing a minimal amount of resources [9]. Those budgets are often implemented using so-called servers that dispatch the available resources to the tasks that are appointed to them.

In general, a server for a shared processing resource, such as a CPU, is characterized by a capacity and a replenishment period [1]. The capacity is the maximum amount of resources (i.e. the maximum amount of processing time) that a server can provide to its associated tasks during its replenishment period. The replenishment period is the minimum time between replenishments of the capacity. Servers typically differ with respect to the amount and moment in time of the replenishments and to the preservation of the remaining capacity when the task is not ready to use it.

In this report, we consider so-called deferrable servers [11]. Deferrable servers are replenished periodically, at fixed intervals of time, and have the following preservation policy. When a deferrable server has access to the shared resource, its capacity is provided if one of the tasks is ready to use it. If the tasks are not ready to use it, the deferrable server suspends its access to the resource, preserving its remaining capacity. Capacity can be preserved until the end of the replenishment period. At the end of the server’s period any remaining capacity is lost.

One of our main contributions, is a formal model of a deferrable server with a single task using the real-time calculus of [12]. Real-time calculus is a branch of network calculus [4], which
uses max-plus algebra for the algebraic analysis of real-time systems. Our specific interest in formalizing the behavior of deferrable servers using this calculus is due to a suspicion that the worst-case response time analysis of tasks with this type of associated server as given in [2] is pessimistic, rather than exact. Using our real-time calculus model of the server, we derive a schedulability theorem for deferrable servers with a single task. Application of our schedulability theorem for a deferrable server with highest priority leads to an example showing that our suspicion is indeed true, and that the analysis in [2] is indeed pessimistic, rather than exact.

The remainder of this report is structured as follows. First, in Section 2, we recapitulate the real-time calculus theory of [12]. Next, in Section 3, we present our extension of this theory with a model of the deferrable server. Our schedulability theorem is the topic of Section 4, and the consequences of our theorem are discussed in Section 5. Section 6 summarizes the main contributions of the report and suggests directions for future research.

2 Recapitulation of real-time calculus

In this section, we recall part of the real-time calculus theory of [12]. The starting point of this calculus, is to consider cumulative requests streams \( R(t) : \mathbb{R} \rightarrow \mathbb{R} \) (from time to amount of work requested) and cumulative resource streams \( C(t) : \mathbb{R} \rightarrow \mathbb{R} \) (from time to available resources) in a system. The request stream models the total amount of requested tasks that have entered the system at a certain time, while the resource stream models the total amount of processing power that has been offered to a server. The total amount of tasks that have been processed is modeled by a cumulative stream \( R'(t) \), while the total amount of resources that remains unused is processed by a cumulative stream \( C'(t) \). For convenience we choose \( C(t) = R(t) = C'(t) = R'(t) = 0 \) whenever \( t < 0 \), reflecting that the system is turned on at \( t = 0 \).

![Figure 1: Basic processing in Real-time Calculus](image)

A single server, without budget and requiring \( K \) resources per requested task, can be depicted as in figure 1. Such a server uses resources to process tasks, so it is obvious that the amount of tasks that is processed by such a server at a certain time \( t \) can never be more than the amount of tasks that was processed already at a time \( u \leq t \), plus the amount of resources that was offered between \( u \) and \( t \), divided by \( K \). If the task buffer is empty at some point, it will even be less. Furthermore, the amount of processed tasks can never be more than the amount of requested tasks. This is captured in the following formula (in which we use \( x \wedge y \) to denote the minimum of \( x \) and \( y \)).

\[
R'(t) \leq \left( R'(u) + \frac{C(t) - C(u)}{K} \right) \wedge R(t).
\]
Using \( R'(u) \leq R(u) \) we easily derive:

\[
R'(t) \leq \left( R(u) + \frac{C(t) - C(u)}{K} \right) \wedge R(t).
\]

And, since this holds for all \( u \leq t \), it also holds for the infimum over \( u \):

\[
R'(t) \leq \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\} \wedge R(t).
\]

Using the special case where \( u = t \), we simplify this to

\[
R'(t) \leq \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}.
\]

If we furthermore assume that tasks may be buffered until resources become available for it, and that a server is eager in the sense that it processes each task as soon as resources are available for it, we can derive a lower bound on \( R'(t) \) as well. We define \( t_0 \) as the latest point before \( t \) at which the resource buffer was empty, assuming \( R(0) = R'(0) \) for convenience.

\[
t_0 \triangleq \sup \{ \tau \leq t \mid R(\tau) = R'(\tau) \}
\]

Between \( t_0 \) and \( t \), the task buffer is always non-empty, which means that all resources that arrive are used for processing tasks. So we find the equality:

\[
R'(t) = \left( R'(t_0) + \frac{C(t) - C(t_0)}{K} \right) \wedge R(t).
\]

For a right-continuous resource stream \( R(t) \) we then find \( R(t_0) = R'(t_0) \) (using the definition of \( t_0 \)) and thus:

\[
R'(t) = \left( R(t_0) + \frac{C(t) - C(t_0)}{K} \right) \wedge R(t),
\]

\[
\geq \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\} \wedge R(t),
\]

\[
= \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}.
\]

In conclusion, \( R'(t) \) is exactly determined by

\[
R'(t) = \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}.
\]

According to [4] the same formula holds if \( R(t) \) is left-continuous, but the proof is more complicated. We expect the same to be true for the theory in this report, but we have not verified this claim and will assume right-continuity throughout this report, for simplicity.

The amount of resources that is left unused can be found easily by subtracting what is used from what is delivered.

\[
C'(t) = C(t) - K \cdot R'(t),
\]

\[
= C(t) - \inf_{u \leq t} \left\{ K \cdot R(u) + C(t) - C(u) \right\},
\]

\[
= \sup_{u \leq t} \left\{ C(u) - K \cdot R(u) \right\}.
\]

In figures 2 and 3 we have depicted an example of a task \( R(t) = 3 \cdot \lceil \frac{t}{3} \rceil \wedge 0 \) with \( K = 1 \), serviced by a resource \( C(t) = 2 \cdot t \wedge 0 \). Note, that by definition \( C'(t) \) is flat whenever \( R'(t) \) rises, and vice versa.
Figure 2: Requested and serviced tasks in RTC

Figure 3: Delivered and unused resources in RTC
3 Deferrable server model

In this section, we model the deferrable server of [8] in a similar way as the ordinary real-time calculus server. We still assume that tasks are processed eagerly by the server, and that tasks are buffered while waiting for resources to arrive. Therefore, the previously derived upper bounds on $R'(t)$ are still valid.

$$R'(t) \leq R(t)$$

$$R'(t) \leq R'(u) + \frac{C(t) - C(u)}{K}, \text{ for all } u \leq t$$

But, additionally, a deferrable server periodically limits the resources it provides to a maximum capacity $Q$. Replenishment of the capacity takes place at the start of each period $T$, which at a time $t$ is determined (right-continuously) by $T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil$. The output $R'(t)$ of a deferrable server cannot be greater than the output $R' \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right)$ at the start of the period plus the maximum amount of tasks $\frac{Q}{K}$ that can be processed within the processing capacity. So, we add the following inequality to capture this behavior.

$$R'(t) \leq R' \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right) + \frac{Q}{K}.$$ 

These three equations hold if and only if:

$$R'(t) \leq \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}$$

$$\land \ R' \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right) + \frac{Q}{K}$$

$$\land \ R' \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right) + \frac{C(t) - C \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right)}{K}.$$ 

Furthermore, for $t_0 \equiv \sup\{ \tau \leq t \mid R(\tau) = R'(\tau) \} \geq 0$ we find

$$R'(t) = R(t_0) + \frac{C(t) - C(t_0)}{K}$$

$$\land \ R' \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right) + \frac{Q}{K}$$

$$\land \ R' \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right) + \frac{C(t) - C \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right)}{K}.$$ 

What results is a recursive specification for the output of the deferrable server

$$R'(t) = \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}$$

$$\land \ R' \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right) + \frac{Q}{K}$$

$$\land \ R' \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right) + \frac{C(t) - C \left( T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil \right)}{K}.$$
To prove that this recursive specification has a unique solution for \( R'(t) \), we rewrite it to

\[
R'(t + T) = \inf_{u \leq t + T} \left\{ R(u) + \frac{C(t + T) - C(u)}{K} \right\}
\]

\[
\wedge \quad R' \left( T \cdot \left[ \frac{t}{T} \right] \right) + \frac{Q}{K}
\]

\[
\wedge \quad R' \left( T \cdot \left[ \frac{t}{T} \right] \right) + \frac{C(t + T) - C \left( T \cdot \left[ \frac{t}{T} \right] \right)}{K}.
\]

From this representation it is clear that, if the solution of the recursive specification is unique upto a point \( t \), then it is unique upto \( t + T \). Furthermore, we find uniqueness for \( t < 0 \) where we have \( R'(t) = 0 \). So, the solution is unique upto 0, and with induction upto \( n \cdot T \) for any \( n \in \mathbb{N} \). We may conclude that \( R'(t) \) is well defined, and may even solve the recursive specification to find:

\[
R'(t) = \inf_{n \in \mathbb{N}, u \leq t \wedge n \cdot T \left[ \frac{t}{T} - n \right]} \left\{ R(u) + \frac{C(t \wedge n \cdot T \left[ \frac{t}{T} - n \right]) - C(u)}{K} \right\}
\]

\[
+ \sum_{m=0}^{n-1} \frac{Q}{K} \wedge \frac{C(t \wedge n \cdot T \left[ \frac{t}{T} - m \right]) - C(t \wedge n \cdot T \left[ \frac{t}{T} - m - 1 \right])}{K}
\}

As before, we find \( C'(t) \) by subtracting what is used from what is delivered. We use \( x \vee y \) to denote the maximum of \( x \) and \( y \).

\[
C'(t) = C(t) - K \cdot R'(t)
\]

\[
= C(t) - K \cdot \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}
\]

\[
\vee \quad C(t) - K \cdot R' \left( T \cdot \left[ \frac{t}{T} - 1 \right] \right) - Q
\]

\[
\vee \quad C(t) - K \cdot R' \left( T \cdot \left[ \frac{t}{T} - 1 \right] \right) - C(t) + C \left( T \cdot \left[ \frac{t}{T} - 1 \right] \right)
\]

\[
= \sup_{u \leq t} \{ C(u) - K \cdot R(u) \}
\]

\[
\vee \quad C' \left( T \cdot \left[ \frac{t}{T} - 1 \right] \right) + C(t) - C \left( T \cdot \left[ \frac{t}{T} - 1 \right] \right) - Q
\]

\[
\vee \quad C' \left( T \cdot \left[ \frac{t}{T} - 1 \right] \right)
\]

\[
= \sup_{n \in \mathbb{N}, u \leq t \wedge n \cdot T \left[ \frac{t}{T} - n \right]} \left\{ C(u) - K \cdot R(u) \right\}
\]

\[
+ \sum_{m=0}^{n-1} \left( C \left( t \wedge n \cdot T \left[ \frac{t}{T} - m \right] \right) - C \left( T \cdot \left[ \frac{t}{T} - m - 1 \right] \right) - Q \right) \vee 0
\]

In figures 4 and 5 we have depicted an example of a task \( R(t) = 3 \cdot \left[ \frac{t}{3} \right] \vee 0 \) with \( K = 1 \), serviced by a resource \( C(t) = 2 \cdot t \vee 0 \), on a deferrable server with \( Q = 2 \) and \( T = 2 \). Note, that as before, \( C'(t) \) is flat whenever \( R'(t) \) rises, and vice versa, but the pattern is different from the normal RTC server.
Figure 4: Requested and serviced tasks in a deferrable server

Figure 5: Delivered and unused resources in a deferrable server
4 Schedulability

The delay of a task entering at time $t$ is the time between its request and its completion. If we assume that tasks are completed in the order in which they arrive, the delay is the earliest time $\tau$ at which $R'(t + \tau) \geq R(t)$. The maximum delay $\Delta$ is then defined by:

$$
\Delta \triangleq \sup_t \inf \{ \tau \mid R'(t) \geq R(t - \tau) \}.
$$

Based on this definition, and the model of a deferrable server found in the previous section, we will now prove the following schedulability theorem. This theorem roughly states that task with a deadline greater than the sum of the minimum interarrival time of the task and the maximum delay between arrival of resources (due to other servers in the network) is schedulable provided that the tasks utilization is smaller than the utilization of the server and smaller than the utilization of the arriving resources.

**Theorem 1 (Schedulability of a deferrable server)** Consider a deferrable server with period $T$ and capacity $Q$. Assume that there is an upper bound $S$ on the arrival time of tasks $R(t)$ such that for all $s \in \mathbb{R}$:

$$
\frac{R(s + S) - R(s)}{S} \leq \frac{Q}{K \cdot T}.
$$

Furthermore, assume that there is a lower bound $U \leq T$ on the arrival times of resources, such that for all $u \in \mathbb{R}$:

$$
\frac{C(u + T) - C(u)}{U} \geq \frac{Q}{K} \text{ and } \frac{C(u + U) - C(u)}{U} \geq \frac{R(s + S) - R(s)}{S}.
$$

Then, all relative deadlines of at least $S + 2 \cdot U$ are met, i.e.

$$
\Delta \triangleq \sup_t \inf \{ \tau \mid R'(t) \geq R(t - \tau) \} \leq S + 2 \cdot U.
$$

**Proof** To prove $\sup_t \inf \{ \tau \mid R'(t) \geq R(t - \tau) \} \leq S + 2 \cdot U$, it is necessary and sufficient to prove that $R'(t) \geq R(t - S - 2 \cdot U)$ for all $t$. For this, we start from our previously obtained solution for $R'(t)$.

$$
R'(t) = \inf_{n \in \mathbb{N}} \inf_{u \leq t \leq T \cdot \lceil \frac{t}{T} \rceil} \left\{ R(u) + \frac{C(t \wedge T \cdot \lfloor \frac{t}{T} \rfloor - n)}{K} - C(u) \right\}
\geq \sum_{m=0}^{n-1} \frac{Q}{K} + \frac{C(t \wedge T \cdot \lfloor \frac{t}{T} \rfloor - m)}{K} - C(T \wedge T \cdot \lfloor \frac{t}{T} \rfloor - m - 1) \right\}
$$

We split off the special cases where $n = 0$ and $m = 0$, and find:

$$
= \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}
\wedge \inf_{n \geq 1} \inf_{u \leq T \cdot \lceil \frac{t}{T} \rceil} \left\{ R(u) + \frac{C(T \cdot \lfloor \frac{t}{T} \rfloor - n)}{K} - C(u) \right\}
\geq \frac{Q}{K} + \frac{C(T) - C(T \cdot \lfloor \frac{t}{T} \rfloor - 1)}{K}
\geq \sum_{m=1}^{n-1} \frac{Q}{K} + \frac{C(T \cdot \lfloor \frac{t}{T} \rfloor - m) - C(T \cdot \lfloor \frac{t}{T} \rfloor - m - 1)}{K} \right\}
$$
Again, we find two cases, depending on whether $Q$ or $C(t) - C(T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil)$ is larger.

$$= \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}$$

$$\wedge \inf_{n \geq 1} \inf_{u \leq T \left\lceil \frac{t}{T} - n \right\rceil} \left\{ R(u) + \frac{C(T \cdot \left\lceil \frac{t}{T} - n \right\rceil) - C(u)}{K} + n \cdot \frac{Q}{K} \right\}$$

$$+ \sum_{m=1}^{n-1} \frac{Q}{K} \wedge \left\{ R(u) + \frac{C(t) - C(T \cdot \left\lceil \frac{t}{T} - 1 \right\rceil) + C(T \cdot \left\lceil \frac{t}{T} - n \right\rceil) - C(u)}{K} + (n-1) \cdot \frac{Q}{K} \right\}$$

Take $u = T \cdot \left\lceil \frac{t}{T} - m - 1 \right\rceil$ in the assumption on resource arrivals to find $C(T \cdot \left\lceil \frac{t}{T} - m \right\rceil) - C(T \cdot \left\lceil \frac{t}{T} - m - 1 \right\rceil) \geq Q$.

$$= \inf_{u \leq t} \left\{ R(u) + \frac{C(t) - C(u)}{K} \right\}$$

$$\wedge \inf_{n \geq 1} \inf_{u \leq T \left\lceil \frac{t}{T} - n \right\rceil} \left\{ R(u) + \frac{C(T \cdot \left\lceil \frac{t}{T} - n \right\rceil) - C(u)}{K} + n \cdot \frac{Q}{K} \right\}$$

Then, we truncate the argument of $C(.)$ to a multiple of $U$ with a convenient remainder. Using monotonicity of $C(t)$ we find:

$$\geq \inf_{u \leq t} \left\{ R(u) + \frac{C \left( \left\lceil \frac{t}{B} \right\rceil \cdot U + u \right) - C(u)}{K} \right\}$$

$$\wedge \inf_{n \geq 1} \inf_{u \leq T \left\lceil \frac{t}{T} - n \right\rceil} \left\{ R(u) + \frac{C \left( \left\lceil \frac{t}{T} - n \right\rceil \cdot U + u \right) - C(u)}{K} + n \cdot \frac{Q}{K} \right\}$$

$$+ \frac{C \left( \left\lceil \frac{t}{T} - n \right\rceil \cdot U + u \right) - C(u)}{K} + (n-1) \cdot \frac{Q}{K}$$

Now, we define $X$ as a lower bound on the utilization of $R(t)$, and find:

$$\inf_{u} \left\{ \frac{C(u + U) - C(u)}{K \cdot U} \right\} \geq \sup_{s} \left\{ \frac{R(s + S) - R(s)}{S} \right\} \triangleq X.$$
Using $X$, we eliminate $C(t)$ and $Q$.

\[
\begin{align*}
&\geq \inf_{u \leq t} \left\{ R(u) + \left[ \frac{t-u}{U} \right] \cdot U \cdot X \right\} \\
&\wedge \inf_{n \geq 1} \inf_{u \leq T} \left[ \frac{t}{T} - n \right] \left\{ R(u) + \left[ \frac{T \cdot \left[ \frac{t}{T} - n \right] - u}{U} \right] \cdot U \cdot X + n \cdot T \cdot X \right\} \\
&\wedge \inf_{n \geq 1} \inf_{u \leq T} \left[ \frac{t}{T} - n \right] \left\{ R(u) + \left[ \frac{t - T \cdot \left[ \frac{t}{T} - 1 \right]}{U} \right] \cdot U \cdot X + \left[ \frac{T \cdot \left[ \frac{t}{T} - n \right] - u}{U} \right] \cdot U \cdot X + (n - 1) \cdot T \cdot X \right\}
\end{align*}
\]

Observe that for all $x$ we have $x \cdot X \geq \left[ \frac{x}{S} \right] \cdot S \cdot X$. And since $S \cdot X$ serves as an upper bound on $R(s + S) - R(s)$, this gives us

\[
\begin{align*}
&\geq \inf_{u \leq t} \left\{ R(u) + \left[ \frac{t-u}{U} \right] \cdot \left[ \frac{U}{S} \right] \cdot S \right\} \\
&\wedge \inf_{n \geq 1} \inf_{u \leq T} \left[ \frac{t}{T} - n \right] \left\{ R(u) + \left[ \frac{T \cdot \left[ \frac{t}{T} - n \right] - u}{U} \right] \cdot \left[ \frac{U}{S} + n \cdot \frac{T}{S} \right] \cdot S \right\} \\
&\wedge \inf_{n \geq 1} \inf_{u \leq T} \left[ \frac{t}{T} - n \right] \left\{ R(u) + \left[ \frac{t - T \cdot \left[ \frac{t}{T} - 1 \right]}{U} \right] \cdot \left[ \frac{U}{S} + T \cdot \left[ \frac{t}{T} - n \right] - u \right] \cdot \left[ \frac{U}{S} + (n - 1) \cdot \frac{T}{S} \right] \cdot S \right\}
\end{align*}
\]

And using monotonicity of $R(t)$, together with the observation that $\lfloor x \rfloor \cdot y \geq x \cdot y - y$ we find:

\[
\begin{align*}
&\geq \inf_{u \leq t} \left\{ R(u) + \left[ \frac{t-u}{U} \right] \cdot U - S \right\} \\
&\wedge \inf_{n \geq 1} \inf_{u \leq T} \left[ \frac{t}{T} - n \right] \left\{ R(u) + \left[ \frac{T \cdot \left[ \frac{t}{T} - n \right] - u}{U} \right] \cdot U + n \cdot T - S \right\} \\
&\wedge \inf_{n \geq 1} \inf_{u \leq T} \left[ \frac{t}{T} - n \right] \left\{ R(u) + \left[ \frac{t - T \cdot \left[ \frac{t}{T} - 1 \right]}{U} \right] \cdot U + \left[ \frac{T \cdot \left[ \frac{t}{T} - n \right] - u}{U} \right] \cdot U + (n - 1) \cdot T - S \right\} \\
&\geq \inf_{u \leq t} \left\{ R(u + t - u - U - S) \right\} \\
&\wedge \inf_{n \geq 1} \inf_{u \leq T} \left[ \frac{t}{T} - n \right] \left\{ R(u + T \cdot \left[ \frac{t}{T} - n \right] - u - U + n \cdot T - S) \right\} \\
&\wedge \inf_{n \geq 1} \inf_{u \leq T} \left[ \frac{t}{T} - n \right] \left\{ R(u + t - T \cdot \left[ \frac{t}{T} - 1 \right] - U + T \cdot \left[ \frac{t}{T} - n \right] - u - U + (n - 1) \cdot T - S) \right\} \\
&= R(t - U - S) \\
&\wedge R(T \cdot \left[ \frac{t}{T} \right] - U - S) \\
&\wedge R(t - 2 \cdot U - S) \\
&= R(t - 2 \cdot U - S)
\end{align*}
\]

Which concludes our proof.

In the special but in practice not uncommon case (see [1]) where the deferrable server has highest priority, the resource stream $C(t)$ can be considered to be linear, i.e. $C(t) = C \cdot t$. As a corollary, we then find that deadlines of at least $S$ are met.
Corollary 1 Consider a highest-priority deferrable server with period $T$ and capacity $Q$. Furthermore, assume that we have an upper bound $S$ on the arrival time of tasks $R(t)$ and a linear resource stream $C(t) = C \cdot t$, such that the respective utilizations satisfy the following inequalities for all $s \in \mathbb{R}$:

\[
\frac{C}{K} \geq \frac{Q}{K \cdot T} \geq \frac{R(s + S) - R(s)}{S}.
\]

Then, all relative deadlines of at least $S$ are met, i.e.

\[
\Delta \triangleq \sup_{t} \inf_{R'} \{ \tau \mid R'(t) \geq R(t - \tau) \} \leq S.
\]

Proof By assumption, we have for all $s \in \mathbb{R}$ that $\frac{R(s+S) - R(s)}{S} \leq \frac{C}{K}$. From linearity, it follows that for all $U > 0$ and all $u \in \mathbb{R}$ we have $\frac{C(u+U) - C(u)}{U} = C$, so $\frac{R(s+S) - R(s)}{S} \leq \frac{C(u+U) - C(u)}{K \cdot U}$ and $C(u + T) - C(u) \geq Q$. Using our main theorem, we find for all $U > 0$, that relative deadlines greater than $S + 2 \cdot U$ are met, and hence $\Delta \triangleq \sup_{t} \inf_{R'} \{ \tau \mid R'(t) \geq R(t - \tau) \} \leq \inf_{U > 0} \{ S + 2 \cdot U \} = S$. Which concludes our proof.

5 Discussion

Using Corollary 1, we will show that the schedulability analysis in [2] for real-time tasks under hierarchical fixed-priority preemptive scheduling and a deferrable server is pessimistic rather than exact. To this end, we consider the special case of a system consisting of a single server and a single task or, equivalently, a system in which a single task is running on a highest priority server. The deadline of the task is chosen equal to its period.

The remainder of this section has the following structure. First, we briefly relate our terminology with the terminology used in [2], and subsequently transcribe and refine Corollary 1 for our system. Next, we recapitulate worst-case response time analysis given in [2] by presenting a dedicated equation for our special case. According to [2], this analysis is exact for Deferrable Servers, i.e. provides a necessary and sufficient schedulability condition for the task. However, our corollary shows that the analysis is pessimistic for Deferrable Servers, which we illustrate by means of an example for our special case. We conclude this section with an implication of our finding.

In [2], a periodic task $\tau$ is characterized by a period (or inter-arrival time) $T^\tau$, a worst-case computation time $C^\tau$, and a relative deadline $D^\tau$. We assume that the task’s period and deadline are equal, i.e. $T^\tau = D^\tau$. A server $\sigma$ is characterized by a replenishment period $T^\sigma$ and a capacity $C^\sigma$. Based on these notions, the utilization $U^\tau$ of the task is given by $\frac{C^\tau}{T^\tau}$ and the utilization $U^\sigma$ of the server by $\frac{C^\sigma}{T^\sigma}$.

The task $\tau$ can be either bound or unbound. The task $\tau$ is bound if it has a period that is an exact multiple of the server’s period and an arrival time that coincides with the replenishment of the server’s capacity. Otherwise $\tau$ is unbound. We assume an unbound task. Without loss of generality, we assume that the server $\sigma$ is replenished for the first time at time $\varphi^\sigma = 0$. Moreover, we assume that $\tau$ is released for the first time at time $\varphi^\tau \geq 0$, i.e. at or after the first replenishment of $\sigma$.

Note, that these parameters of the task and the server have the same dimension, namely time. So we choose $K = 1$, dimensionless, in our real-time calculus model, and we choose $C(t) = t$.
to reflect the incoming resource stream. With this terminology in place, we can write

\[
R(t) = C^\tau \cdot \left\lceil \frac{t - \varphi^\tau}{T^\tau} \right\rceil \\
C(t) = t \\
Q = C^\sigma \\
S = T^\tau \\
T = T^\sigma
\]

For our system, we can now transliterate and refine Corollary 1.

**Corollary 2** Consider a highest-priority deferrable server \( \sigma \) with period \( T^\sigma \) and capacity \( C^\sigma \). Furthermore, assume that the server is associated with a periodic task \( \tau \) with period \( T^\tau \) and worst-case computation time \( C^\tau \), where the first release of \( \tau \) takes place at or after the first replenishment of \( \sigma \). When the respective utilizations satisfy the following inequality

\[
U^\tau \leq U^\sigma \leq 1, \tag{1}
\]

the deadline \( D^\tau = T^\tau \) of \( \tau \) is met.

Note that our transliterated corollary holds for both a bound task and an unbound task. Furthermore, note that (1) is a necessary and sufficient (i.e. exact) schedulability condition for both the task and the server. Finally, note that

\[
U^T \leq U^\sigma \leq 1, \tag{2}
\]

is a necessary schedulability condition for a set \( T \) of independent hard real-time tasks with utilization \( U^T \) with an associated server \( \sigma \) with utilization \( U^\sigma \).

We will now derive a schedulability condition for our system from [2] starting from an equation to determine the task’s worst-case response time. The task’s worst-case response time \( WR^\tau \) is the longest possible time from its arrival to its completion. Similarly, the server’s worst-case response time \( WR^\sigma \) is the longest possible time from the server being replenished to its capacity being exhausted, given the task is ready to use all of its capacity. The task is said to be schedulable if (and only if)

\[
WR^\tau \leq D^\tau.
\]

Similarly, the server is schedulable if (and only if) \( WR^\sigma \leq T^\sigma \).

For our system, the server is schedulable when \( C^\sigma \leq T^\sigma \). Based on [2], we derive for our system that \( WR^\tau \) is given by

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

which leads to the following condition for schedulability of a task with a deadline \( D^\tau \) equal to its period \( T^\tau \)

\[
C^\tau + \left\lceil \frac{C^\tau}{C^\sigma} \right\rceil (T^\sigma - C^\sigma) \leq T^\tau. \tag{4}
\]

Now, as an example, we fix \( C^\tau \) and \( T^\tau \) and plot the minimum utilization \( U^\sigma_{\text{min}} \) of the server as a function of \( T^\sigma \), i.e. we plot

\[
U^\sigma_{\text{min}}(T^\sigma) = \min \left\{ \frac{C^\sigma}{T^\sigma} \mid T^\sigma \geq C^\tau + \left\lceil \frac{C^\tau}{C^\sigma} \right\rceil (T^\sigma - C^\sigma), C^\sigma \geq 0 \right\}.
\]
The result for $C^\tau = 2$ and $T^\tau = 5$, obtained using Mathematica, is depicted in figure 6. The horizontal line in this figure, shows the utilization $U^\tau = \frac{C^\tau}{T^\tau}$ of the task.

The figure illustrates that according to [2] only for values of $T^\sigma$ equal to an integral fraction of $T^\tau$, i.e. $T^\sigma = \frac{T^\tau}{n}$ for $n \in \mathbb{N}^+$, the minimum server utilization $U^\sigma_{\text{min}}$ needed for schedulability is equal to the task utilization $U^\tau$. For other values of $T^\sigma$, $U^\sigma_{\text{min}}$ is higher than $U^\tau$. However, according to our theorem, the server utilization $U^\sigma$ may be chosen equal to the task utilization $U^\tau$ when using a deferrable server, irrespective of $T^\sigma$. From this example, we conclude that the schedulability condition expressed by (4) is sufficient but not necessary for an unbound task with an associated Deferrable Server. As a result, equation (3) is pessimistic, and the worst-case response time analysis presented in [2] for unbound tasks with associated deferrable servers is therefore pessimistic and not exact.

According to [2], their worst-case response time analysis is exact for both deferrable servers and periodic servers, where the latter are another kind of server with a subtly different budgeting strategy. Based on that result, they claim that periodic servers dominate deferrable servers with respect to schedulability of tasks, i.e. “there are no systems . . . that can be scheduled using a set of deferrable servers that cannot also be scheduled using an equivalent set of periodic servers”. However, because the analysis for deferrable servers is pessimistic, that claim looses part of its underpinning, and therefore need no longer hold.

6 Conclusion and future research

We presented a formal model of a deferrable server associated with a single task, using the real-time calculus of [12]. Using this model we derived a schedulability theorem stating that a task with a deadline greater than the sum of the minimum interarrival time of the task and the maximum delay between arrival of resources (due to other servers in the network) is schedulable provided that the tasks utilization is smaller than the utilization of the server and smaller than the utilization of the arriving resources. Application of this theorem to the special case of a highest-priority server has lead to a counterexample for the claim in [2] that
the worst-case response time analysis for tasks under deferrable servers given in that paper is exact. Exact worst-case response time analysis therefore still requires further research.

In this report, we used real-time calculus as an aid to proof our schedulability theorem for deferrable servers. The treatment of deferrable servers given in this report is far from complete, however. To fully embed our model in the real-time calculus framework, we must at least show how so-called service curves [4] are transformed by a deferrable server. Service curves define upper and lower bounds on the input streams rather than giving the streams exactly. The original real-time calculus model also allows given service curves on the input to be transformed to service curves on the output. Thus, more general claims about, for example, hard bounds on the latency and throughput of a system can be made. Initial investigations suggest that a service curve transformation is possible for deferrable servers, but the recursion over $n$ must be controlled in some way to make the transformation tractable.

Finally, the treatment of budgeting servers in general does not stop with the treatment of deferrable servers. First of all, there is a great variety of budgeting servers already in use in practice and theory. And secondly, the use of budgeting servers has lead to the introduction of a hierarchical approach to scheduling. If multiple tasks are run on a single server, one may first divide the available resources over the servers, and in a second stage schedule the tasks that run on each server independently of what runs on the other servers. Recent theory about the schedulability of tasks under different kinds of budgeting servers was presented in [2, 3, 7, 10]. Our real-time calculus models are not yet ready to formalize this hierarchical approach, because we have assumed that each server processes a single task.

References


