The Sixteenth IEEE International Conference on Embedded and Real-Time Computing Systems and Applications

Generalizing Response-Time Analysis

Victor Pollex, Steffen Kollmann and Frank Slomka
Institute of Embedded Systems / Real-Time Systems
Ulm University
{firstname}.{lastname}@uni-ulm.de

***Abstract*—In real-time theory, basically two approaches for the computation of response-times exist. One of them is the busy window method, the other is the real-time calculus, an extension of the network calculus. While both can be used to compute the bounds of response-times, they have different properties that make them suitable for different system architectures. The busy window approach on the one hand is able to obtain tight bounds for scheduling policies like round-robin. It is also capable of considering offsets, therefore delivering better results in the relevant cases. Hierarchical scheduling on the other hand can be better accounted for by the real-time calculus, where this is an inherent feature of the underlying concept. The approach we present in this paper takes the theory of hierarchy from the real-time calculus and uses it to generalize the response-time analysis. This is implemented as an extension of the busy window method, which enables it to analyze scheduling hierarchies of an arbitrary depth.**

I. Introduction

The complexity of embedded systems has grown rapidly in the last years e. g. in the automotive industry where many functions and applications are distributed across several electronic control units (ECUs). This is among other things due to the layout of the sensors and actuators, for instance in electro-hydraulic brake systems or driver assistance systems.

The ECUs communicate over several busses like FlexRay [1]. This bus uses basically a time division multiple access (TDMA) policy for its arbitration. A cycle consists of various consecutive segments which can include a static segment and a dynamic segment. Both segments are divided into slots. In the static segment these slots are of fixed size, whereas they can dynamically expand in the dynamic segment, up to its complete length. Spontaneous or sporadic messages are usually sent over the dynamic segment which basically follows a fixed-priority non-preemptive policy as its arbitration. Therefore the arbitration of FlexRay can be treated as a hierarchical scheduling policy.

The automotive domain is only one field where hierarchical scheduling is encountered. Other examples are real-time systems using one of the various server models, like the periodic server or the deferrable server [2]. These are used to handle both strict periodic and spontaneous tasks. More examples are described in [3, p. 3].

To verify that a system meets the real-time constraints, a schedulability analysis is performed. Many approaches used for the analysis are based on the the busy window introduced by Lehoczky [4]. To the best of our knowledge none of those approaches use a general method to handle hierarchical scheduling. The main challenge is coping with all the possible combinations of scheduling policies.

A different approach is the real-time calculus, which is based on the network calculus. By considering each task independently, the real-time calculus is capable of handling hierarchical scheduling. However, the schedulability analysis is not as good as the approaches based on the busy window when e.g. a round robin scheduling policy or offset relations are involved. It is therefore desirable that the approaches based on the busy window are capable of handling hierarchical scheduling with a general method. We will achieve this by deriving from the real-time calculus a more general form of the response-time analysis that was introduced by Lehoczky [4].

The paper is organized as follows: in section II an overview of the related work is presented. The computational model and the assumptions are introduced in section III. In section IV the generalized response-time analysis is developed, followed by various scheduling policies in section V which can be used in the analysis. After an example in section VI, the work closes with a conclusion.

II. Related Work

The methods for exact real-time analysis of distributed systems can be mainly divided into two groups. One is the real-time calculus [5] based on the network calculus [6], and the other one is the holistic analysis by Tindell and Clark [7] based on the response-time analysis introduced by Lehoczky [4]. Since then, many improvements have been achieved.

The SymTA/S approach [8] based on the response-time analysis is one of the established methodologies. Especially, offsets between tasks [9] and the round-robin policy [10] can be handled very well by this methodology.

The consequent advancement of the response-time analysis has lead to an expressive analysis which can handle many different system architectures. Special approaches were developed to model hierarchical scheduling: Seawong et al. have shown in [11] how hierarchical preemptive scheduling can be

This work is supported in part by the Carl Zeiss Foundation.

1533-2306/10 $26.00 © 2010 IEEE
DOI 10.1109/RTCSA.2010.36

modeled for a limited number of scheduling policies. Almeida has described in [12] how to derive the response time for deferrable servers by using capacity functions. Additionally, Davis et al. [13] have shown how the classical response time analysis can be used to model sporadic and periodic servers. But all those approaches are only solutions for special cases. A more general approach has been used by Naedele et al. [14]. In this paper, a simple schedulability test is derived from the real-time calculus. But this approach is also not appropriate for general hierarchical scheduling, because only strict periodic tasks with fixed priorities are considered.

The real-time calculus [15] is based on arrival and service curves. These are used via the min-/max-plus algebra [6] in order to determine the worst-case response times in distributed systems. The concept of service curves allows to handle the different scheduling policies. A good overview is given in [5]. This is done by providing the bounding tasks' capacity for each task. Describing hierarchical scheduling is very simple, because service curves are an integral part of the model. In [16] it was shown how the hierarchical policy of a deferrable server can be described by the real-time calculus. But not all scheduling policies can be analyzed exactly. For example, round-robin can only be approximated by TDMA [17]. Furthermore, task contexts like offsets between tasks can not be included.

One approach to cope with this problem is to combine the real-time calculus and the response time analysis. This is done by modeling hierarchical scheduling with the real-time calculus, round-robin systems with the classical task system used by the response-time analysis as introduced in [17]. But the disadvantage of this concept is the conversion of the event models from one model into the other, which results in a loss of accuracy.

III. Computational Model

In this section we restate the computational model used in real-time calculus [5] and we introduce some additional functions. The model mainly consists of different mathematical functions having certain properties which will be described in the following.

A. General Curves

The real-time calculus models certain aspects of the system by bounding them through lower and upper curves. All lower and upper curves have common properties, respectively. We extend these properties as introduced in [15] by following definitions:

Definition 1 (Lower Curve)**.** A lower curve is a function $f^- : A \to B$ with $A, B \subseteq \mathbb{R}_0^+$ that vanishes at the origin, is monotonically non-decreasing and is superadditive. For all $a_1, a_2 \in A$ and w.l.o.g. $a_1 < a_2$ a lower curve f^- satisfies:

$$\begin{aligned} f^-(0) &= 0 \\ f^-(a_1) &\leq f^-(a_2) \\ f^-(a_1 + a_2) &\geq f^-(a_1) + f^-(a_2) \end{aligned}$$

Definition 2 (Upper Curve)**.** An upper curve is a function $f^+ : A \to B$ with $A, B \subseteq \mathbb{R}_0^+$ that vanishes only at the origin, is monotonically non-decreasing and is subadditive. For all $a_1, a_2 \in A$ and w.l.o.g. $a_1 < a_2$ an upper curve f^+ satisfies:

$$\begin{aligned} f^+(0) &= 0 \\ f^+(a_2) &> 0 \\ f^+(a_1) &\leq f^+(a_2) \\ f^+(a_1 + a_2) &\leq f^+(a_1) + f^+(a_2) \end{aligned}$$

B. Specific Curves

The first aspect that is modeled are the arrival curves which are defined as follows:

Definition 3 (Event-Based Arrival Curves)**.** Let $R[s, t)$ denote the number of events that arrive on an event stream in the time interval $[s, t)$. Then the corresponding lower and upper arrival curves are denoted as $\overline{\alpha}^- : \mathbb{R}_0^+ \to \mathbb{N}_0$ and $\overline{\alpha}^+ : \mathbb{R}_0^+ \to \mathbb{N}_0$, respectively, and satisfy $\forall s, t \in \mathbb{R}_0^+$ where $s \leq t$:

$$\overline{\alpha}^-(t - s) \leq R[s, t) \leq \overline{\alpha}^+(t - s)$$

A specific event model used in literature is the periodic model with jitter and minimum distance [8]. This model can be easily described by arrival curves. Given the parameters (p, j, d), where p is the period, j the jitter and d the minimum distance of events, the corresponding arrival curves are specified as follows:

$$\overline{\alpha}^-(\Delta) = \max\left\{ \left\lfloor \frac{\Delta - j}{p} \right\rfloor, 0 \right\} \tag{1}$$

$$\overline{\alpha}^+(\Delta) = \min\left\{ \left\lceil \frac{\Delta + j}{p} \right\rceil, \left\lceil \frac{\Delta}{d} \right\rceil \right\} \tag{2}$$

where Δ is the length of the interval considered.

The next aspect that is modeled are the available resources. They are modeled with service curves:

Definition 4 (Resource-Based Service Curves)**.** Let $C[s, t)$ denote the amount of demand that a resource can process in the time interval $[s, t)$. Then the corresponding lower and upper service curves are denoted as $\beta^- : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ and $\beta^+ : \mathbb{R}_0^+ \to \mathbb{R}_0^+$, respectively, and satisfy $\forall s, t \in \mathbb{R}_0^+$ where $s \leq t$:

$$\beta^-(t - s) \leq C[s, t) \leq \beta^+(t - s)$$

It is assumed that the image of both the lower and upper service curve is the set $\mathbb{R}_0^+$.

Note that β is defined in resource units, whereas $\overline{\alpha}$ is defined in event units. Therefore α describes a requested computation demand. It is commonly assumed, that the resource used is an unit processor. For every unit of time an unit of resource is available. This can be easily described by service curves as follows:

$$\beta^-(\Delta) = \Delta \tag{3}$$

$$\beta^+(\Delta) = \Delta \tag{4}$$

The arrival curves have been defined in event units and the service curves in resource units, therefore a transformation between both units is needed. This can be achieved by using workload curves:

Definition 5 (Workload Curves)**.** Let $W(u)$ denote the total resource demand created on a component by u consecutive events of an incoming event stream. Then the corresponding lower and upper workload curves are denoted as $\gamma^- : \mathbb{N}_0 \to \mathbb{R}_0^+$ and $\gamma^+ : \mathbb{N}_0 \to \mathbb{R}_0^+$, respectively, and satisfy $\forall u, v \in \mathbb{N}_0$ where $u \leq v$:

$$\gamma^-(v-u) \leq W(v) - W(u) \leq \gamma^+(v-u)$$

Using the workload curves the event-based arrival curves can now be transformed into resource-based arrival curves as follows:

$$\alpha^-(\Delta) = \gamma^-(\overline{\alpha}^-(\Delta)) \tag{5}$$

$$\alpha^+(\Delta) = \gamma^+(\overline{\alpha}^+(\Delta)) \tag{6}$$

A common notion in literature is to specify the best-case execution demand c^- (BCED) and the worst-case execution demand c^+ (WCED) an event causes. It is then assumed that in the best case every event only causes a demand of c^- and likewise in the worst case it is assumed that every event causes a demand of c^+. This behavior can be easily described by workload curves as follows:

$$\gamma^-(k) = k \cdot c^- \tag{7}$$

$$\gamma^+(k) = k \cdot c^+ \tag{8}$$

where $k \in \mathbb{N}_0$ is the number of consecutive events.

Up to this point, the curves introduced are the same as those used in the real-time calculus. Additional definitions of curves are needed for the generalized response-time analysis. First the pseudo-inverse of the upper arrival curve is defined. This curve denotes the length an interval has to be at least in size for a given amount of events to occur in it.

Definition 6 (Pseudo-Inverse Upper Arrival Curve)**.** Let $\overline{\alpha}^+(\Delta)$ be an upper arrival curve, then the pseudo-inverse curve ${\overline{\alpha}^+}^{-1}(k)$ is defined as follows:

$${\overline{\alpha}^+}^{-1}(k) = \inf_{\Delta \in \mathbb{R}_0^+} \left\{ \Delta \,\middle|\, k \leq \overline{\alpha}^+(\Delta) \right\}$$

If the periodic model with jitter and minimum distance (p, j, d) is given, the inverse of the upper arrival curve is given by:

$${\overline{\alpha}^+}^{-1}(k) = \begin{cases} 0 & \text{if } k = 0 \\ \max\{(k-1)p - j, (k-1)d\} & \text{if } k > 0 \end{cases}$$

Second the pseudo-inverse of the lower service curve is described. This curve denotes the length an interval has to be at most in size for a given amount of resource units to be available in it.

Definition 7 (Pseudo-Inverse Lower Service Curve)**.** Let β^- be a lower service curve, then the pseudo-inverse curve ${\beta^-}^{-1}$ is defined as follows:

$${\beta^-}^{-1}(c) = \inf_{\Delta \in \mathbb{R}_0^+} \left\{ \Delta \,\middle|\, c \leq \beta^-(\Delta) \right\}$$

C. Processing Component and Scheduling Domain

Next the greedy processing component is introduced, an abstract component used in real-time calculus to model the execution of a task on a resource as given in [18].

Definition 8 (Greedy Processing Component)**.** Let τ denote a greedy processing component (GPC).

$$\tau = ((\overline{\alpha}^-, \overline{\alpha}^+), (\beta^-, \beta^+), (\gamma^-, \gamma^+))$$

Whenever an event occurs, described by the corresponding arrival curves $\overline{\alpha}_\tau^-$ and $\overline{\alpha}_\tau^+$, a job of the task the GPC represents is created to process the event. Events are processed in a first-in first-out order. The demand created by the events, described by the corresponding workload curves γ_τ^- and γ_τ^+, is processed according to the availability of resources described by the corresponding service curves β_τ^- and β_τ^+. After an event is processed, a new event is created which possibly triggers any succeeding component.

The relation between the arrival curve, the service curve and the outgoing service curve is given by:

$$\beta_\tau^{'-}(\Delta) = \sup_{0 \leq \lambda \leq \Delta} \left\{ \beta_\tau^-(\lambda) - \alpha_\tau^+(\lambda) \right\}$$

Now two characteristic values are introduced, the utilization and the capacity of a GPC τ. The utilization U_τ denotes the average amount of resource units the GPC τ demands per time unit, whereas the capacity C_τ denotes the average amount of resource units available per time unit to process the demand caused by the GPC τ.

$$U_\tau = \lim_{\Delta \to \infty} \frac{\alpha_\tau^+(\Delta)}{\Delta} \qquad C_\tau = \lim_{\Delta \to \infty} \frac{\beta_\tau^-(\Delta)}{\Delta}$$

It is assumed that the utilization of a GPC is less than the capacity available for it. This guarantees that the delay experienced by an event is bounded.

$$U_\tau < C_\tau \tag{9}$$

Definition 9 (Scheduling Domain)**.** Let Γ denote a scheduling domain consisting of a set of GPCs and resource-based service curves

$$\Gamma = \left(\{\tau_1, \ldots, \tau_n\}, (\beta^-, \beta^+)\right)$$

A scheduling domain is an instance of a scheduling policy. The set of GPCs which comprises the scheduling domain are those that are scheduled by said instance. The scheduling domain distributes the available resources, described by the service curves β_Γ^- and β_Γ^+, accordingly to its policy.

Each scheduling domain may be embedded in another scheduling domain, forming scheduling hierarchies. For a parent scheduling domain the embedded scheduling domain is treated as a GPC which is given resources accordingly to the parents scheduling policy.

IV. Response-Time Analysis

Using the models introduced in section III, we will now derive the equations representing a more general response-time analysis. We start by restating the bound of the delay a job of a task r_τ can experience according to the network calculus. This is described by the greatest horizontal deviation between the upper arrival curve α_τ^+ and the lower service curve β_τ^- [6, p. 28]:

$$r_\tau \leq \sup_{\lambda \in \mathbb{R}_0^+} \left\{ \inf_{\mu \in \mathbb{R}_0^+} \left\{ \mu \,\middle|\, \alpha_\tau^+(\lambda) \leq \beta_\tau^-(\lambda + \mu) \right\} \right\} \tag{10}$$

Using the pseudo-inverse of the service curve, the horizontal deviation in (10) can also be expressed as follows [6, p. 155]:

$$\sup_{\lambda \in \mathbb{R}_0^+} \left\{ {\beta_\tau^-}^{-1}(\alpha_\tau^+(\lambda)) - \lambda \right\} \tag{11}$$

We will now rewrite (11) to represent a more general form of the response-time analysis by the following theorem:

Theorem 1. *The response time r_τ of task τ is bounded from above by:*

$$r_\tau \leq \sup_{k \in \mathbb{N}_0} \left\{ {\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) - {\overline{\alpha}_\tau^+}^{-1}(k) \right\} \tag{12}$$

Proof: Let A_k be the preimage of $\overline{\alpha}_\tau^+$ at k:

$$A_k = \left\{ \lambda \in \mathbb{R}_0^+ \,\middle|\, \overline{\alpha}_\tau^+(\lambda) = k \right\} \quad \forall k \in \mathbb{N}_0$$

Due to $\overline{\alpha}_\tau^+$ being monotonically non-decreasing, the preimages are disjoint and the union of all preimages A_k is the set $\mathbb{R}_0^+$

$$\bigcup_{k \in \mathbb{N}_0} A_k = \mathbb{R}_0^+$$

therefore (11) can be rewritten as follows:

$$\sup_{k \in \mathbb{N}_0} \left\{ \sup_{\lambda \in A_k} \left\{ {\beta_\tau^-}^{-1}(\alpha_\tau^+(\lambda)) - \lambda \right\} \right\} \tag{13}$$

Obviously $\overline{\alpha}_\tau^+(\lambda) = k$ for all $\lambda \in A_k$ therefore $\alpha_\tau^+(\lambda) = \gamma_\tau^+(\overline{\alpha}_\tau^+(\lambda)) = \gamma_\tau^+(k)$. Because $\overline{\alpha}_\tau^+$ is monotonically non-decreasing and the interval considered is a left-closed, right-open interval, the preimage A_k of $\overline{\alpha}_\tau^+$ can also be stated as follows:

$$A_k = \begin{cases} \{0\} & \text{if } k = 0 \\ \left({\overline{\alpha}_\tau^+}^{-1}(k), {\overline{\alpha}_\tau^+}^{-1}(k+1)\right] & \text{if } k \in \mathbb{N} \end{cases}$$

thus

$$\begin{aligned} \sup_{\lambda \in A_k} \left\{ {\beta_\tau^-}^{-1}(\alpha_\tau^+(\lambda)) - \lambda \right\} &= \sup_{\lambda \in A_k} \left\{ {\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) - \lambda \right\} \\ &= {\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) + \sup_{\lambda \in A_k} \{-\lambda\} \\ &= {\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) - \inf_{\lambda \in A_k} \{\lambda\} \\ &= {\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) - {\overline{\alpha}_\tau^+}^{-1}(k) \end{aligned}$$

Therefore (13) can also be expressed as follows:

$$\sup_{k \in \mathbb{N}_0} \left\{ {\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) - {\overline{\alpha}_\tau^+}^{-1}(k) \right\}$$

■

Using the terminology of Tindell et al. [19], ${\beta_\tau^-}^{-1}(\gamma_\tau^+(k))$ is the time when the k-th invocation finishes and ${\overline{\alpha}_\tau^+}^{-1}(k)$ is the release time of the k-th invocation, thus the difference is the response time of the k-th invocation. Therefore, (12) describes the supremum of the response times of all invocations of task τ, which is a bound on the worst-case response time that a job of task τ can experience.

To compute the worst-case response time according to (12) an infinite amount of jobs has to be analyzed. This is neither possible nor necessary as is shown with following theorem.

Theorem 2. *The upper bound on the response time r_τ of a task τ is given by:*

$$\begin{aligned} r_\tau &\leq \max_{k \in [1..m]} \left\{ {\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) - {\overline{\alpha}_\tau^+}^{-1}(k) \right\} \\ m &= \min_{k \in \mathbb{N}} \left\{ k \,\middle|\, {\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) \leq {\overline{\alpha}_\tau^+}^{-1}(k+1) \right\} \end{aligned}$$

Proof: According to lemma 4 the pseudo-inverse of the lower service curve is an upper curve (definition 2). The composition of the pseudo-inverse of the lower service curve and the upper workload curve is again an upper curve according to lemma 5, see appendix, and is therefore positive except at the origin. Thus ${\beta_\tau^-}^{-1}(\gamma_\tau^+(1))$ is positive. The pseudo-inverse of the upper arrival curve vanishes at $k = 1$. Therefore the difference ${\beta_\tau^-}^{-1}(\gamma_\tau^+(k)) - {\overline{\alpha}_\tau^+}^{-1}(k)$ is positive at $k = 1$ and $k = 0$ can be omitted.

Let $f_\tau(\lambda) = {\beta_\tau^-}^{-1}(\alpha_\tau^+(\lambda))$ and let $\lambda_1 > 0$ be a fix-point of f. For all $\lambda \in \mathbb{R}_0^+$ where $\lambda = k\lambda_1 + \lambda_2$, $k \in \mathbb{N}_0$ and

$0 \leq \lambda_2 < \lambda_1$ it follows with lemma 3, 4 and 5:

$$\begin{aligned} & f_\tau(\lambda) - \lambda \\ &= f_\tau(k\lambda_1 + \lambda_2) - (k\lambda_1 + \lambda_2) \\ &\leq f_\tau(k\lambda_1) + f_\tau(\lambda_2) - (k\lambda_1 + \lambda_2) \\ &\leq k\lambda_1 + f_\tau(\lambda_2) - (k\lambda_1 + \lambda_2) \\ &= f_\tau(\lambda_2) - \lambda_2 \end{aligned}$$

This means that for any λ after the fix-point λ_1 there exists a λ_2 before the fix-point where the horizontal deviation is greater. Therefore the supremum (11) lies in the range $[0, \lambda_1]$, where $\lambda_1 > 0$ is the smallest fix-point. Hence (11) can be rewritten as follows:

$$\sup_{\lambda \in \mathbb{R}_0^+} \left\{ \beta_\tau^{-^{-1}}(\alpha_\tau^+(\lambda)) - \lambda \right\} = \sup_{0 \leq \lambda \leq \lambda_1} \left\{ \beta_\tau^{-^{-1}}(\alpha_\tau^+(\lambda)) - \lambda \right\}$$

To apply this to (12), the smallest k has to be determined for which the preimage A_k contains the smallest positive fix-point of $\beta_\tau^{-^{-1}} \circ \alpha_\tau^+$.

$$\begin{aligned} m &= \min_{k \in \mathbb{N}} \left\{ k \,\middle|\, \exists \lambda \in A_k : \beta^{-^{-1}}(\alpha^+(\lambda)) = \lambda \right\} \\ &= \min_{k \in \mathbb{N}} \left\{ k \,\middle|\, \exists \lambda \in A_k : \beta^{-^{-1}}(\gamma^+(k)) = \lambda \right\} \\ &= \min_{k \in \mathbb{N}} \left\{ k \,\middle|\, \beta^{-^{-1}}(\gamma^+(k)) \leq \overline{\alpha}^{+^{-1}}(k+1) \right\} \end{aligned}$$

Concluding (12) can now be rewritten as follows:

$$\begin{aligned} r_\tau &\leq \max_{k \in [1..m]} \left\{ \beta_\tau^{-^{-1}}(\gamma_\tau^+(k)) - \overline{\alpha}_\tau^{+^{-1}}(k) \right\} \\ m &= \min_{k \in \mathbb{N}} \left\{ k \,\middle|\, \beta^{-^{-1}}(\gamma^+(k)) \leq \overline{\alpha}^{+^{-1}}(k+1) \right\} \end{aligned}$$

■

With theorem 2 we have obtained a more general form on the bound of the worst-case response time. The main issue in computing the bound of the worst-case response time is the pseudo-inverse of the lower service curve $\beta^{-^{-1}}$. This strongly depends on the scheduling policy applied to the tasks.

V. Scheduling Policies

In this section it is shown how the necessary pseudo-inverse service curve of a GPC can be computed for various scheduling policies.

A. Fixed-Priority Preemptive Scheduling (FPPS)

Let Γ be a scheduling domain which uses fixed priorities to schedule the GPCs within its domain. Each GPC τ is assigned a unique priority ϕ_τ within the scheduling domain and can be preempted at any time. Let the set of GPCs $\{\tau_1, \ldots, \tau_n\}$ be ordered by decreasing priority. τ_1 has the highest priority and τ_n has the lowest priority.

The relation between incoming and outgoing service curves in a FPPS domain is as follows:

$$\beta_{\tau_i}^- = \begin{cases} \beta_\Gamma^- & \text{if } i = 1 \\ \beta_{\tau_{i-1}}'^- & \text{if } i > 1 \end{cases}$$

The lower service curve of the GPC with the highest priority equals the lower service curve of the scheduling domain. For any GPC τ_i with a lower priority the lower service curve equals the outgoing lower service curve of the GPC with the next higher priority. Using both relations it is possible to describe the lower service curve for any GPC of a FPPS domain as stated by following lemma:

Lemma 1. *Let Γ be a FPPS domain. The lower service curve of GPC τ_i is given by:*

$$\beta_{\tau_i}^-(\Delta) = \sup_{0 \leq \lambda \leq \Delta} \left\{ \beta_\Gamma^-(\lambda) - \sum_{j=1}^{i-1} \alpha_{\tau_j}^+(\lambda) \right\} \tag{14}$$

Proof: Let $i = 1$, then (14) holds:

$$\beta_{\tau_1}^-(\Delta) = \sup_{0 \leq \lambda \leq \Delta} \left\{ \beta_\Gamma^-(\lambda) \right\} = \beta_\Gamma^-(\Delta)$$

Assume (14) holds for some $i \geq 1$, then it follows that (14) holds for $i + 1$:

$$\begin{aligned} \beta_{\tau_{i+1}}^- &= \beta_{\tau_i}'^- \\ &= \sup_{0 \leq \lambda \leq \Delta} \left\{ \beta_{\tau_i}^-(\lambda) - \alpha_{\tau_i}^+(\lambda) \right\} \\ &= \sup_{0 \leq \lambda \leq \Delta} \left\{ \sup_{0 \leq \mu \leq \lambda} \left\{ \beta_\Gamma^-(\mu) - \sum_{j=1}^{i-1} \alpha_{\tau_j}^+(\mu) \right\} - \alpha_{\tau_i}^+(\lambda) \right\} \\ &= \sup_{0 \leq \lambda \leq \Delta} \left\{ \sup_{0 \leq \mu \leq \lambda} \left\{ \beta_\Gamma^-(\mu) - \sum_{j=1}^{i-1} \alpha_{\tau_j}^+(\mu) - \alpha_{\tau_i}^+(\lambda) \right\} \right\} \\ &= \sup_{0 \leq \mu \leq \Delta} \left\{ \sup_{\mu \leq \lambda \leq \Delta} \left\{ \beta_\Gamma^-(\mu) - \sum_{j=1}^{i-1} \alpha_{\tau_j}^+(\mu) - \alpha_{\tau_i}^+(\lambda) \right\} \right\} \\ &= \sup_{0 \leq \mu \leq \Delta} \left\{ \beta_\Gamma^-(\mu) - \sum_{j=1}^{i-1} \alpha_{\tau_j}^+(\mu) + \sup_{\mu \leq \lambda \leq \Delta} \left\{ -\alpha_{\tau_i}^+(\lambda) \right\} \right\} \\ &= \sup_{0 \leq \mu \leq \Delta} \left\{ \beta_\Gamma^-(\mu) - \sum_{j=1}^{i-1} \alpha_{\tau_j}^+(\mu) - \alpha_{\tau_i}^+(\mu) \right\} \\ &= \sup_{0 \leq \mu \leq \Delta} \left\{ \beta_\Gamma^-(\mu) - \sum_{j=1}^{i} \alpha_{\tau_j}^+(\mu) \right\} \end{aligned}$$

Thus (14) holds for every i. ■

With lemma 1 the pseudo-inverse lower service curve of a GPC τ_i is derived by following theorem:

Theorem 3. *Given the lower service curve $\overline{\beta_\Gamma^-}$ of a FPPS domain and the upper arrival curve of all GPCs $\alpha_{\tau_1}^+, \ldots, \alpha_{\tau_n}^+$ scheduled by the FPPS, the length of the interval in which d resource units are available for the GPC with i-highest priority is bounded by*

$$\beta_{\tau_i}^{-^{-1}}(d) = \min_{\Delta \in \mathbb{R}_0^+} \left\{ \Delta \,\middle|\, \beta_\Gamma^-(\Delta) = d + \sum_{j=1}^{i-1} \alpha_{\tau_j}^+(\Delta) \right\} \tag{15}$$

Proof:

$$
\begin{aligned}
&{\beta_{\tau_i}^{-}}^{-1}(d)\\
&= \inf_{\Delta\in\mathbb{R}_0^+} \left\{\Delta \,\middle|\, d \le \beta_{\tau_i}^{-}(\Delta)\right\}\\
&= \inf_{\Delta\in\mathbb{R}_0^+} \left\{\Delta \,\middle|\, d \le \sup_{0\le\lambda\le\Delta} \left\{\beta_\Gamma^-(\lambda) - \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\lambda)\right\}\right\}\\
&= \inf_{\Delta\in\mathbb{R}_0^+} \left\{\Delta \,\middle|\, d \le \beta_\Gamma^-(\Delta) - \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\Delta)\right\}\\
&= \inf_{\Delta\in\mathbb{R}_0^+} \left\{\Delta \,\middle|\, \beta_\Gamma^-(\Delta) \ge d + \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\Delta)\right\}
\end{aligned}
$$

Due to the assumption that the image of β_Γ^- is the set $\mathbb{R}_0^+$, the inequality can be replaced by an equality and the infimum can be replaced by a minimum.

$$
= \min_{\Delta\in\mathbb{R}_0^+} \left\{\Delta \,\middle|\, \beta_\Gamma^-(\Delta) = d + \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\Delta)\right\}
$$

■

Equation (15) describes a fix-point which can be computed with following recurrence relation:

$$
\vartheta_{\tau_i}^{\#n} = \begin{cases} {\beta_\Gamma^-}^{-1}(d) & \text{if } n = 0\\ {\beta_\Gamma^-}^{-1}(d + \sum\limits_{j=1}^{i-1}\alpha_{\tau_j}^+(\vartheta_{\tau_i}^{\#n-1})) & \text{if } n > 0 \end{cases} \tag{16}
$$

With theorem 2 and 3 and by chosing the corresponding functions for the arrival and service curves, the equations to find the worst-case response times as described by [20], [21], and [19] can be obtained.

Example. We will now exemplarily derive the analysis given by Tindell et al. in [19]. The functions to represent the strict periodical model, the unit processor, and the execution demand are as follows:

$$
\begin{aligned}
&\overline{\alpha}_\tau^+(\Delta) = \left\lceil \frac{\Delta}{p_\tau} \right\rceil && {\overline{\alpha}_\tau^+}^{-1}(k) = (k-1)\cdot p_\tau\\
&\beta_\Gamma^-(\Delta) = \Delta && {\beta_\Gamma^-}^{-1}(d) = d\\
&\gamma_\tau^+(k) = k\cdot c_\tau^+
\end{aligned}
$$

Using theorem 2 and 3 we obtain the following equations:

$$
\begin{aligned}
r_{\tau_i} &\le \max_{k\in[1..m]} \left\{\vartheta_{\tau_i}(k) - {\overline{\alpha}_{\tau_i}^+}^{-1}(k)\right\}\\
m &= \min_{k\in\mathbb{N}} \left\{k \,\middle|\, \vartheta_{\tau_i}(k) \le {\overline{\alpha}_{\tau_i}^+}^{-1}(k+1)\right\}
\end{aligned}
$$

where

$$
\begin{aligned}
\vartheta_{\tau_i}(k) &= {\beta_{\tau_i}^-}^{-1}(\gamma_{\tau_i}^+(k))\\
&= \min_{\Delta\ge 0} \left\{\Delta \,\middle|\, \Delta = k\cdot c_{\tau_i}^+ + \sum_{j=1}^{i-1}\left(\left\lceil \frac{\Delta}{p_{\tau_j}} \right\rceil \cdot c_{\tau_j}^+\right)\right\}
\end{aligned}
$$

which is equivalent to the analysis given in [19].

B. Fixed-Priority Non-Preemptive Scheduling (FPNP)

For a FPNP domain, the same assumptions as for a FPPS domain are made, except that the GPCs cannot be preempted. Due to the similarity between FPPS and FPNP, the lower service curve and the inverse of the lower service curve are also similar as described by following lemma and theorem:

Lemma 2. *Given the lower service curve $\overline{\beta}_\Gamma^-$ of the scheduling domain and the upper arrival curves of all higher priority GPCs $\overline{\alpha}_{\tau_1}^+,\dots,\overline{\alpha}_{\tau_{i-1}}^+$, the lower service curve for GPC τ_i is given by:*

$$
\beta_{\tau_i}^-(\Delta) = \max\left\{0, \sup_{0\le\lambda\le\Delta}\left\{\beta_\Gamma^-(\lambda) - \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\lambda)\right\} - b_i\right\} \tag{17}
$$

with

$$
b_i = \max_{i<j\le n}\left\{\gamma_{\tau_j}^+(1)\right\}
$$

Proof: The proof is given in [18]. ■

Theorem 4. *Given the lower service curve $\overline{\beta}_\Gamma^-$ of a FPNS domain and the upper arrival curve of all GPCs $\alpha_{\tau_1}^+,\dots,\alpha_{\tau_n}^+$ scheduled by the FPNS, the length of the interval in which d resource units are available for the GPC with i-highest priority is bounded by*

$$
{\beta_{\tau_i}^-}^{-1}(d) = \min_{\Delta\in\mathbb{R}_0^+}\left\{\Delta \,\middle|\, \overline{\beta}_\Gamma^-(\Delta) = d + b_i + \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\Delta)\right\} \tag{18}
$$

with

$$
b_i = \max_{i<j\le n}\{\gamma_{\tau_j}^+(1)\}
$$

Proof:

$$
\begin{aligned}
&{\beta_{\tau_i}^-}^{-1}(d)\\
&= \inf_{\Delta\in\mathbb{R}_0^+}\left\{\Delta \,\middle|\, d \le \beta_{\tau_i}^-(\Delta)\right\}\\
&= \inf_{\Delta\in\mathbb{R}_0^+}\left\{\Delta \,\middle|\, d \le \sup_{0\le\lambda\le\Delta}\left\{\beta_\Gamma^-(\lambda) - \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\lambda)\right\} - b_i\right\}\\
&= \inf_{\Delta\in\mathbb{R}_0^+}\left\{\Delta \,\middle|\, d \le \beta_\Gamma^-(\Delta) - \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\Delta) - b_i\right\}\\
&= \inf_{\Delta\in\mathbb{R}_0^+}\left\{\Delta \,\middle|\, \beta_\Gamma^-(\Delta) \ge d + b_i + \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\Delta)\right\}\\
&= \min_{\Delta\in\mathbb{R}_0^+}\left\{\Delta \,\middle|\, \beta_\Gamma^-(\Delta) = d + b_i + \sum_{j=1}^{i-1}\alpha_{\tau_j}^+(\Delta)\right\}
\end{aligned}
$$

■

A very similar recurrence relation like (16) can be used to compute (18).

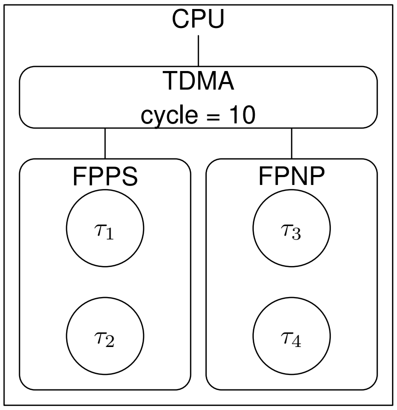

Figure 1. Example System

C. Time Division Multiple Access (TDMA)

Let Γ be a scheduling domain which uses a TDMA policy to schedule the GPCs within its domain. The amount of resources which are available to the scheduling domain is limited by a lower service curve β^-_Γ. The cycle length that the scheduler uses is given by c. Each GPC τ within the scheduling domain has a slot of length s_τ assigned. If a GPC demands an amount of d resource units, the total amount of resources needed is given by [8]:

$$\left\lceil \frac{d}{s_\tau} \right\rceil (c - s_\tau) + d$$

Therefore the inverse of the lower service curve of a GPC τ is given by:

$${\beta^-_\tau}^{-1}(d) = {\beta^-_\Gamma}^{-1}\left(\left\lceil \frac{d}{s_\tau} \right\rceil (c - s_\tau) + d \right) \tag{19}$$

VI. Example

In this section we will show how to use the generalized worst-case response-time analysis given by theorem 2. In [17] an example of a distributed system was given in which one resource uses hierarchical scheduling. To be able to perform a response-time analysis, the resource using the hierarchical scheduling had to be substituted by resources with a single scheduling domain. These scheduling domains had to be chosen carefully, because they had to have the same behavior as the original scheduling domains. The example that will be used to show the response-time analysis of the scheduling policies discussed in section V is a modified version of the example given in [17].

Table I
Parameters of GPCs of example system

GPC	p	j	d	c^+
τ_1	150	450	0	20
τ_2	150	370	8	20
τ_3	250	125	0	15
τ_4	250	281	5	3

The example is shown in Fig. 1. The resource is a unit processor, and therefore the service curves for the top most scheduling domain Γ_1 are obtained by using (3) and (4). The inverse of the lower service curve ${\beta^-_{\Gamma_1}}^{-1}$ is thus the identity function. TDMA with a cycle of 10 resource units is used as scheduling policy for the domain Γ_1. The cycle is divided into two slots. One with six resource units and another one with four resource units.

$$\Gamma_1 = \{\Gamma_2, \Gamma_3\}$$
$${\beta^-_{\Gamma_1}}^{-1}(d) = d$$
$$c = 10,\ s_{\Gamma_2} = 6,\ s_{\Gamma_3} = 4$$

Inside the slot with six resource units, a scheduling domain Γ_2 with a FPPS policy is used. In the other slot, a scheduling domain with FPNP policy is used. Both domains consist of two GPCs.

$$\Gamma_2 = \{\tau_1, \tau_2\} \quad \Gamma_3 = \{\tau_3, \tau_4\}$$

The inverse of the lower service curve for the scheduling domains Γ_2 and Γ_3 can be obtained by using (19).

$$\begin{aligned} {\beta^-_{\Gamma_2}}^{-1}(d) &= {\beta^-_{\Gamma_1}}^{-1}\left(\left\lceil \frac{d}{s_{\Gamma_2}} \right\rceil (c - s_{\Gamma_2}) + d \right) = \left\lceil \frac{d}{6} \right\rceil 4 + d \\ {\beta^-_{\Gamma_3}}^{-1}(d) &= {\beta^-_{\Gamma_1}}^{-1}\left(\left\lceil \frac{d}{s_{\Gamma_3}} \right\rceil (c - s_{\Gamma_3}) + d \right) = \left\lceil \frac{d}{4} \right\rceil 6 + d \end{aligned}$$

To describe the event streams of the GPCs, the periodic model with jitter and minimum distance is used. The corresponding upper arrival curve can be obtained with (2). For workload transformation the notion of worst-case execution demand c^+ is used. The corresponding upper workload curve can be obtained by using (8). The parameters for each GPC are listed in table I. Now the worst-case response times of the GPCs τ_1, τ_2 and τ_3 will be computed. The scheduling domain of τ_1 and τ_2 uses a FPPS policy and the scheduling domain of τ_3 uses a FPNP policy, therefore (15) and (18) are used to compute the inverse of the lower service curve. Let $\vartheta_{\tau_i}(k)$ be the length of the interval in which the demand caused by k consecutive events of τ_i is guaranteed to be available.

$$\begin{aligned} \vartheta_{\tau_1}(k) &= {\beta^-_{\tau_1}}^{-1}(\gamma^+_{\tau_1}(k)) \\ &= \min_{\Delta \in \mathbb{R}^+_0} \left\{ \Delta \,\middle|\, \beta^-_{\Gamma_2}(\Delta) = k \cdot c^+_{\tau_1} \right\} \\ &= {\beta^-_{\Gamma_2}}^{-1}(k \cdot c^+_{\tau_1}) \\ \vartheta_{\tau_2}(k) &= {\beta^-_{\tau_2}}^{-1}(\gamma^+_{\tau_2}(k)) \\ &= \min_{\Delta \in \mathbb{R}^+_0} \left\{ \Delta \,\middle|\, \beta^-_{\Gamma_2}(\Delta) = k \cdot c^+_{\tau_2} + \sum_{j=1}^{1} \alpha^+_{\tau_j}(\Delta) \right\} \\ &= \min_{\Delta \in \mathbb{R}^+_0} \left\{ \Delta \,\middle|\, \beta^-_{\Gamma_2}(\Delta) = k \cdot c^+_{\tau_2} + \alpha^+_{\tau_1}(\Delta) \right\} \end{aligned}$$

Table II
RESPONSE-TIME ANALYSIS FOR GPC τ_1 OF EXAMPLE SYSTEM

k	$\vartheta_{\tau_1}(k)$	$r_{\tau_1}(k)$	${\overline{\alpha}_{\tau_1}^{+}}^{-1}(k+1)$
1	36	36	0
2	68	68	0
3	100	100	0
4	136	136	150

$$\begin{aligned}\vartheta_{\tau_3}(k) &= {\beta_{\tau_3}^{-}}^{-1}(\gamma_{\tau_3}^{+}(k))\\ &= \min_{\Delta\in\mathbb{R}_0^{+}}\left\{\Delta \,\middle|\, \beta_{\Gamma_3}^{-}(\Delta) = k\cdot c_{\tau_3}^{+} + c_{\tau_4}^{+}\right\}\\ &= {\beta_{\Gamma_3}^{-}}^{-1}(k\cdot c_{\tau_3}^{+} + c_{\tau_4}^{+})\end{aligned}$$

$\vartheta_{\tau_1}(k)$ and $\vartheta_{\tau_3}(k)$ can be computed directly, whereas for $\vartheta_{\tau_2}(k)$ the recurrence relation (16) with $d = k\cdot c_{\tau_2}^{+}$ can be used. The response-time $r_{\tau_i}(k)$ can then be described as follows

$$r_{\tau_i}(k) = \vartheta_{\tau_i}(k) - {\overline{\alpha}_{\tau_i}^{+}}^{-1}(k)$$

The response-time analysis for GPC τ_1, τ_2 τ_3 is shown in tables II, III, and IV respectively. The first column of the tables denotes the number of consecutive events k considered and the corresponding response-time $r_{\tau_i}(k)$ is denoted in the second to last column. Due to the usage of the recurrence relation for the response-time analysis of τ_2, the columns 2-5 of table III list the single steps in the recurrence relation.

VII. CONCLUSION

It has been shown how a hierarchical response-time analysis can be directly derived from the theory of the real-time calculus. This new approach allows to consider hierarchical scheduling in classical response time analysis in general. In a detailed discussion it has been described how this methodology can be used for different scheduling policies as fixed-priority preemptive scheduling (FPPS), fixed-priority non-preemptive scheduling (FPNP), and time division multiple access (TDMA). The approach can easily be used to handle other scheduling policies like round-robin and first-come first-serve. The paper closes with a modified example from literature.

Table III
RESPONSE-TIME ANALYSIS FOR GPC τ_2 OF EXAMPLE SYSTEM

k	n	$\vartheta_{\tau_2}^{\#n}(k)$	$\alpha_{\tau_1}^{+}(\vartheta_{\tau_2}^{\#n}(k))$	$\vartheta_{\tau_2}^{\#n+1}(k)$	$r_{\tau_2}(k)$	${\overline{\alpha}_{\tau_2}^{+}}^{-1}(k+1)$
	1	36	80	168		
1	2	168	100	200	200	8
	3	200	100	200		
	1	68	80	200		
2	2	200	100	236	228	16
	3	236	100	236		
	1	100	80	236		
3	2	236	100	268	252	80
	3	268	100	268		
	1	136	80	268		
4	2	268	100	300	220	230
	3	300	100	300		
	1	168	100	336		
5	2	336	120	368	138	380
	3	368	120	368		

Table IV
RESPONSE-TIME ANALYSIS FOR GPC τ_3 OF EXAMPLE SYSTEM

k	$\vartheta_{\tau_3}(k)$	$r_{\tau_3}(k)$	${\overline{\alpha}_{\tau_3}^{+}}^{-1}(k+1)$
1	48	48	125

For advanced research in this area it is necessary to cover different response-time analysis approaches by deriving new analysis methods from the powerful mathematical approach of the real-time calculus.

APPENDIX

Lemma 3. *Let f be an upper curve and let a be a fix-point of f, then the following inequality holds $\forall k\in\mathbb{N}_0$:*

$$f(k\cdot a)\le k\cdot a \qquad (20)$$

Proof: Let $k=0$, then (20) holds

$$0 = f(0)\le 0$$

Assume (20) holds for some $k\ge 0$, then it follows that (20) also holds for $k+1$

$$\begin{aligned}f((k+1)\cdot a) &= f(k\cdot a + a)\\ &\le f(k\cdot a) + f(a)\le k\cdot a + a = (k+1)\cdot a\end{aligned}$$

Therefore (20) holds $\forall k\in\mathbb{N}_0$. ■

Lemma 4. *Given a lower curve $f^{-}: A\to B$, the pseudo-inverse curve ${f^{-}}^{-1}: B\to A$ is an upper curve.*

Proof: The lower curve vanishes at the origin and is monotonically non-decreasing, therefore it follows that the pseudo-inverse also vanishes at the origin

$${f^{-}}^{-1}(0) = \inf_{a\in A}\left\{a \,\middle|\, 0\le f^{-}(a)\right\} = 0$$

Assume $\exists b>0$ with

$${f^{-}}^{-1}(b) = 0$$

then it follows

$$\inf_{a\in A}\left\{a \,\middle|\, b\le f^{-}(a)\right\} = 0$$

that $\forall a > 0 \; f^{-}(a) \geq b$. Assume $a_1, a_2 > 0$, then it must satisfy

$$f(a_1 + a_2) \geq f(a_1) + f(a_2) \geq f(a_1) + b$$

This is only possible if

$$f(a) = \infty \quad \forall a > 0$$

which is not a valid lower curve, thus it follows that the pseudo-inverse of a lower curves only vanishes at the origin.

The lower curve is monotonically non-decreasing, therefore it follows that the pseudo-inverse is also monotonically non-decreasing. Let $b_1 < b_2$ then

$$\begin{aligned} f^{-^{-1}}(b_1) &= \inf_{a \in A} \left\{ a \left| b_1 \leq f^{-}(a) \right. \right\} \\ &\leq \inf_{a \in A} \left\{ a \left| b_2 \leq f^{-}(a) \right. \right\} = f^{-^{-1}}(b_2) \end{aligned}$$

The proof that the pseudo-inverse of a superadditive function is subadditive is given in [22]. ■

Lemma 5. *Upper curves are closed under composition. Given two upper curves f and g then $f \circ g$ is an upper curve.*

Proof: Let $f : B \to C$ and $g : A \to B$ be upper curves. Furthermore, let $a_1, a_2 \in A$ and without loss of generality let $a_1 < a_2$.

First we show that the composition $f \circ g$ vanishes at the origin

$$f(g(0)) = f(0) = 0$$

second the monotonicity is shown

$$a_1 < a_2 \Rightarrow g(a_1) \leq g(a_2) \Rightarrow f(g(a_1)) \leq f(g(a_2))$$

and last the subadditivity is shown

$$f(g(a_1 + a_2)) \leq f(g(a_1) + g(a_2)) \leq f(g(a_1)) + f(g(a_2))$$

Since the composition has all three properties, it is an upper curve. ■

References

[1] FlexRay, http://www.flexray.com.

[2] G. C. Buttazzo, *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*. Norwell, MA, USA: Kluwer Academic Publishers, 1997.

[3] J. D. Regehr, "Using Hierarchical Scheduling to Support Soft Real-Time Applications in General-Purpose Operating Systems," Ph.D. dissertation, University of Virginia, 2001.

[4] J. P. Lehoczky, "Fixed priority scheduling of periodic task sets with arbitrary deadlines," in *Proceedings of the 11th IEEE Real-Time Systems Symposium*, December 1990, pp. 201–209.

[5] E. Wandeler, "Modular Performance Analysis and Interface-Based Design for Embedded Real-Time Systems," Ph.D. dissertation, Swiss Federal Institute of Technology Zurich, 2006.

[6] J.-Y. Le Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Springer Verlag, 2001.

[7] K. Tindell and J. Clark, "Holistic schedulability analysis for distributed hard real-time systems," *Microprocessing and Microprogramming*, vol. 40, pp. 117–134, 1994.

[8] K. Richter, "Compositional Scheduling Analysis Using Standard Event Models - The SymTA/S Approach," Ph.D. dissertation, University of Braunschweig, 2005.

[9] R. Henia and R. Ernst, "Context-aware scheduling analysis of distributed systems with tree-shaped task-dependencies," in *DATE '05: Proceedings of the conference on Design, Automation and Test in Europe*, 2005, pp. 480–485.

[10] R. Racu, L. Li, R. Henia, A. Hamann, and R. Ernst, "Improved response time analysis of tasks scheduled under preemptive round-robin," in *Proceedings of the International Conference on Hardware-Software Codesign and System Synthesis*, 2007, pp. 179 – 184.

[11] S. Saewong, R. R. Rajkumar, J. P. Lehoczky, and M. H. Klein, "Analysis of Hierarchical Fixed-Priority Scheduling," *Real-Time Systems, Euromicro Conference on*, p. 173, 2002.

[12] L. Almeida, "Response time analysis and server design for hierarchical scheduling," in *proceedings of the IEEE Real-Time Systems Symposium Work-in-Progress*. Citeseer, 2003.

[13] R. Davis and A. Burns, "Hierarchical fixed priority pre-emptive scheduling," in *Real-Time Systems Symposium, 2005. RTSS 2005. 26th IEEE International*, Dec. 2005, pp. 389–398.

[14] M. Naedele, L. Thiele, and M. Eisenring, "Characterizing Variable Task Releases and Processor Capacities," TIK-Report 45. Computer Engineering and Networks Lab, Swiss Federal Instiute of of Technology, Tech. Rep., 1999.

[15] S. Chakraborty, S. Künzli, L. Thiele, A. Herkersdorf, and P. Sagmeister, "Performance evaluation of network processor architectures: combining simulation with analytical estimation," *Comput. Netw.*, vol. 41, no. 5, pp. 641–665, 2003.

[16] P. J. L. Cuijpers and R. J. Bril, "Towards budgeting in real-time calculus: Deferrable servers," in *FORMATS*, 2007, pp. 98–113.

[17] S. Kuenzli, A. Hamann, R. Ernst, and L. Thiele, "Combined approach to system level performance analysis of embedded systems," in *CODES+ISSS '07: Proceedings of the 5th IEEE/ACM international conference on Hardware/software codesign and system synthesis*. New York, NY, USA: ACM, 2007, pp. 63–68.

[18] W. Haid and L. Thiele, "Complex Task Activation Schemes in System Level Performance Analysis," in *CODES+ISSS '07: Proceedings of the 5th IEEE/ACM International Conference on Hardware/Software Codesign and System Synthesis*. New York, NY, USA: ACM, 2007, pp. 173–178.

[19] K. W. Tindell, A. Burns, and A. J. Wellings, "An Extendible Approach for Analyzing Fixed Priority Hard Real-Time Tasks," *Real-Time Systems*, vol. 6, no. 2, pp. 133–151, March 1994.

[20] M. Joseph and P. Pandya, "Finding Response Times in a Real-Time System," *The Computer Journal*, vol. 29, no. 5, pp. 390–395, 1986. [Online]. Available: http://comjnl.oxfordjournals.org/cgi/content/abstract/29/5/390

[21] N. Audsley, A. Burns, M. Richardson, K. Tindell, and A. J. Wellings, "Applying New Scheduling Theory to Static Priority Pre-Emptive Scheduling," *Software Engineering Journal*, vol. 8, pp. 284–292, 1993.

[22] L. P. Østerdal, "Subadditive functions and their (pseudo-) inverses," *Journal of Mathematical Analysis and Applications*, vol. 317, no. 2, pp. 724–731, 2006.