Response time analysis in distributed real-time systems
An improvement based on best-case finalization time analysis

R.J. Bril · L. Cucu-Grosjean · J. Goossens

1 Introduction

Existing end-to-end response time analysis in distributed real-time systems [2], where the
finalization of one task on a processor activates another task on another processor, is pes-
simistic. By “pessimistic” we mean that not all systems deemed to be unschedulable by the
analysis are in fact unschedulable. This pessimism has two causes: (i) the existing analysis
is based on best-case response times rather than best-case finalization times and (ii) those
best-case response times are based on analysis for (worst-case) deadlines at most equal to
periods minus (absolute) activation jitter [1]. In this paper, we present analytical means to
determine best-case finalization times of independent real-time tasks with deadlines larger
than periods minus activation jitter under uniprocessor fixed-priority preemptive scheduling
(FPPS) and arbitrary phasing, allowing an improvement of the existing analysis. We will
illustrate the improvement by means of an example.

We assume a single processor and a set $T$ of $n$ periodically released, independent tasks
$\tau_1, \tau_2, \ldots, \tau_n$ with unique, fixed priorities. At any moment in time, the processor executes
the highest priority task that has work pending, i.e. tasks are scheduled using FPPS.

Each task $\tau_i$ generates an infinite sequence of jobs $\iota_{ik}$ with $k \in \mathbb{Z}$. The inter-activation
times of $\tau_i$ are characterized by a (fixed) period $T_i \in \mathbb{R}^+$ and an (absolute) activation jitter
$A_{J_i} \in \mathbb{R}^+ \cup \{0\}$, where $A_{J_i} < T_i$. Moreover, $\tau_i$ is characterized by a (fixed) computation time
$C_i \in \mathbb{R}^+$, a phasing $\phi_i \in \mathbb{R}$, a (relative) worst-case deadline $W_{D_i} \in \mathbb{R}^+$, and a (relative)
best-case deadline $B_{D_i} \in \mathbb{R}^+ \cup \{0\}$, where $B_{D_i} \leq W_{D_i}$. The set of phasings $\phi_i$ is termed
the phasing $\phi$ of the task set $T$. We assume that we have no control over the phasing $\phi$, so any
arbitrary phasing may occur. The deadlines $B_{D_i}$ and $W_{D_i}$ are relative to the activations.

Reinder J. Bril
Technische Universiteit Eindhoven (TU/e), Mathematics and Computer Science Department, Den Dolech 2,
E-mail: R.J.Bril@tue.nl

Liliana Cucu-Grosjean
INRIA Nancy-Grand Est, TRIO team, 615 rue du Jardin Botanique, Villers les Nancy, 54600, France.
E-mail: Liliana.Cucu@loria.fr

Joël Goossens
Université Libre de Bruxelles (ULB), Computer Science Department, Boulevard du Triomphe - C.P.212, 1050
Brussels, Belgium.
E-mail: Joel.Goossens@ulb.ac.be
Note that the activations of \( \tau_i \) do not necessarily take place strictly periodically with period \( T_i \), but somewhere in an interval of length \( AJ_i \) that is repeated with period \( T_i \). The activation times \( a_{ik} \) of \( \tau_i \) satisfy \( \sup_{\ell} (a_{ik}(\varphi_i) - a_{i(\varphi_i)} - (k - \ell) T_i) \leq AJ_i \), where \( \varphi_i \) denotes the start of the interval in which job zero is activated, i.e. \( \varphi_i + k T_i \leq a_{ik} \leq \varphi_i + k T_i + AJ_i \). A task with activation jitter equal to zero is termed a \textit{strictly periodic} task.

\[ \text{Fig. 1: Basic model for a periodic task } \tau_i \text{ with (absolute) activation jitter } AJ_i. \]

The \textit{(relative) finalization time} \( F_{ik} \) of job \( \tau_{ik} \) is defined relative to the start of the interval in which \( \tau_{ik} \) is activated, i.e. \( F_{ik} = f_{ik} - (\varphi_i + k T_i) \). The \textit{active interval} of job \( \tau_{ik} \) is defined as the time span between the activation time of that job and its finalization time, i.e. \( [a_{ik}, f_{ik}] \). The \textit{response time} \( R_{ik} \) of job \( \tau_{ik} \) is defined as the length of its active interval, i.e. \( R_{ik} = f_{ik} - a_{ik} \).

Figure 1 illustrates the above basic notions for an example job of a periodic task \( \tau_i \). Whereas \( F_{ik} = R_{ik} \) holds for a strictly periodic task \( \tau_i \), the following relation holds in general

\[ F_{ik} \geq R_{ik}. \]

For notational convenience, we assume that the tasks are given in order of decreasing priority, i.e. task \( \tau_1 \) has highest priority and task \( \tau_n \) has lowest priority.

\[ \text{Legend:} \]
- \( \varphi \) (absolute) activation jitter
- \( \tau \) preemptions by higher priority tasks
- \( \ell \) execution
- \( \text{a} \) release
- \( \text{w} \) (absolute) worst-case deadline
- \( \text{r} \) (absolute) best-case deadline

\[ \text{Fig. 1: Basic model for a periodic task } \tau_i \text{ with (absolute) activation jitter } AJ_i. \]

\[ \text{2 Existing results} \]

The \textit{worst-case response time} \( WR_i \) and the \textit{best-case response time} \( BR_i \) of a task \( \tau_i \) are the largest and the smallest (relative) response time of any of its jobs, respectively, i.e. \( WR_i \overset{\text{def}}{=} \sup_{\varphi} R_{ik}(\varphi) \) and \( BR_i \overset{\text{def}}{=} \inf_{\varphi} R_{ik}(\varphi) \). For worst-case deadlines at most equal to periods minus activation jitter, i.e. \( WD_i \leq T_i - AJ_i \), \( BR_i \) is given by the \textit{largest} \( x \in \mathbb{R}^+ \) that satisfies

\[ x = C_i + \sum_{j \neq i} \left( \frac{x - AJ_j}{T_j} \right)^+ C_j. \]

Here, the notation \( w^+ \) stands for \( \max(w, 0) \), which is used to indicate that the number of preemptions of tasks with a higher priority than \( \tau_i \) can not become negative. To calculate \( BR_i \), we can use an iterative procedure based on recurrence relationships, starting with an upper bound, e.g. \( WR_i \).

The \textit{worst-case finalization time} \( WF_i \) and the \textit{best-case finalization time} \( BF_i \) of a task \( \tau_i \) are the largest and the smallest (relative) finalization time of any of its jobs, respectively, i.e. \( WF_i \overset{\text{def}}{=} \sup_{\varphi} F_{ik}(\varphi) \) and \( BF_i \overset{\text{def}}{=} \inf_{\varphi} F_{ik}(\varphi) \). The \textit{worst-case (absolute) finalization jitter} \( FJ_i \) of task \( \tau_i \) is the largest difference between the finalization times of any two of its jobs, i.e.

\[ FJ_i \overset{\text{def}}{=} \sup_{\varphi, k, \ell} (F_{ik}(\varphi) - F_{i(\varphi)}). \]
Finalization jitter analysis presented in [2] is based on $FJ_i \leq WF_i - BR_i$, where $BR_i$ is determined using Equation (2).

**3 Contributions**

From Equation (3) we derive $FJ_i \leq WF_i - BF_i$. The finalization jitter analysis presented in [2] is therefore pessimistic for two reasons; firstly $BF_i \geq BR_i$ for $AJ_i > 0$; see Equation (1) and secondly $BR_i$ as determined by Equation (2) is pessimistic for worst-case deadlines larger than periods minus activation jitter [1]. We will illustrate this by means of an example and subsequently present a conjecture for best-case finalization time analysis.

Table 1 presents the characteristics of our example task set $T_1$ consisting of three tasks, and Figure 2 shows a time-line for $T_1$ with $BF_3 = 3$ and $BR_3 = 2.4$ of task $\tau_3$, hence $BF_3 > BR_3$. Using Equation (2) yields a value $BR_3 = 2$, which is pessimistic, i.e. too small. A conjecture for exact best-case response time analysis of tasks with arbitrary deadlines that are scheduled using FPPS has been presented in [4].

<table>
<thead>
<tr>
<th>Task</th>
<th>$T$</th>
<th>$C$</th>
<th>$AJ$</th>
<th>$BF$</th>
<th>$BR$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>5</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>7</td>
<td>2</td>
<td>0.6</td>
<td>3</td>
<td>2.4</td>
</tr>
</tbody>
</table>

**Table 1** Task characteristics of $T_1$ and values for best-case finalization times and response times.

**Fig. 2** Time-line for $T_1$ with a best-case finalization time $BF_3 = 3$ and a best-case response time $BR_3 = 2.4$ for job $\tau_3$ of task $\tau_1$.

**Conjecture 1** The best-case finalization time $BF_i$ of task $\tau_i$ with $T_i - AJ_i < WD_i$ is given by

$$BF_i = \max_{0 \leq k < w_{\ell_i}} (BR'_i((k+1)C'_i) - kT_i).$$  

where $w_{\ell_i}$ is the worst-case number of jobs of $\tau_i$ in a level-$i$ active period\(^\dagger\), and $BR'_i((k+1)C'_i)$ is the best-case response time of a task $\tau'_i$ with a computation time $C'_i = (k+1)C_i$, a period equal to its worst-case deadline, i.e. $T'_i = WD'_i$, a worst-case deadline $WD'_i$ given by

$$WD'_i = WD_i + \begin{cases} 0, & k = 0 \\ kT_i - AJ_i, & \text{otherwise} \end{cases}.$$  

\(^\dagger\) A level-$i$ active period is the longest interval in which the sum of pending loads is higher than 0 for tasks with a priority equal to or higher than the priority of task $\tau_i$; see [3]. The length of the longest level-$i$ active period is finite for all $1 \leq i \leq n$ when either (i) the utilization factor $U^T = \sum_{1 \leq j \leq n} C_j$ is smaller than 1 or when (ii) $U^T$ is equal to 1, the activation jitter of all tasks of $\mathcal{T}$ are equal to zero, and the least common multiple of the periods of all tasks of $\mathcal{T}$ exists.
and a best-case deadline $BD_i$ equal to its computation time, i.e. $BD_i = (k + 1)C_i$.

Based on Conjecture 1, we find $w\ell_3 = 3$ and $BF_3 = \max(2, 9 - 7, 17 - 14) = 3$.

Note that for $w\ell_i = 1$, (4) becomes equal to the solution of (2). Hence, the conjecture therefore applies for tasks with arbitrary deadlines.

References