



**HAL**  
open science

## **Demo Abstract: Bounding Deadline Misses for Weakly-Hard Real-Time Systems Designed in CAPELLA**

Rafik Henia, Laurent Rioux, Nicolas Sordon, Zain a H Hammadeh, Rolf Ernst, Sophie Quinton

► **To cite this version:**

Rafik Henia, Laurent Rioux, Nicolas Sordon, Zain a H Hammadeh, Rolf Ernst, et al.. Demo Abstract: Bounding Deadline Misses for Weakly-Hard Real-Time Systems Designed in CAPELLA. RTAS 2017 - IEEE Real-Time and Embedded Technology and Applications Symposium, Apr 2017, Pittsburgh, United States. , pp.1-2. hal-01674754

**HAL Id: hal-01674754**

**<https://inria.hal.science/hal-01674754>**

Submitted on 3 Jan 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Bounding Deadline Misses for Weakly-Hard Real-Time Systems Designed in CAPELLA

Rafik HENIA\*, Laurent RIOUX\*, Nicolas SORDON\*, Zain A. H. HAMMADEH†, Rolf ERNST†, Sophie QUINTON‡

\*Thales Research & Technology, France, {rafik.henia, laurent.rioux, nicolas.sordon}@thalesgroup.com

†TU Braunschweig, Germany, {hammadeh, ernst}@ida.ing.tu-bs.de

‡Inria Grenoble Rhône-Alpes, France, sophie.quinton@inria.fr

Real-time systems with functional dependencies between tasks often require guarantees on end-to-end delays. For many of these systems, end-to-end deadline misses are accepted if one can limit their frequency. Such systems are called weakly-hard. Recent work has shown that typical worst-case analysis (TWCA) can compute an upper bound on the number of potential deadline misses in a sequence of activations of a task chain. In a joint collaboration between Thales and TU Braunschweig, the use of TWCA to limit the number of deadline misses in an aerial video tracking (AVT) system was evaluated. The AVT case-study, the complete automated model-based tool chain from the design environment to the timing verification using TWCA, as well as the results of the evaluation will be presented in the demonstration. The tool chain involves four tools: the design modeling tool CAPELLA extended by a performance viewpoint which allows annotating the design model with timing properties needed to perform TWCA, the pivot model TEMPO which handles mismatches between the semantics of the design model and the semantics of the model used in TWCA, the scheduling analysis tool pyCPA that performs TWCA and finally the graphical tool TimingGraphics used to visualize the TWCA results. To show the pertinence of the use of TWCA, we will also compare in the demonstration the obtained results with those obtained using worst-case analysis and simulation.

*Keywords— model-based design; model transformation; weakly-hard real-time systems; scheduling analysis;*

## I. INTRODUCTION

In real-time systems, it is not uncommon to tolerate deadline misses for task chains, if their number can be bounded. Such systems are called weakly-hard [1]. Classic worst-case analysis would not be able to take into account the weakly-hard nature of these systems, and would just reveal that some deadlines may not be met in the worst-case. This would lead to an overestimation of the required computing resources to safely run the system, which could be delicate, especially in the case of a system with a high CPU load. It is therefore important to design these systems so as to preserve weakly-hard rather than hard guarantees. TWCA can help providing such guarantees by calculating an upper bound on the number of deadline misses in a sequence of consecutive executions of a task (“at most  $m$  out of  $k$  deadlines of the task may be missed”). Recently, the TWCA approach was extended to handle end-to-end delays of task chains [2], thus making its use in industry more attractive. To evaluate the pertinence of using TWCA in industry, Thales and TU Braunschweig set up a tool-chain (Figure 1) to compute upper bounds on deadline misses in an AVT system with weakly-hard end-to-end timing

constraints. Starting from an initial system model in the model-based engineering workbench CAPELLA [3], a performance viewpoint was developed to annotate the model with various timing and scheduling properties, such as execution times, periods, priorities, weakly-hard (“ $m$  misses out of  $k$  deadlines”) constraints, etc. Then, a bridge was implemented to transform the annotated design model into a model that is analyzable by the scheduling analysis tool pyCPA [4]. The transformation is performed via the pivot model TEMPO [5] that ensures compatibility between the semantics of the design model and analysis model by performing model transformations, when needed, according to pre-defined rules. After running the analysis in pyCPA, results are sent back to CAPELLA and graphically displayed, using the tool TimingGraphics [6]. To assess the pertinence of TWCA, we have compared the results with those obtained using a Thales in-house tool for worst-case scheduling analysis and simulation, called TEMPO Verifier. All analyses and simulation are run on the same model.

In the demonstration, we will present the AVT system modeled in CAPELLA (Figure 2). Here, we will focus in particular on the implemented performance viewpoint. Then we will run the TWCA analysis in pyCPA and load the results back into CAPELLA. The whole tool chain is automated and can be run using a push button command in CAPELLA. All generated intermediate models will be presented in the demonstration. We will also compare the results with those obtained using TEMPO Verifier. The individual building blocks of the tool chain as well as the AVT case-study are briefly described in the next sections.

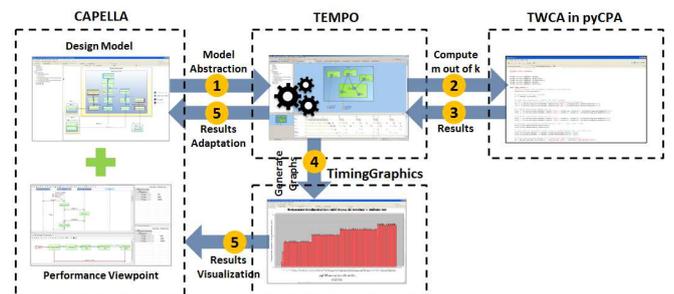


Figure 1: Tool chain to compute an upper bound on deadline misses for weakly-hard systems designed in CAPELLA

## II. AERIAL VIDEO TRACKING CASE-STUDY

The system purpose is to detect and track moving objects, e.g. vehicles on a roadway. AVT systems are mission critical real-time systems since they embed intelligence, surveillance,

reconnaissance and tactical applications characterized by hard and weakly hard constraints on timing. For simplicity's sake, the case-study we present at the demonstration is limited to two subsystems of the AVT system running on a single processor according to the static priority scheduling policy: a periodic activated video frame processing application and a sporadic activated tracking and camera control application.

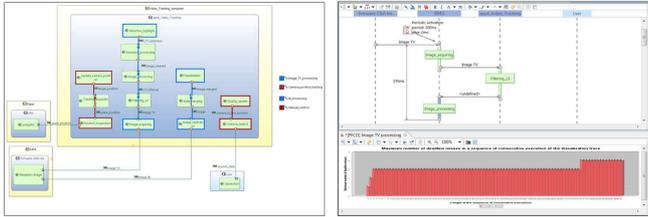


Figure 2: Design model (left fig.), performance viewpoint and TWCA result illustration (right fig.) for the aerial video tracking system in CAPELLA

### III. CAPELLA

CAPELLA is an Open Source solution hosted at [polarsys.org](http://polarsys.org). It provides a process and tooling for graphical modeling of systems, hardware or software architectures, in accordance with the principles defined by the model-based engineering method ARCADIA [7]. CAPELLA is mainly used for modeling complex and safety-critical real-time embedded systems in various industries such as aerospace, avionics, transportation and automotive. CAPELLA can be specialized, according to the concept of "Viewpoint", to manage information specific to a particular engineering need, e.g. non-functional properties such as timing performance.

### IV. PERFORMANCE VIEWPOINT

In order to apply scheduling analysis to any design model, the latter must be annotated with information describing the timing characteristics of tasks (e.g. execution times, activation periods, etc.), the behavior of the application (e.g. functional dependencies between tasks, end-to-end deadlines, etc.) and the characteristics of the hardware platform (e.g. CPU speed, scheduling policy, etc.). The UML Profile for MARTE [8] is a key technology for this purpose. It allows extending design models with concepts modeling real-time behavior, thus making possible the extraction of a scheduling analysis model from a design model. We therefore selected concepts from MARTE to develop the CAPELLA performance viewpoint. In addition to these concepts, the performance viewpoint we developed includes concepts specific to weakly-hard systems and required by the TWCA, in particular the « at most  $m$  deadline misses out of  $k$  consecutive executions » constraint.

### V. PIVOT MODEL TEMPO

Scheduling analysis is very often not directly applicable to conceptual design models due to the semantic mismatch between the latter and the variety of scheduling analysis models known from the classical real time systems research and represented by academic [4][9] and commercial tools [10]. For instance, in the common scheduling analysis models, a standard assumption is that a task writes its output data at the end of its execution. This is not always the case in design

models, since very often, calls are either synchronous (blocking) or asynchronous (non-blocking). As a consequence, a task may write data into the input of a connected task at any instant during its execution and not necessarily at the end. TEMPO allows adapting design models to the semantics of the chosen scheduling analysis models through a set of transformation rules. Such transformations preserve the timing behavior modeled in the conceptual design. After performing scheduling analysis, the obtained results are, in turn, adapted back to the semantics of the design model.

### VI. TYPICAL WORST-CASE ANALYSIS IN PYCPA

TWCA is a technique to compute an upper bound on the number of deadline misses in a sequence of consecutive executions of a task. Recently TWCA was extended at TU Braunschweig to compute an upper bound on the number of deadline misses in any sequence of consecutive executions of a task chain under fixed priority preemptive scheduling in a uniprocessor system [2]. The extended TWCA is implemented as a plugin for the scheduling analysis tool pyCPA.

### VII. TIMINGGRAPHICS

TimingGraphics is a visualization tool allowing the illustration of the temporal behavior of real-time systems through a set of graphs [6]. In the context of this work, Gantt-charts and histograms were used respectively to visualize the worst-case end-to-end delay scenarios of the task chains, as well as illustrate the maximum number of end-to-end deadline misses as a function of the number of consecutive executions.

### VIII. DEMONSTRATION

The evaluation of TWCA that we propose and show in this demonstration is the result of a fruitful collaboration between industry and academia, namely Thales and TU Braunschweig. It represents a contribution to the industrial exploitation of model-driven technologies and scheduling analysis in the design of weakly-hard real-time systems. An industrial case-study is available as hands-on demonstration of the quality of the proposed technical solution.

### REFERENCES

- [1] G. Bernat, A. Burns, and A. Llamas. Weakly hard real-time systems. *IEEE Trans. Computers*, 50(4):308–321, 2001.
- [2] Z. A. H. Hammadeh, S. Quinton, R. Henia, L. Rioux and R. Ernst. Bounding Deadline Misses in Weakly-Hard Real-Time Systems with Task Dependencies. In *Proc. of DATE*, March 2017
- [3] CAPELLA: <http://www.polarsys.org/capella/>
- [4] PyCPA: J. Diemer, P. Axer, and R. Ernst. Compositional performance analysis in python with pycpa. In *Proc. of WATERS*, July 2012.
- [5] R. Henia, L. Rioux, N. Sordon, G. E. Garcia and M. Panunzio. Integrating Formal Timing Analysis in the RealTime Software Development Process. In *Proc. of WOSP*, Januar 2015.
- [6] TimingGraphics: <http://www.linkconet.com/en/TimingGraphics.html>
- [7] ARCADIA : <http://polarsys.org/capella/arcadia.html>
- [8] MARTE: <http://www.omg.org/omgmarte/>
- [9] M. González Harbour, J.J. Gutiérrez, J.C.Palencia and J.M.Drake. MAST: Modeling and Analysis Suite for Real - Time Applications. In *Proc. of ECRTS*, June 2001.
- [10] SymTA/S: <https://www.symtavision.com/symtas.html>.