RESEARCH CENTRE

Inria Paris Centre

2023
ACTIVITY REPORT

Project-Team

KOPERNIC

Keeping worst case reasoning for different criticalities

DOMAIN

Algorithmics, Programming, Software and Architecture

THEME

Embedded and Real-time Systems

Contents

Project-Team KOPERNIC

Creation of the Project-Team: 2021 October 01

Keywords

Computer sciences and digital sciences

A1.1.1. – Multicore, Manycore

A1.5. – Complex systems

A1.5.1. – Systems of systems

A1.5.2. – Communicating systems

A2.3. – Embedded and cyber-physical systems

A2.3.1. – Embedded systems

A2.3.2. – Cyber-physical systems

A2.3.3. – Real-time systems

A2.4.1. – Analysis

Other research topics and application domains

B5.2. – Design and manufacturing

B5.2.1. – Road vehicles

B5.2.2. – Railway

B5.2.3. – Aviation

B5.2.4. – Aerospace

B6.6. – Embedded systems

1 Team members, visitors, external collaborators

Research Scientist

- Liliana Cucu [Team leader, INRIA, Senior Researcher, HDR]

Faculty Member

- Avner Bar-Hen [CNAM, Professor Delegation, until Mar 2023, HDR]

PhD Students

- Chiara Daini [INRIA, until Apr 2023]
- Ismail Hawila [StatInf, CIFRE]
- Marwan Wehaiba El Khazen [Statinf, CIFRE]
- Kevin Zagalo [INRIA, until Sep 2023]

Technical Staff

- Masum Bin Alam [INRIA, Engineer, from Nov 2023]
- Rihab Bennour [INRIA, Engineer, until May 2023]

Interns and Apprentices

- Marc-Antoine Auvray [INRIA, Apprentice, until Aug 2023]
- Myriam Mabrouki [INRIA, Intern, from Jul 2023 until Aug 2023]

Administrative Assistants

- Nelly Maloisel [INRIA]
- Christelle Rosello [INRIA, from Apr 2023]

External Collaborators

- Slim Ben Amor [StaInf]
- Adriana Gogonel [Statinf]
- Kossivi Kougblenou [StatInf]
- Yves Sorel [INRIA]

2 Overall objectives

The Kopernic members are focusing their research on studying **time for embedded communicating systems**, also known as cyber-physical systems. More precisely, the team proposes a **system-oriented solution** to the problem of studying time properties of the cyber components of a CPS. The solution is expected to be obtained by composing probabilistic and non-probabilistic approaches for CPSs. Moreover, statistical approaches are expected to validate existing hypotheses or propose new ones for the models considered by probabilistic analyses [3, 4].

The term cyber-physical systems refers to a new generation of systems with integrated computational and physical capabilities that can interact with humans through many new modalities [15]. A defibrillator, a mobile phone, an autonomous car or an aircraft, they all are CPSs. Beside constraints like power consumption, security, size and weight, CPSs may have cyber components required to fulfill their functions within a limited time interval (a.k.a. time dependability), often imposed by the environment, e.g., a physical process controlled by some cyber components. The appearance of communication channels between cyber-physical components, easing the CPS utilization within larger systems, forces cyber components with high criticality to interact with lower criticality cyber components. This interaction is completed by external events from the environnement that has a time impact on the CPS. Moreover, some programs of the cyber components may be executed on time predictable processors and other programs on less time predictable processors.

Different research communities study separately the three design phases of these systems: the modeling, the design and the analysis of CPSs [27]. These phases are repeated iteratively until an appropriate solution is found. During the first phase, the behavior of a system is often described using model-based methods. Other methods exist, but model-driven approaches are widely used by both the research and the industry communities. A solution described by a model is proved (functionally) correct usually by a formal verification method used during the analysis phase (third phase described below).

During the second phase of the design, the physical components (e.g., sensors and actuators) and the cyber components (e.g., programs, messages and embedded processors) are chosen often among those available on the market. However, due to the ever increasing pressure of smartphone market, the microprocessor industry provides general purpose processors based on multicore and, in a near future, based on manycore processors. These processors have complex architectures that are not time predictable due to features like multiple levels of caches and pipelines, speculative branching, communicating through shared memory or/and through a network on chip, internet, etc. Due to the time unpredictability of some processors, nowadays the CPS industry is facing the great challenge of estimating worst case execution times (WCETs) of programs executed on these processors. Indeed, the current complexity of both processors and programs does not allow to propose reasonable worst case bounds. Then, the phase of design ends with the implementation of the cyber components on such processors, where the models are transformed in programs (or messages for the communication channels) manually or by code generation techniques [18].

During the third phase of analysis, the correctness of the cyber components is verified at program level where the functions of the cyber component are implemented. The execution times of programs are estimated either by static analysis, by measurements or by a combination of both approaches [36].

These WCETs are then used as inputs to scheduling problems [29], the highest level of formalization for verifying the time properties of a CPS. The programs are provided a start time within the schedule together with an assignment of resources (processor, memory, communication, etc.). Verifying that a schedule and an associated assignment are a solution for a scheduling problem is known as a schedulability analysis.

The current CPS design, exploiting formal description of the models and their transformation into physical and cyber parts of the CPS, ensures that the functional behavior of the CPS is correct. Unfortunately, there is no formal description guaranteeing today that the execution times of the generated programs is smaller than a given bound. Clearly all communities working on CPS design are aware that **computing takes time** [26], but there is **no CPS solution guaranteeing time predictability** of these systems as **the processors appear late within the design phase** (see Figure 1). Indeed, the choice of the processor is made at the end of the CPS design process, after writing or generating the programs.

Since the processor appears late within the CPS design process, the CPS designer in charge of estimating the worst case execution time of a program or analyzing the schedulability of a set of programs inherits a difficult problem. The Kopernic main purpose is the proposition of compositional rules with respect to the time behaviour of a CPS, allowing to restrain the CPS design to analyzable instances of the WCET estimation problem and of the schedulability analysis problem.

With respect to the WCET estimation problem, we say that a rule $\circ$ is compositional for any two sets of measured execution times $\mathscr{C}_1$ and $\mathscr{C}_2$ of a program A, and a WCET statistical estimator p, if we obtain a safe WCET estimation for A from $p(\mathscr{C}_1 \circ \mathscr{C}_2)$. For instance, $\mathscr{C}_1$ may be the set of measured execution times of the program A while all processor features except the local cache L1 are deactivated, while $\mathscr{C}_2$ is obtained, similarly, with a shared L2 cache activated. We consider that the variation of all input variables of the program A follows the same sequence of values, when measuring the execution time of the program p. With respect to the schedulability analysis problem, we are interested in analyzing graphs of communicating programs. A program A communicates with a program B if input variables of

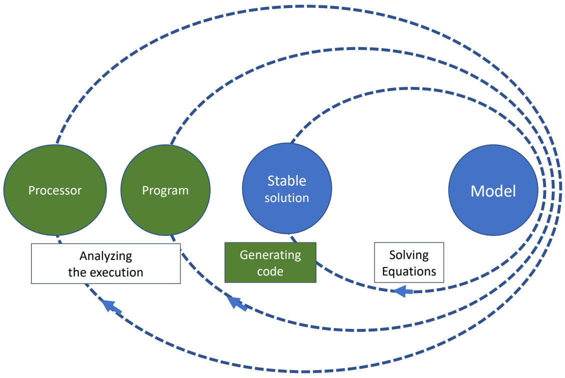

Figure 1: The CPSs design: from models towards analyzing the time properties of the cyber components

the program B are among output variables of the program A. A graph of communicating programs is a direct acyclic graph with programs as vertices. An edge from a program A to program B is defined if A communicates with B. The end to end response time of such graph is the longest path from any source vertex to any sink vertex of the graph, if there is, at least one path between these two vertices. A rule $\bigodot$ is compositional for any set of measured response times $\mathscr{R}_A$ of program A, any set of measured response times $\mathscr{R}_B$ and a schedulability analysis S if we obtain a safe schedulability analysis from $S(\mathscr{R}_A \bigodot \mathscr{R}_B)$.

Before enumerating our scientific objectives, we introduce the concept of variability factors. More precisely, the time properties of a cyber component are subject to variability factors. We understand by variability the distance between the smallest value and the largest value of a time property. With respect to the time properties of a CPS, the factors may be classified in three main classes:

- program structure: for instance, the execution time of a program that has two main branches is obtained, if appropriate composition principles apply, as the maximum between the largest execution time of each branch. In this case the branch is a variability factor on the execution time of the program;
- processor structure: for instance, the execution time of a program on a less predictable processor (e.g., one core, two levels of cache memory and one main memory) will have a larger variability than the execution time of the same program executed on a more predictable processor (e.g., one core, one main memory). In this case the cache memory is a variability factor on the execution time of the program;
- execution environment: for instance, the appearance of a pedestrian in front of a car triggers the execution of the program corresponding to the brakes in an autonomous car. In this case the pedestrian is a variability factor for the triggering of the program.

We identify three main scientific objectives to validate our research hypothesis. The three objectives are presented from program level, where we use statistical approaches, to the level of communicating programs, where we use probabilistic and non-probabilistic approaches.

The Kopernic scientific objectives are:

- **[O1] worst case execution time estimation of a program** - modern processors induce an increased variability of the execution time of programs, making difficult (or even impossible) a complete static analysis to estimate such worst case. Our objective is to propose a solution composing probabilistic and non-probabilistic approaches based both on static and on statistical analyses by answering the following **scientific challenges**:

  1. **a classification of the variability factors of execution times** of a program with respect to the processor features. The difficulty of this challenge is related to the definition of an element belonging to the set of variability factors and its mapping to the execution time of the program.

2. **a compositional rule** of statistical models associated to each variability factor. The difficulty of this challenge comes from the fact that a global maximum of a multicore processor cannot be obtained by upper bounding the local maxima on each core.

- **[O2] deciding the schedulability of all programs running within the same cyber component, given an energy budget** - in this case the programs may have different time criticalities, but they share the same processor, possibly multicore[1]. Our objective is to propose a solution composing probabilistic and non-probabilistic approaches based on answers to the following **scientific challenges**:

  1. **scheduling algorithms taking into account the interaction between different variability factors**. The existence of time parameters described by probability distributions imposes to answer to the challenge of revisiting scheduling algorithms that lose their optimality even in the case of an unicore processor [30]. Moreover, the multicore partionning problem is recognized difficult for the non-probabilistic case [34];
  2. **schedulability analyses** based on the algorithms proposed previously. In the case of predictable processors, the schedulability analyses accounting for operating systems costs increase the time dependability of CPSs [32]. Moreover, in presence of variability factors, the composition property of non-probabilistic approaches is lost and new principles are required.

- **[O3] deciding the schedulability of all programs communicating through predictable and non-predictable networks, given an energy budget** - in this case the programs of the same cyber component execute on the same processor and they may communicate with the programs of other cyber components through networks that may be predictable (network on chip) or non-predictable (internet, telecommunications). Our objective is to propose a solution to this challenge by analysing schedulability of programs, for which existing (worst case) probabilistic solutions exist [31], communicating through networks, for which probabilistic worst-case solutions [19] and average solutions exist [28].

3 Research program

The research program for reaching these three objectives is organized according three main research axes

- Worst case execution time estimation of a program, detailed in Section 3.1;
- Building measurement-based benchmarks, detailed in Section 3.2;
- Scheduling of graph tasks on different resources within an energy budget in Section 3.3.

3.1 Worst case execution time estimation of a program

The temporal study of real-time systems is based on the estimation of the bounds for their temporal parameters and more precisely the WCET of a program executed on a given processor. The main analyses for estimating WCETs are static analyses [36], dynamic analyses [20], also called measurement-based analyses, and finally hybrid analyses that combine the two previous ones [36].

The Kopernic approach for solving the WCET estimation problem is based on (i) the identification of the impact of variability factors on the execution of a program on a processor and (ii) the proposition of compositional rules allowing to integrate the impact of each factor within a WCET estimation. Historically, the real-time community had illustrated the distribution of execution times for programs as heavy-tailed ones as intuitively the large values of execution times of programs are agreed to have a low probability of appearance. For instance Tia et al. are the first underlining this intuition within a paper introducing execution times described by probability distributions within a single core schedulability analysis [35]. Since [35], a low probability is associated to large values of execution times of a program executed on a single core processor. It is, finally, in 2000 that the group of Alan Burns, within the thesis of Stewart Edgar [21], formalizes this property as a conjecture indicating that a maximal bound on the execution times of a program may be estimated by the Extreme Value Theory [24]. No mathematical definition of

[1] This case is referred as a mixed criticality approach.

what represents this bound for the execution time of a program has been proposed at that time. Two years later, a first attempt to define this bound has been done by Bernat et al. [17], but the proposed definition is extending the static WCET understanding as a combination of execution times of basic blocks of a program. Extremely pessimistic, the definition remains intuitive, without associating a mathematical description. After 2013, several publications from Liliana Cucu-Grosjean's group at Inria Nancy introduce a mathematical definition of a probabilistic worst-case execution time, respectively, probabilistic worst-case response time, as an appropriate formalization for a correct application of the Extreme Value Theory to the real-time problems [2, 1, 5].

We identify the following open research problems related to the first research axis:

1. the generalization of modes analysis to multi-dimensional, each dimension representing a program when several programs cooperate;

2. the proposition of a rules set for building programs that are time predictable for the internal architecture of a given single core and, then, of a multicore processor;

3. modeling the impact of processor features on the energy consumption to better consider both worst case execution time and schedulability analyses considered within the third research axis of this proposal.

3.2 Building measurement-based benchmarks

The real-time community is facing the lack of benchmarks adapted to measurement-based analyses. Existing benchmarks for the estimation of WCET [33, 25, 22] have been used to estimate WCETs mainly for static analyses. They contain very simple programs and are not accompanied by a measurement protocol. They do not take into account functional dependencies between programs, mainly due to shared global variables which, of course, influence their execution times. Furthermore, current benchmarks do not take into account interferences due to the competition for resources, e.g., the memory shared by the different cores in a multicore. On the other hand, measurement-based analyses require execution times measured while executing programs on embedded processors, similar to those used in the embedded systems industry. For example, the mobile phone industry uses multicore based on non predictable cores with complex internal architecture, such as those of the ARM Cortex-A family. In a near future, these multicore will be found in critical embedded systems found in application domains such as avionics, autonomous cars, railway, etc., in which the team is deeply involved. This increases dramatically the complexity of measurement-based analyses compared to analyses performed on general purpose personal computers as they are currently performed.

We understand by measurement-based benchmarks a 3-uple composed by a program, a processor and a measurement protocol. The associated measurement protocols should detail the variation of the input variables (associated to sensors) of these benchmarks and their impact on the output variables (associated to actuators), as well as the variation of the processor states.

Proposing reproducibility and representativity properties that measurement-based benchmarks should follow is the strength of this research axis [7]. We understand by the reproducibility, the property of a measurement protocol to provide the same ordered set of execution times for a fixed pair (program, processor). We understand by the representativity, the existence of a (sufficiently small) number of values for the input variables allowing a measurement protocol to provide an ordered set of execution times that ensure a convergence for the Extreme Value Index estimators.

Within this research axis we identify the following open problems:

1. proving reproducibility and representativity properties while extending current benchmarks from predictable unicore processors (e.g., ARM Cortex-M4) to non predictable ones (e.g., ARM Cortex-A53 or Cortex-A7);

2. proving reproducibility and representativity properties while extending unicore benchmarks to multicore processors. In this context, we face the supplementary difficulty of defining the principles that an operating system should satisfy in order to ensure a real-time behaviour.

3.3 Scheduling of graph tasks on different resources within an energy budget

Following the model-driven approach, the functional description of the cyber part of the CPS, is performed as a graph of dependent functions, e.g., a block diagram of functions in Simulink, the most widely used modeling/simulation tool in industry. Of course, a program is associated to every function. Since the graph of dependent programs becomes a set of dependent tasks when real-time constraints must be taken into account, we are facing the problem of verifying the schedulability of such dependent task sets when it is executed on a multicore processor.

Directed Acyclic Graphs (DAG) are widely used to model different types of dependent task sets. The typical model consists of a set of independent tasks where every task is described by a DAG of dependent sub-tasks with the same period inherited from the period of each task [16]. In such DAG, the sub-tasks are vertices and edges are dependencies between sub-tasks. This model is well suited to represent, for example, the engine controller of a car described with Simulink. The multicore schedulability analysis may be of two types, global or partitionned. To reduce interference and interactions between sub-tasks, we focus on partitioned scheduling where each sub-task is assigned to a given core [23, 6, 8].

In order to propose a general DAG task model, we identify the following open research problems:

1. solving the schedulability problem where probabilistic DAG tasks are executed on predictable and non predictable processors, and such that some tasks communicate through predictable networks, e.g., inside a multicore or a manycore processor, and non-predictable networks, e.g., between these processors through internet. Within this general schedulability problem; we consider five main classes of scheduling algorithms that we adapt to solve probabilistic DAG task scheduling problems. We compare the new algorithms with respect to their energy-consumption in order to propose new versions with a decreased energy consumption by integrating variation of frequencies for processor features like CPU or memory accesses.

2. the validation of the proposed framework on our multicore drone case study. To answer to the challenging objective of proposing time predictable platforms for drones, we currently migrate the PX4-RT programs on heterogeneous architectures. This includes an implementation of the scheduling algorithms detailed within this research axis within current operating system, NuttX[2].

4 Application domains

4.1 Avionics

Time critical solutions in this context are based on temporal and spatial isolation of the programs and the understanding of multicore interferences is crucial. Our contributions belong mainly to the solutions space for the objective [O1] identified previously.

4.2 Railway

Time critical solutions in this context concern both the proposition of an appropriate scheduler and associated schedulability analyses. Our contributions belong to the solutions space of problems dealt within objectives [O1] and [O2] identified previously.

4.3 Autonomous cars

Time critical solutions in this context concern the interaction between programs executed on multicore processors and messages transmitted through wireless communication channels. Our contributions belong to the solutions space of all three classes of problems dealt within all three Kopernic objectives identified previously.

[2] For more details, see the NuttX webpage

4.4 Drones

As it is the case of autonomous cars, there is an interaction between programs and messages, suggesting that our contributions in this context belong to the solutions space of all three classes of problems dealt within the objectives identified previously.

5 Social and environmental responsibility

5.1 Impact of research results

The Kopernic members provide theoretical means to decrease both the processor utilization and the energy consumption. Such gain is estimated within 30% to 60% utilization gain for existing architectures or energy consumption for new architectures by decreasing the number of necessary cores.

6 Highlights of the year

6.1 Awards

Liliana Cucu-Grosjean and Adriana Gogonel have been nominated among the 100 most innovative French inventors of the year 2023 by the newpaper La Tribune (France). Liliana Cucu-Grosjean has been nominated among the 100 most influent Romanian in 2023 by the newpaper Newsweek (Romania). StatInf, a spin-off of the Kopernic team has been awarded within the TechForFuture contest in Paris at the category "Industrie du Futur"

6.2 Keynotes

Liliana Cucu-Grosjean has gave a keynote entitled "Probabilities – a means to gain time and space when designing CPS" at the 2023 edition of the Hipeac conference

6.3 Nominations

Liliana Cucu-Grosjean has been nominated to run for the election and have been elected vice-chair of the IEEE Technical Community on Real-Time Systems (TCRTS). She has started her office on January, 1st of 2024 as the first female vice-chair of this IEEE committee after more than 40 years of existence. According to the IEEE TCRTS rules, she becomes chair of this committee on January, 1st, 2026 for a period of two years.

7 New software, platforms, open data

7.1 New software

7.1.1 SynDEx

Keywords: Distributed, Optimization, Real time, Embedded systems, Scheduling analyses

Scientific Description: SynDEx is a system level CAD software implementing the AAA methodology for rapid prototyping and for optimizing distributed real-time embedded applications. It is developed in OCaML.

Architectures are represented as graphical block diagrams composed of programmable (processors) and non-programmable (ASIC, FPGA) computing components, interconnected by communication media (shared memories, links and busses for message passing). In order to deal with heterogeneous architectures it may feature several components of the same kind but with different characteristics.

Two types of non-functional properties can be specified for each task of the algorithm graph. First, a period that does not depend on the hardware architecture. Second, real-time features that depend on the different types of hardware components, ranging amongst execution and data transfer time, memory, etc.. Requirements are generally constraints on deadline equal to period, latency between any pair of tasks in the algorithm graph, dependence between tasks, etc.

Exploration of alternative allocations of the algorithm onto the architecture may be performed manually and/or automatically. The latter is achieved by performing real-time multiprocessor schedulability analyses and optimization heuristics based on the minimization of temporal or resource criteria. For example while satisfying deadline and latency constraints they can minimize the total execution time (makespan) of the application onto the given architecture, as well as the amount of memory. The results of each exploration is visualized as timing diagrams simulating the distributed real-time implementation.

Finally, real-time distributed embedded code can be automatically generated for dedicated distributed real-time executives, possibly calling services of resident real-time operating systems such as Linux/RTAI or Osek for instance. These executives are deadlock-free, based on off-line scheduling policies. Dedicated executives induce minimal overhead, and are built from processor-dependent executive kernels. To this date, executives kernels are provided for: TMS320C40, PIC18F2680, i80386, MC68332, MPC555, i80C196 and Unix/Linux workstations. Executive kernels for other processors can be achieved at reasonable cost following these examples as patterns.

Functional Description: Software for optimising the implementation of embedded distributed real-time applications and generating efficient and correct by construction code

URL: http://www.syndex.org

Contact: Yves Sorel

Participant: Yves Sorel

7.1.2 EVT Kopernic

Keywords: Embedded systems, Worst Case Execution Time, Real-time application, Statistics

Scientific Description: The EVT-Kopernic tool is an implementation of the Extreme Value Theory (EVT) for the problem of the statistical estimation of worst-case bounds for the execution time of a program on a processor. Our implementation uses the two versions of EVT - GEV and GPD - to propose two independent methods of estimation. Their results are compared and only results that are sufficiently close allow to validate an estimation. Our tool is proved predictable by its unique choice of block (GEV) and threshold (GPD) while proposant reproducible estimations.

Functional Description: EVT-Kopernic is tool proposing a statistical estimation for bounds on worst-case execution time of a program on a processor. The estimator takes into account dependences between execution times by learning from the history of execution, while dealing also with cases of small variability of the execution times.

URL: http://www.statinf.fr

Contact: Adriana Gogonel

Participants: Adriana Gogonel, Liliana Cucu

7.2 Open data

The Kopernic members contribute to the effort of reproducing research results and numerical evaluation by proposing dynamic benchmarks for statistical analysis of measured execution times, memory accesses and other traces obtained during the Hardware in the Loop execution of the open source programs PX4-RT. A set of measurements obtained from the execution of the PX4-RT programs on the Pixhawk board are shared with the community under the name of KDBench, a.k.a, Kopernic Dynamic Benchmarks. The data related to KDBench are available at https://team.inria.fr/kopernic/kdbench/ and the contact person is Liliana Cucu-Grosjean.

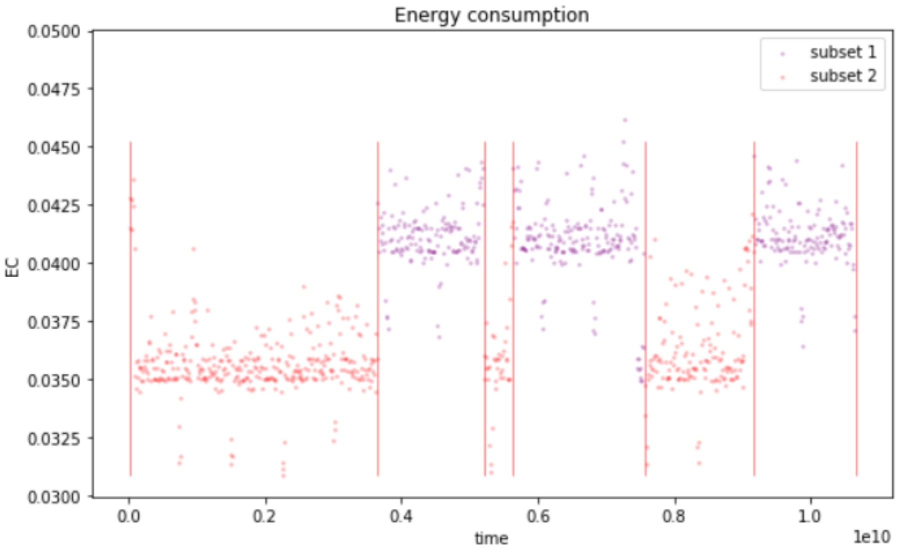

Figure 2: Different execution profiles of the Dec program on an ARM microcontroller

8 New results

During this year, the results of Kopernic members have covered all Kopernic three research axes.

8.1 Worst case execution time estimation of a program

Participants: Slim Ben Amor, Rihab Bennour, Liliana Cucu-Grosjean, Adriana Gogonel, Kossivi Kougblenou, Marwan Wehaiba El Khazen.

We consider WCET statistical estimators that are based on the utilization of the Extreme Value Theory [24]. Compared to existing methods [36], our results require the execution of the program under study on the targeted processor or at least a cycle-accurate simulator of this processor. We concentrate on the separation of hardware and software impacts [9], where the correct application of the Extreme Value Theory brings an important support [10].

The originality of this year results concerns the migration from the WCET estimation to the estimation of the worst-case energy consumption [12]. More precisely, we propose a new statistical model introducing the impact of both software and hardware events on the estimation of worst-case energy consumption of programs on embedded processors. In order to achieve this purpose, we build a framework to better understand the representativeness of measurements with respect to both software and hardware events. During this year, we test this framework on both execution times and energy consumption data for existing benchmarks as a first step towards a complete statistical worst-case energy consumption model. We denote by WCEC the worst-case energy consumption of a program.

Another original aspect of this work is the definition of critical paths in the context of worst-case statistical estimations. Indeed, while a static analysis calculates a worst-case value based on a critical path, in statistical-based estimators we define a similar concept of critical paths. For a program A executed on a processor Π and a measurement protocol $\mathscr{M}$, we obtain an ordered sequences of paths $\left(P_j\right)_j$ and of execution times or energy consumption values denoted by V^*, depending on the choice of target variable (WCET or WCEC).

The Worst-Case Execution Time of a program A is defined as its largest execution time for any valid execution scenario S, while the probabilistic worst-case execution time (pWCET) $\mathscr{C}_{\mathscr{T}}$ of a program is an upper bound on all possible probabilistic execution times $\mathscr{C}_{\mathscr{T}i}$ for all possible execution scenarios $S_i, \forall i \geq 1$ (Each scenario S_i thus defines a probability distribution of execution times $\mathscr{C}_{\mathscr{T}i}$). The relation

$\geq$ describes the relation between the probabilistic execution times (pET) of a program and its pWCET, $\mathscr{C}_{\mathscr{T}} \geq \mathscr{C}_{\mathscr{T}\,i}, \forall i$, defined as follows. One writes $\mathscr{C}_{\mathscr{T}} \geq \mathscr{C}_{\mathscr{T}\,i}$ or $\mathscr{C}_{\mathscr{T}}$ is said to be worse than $\mathscr{C}_{\mathscr{T}\,i}$ if its complementary cumulative distribution function (survival function 1-CDF) has a higher or equal probability associated to each possible value, i.e., $P(\mathscr{C}_{\mathscr{T}} \geq c) \geq P(\mathscr{C}_{\mathscr{T}\,i} \geq c), \forall c$ and $\forall i \geq 1$.

Based on the previous (p)WCET definitions, we may define the analogous concepts of the worst case energy consumption (WCEC) and of the probabilistic WCEC (pWCEC) of a program as follows. A **Worst-Case Energy Consumption** of a program is its largest energy consumption during the execution of that program for any valid execution scenario, while the probabilistic worst-case energy consumption $\mathscr{C}_{\mathscr{E}}$ of a program is an upper bound on all possible probabilistic energy consumption profiles $\mathscr{C}_{\mathscr{E}\,i}$ for all possible execution scenarios $S_i, \forall i \geq 1$. The relation $\geq$ describes, as with pETs, the relation between the probabilistic energy consumption (pEC) of a program and its pWCEC, such that $\mathscr{C}_{\mathscr{E}} \geq \mathscr{C}_{\mathscr{E}\,i}, \forall i$. We say that a path P_j of a program A is critical with respect to a pWCET estimation of A if, at least, one measurement of the execution of that path appears within the set of measurements building the pWCET estimate of A. Similarly, we define critical paths wrt the pWCEC estimation. A path P_j of a program A is critical with respect to a pWCEC estimation of A if, at least, one measurement of the energy consumption associated to the execution of that path appears within the set of measurements building the pWCEC estimate of A. For instance, in Figure 2, we illustrate for the program Dec [25] executed on an ARM microcontroller, by two colors different measured energy consumption, where points with the same color identify paths that are equivalent with respect to the worst-case energy consumption estimation.

8.2 Building measurement-based benchmarks: KDBench

Participants: Slim Ben Amor, Rihab Bennour, Masum Bin Alam, Liliana Cucu-Grosjean, Ismail Hawila, Yves Sorel, Marwan Wehaiba El Khazen.

KDBench, our measurement-based benchmarks, are obtained by modifying open-source programs of the autopilot PX4 designed to control a wide range of air, land, sea and underwater drones. More precisely, the studied programs are executed on a standard Pixhawk drone board based on a predictable single core processor ARM Cortex-M4 and during the CEOS project we have transformed this set of programs into a set of dependent tasks that satisfies real-time constraints, leading to a new version of PX4, called PX4-RT. As usual, the set of dependent real-time tasks is defined by a data dependency graph. An interested reader may refer to the web page of the Kopernic team at The KDBench website.

During this year, the data associated to the KDBench has been collected from 3 flights according to different scenarios, where we vary from one scenario to another the periods of programs. We consider 4 different scenarios per flight and we obtain 12 sets of collected data, 4 sets for each flight. Moreover, different data are extracted from the benchmark after the flight completes.
We collect data from three different flight that we generate by using the QGroundControl software which provides planning for MAVlink enabled drones. The three flights are described as follows:

1. during the first flight *short_flight*, the drone follows a simple line trajectory and the duration of the flight is 23s,

2. for the second *medium_flight* we choose a flight with a sudden change of the drone orientation, in addition to other waypoints where we change the altitude. The duration of this flight is around 43s,

3. the third *long_flight* has more waypoints than the second flight and its duration is 2min 48s.

More precisely, we have considered the execution of 9 real-time tasks (or programs) of PX4-RT. Their list is detailed in the table below together with their periods of activation. In the first line of this table, tasks are ordered according to their priority from the highest priority (on the left of the table) to the lowest priority (on the right of the table). For instance, the task Sensors has the highest priority while Commander has the lowest priority. The last 4 lines of this table contain the periods of activation of each task within the 4 considered measuring flight scenarios, where the periods T_i are indexed according to the scenario number i.

Periods Tasks	Sensors	EKF2	Rate	Attitude	Position	Flight manager	Hover thrust	Nav	Cmd
T_1(ms)	10	10	15	15	15	15	15	25	50
T_2 (ms)	4	4	4	4	4	4	4	4	4
T_3(ms)	8	8	8	8	8	8	8	8	8
T_4(ms)	12	12	12	12	12	12	12	12	12

Table 1: List of real-time control tasks with their associated periods

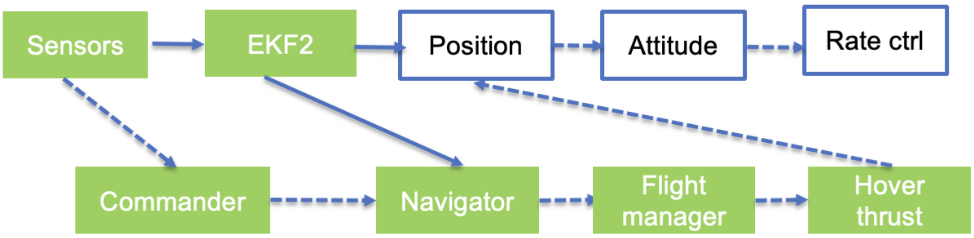

Figure 3: Data and precedence constraints represented by a DAG-based task model for KDBench programs

8.3 Scheduling of graph tasks on different resources within an energy budget

Participants: Slim Ben Amor, Liliana Cucu-Grosjean, Ismail Hawila, Kossivi Kougblenou, Yves Sorel, Kevin Zagalo.

Due to widespread of multicore processors on embedded and real-time systems, we concentrate our work on the study of the schedulability of graph tasks on such processors. We consider preemptive (both global and partitionned) fixed-priority scheduling policies. We monitor, when possible, the energy consumption required to meet deadlines as a metric comparing the efficiency of scheduling policies.

Given the difficulty of our scheduling problem, we have considered the single processor case for of independent tasks for which a feasibility interval has been proved [11, 14]. To this problem we have added the precedence constraints and we study the specific case of control real-time tasks as an important mean to understand how probabilities may be associated to the inter-arrival times between consecutive instances of the same program or task as well as understanding the motivation for precedence constraints between tasks and/or their instances.

With respect to the proposition of a new task model merging scheduling and WCET concerns for real-time control systems, our purpose is that this new task model fulfills the following expectations:

1. the data and/or precedence constraints between different control tasks as well as non-control tasks are considered;
2. the impact of data constraints on the execution times of control tasks is considered;
3. the impact of period variation between dependent control tasks is considered.

Such expectations applied to the PX4-RT programs require to indicate the appropriate precendence constraints. In Figure 3, we represent by continuous lines edges that depict precedence constraints between tasks and by dotted lines edges that depict the data constraints for the PX4-RT program [13].

9 Bilateral contracts and grants with industry

9.1 CIFRE Grant funded by StatInf

Participants: Liliana Cucu-Grosjean, Adriana Gogonel, Marwan Wehaiba El Khazen.

A CIFRE agreement between the Kopernic team and the start-up StatInf has started on October 1st, 2020. Its funding is related to study the evolution of WCET models to consider the energy consumption according to Kopernic research objectives.

9.2 CIFRE Grant funded by StatInf

Participants: Liliana Cucu-Grosjean, Slim Ben Amor, Ismail Hawila, Yves Sorel.

A CIFRE agreement between the Kopernic team and the start-up StatInf has started on October 1st, 2022. Its funding is related to study the relation between the control theory robustness and the schedulability problem using probabilistic descriptions according to Kopernic research objectives.

10 Partnerships and cooperations

10.1 International initiatives

10.1.1 Associate Teams in the framework of an Inria International Lab or in the framework of an Inria International Program

Kepler

Participants: Slim Ben Amor, Liliana Cucu-Grosjean, Adriana Gogonel, Ismail Hawila, Kossivi Kougblenou, Yves Sorel, Marwan Wehaiba.

Title: Probabilistic foundations for time, a key concept for the certification of cyber-physical systems

Partner Institution(s):
- Universidade Federal da Bahia (Brezil)

Since 2020/5 years

10.2 International research visitors

10.2.1 Visits to international teams

Research stays abroad Marwan Wehaiba has visited the UFBA team during November 2023 while working on the results presented in [12].

10.2.2 Other european programs/initiatives

The Kopernic members participate to the EU COST action CERCIRAS

10.3 National initiatives

10.3.1 PSPC

STARTREC The STARTREC project has been funded by the PSPC call until September 2023. Its partners are Easymile, StatInf, Trustinsoft, Inria and CEA. Its objective is the proposition of ISO26262 compliant arguments for the autonomous driving. The results are described within Section 8.

11 Dissemination

Participants: Liliana Cucu-Grosjean, Adriana Gogonel, Yves Sorel.

11.1 Promoting scientific activities

11.1.1 Scientific events: selection

Chair of conference program committees Liliana Cucu-Grosjean has been the Embedded Systems track chair at DATE 2023

Member of the conference program committees All Kopernic members are regular PC members for relevant conferences like IEEE RTSS, RTAS, ECRTS, DATE, ETFA, WFCS and RTNS.

11.1.2 Journal

Member of the editorial boards Liliana Cucu-Grosjean is associated editor at the Journal of Systems Architecture

Reviewer - reviewing activities All Kopernic members are regularly serving as reviewers for the main journals of our domain: Journal of Real-Time Systems, IEEE Transactions on Computer Science, Information Processing Letter, Journal of Heuristics, Journal of Systems Architecture, Journal of Signal Processing Systems, Leibniz Transactions on Embedded Systems, IEEE Transactions on Industrial Informatics, etc.

11.1.3 Invited talks

Liliana Cucu-Grosjean has been keynote at the HIPEAC 2023 Conference in Toulouse

11.1.4 Leadership within the scientific community

Liliana Cucu-Grosjean has been nominated to run for the election and have been elected vice-chair of the IEEE Technical Community on Real-Time Systems (TCRTS). She has started her office on January, 1st of 2024 as the first female vice-chair of this IEEE committee after more than 40 years of existence. According to the IEEE TCRTS rules, she becomes chair of this committee on January, 1st, 2026 for a period of two years.

11.1.5 Scientific expertise

- Yves Sorel is a member of the Steering Committee of Advanced engineering and computing Hub of Systematic Paris-Region Cluster
- Yves Sorel is a member of the Steering Committee Orientations and Programs of SystemX Institute for Technological Research (IRT).

11.1.6 Research administration

Yves Sorel is a member of the CDT Paris center commission

11.2 Teaching - Supervision - Juries

11.2.1 Teaching

Liliana Cucu-Grosjean, Initiation to the research, MSc degree on Embedded Systesm at University of Saclay

11.2.2 Supervision

- Ismail Hawila, Multicore scheduling of real-time control systems of probabilisic tasks with precedence constraints, Sorbonne university, started on October 2022, supervised by Liliana Cucu-Grosjean and Slim Ben Amor (StatInf)
- Marwan Wehaiba El Khazen, Statistical models for optimizing the energy consumption of cyber-physical systems, Sorbonne university, started on October 2020, supervised by Liliana Cucu- Grosjean and Adriana Gogonel (StatInf) with a defense expected in September 2024.
- Kevin Zagalo,Statistical predictability of cyber-physical systems,Sorbonne University, PhD thesis defended on September 23rd, 2023, supervised by Liliana Cucu and Prof. Avner Bar-Hen (CNAM).

11.2.3 Juries

- Liliana Cucu-Grosjean has been chair of the PhD defense committee of:
  - Hadjer Benmeziane, supervised by Smail Niar, Kaoutar El Maghraoui and Hamza Ouarnoughi at Université Polytechnique Hauts-de-France, defense on August 30th, 2023
  - Nan Li, supervised by Laurent Pautet and Eric Goubault at Institut polytechnique de Paris, defense on March 24th, 2023
- Liliana Cucu-Grosjean has been a reviewer for the following theses of:
  - Frédéric Ridouard, at Ecole nationale supérieure de mécanique et d'aérotechnique, HDR defense on November 27th, 2023
  - Gabriella Bettonte, supervised by Stéphane Louise at Universitéy of Paris-Saclay, PhD defense on January 12th, 2023
  - Ill-Ham Atchadam, supervised by Frank Singhoff at University of Brest, PhD defense on March 23rd, 2023
- Liliana Cucu-Grosjean has been member of the PhD defense jury of:
  - Matheus Ladeira Boechat Lemos supervised by Emmanuel Grolleau and Yassine Ouhammou at Ecole nationale supérieure de mécanique et d'aérotechnique, defense on November 11th, 2023

11.3 Popularization

11.3.1 Internal or external Inria responsibilities

Liliana Cucu-Grosjean is the Inria national harassment referee within the FS-CSA

11.3.2 Articles and contents

- Liliana Cucu-Grosjean shares her vision on the future of embedded systems to Hipeac vision readers
- Liliana Cucu-Grosjean shares her experience as woman in Computer Science to the EU project Admorph participants.

11.3.3 Interventions

- Adriana Gogonel has participated to the panel "Competences challenges" organized during the Fête des Start-ups at the Cybersecurity Campus
- Adriana Gogonel has participated in the expert panel "Skilled Labour Shortage" in Nuremberg at the Embedded World Conference
- Adriana Gogonel has participated in the panel "The place of women as deeptech entrepreneurs in France" organized by Starburst at the event Meet up Atechna, Paris

12 Scientific production

12.1 Major publications

[1] L. Cucu-Grosjean and A. Gogonel. 'Simulation Device'. FR2016/050504 (France). Mar. 2016. URL: https://hal.archives-ouvertes.fr/hal-01666599.

[2] L. Cucu-Grosjean, L. Santinelli, M. Houston, C. Lo, T. Vardanega, L. Kosmidis, J. Abella, E. Mezzetti, E. Quiñones and F. J. Cazorla. 'Measurement-Based Probabilistic Timing Analysis for Multi-path Programs'. In: *the 24th Euromicro Conference on Real-Time Systems, ECRTS*. 2012, pp. 91–101.

[3] R. Davis and L. Cucu-Grosjean. 'A Survey of Probabilistic Schedulability Analysis Techniques for Real-Time Systems'. In: *Leibniz Transactions on Embedded Systems* 6.1 (2019), p. 53. DOI: 10.4230/LITES-v006-i001-a004. URL: https://inria.hal.science/hal-02158985.

[4] R. Davis and L. Cucu-Grosjean. 'A Survey of Probabilistic Timing Analysis Techniques for Real-Time Systems'. In: *Leibniz Transactions on Embedded Systems* 6.1 (2019), p. 60. DOI: 10.4230/LITES-v006-i001-a003. URL: https://inria.hal.science/hal-02158973.

[5] A. Gogonel and L. Cucu-Grosjean. 'Dispositif de caractérisation et/ou de modélisation de temps d'exécution pire-cas'. 1000408053 (France). June 2017. URL: https://hal.archives-ouvertes.fr/hal-01666535.

[6] T. Kloda, A. Bertout and Y. Sorel. 'Latency analysis for data chains of real-time periodic tasks'. In: *the 23rd IEEE International Conference on Emerging Technologies and Factory Automation, ETFA'18*. Sept. 2018.

[7] C. Maxim, A. Gogonel, I. M. Asavoae, M. Asavoae and L. Cucu-Grosjean. 'Reproducibility and representativity: mandatory properties for the compositionality of measurement-based WCET estimation approaches'. In: *SIGBED Review* 14.3 (2017), pp. 24–31.

[8] S. E. Saidi, N. Pernet and Y. Sorel. 'Scheduling Real-time HiL Co-simulation of Cyber-Physical Systems on Multi-core Architectures'. In: *the 24th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*. Aug. 2018.

12.2 Publications of the year

International journals

[9] T. N. C. Andrade, G. Lima, V. M. C. Lima, S. Bem-Amor, I. Hawila and L. Cucu-Grosjean. 'On the impact of hardware-related events on the execution of real-time programs'. In: *Design Automation for Embedded Systems* 27.4 (31st Dec. 2023), pp. 275–302. DOI: 10.1007/s10617-023-09281-9. URL: https://inria.hal.science/hal-04442548.

[10] J. Vasconcelos, G. Lima, M. Wehaiba El Khazen, A. Gogonel and L. Cucu-Grosjean. 'On vulnerabilities in EVT-based timing analysis: an experimental investigation on a multi-core architecture'. In: *Design Automation for Embedded Systems* (17th Oct. 2023). DOI: 10.1007/s10617-023-09277-5. URL: https://inria.hal.science/hal-04468516.

[11] K. Zagalo, Y. Abdeddaïm, A. Bar-Hen and L. Cucu-Grosjean. 'Response Time Stochastic Analysis for Fixed-Priority Stable Real-Time Systems'. In: *IEEE Transactions on Computers* (1st Jan. 2023), pp. 1–12. DOI: 10.1109/TC.2022.3211421. URL: https://inria.hal.science/hal-03797980.

International peer-reviewed conferences

[12] M. W. E. Khazen, S. B. Amor, K. Kougblenou, A. Gogonel and L. Cucu-Grosjean. 'Work in progress: Towards a statistical worst-case energy consumption model'. In: 2023 IEEE 29th Real-Time and Embedded Technology and Applications Symposium (RTAS). San Antonio, France: IEEE, 9th May 2023, pp. 333–336. DOI: 10.1109/RTAS58335.2023.00034. URL: https://inria.hal.science/hal-04468460.

Conferences without proceedings

[13] I. Hawila, L. Cucu-Grosjean, S. B. Amor and Y. Sorel. ‘Towards a new task model for merging control theory and real-time scheduling problems’. In: 16th Junior Researcher Workshop on Real-Time Computing. Dortmund, Germany, 7th June 2023. URL: https://inria.hal.science/hal-04153680.

Doctoral dissertations and habilitation theses

[14] K. Zagalo. ‘Stochastic analysis of stationary real-time systems’. Sorbonne Université, 28th Sept. 2023. URL: https://theses.hal.science/tel-04378907.

12.3 Cited publications

[15] R. Baheti and H. Gill. *Cyber-physical systems.* IEEE, 2011.

[16] S. Baruah, V. Bonifaci, A. Marchetti-Spaccamela, L. Stougie and A. Wiese. ‘A Generalized Parallel Task Model for Recurrent Real-time Processes’. In: *2012 IEEE 33rd Real-Time Systems Symposium (RTSS).* 2012, pp. 63–72.

[17] G. Bernat, A. Colin and S. M. Petters. ‘WCET Analysis of Probabilistic Hard Real-Time System’. In: *Proceedings of the 23rd IEEE Real-Time Systems Symposium (RTSS’02).* IEEE Computer Society, 2002, pp. 279–288.

[18] T. Bourke, J. Colaço, B. Pagano, C. Pasteur and M. Pouzet. ‘A Synchronous-Based Code Generator for Explicit Hybrid Systems Languages’. In: *Compiler Construction - 24th International Conference, CC, Joint with ETAPS.* 2015, pp. 69–88.

[19] L. Cucu. ‘Preliminary results for introducing dependent random variables in stochastic feasiblity analysis on CAN’. In: *the WIP session of the 7th IEEE International Workshop on Factory Communication Systems (WFCS).* 2008.

[20] R. I. Davis and L. Cucu-Grosjean. ‘A Survey of Probabilistic Timing Analysis Techniques for Real-Time Systems’. In: *LITES* 6.1 (2019), 03:1–03:60.

[21] S. Edgar and A. Burns. ‘Statistical Analysis of WCET for Scheduling’. In: *the 22nd IEEE Real-Time Systems Symposium (RTSS).* 2001, pp. 215–225.

[22] H. Falk, S. Altmeyer, P. Hellinckx, B. Lisper, W. Puffitsch, C. Rochange, M. Schoeberl, R. B. Sorensen, P. Wägemann and S. Wegener. ‘TACLeBench: A Benchmark Collection to Support Worst-Case Execution Time Research’. In: *16th International Workshop on Worst-Case Execution Time Analysis (WCET).* Vol. 55. OASICS. 2016, 2:1–2:10.

[23] J. Fonseca, G. Nelissen, V. Nelis and L. Pinho. ‘Response time analysis of sporadic DAG tasks under partitioned scheduling’. In: *11th IEEE Symposium on Industrial Embedded Systems (SIES).* May 2016, pp. 1–10.

[24] S. J. Gil, I. Bate, G. Lima, L. Santinelli, A. Gogonel and L. Cucu-Grosjean. ‘Open Challenges for Probabilistic Measurement-Based Worst-Case Execution Time’. In: *Embedded Systems Letters* 9.3 (2017), pp. 69–72.

[25] J. Gustafsson, A. Betts, A. Ermedahl and B. Lisper. ‘The Mälardalen WCET Benchmarks: Past, Present And Future’. In: *10th International Workshop on Worst-Case Execution Time Analysis (WCET).* Vol. 15. OASICS. 2010, pp. 136–146.

[26] E. Lee. ‘Computing Needs Time’. In: *Communications of ACM* 52.5 (2009).

[27] E. Lee and S. Seshia. *Introduction to embedded systems - a cyber-physical systems approach.* MIT Press, 2017.

[28] J. Lehoczky. ‘Real-Time Queueing Theory’. In: *the 10th IEEE Real-Time Systems Symposium (RTSS).* 1996.

[29] S. B. M. Bertogna and G. Buttazzo. *Multiprocessor Scheduling for Real-Time Systems.* Springer, 2015.

[30] D. Maxim, O. Buffet, L. Santinelli, L. Cucu-Grosjean and R. I. Davis. 'Optimal Priority Assignment Algorithms for Probabilistic Real-Time Systems'. In: *the 19th International Conference on Real-Time and Network Systems (RTNS)*. 2011.

[31] D. Maxim and L. Cucu-Grosjean. 'Response Time Analysis for Fixed-Priority Tasks with Multiple Probabilistic Parameters'. In: *the IEEE Real-Time Systems Symposium (RTSS)*. 2013.

[32] F. Ndoye and Y. Sorel. 'Monoprocessor Real-Time Scheduling of Data Dependent Tasks with Exact Preemption Cost for Embedded Systems'. In: *the 16th IEEE International Conference on Computational Science and Engieering (CSE)*. 2013.

[33] F. Nemer, H. Cassé, P. Sainrat, J. P. Bahsoun and M. D. Michiel. 'PapaBench: a Free Real-Time Benchmark'. In: *6th Intl. Workshop on Worst-Case Execution Time (WCET) Analysis*. Vol. 4. OASICS. 2006.

[34] S. E. Saidi, N. Pernet and Y. Sorel. 'Automatic Parallelization of Multi-Rate FMI-based Co-Simulation On Multi-core'. In: *the Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium*. 2017.

[35] T. Tia, Z. Deng, M. Shankar, M. Storch, J. Sun, L. Wu and J. Liu. 'Probabilistic Performance Guarantee for Real-Time Tasks with Varying Computation Times'. In: *IEEE Real-Time and Embedded Technology and Applications Symposium*. 1995.

[36] R. Wilhelm, J. Engblom, A. Ermedahl, N. Holsti, S. Thesing, D. Whalley, G.Bernat, C. Ferdinand, R.Heckmann, T. Mitra, F. Mueller, I. Puaut, P. Puschner, G. Staschulat and P. Stenströem. 'The worst-case execution time problem: overview of methods and survey of tools'. In: *Trans. on Embedded Computing Systems* 7.3 (2008), pp. 1–53.