Netherlands Organisation for Scientific Research

Vernieuwingsimpuls
Innovational Research Incentives Scheme
Grant application form 2015

Veni scheme

Registration form (basic details)

1a. Details of applicant

Title:	Dr.-Ing.
First name:	Sebastian
Initials:	S.J.A.
Prefix:	—
Surname:	Altmeyer
Male/female:	male
Address for correspondence:	Informatics Institute, University of Amsterdam Science Park 904, 1098 XH, Amsterdam
Preference for correspondence in English:	Yes
Telephone:	020.525.8735
Email:	altmeyer@uva.nl
Website:	staff.fnwi.uva.nl/s.j.altmeyer/
Use of extension clause:	No

1b. Title of research proposal

The time is now: Timing Verification for Safety-Critical Multi-Cores

1c. Summary of research proposal

In our everyday life, we interact with a huge number of computer systems embedded into larger devices. Examples are phones, cars, fridges, air-planes and many more. Many of these devices are safety-critical real-time systems. Real-time means that the correctness of a system is not only a functional (the right result), but also an extra-functional property (the right result at the right time). Safety-critical means that a single failure—such as a wrong timing—may lead to a catastrophe and the loss of life: An airbag controller, for instance, has not only to decide whether or not to inflate the airbag, but has to do so before the driver's head hits the steering wheel. Safety-critical real-time systems thus undergo a thorough timing verification to fulfil highest validation requirements. Timing verification is traditionally a two-step process. The execution time of each individual software task is derived independently and then given to the schedulability analysis, which analyses the combination of the tasks on the system. While this timing verification is valid for state-of-the-art embedded processors, it fails for modern multi-core systems optimized towards the average-case and high-performance computing. Due to interferences on shared resources (bus, cache, etc.), the timing of an individual task highly depends on the tasks running in parallel on the other cores and cannot be analysed independently anymore. Multi-core architectures, however, are the only viable solution to accommodate performance demands of modern computer systems in general and of modern embedded real-time systems in particular. In this project, I will develop a timing verification for real-time multi-cores. Interference on shared resources will be modelled explicitly and reflected in the task model, i.e., in the parameters describing the tasks' timing behaviour. Theoretical schedulability results—valid for the obsolete model—will be put to a test and updated for the interference-aware task model.

1d. Keywords

Real-Time Systems, Real-Time Scheduling, Timing Validation, Computer Architecture, Multi-Core Systems, Safety-Critical Systems

1e. Current institution of employment

Informatics Institute, University of Amsterdam

1f. Host institution

Informatics Institute, University of Amsterdam

1g. NWO Division

NWO EW

1h. Main field of research

16.10.00 Computer systems, architectures, networks

1i. Public summary of the research proposal

Safety-critical computer systems embedded into cars or air-planes must work correctly, as a single failure or a wrong timing may have catastrophic consequences and cost lives. This research aims at guaranteeing the correct timing behaviour of such safety-critical systems for modern processor architectures by devising new mathematical models and tools.

Research proposal

2a. Description of the proposed research

2a1. Overall aim and key objectives

The most common computer systems are not personal computers or servers, but embedded computing systems [20]. Designing such systems differs fundamentally from traditional design methods for general-purpose computer architectures or software, as embedded systems are subject to specific requirements, especially in terms of efficiency (cost, code size, weight and energy) and dependability (a failure may not only cause a reboot of the system but harm to the users) [21]. Yet, also these systems are exposed to an increasing demand for a higher performance [20]. Think for instance of the collision avoidance systems in modern cars, where huge amounts of data (video stream of the street) needs to be processed and correct decisions based on this data must be taken in real-time to prevent accidents [15].

As timing behavior is a major property of such systems, proving timing correctness is of utter importance during the development process. Functionality of an embedded system is typically implemented by a set of tasks distributed to the available hardware and processing units. Proving timing correctness of the complete system is traditionally a two-step approach as depicted in Figure 1:

Timing analysis derives upper bounds on the execution times of tasks in isolation, called worst-case execution times (WCET). A static timing analysis–able to fulfill the highest certification requirements—employs an abstract model of the hardware to characterize the timing behaviour of a task.

Scheduling Analysis determines if each task complies with its timing constraints when scheduled according to a predefined scheduling policy. Timing constraints are typically defined by a task's period and a task's deadline, both determined by the physical environment. Hence, tasks are assumed to be fully characterized by a triple consisting of a period, a deadline and an execution time bound, i.e. the WCET of the task.

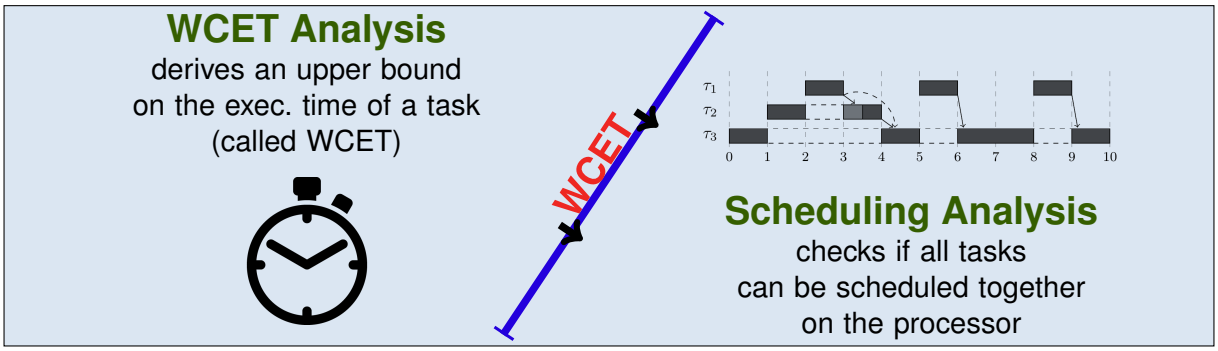


Figure 1: Traditional 2-Step Timing Verification

Traditionally, the complete **interface between timing analysis and scheduling analysis** is given by bounds on the tasks' execution times. Other input to the scheduling analysis—such as deadlines or periods of tasks—are dictated by the system's environment and bypass the timing analysis.

Scientific relevance and challenges

As power dissipation became a limiting factor for increasing clock rates, the need for higher performance is accommodated by multi-core systems. While such multi-cores are state-of-the-art in high-performance and end-consumer devices since around 2005, the specific needs of real-time embedded systems impede their wide-spread use in this domain—especially for safety-critical real-time systems such as the aforementioned collision avoidance system. These systems are subject to stringent certification and validation requirements, which in turn, prove to be strong barriers for multi-core architectures: Predicting the timing behaviour and providing timing guarantees is not yet possible, largely preventing multi-core systems to enter the field of safety-critical and hard real-time applications.

At the very heart of the problem lies the interference of tasks which are modelled as independent entities, but when executed on a multi-core system compete for the very same common resources: foremost *shared bus* and *shared memories*. The consequences of these interferences are twofold:

1. The high-level task model (consisting of period T, deadline D and WCET C) used within multi-core scheduling theory does not reflect the behaviour on modern multi-core architectures. Theoretical schedulability results based on the simplistic task model do not transfer to reality.
2. The traditional 2-step timing verification fails as tasks' execution times can not be analysed independently. As a result, multi-core processors used in hard real-time systems are highly underutilized to compensate for the uncertainty and variance in the execution time. A remedy that often completely negates the potential performance benefit of multi-core systems.

Overall aim

The aim of the project is to develop a **timing verification framework for real-time multi-cores** as depicted in Figure 2. Explicit modelling of interference on shared resources will replace the traditional 2-step timing verification and enable (i) a **schedulability theory with a realistic task model** and (ii) a mathematically profound **classification of multi-core hardware components** with respect to predictability and usability for hard real-time systems. The sources of interference and thus the design-space of the multi-core systems considered within this project are: the memory hierarchy, shared caches, and bus topology and arbiter.

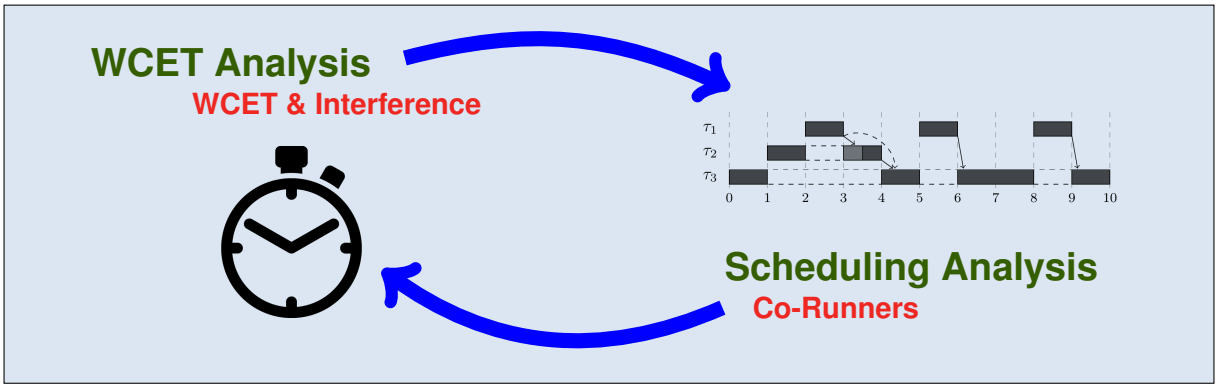


Figure 2: Combined Timing Verification for Multi-Core Systems

Key objectives

Obj.1 *Static Interference analysis*
Worst-case execution time analyses for uni-processor systems derive a single upper bound on the execution time of a task (WCET) and the corresponding execution path, where the task exhibits the worst-case behaviour. As a task's execution time on a multi-core system depends on tasks running in parallel on the other cores, a single upper bound on the execution time is insufficient. Key Objective 1 consists of extending timing analysis in two aspects: (a) to account for dependency with co-running tasks, and (b) to provide—in addition to the execution time bound—information on the accesses to/usage of shared resources.

Obj.2 *Interference-aware multi-core scheduling analysis*
Multi-core scheduling analyses rely on a highly simplified and restricted task model, usually consisting only of period/deadline/execution time. Key Objective 2 consists of extending the traditional task model by additional terms explicitly representing the interference on shared resources. I will explore the consequences on theoretical schedulability results which were obtained for the simplistic task model (optimality of scheduling algorithms/priority assignments) and re-evaluate the relative performance of different scheduling policies for the realistic task model, which accurately represents the task behaviour on modern multi-core architectures.

Obj.3 *Classification of hardware components*
A wide selection of multi-core components (e.g. bus topologies, bus arbiter, memory hierarchy) are available and used in various multi-core architectures; most of which designed for high (average) performance. Key objective 3 consists of a classification of these hardware components (based on the timing verification framework and the interference-aware schedulability theory) in terms of predictability, performance and analysability. This classification thus provides means to argue about the suitability of multi-core components for hard real-time systems in a mathematically rigorous and formal manner.

Originality and innovative character

The research is highly innovative as it targets the complete timing verification process at once (instead of either the WCET or the scheduling analysis) and aims at replacing the traditional process by an integrated method. The interferences on shared resources will be treated as first-class citizens and genuine components of the task model. This allows me to study the impact of the interferences on the schedulability in a mathematically rigorous fashion and to develop a timing verification framework for realistic multi-core architectures.

The project thus differs fundamentally from previous efforts on multi-core timing analysis. Worst-case execution time analyses [26]—successfully applied for uni-processors—fail as the complexity of modern multi-core systems sets a limit to timing analysability and renders the state-of-the-art timing analyses practically infeasible. Multi-core schedulability theory (see [13]) has been developed when interferences on shared resource could be ignored and still relies on a simple task model with a single bound to represent a task's execution time. Efforts to include interferences within the timing verification are limited to individual components only (either bus or cache [24, 17]) and/or treat interferences in a highly abstract fashion [22, 16, 19]. Consequently, multi-core timing analyses either rely on unrealistic assumptions or are too pessimistic [2]. The community currently aims at developing predictable multi-cores [11, 25] or components thereof [3, 23]. Even though design guidelines for multi-core systems have emerged [12, 27],

the research community relies on a hand-waving and vague argumentation for or against certain components, without a clear classification and justification thereof.

Methods and techniques

The timing verification for *pre-emptively scheduled uni-processors with caches* [4] provides a guide-line for the integrated multi-core timing verification (see Figure 2). Pre-emptively scheduled systems with caches violate the independence-assumption of the timing analysis due to interference on the cache [6]. On a task pre-emption, the pre-empting task disrupts the current processor state and evicts cached data of the pre-empted task. After preemption, the execution time of the pre-empted task depends on whether previously cached data is still resident in cache. The additional execution time due to preemption is denoted *cache-related preemption delay (CRPD)* [7]. The CRPD highly depends on the actual pre-emption point, pre-empted task and the set of (potentially nested) pre-empting tasks. Using a single bound for the CRPD (independent of the aforementioned factors) leads to a significant over-approximation and unacceptable pessimism [9]. Timing verification for pre-emptive systems has been successfully realized by (i) extending timing analysis to provide metrics for the cache-reuse (the set of useful cache blocks) [5] and the memory footprint (set of evicting cache blocks) [10] of each task and (ii) by integrating these metrics into the scheduling analysis [8, 18].

Timing verification for multi-core systems will rely on a similar structure. The WCET analysis will derive bounds on the tasks' execution times (without any interference on shared resources), on the memory usage, the reuse patterns and the bus usage. The different factors are highly-interdependent as for instance, the worst-case program path in terms of pure execution time may differ from the worst-case path in terms of memory usage. Also the amount of bus accesses depends not only on the task itself, but also on the interference on shared local caches and thus, on other tasks running on the same core. The scheduling analysis will be extended to include the interference on the various resources and will not only provide a bound on the tasks' response times but also on the set of tasks running in parallel. This way creating a feed-back loop: the interference bounds obtained from the WCET analysis can be refined by the set of co-running tasks and the refined bounds on the interference can in turn be used to refine the set of co-running tasks. The traditional term of WCET will be obsolete and only refer to the pure computational demand without interference.

To isolate the various sources of interference, I will use a **gradual refinement** of the task model as depicted in Figure 3. The figure exemplifies the interference-aware timing verification based on response-time analysis for fixed-priority pre-emptive scheduling and static mapping of tasks to cores. The response time R_i of a task i denotes the maximal time in between a task's release and its completion and can be computed using the recursive equation on the right, where $hp(i)$ denotes the set of tasks with higher priority than i scheduled on the same core. C_i denotes the worst-case execution time of task i, T_j the period of task j and $\lceil R_i/T_j \rceil$ the number of invocations of task j during the response time R_i. The basic equation is only valid under the assumption that no interference on shared resources other than the core itself influences the tasks' execution times. I.e. for an abstract model of the multi-core architectures with *perfect* shared components able to avoid all interference. This abstract processor model serves as the starting point for the gradual refinement, which replaces successively abstract, perfect components by their concrete implementations. In each refinement step, (i) a task parameter is included to reflect the additional execution time due to interferences on the concretized component, and (ii) the impact on the overall schedulability is analysed.

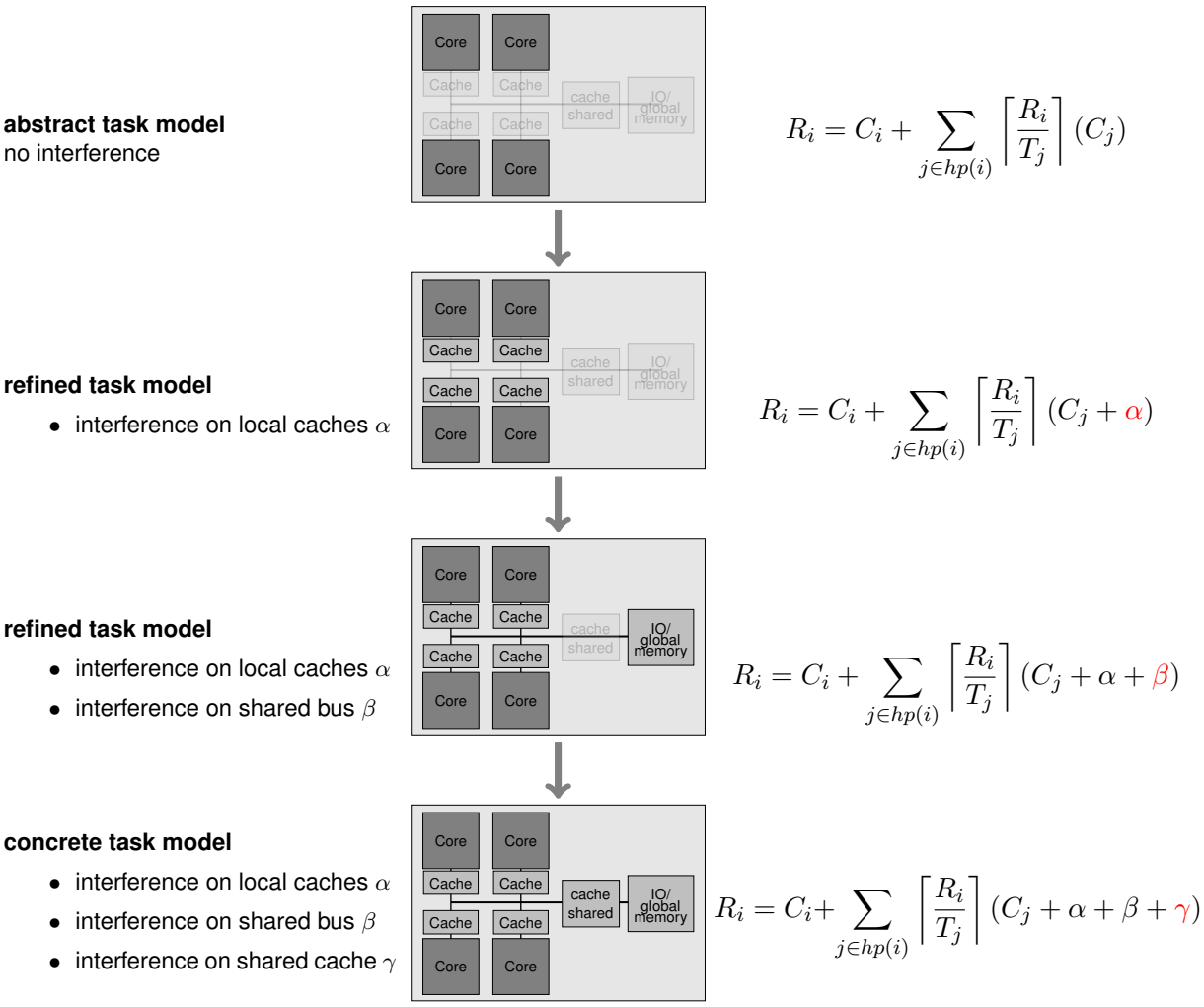


Figure 3: Gradual Refinement

2a2. Research plan

Practical timetable/timeline over the grant period
The project consists of five work-packages.

Work-package 1 Static Analysis of Bus/Memory/Cache usage

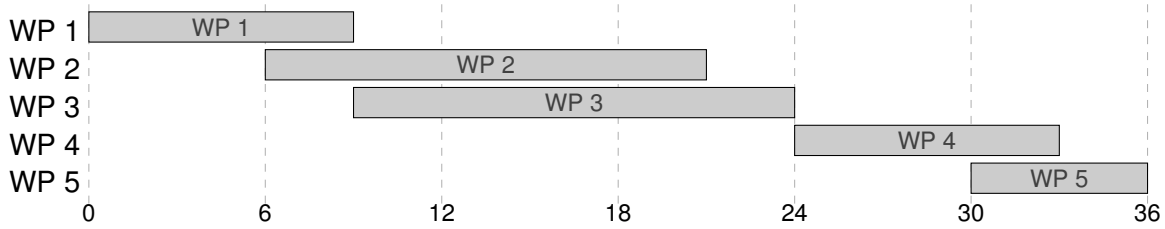
Work-package 2 Analysis of the interdependency of various interferences

Work-package 3 Interference-aware Scheduling Analysis

Work-package 4 Classification of multi-core components

Work-package 5 Prototype dissemination

The following gantt-chart describes the chronological order of the work-packages:



Local, national and international collaboration
Realization of the projects requires input from experts in the following domains

1. Multi-core/Hardware Design
2. Analysis of Real-Time systems
3. Real-Time Scheduling

which are fully covered by the existing collaborations with the following groups:

- *Real-Time Systems Group* at the *University of York* headed by Alan Burns which are the leading experts on real-time scheduling [13].
- *Compiler Design Lab* headed by Reinhard Wilhelm and the *Real-Time and Embedded Systems Group* headed by Jan Reineke at the *Saarland University* where the industry standard for static timing analysis [14] has been developed.
- *Computer Systems Architecture Group* led by Andy Pimentel at the *University of Amsterdam* where expertise on actual multi-core designs and implementation is abundant.

I am furthermore in contact with experts on timing analysis as member of the ICT COST-Action Timing Analysis on Code-Level (TACLe).

2b. Knowledge utilisation

The embedded market is one of the largest information-technology sectors in Europe. It is still expected to grow considerably [1] and already now contributes substantially to Europe's economic strength [20]. Furthermore, embedded systems have a significant impact on other industrial sectors such as the automotive industry, industrial automation and avionics, where the innovation and progress is mainly driven by embedded systems. The majority of embedded systems in these areas are real-time and safety-critical systems. The European economy is thus highly dependent on cutting-edge research in the area of embedded systems, especially when considering the strong worldwide competition [20].

With the increasing complexity of modern-day embedded and safety-critical systems, the performance of traditional single-core processor systems has reached its limit. Multi-core systems, which are already common in high-performance and personal computing, are thus widely considered the basis for future embedded systems. The inherent problem of this development is that multi-core systems not only offer higher performance, but also a higher variability of the execution times and a strongly increased level of parallelism. Consequently, the timing behaviour of such systems is considered unpredictable and providing timing guarantees is not yet possible. This prevents the use of multi-core systems in safety-critical devices even though performance demands create the necessity. In recent years, this problem has received significant attention

and is currently the hot topic in the real-time research community. Up to now, no sophisticated solutions exist and multi-cores system cannot be utilized.

The research project tackles the very fundamental concepts of the timing verification: it replaces the traditional two-step process by an integrated timing verification where the interferences on shared resources (i) are modelled explicitly and (ii) are considered a first-class citizen of the task model. It thus explores a novel and promising research direction which aims at one of the most pressing open problems in the area of safety-critical real-time systems. The research will enable the use of multi-core architectures in safety-critical real-time systems including the performance benefits of multi-cores: lower power-consumption and lower hardware costs at higher computational power.

I will target knowledge utilization using three different paths:

Publications: Using the traditional way of publishing scientific results, I will disseminate the results at the top conferences and journals. For the theoretical results on the multi-core schedulability with interferences, I will aim at the main conferences for the real-time community, foremost *RTSS* and *ECRTS*. For the practical results of the project, I will mainly target venues which are intended not only for a scientific but also an industrial audience, e.g. *DATE* and *DAC*.

Software Release: A main result of the project will be a prototype implementation of the integrated timing verification framework. To alleviate the high implementation efforts, I will implement this framework using open-source components and use existing compiler and timing analyses frameworks. I intend to make the implementation of the timing verification framework publicly available. The software for the complete framework will only be released at a late stage of the project, yet also with the possibility to provide the interference analysis and the interference-aware scheduling analysis as intermediate stand-alone tools. The prototype implementation will provide means to validate the theoretical results and to explore the usability.

Industrial Use Cases: Due to the high competition on the market of safety-critical systems, the access to real-life code is very restricted. The available benchmarks—needed to validate and test the timing verification framework—are limited in size and relevance. I will thus seek contact with potential industrial partners by exploiting existing networks (of myself and my academic partners) with companies such as Daimler Benz AG, European Space Research and Technology Centre (ESTEC), Koninklijke Philips N.V. and ASML. The high significance of the proposed research to industry and many Dutch and European companies will ease this search. Industry contact will already be useful at early stages of the project to guide the development towards practicability. The case studies can only be conducted after the software is written and thus have to be postponed to the third year.

Furthermore, I aim to initiate contact with the leading companies on timing verification for embedded systems (ABSINT GmbH and Rapita Systems Ltd.) to target not only the users of timing verification, but also the vendors of the respective tools. This will allow me to explore potential collaboration and lay a basis for a successful knowledge transfer of the resulting theoretical insights and the timing verification framework.

This threefold knowledge utilization is aimed to establish a lasting impact of the project, both in the academic and the industrial world.

2c. Number of words used: section 2a: 2000 (max. 2000 words)

Number of words used: section 2b: 750 (max. 750 words)

2d. Literature references

References

[1] Information society and european commission, “building artemis”, 2004.

[2] A. Abel, F. Benz, J. Doerfert, B. Dörr, S. Hahn, F. Haupenthal, M. Jacobs, A. Moin, J. Reineke, B. Schommer, and R. Wilhelm. Impact of resource sharing on performance and performance prediction: A survey. In P. D’Argenio and H. Melgratti, editors, *CONCUR 2013 – Concurrency Theory*, volume 8052 of *Lecture Notes in Computer Science*, pages 25–43. 2013.

[3] B. Akesson and K. Goossens. Architectures and modeling of predictable memory controllers for improved system integration. In *DATE’11*, pages 851–856, 2011.

[4] S. Altmeyer. *Analysis of Preemptively Scheduled Hard Real-time Systems*. epubli GmbH, 2013.

[5] S. Altmeyer and C. Burguière. A new notion of useful cache block to improve the bounds of cache-related preemption delay. In *Proceedings of the 21st Euromicro Conference on Real-Time Systems ECRTS ’09*, pages 109–118. IEEE Computer Society, July 2009.

[6] S. Altmeyer and C. Burguière. Influence of the task model on the precision of scheduling analysis for preemptive systems. In R. I. Davis and N. Fisher, editors, *Proceedings of the 1st International Real-Time Scheduling Open Problems Seminar*, pages 5–6, July 2010.

[7] S. Altmeyer and C. Maiza. Cache-related preemption delay via useful cache blocks: Survey and redefinition. *Journal of Systems Architecture*, 2010.

[8] S. Altmeyer and C. Maiza. Influence of the task model on the precision of scheduling analysis for preemptive systems – status report. In R. I. Davis and N. Fisher, editors, *Proceedings of the 2nd International Real-Time Scheduling Open Problems Seminar*, pages 1–2, July 2011.

[9] A. Bastoni, B. Brandenburg, and J. Anderson. Cache-related preemption and migration delays: Empirical approximation and impact on schedulability. In *6th International Workshop on Operating Systems Platforms for Embedded Real-Time Applications*, OSPERT ’10, pages 33–44, July 2010.

[10] J. Busquets-Mataix, J. Serrano, R. Ors, P. Gil, and A. Wellings. Adding instruction cache effect to schedulability analysis of preemptive real-time systems. In *Proceedings of the 2nd IEEE Real-Time Technology and Applications Symposium*, RTAS ’96, pages 204–214, April 1996.

[11] F. J. Cazorla, E. Quiñones, T. Vardanega, L. Cucu, B. Triquet, G. Bernat, E. D. Berger, J. Abella, F. Wartel, M. Houston, L. Santinelli, L. Kosmidis, C. Lo, and D. Maxim. Proartis: Probabilistically analyzable real-time systems. *ACM Trans. Embedded Comput. Syst.*, 12(2s):94, 2013.

[12] C. Cullmann, C. Ferdinand, G. Gebhard, D. Grund, C. Maiza, J. Reineke, B. Triquet, S. Wegener, and R. Wilhelm. Predictability considerations in the design of multi-core embedded systems. *Ingénieurs de l'Automobile*, 807:36–42, September 2010.

[13] R. I. Davis and A. Burns. A survey of hard real-time scheduling for multiprocessor systems. *ACM Comput. Surv.*, 43(4):35:1–35:44, Oct. 2011.

[14] C. Ferdinand and R. Heckmann. ait: worst case execution time prediction by static program analysis. In *IFIP Congress Topical Sessions*, pages 377–384, 2004.

[15] S. K. Gehrig and F. Stein. Cartography and dead reckoning using stereo vision for an autonomous car. In *ICIP (4)*, pages 30–34, 1999.

[16] K. Lampka, G. Giannopoulou, R. Pellizzoni, Z. Wu, and N. Stoimenov. A formal approach to the wcrt analysis of multicore systems with memory contention under phase-structured task sets. *Real-Time Syst.*, 50(5-6):736–773, Nov. 2014.

[17] Y. Liang, H. Ding, T. Mitra, A. Roychoudhury, Y. Li, and V. Suhendra. Timing analysis of concurrent programs running on shared cache multi-cores. *Real-Time Systems*, 48(6):638–680, 2012.

[18] W. Lunniss, S. Altmeyer, C. Maiza, and R. I. Davis. Integrating cache related pre-emption delay analysis into edf scheduling. In *Proceedings of the 19th Real-Time and Embedded Technology and Applications Symposium RTAS'13*, April 2013.

[19] M. Lv, W. Yi, N. Guan, and G. Yu. Combining abstract interpretation with model checking for timing analysis of multicore software. In *Proceedings of the 2010 31st IEEE Real-Time Systems Symposium*, RTSS '10, pages 339–349, Washington, DC, USA, 2010. IEEE Computer Society.

[20] K. B. J. M. M. Duranton, D.Blac-Schaffer. The hipeac vision for advanced computing in horizon 2020, 2013.

[21] P. Marwedel. *Embedded System Design*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[22] G. Nelissen, K. Bletsas, and J. Goossens. Cpmd-mindful task assignment for nps-f. *Real-Time Syst.*, 50(5-6):585–591, Nov. 2014.

[23] E. Quinones, E. D. Berger, G. Bernat, and F. J. Cazorla. Using randomized caches in probabilistic real-time systems. In *ECRTS 2009*, pages 129–138.

[24] A. Schranzhofer, J.-J. Chen, and L. Thiele. Timing analysis for tdma arbitration in resource sharing systems. In *IEEE Real-Time and Embedded Technology and Applications Symposium*, pages 215–224, 2010.

[25] T. Ungerer, F. Cazorla, P. Sainrat, G. Bernat, Z. Petrov, C. Rochange, E. Quinones, M. Gerdes, M. Paolieri, J. Wolf, H. Casse, S. Uhrig, I. Guliashvili, M. Houston, F. Kluge, S. Metzlaff, and J. Mische. Merasa: Multicore execution of hard real-time applications supporting analyzability. *IEEE Micro*, 30(5):66–75, 2010.

[26] 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, J. Staschulat, and P. Sten-

ström. The worst-case execution-time problem: overview of methods and survey of tools. *ACM Trans. Embed. Comput. Syst.*, 7(3):36:1–36:53, May 2008.

[27] R. Wilhelm, D. Grund, J. Reineke, M. Schlickling, M. Pister, and C. Ferdinand. Memory hierarchies, pipelines, and buses for future architectures in time-critical embedded systems. *Trans. Comp.-Aided Des. Integ. Cir. Sys.*, 28(7):966–978, 2009.

Cost estimates

3a. Budget

	Description			Year 1	Year 2	Year 3	Total
Staff		FTE	Months				
WP	Applicant	0	0	67,117	69,232	73,091	209,440
NWP							
Total Staff		0	0	67,117	69,232	73,091	209,440
Equipment							
Total Equipment							
Investments							
Total Investments							
Materials				2,750	2,750	2,750	8,250
Total Materials				2,750	2,750	2,750	8,250
Travels				10,770	10,770	10,770	32,310
Total Travels				10,770	10,770	10,770	32,310
Other							
Total Other							
Grand Total				80,637	82,752	86,611	250,000

3b. Cofinancing 'in kind'

Cofinancer/party	Description	Estimated value in Euro
-	-	-

3c. Cofinancing 'in cash'

Cofinancer/party	Description	Estimated value in Euro
-	-	-

3d. Totals

Grand total	250.000
Requested budget	250.000

3e. Intended starting date

September 1st, 2015

3f. Have you submitted the same idea elsewhere or have you requested any additional grants for this project either from NWO or from any other institution?

No

Curriculum vitae

4a. Personal details

Title(s), initial(s), first name, surname:	Dr.-Ing., S.J., Sebastian, Altmeyer
Gender:	male
Date and place of birth:	19/9/82, Saarbrücken, Germany
Nationality:	German
Birth country/countries of parents:	Germany

4b. Master's ('doctoraal')

University of Higher Education:	Saarland University
Date:	11/11/06
Main subject:	Parametric WCET Analysis, Parameter Framework and Parametric Path Analysis

4c. Doctorate

University of Higher Education:	Saarland University
Completion date:	25/10/12
Supervisor ('Promotor'):	Prof. Dr. Dr. hc Reinhard Wilhelm
Title of thesis:	Analysis of Preemptively Scheduled Hard Real-Time Systems

4d. Work experience since completing your PhD

Position	Period	fte	Type	Institution
Postdoc	1/1/13 – 31/12/15	1	temporary	University of Amsterdam

Work experience in months spent since completing your PhD

Experience	Number of months
Research activities	24
Education	-
Care or sick leave	-
Management tasks	-
Other, please specify	-

4e. Academic staff supervised

	Role as (co-) supervisor
PhDs Ongoing	1
Completed	-
Subtotal PhDs	1
Postdocs	-
Subtotal postdocs	-
Master students	-
Subtotal master students	-
Other	2
Subtotal other	2

4f. Brief summary of research over the last five years

My research is driven by the genuine interest in computing systems that surround us and on which we entrust our lives on a daily basis. I have thus worked on various aspects of the design, optimization and verification of these systems. In particular, I have improved the applicability of static timing analysis. The major achievements in this aspect are a symbolic timing analysis framework and the static timing analysis for preemptively scheduled systems. The symbolic timing analysis allows parametric in addition to static loop bounds and computes a loop bound function parametric in these bounds. This framework thus lifts the restriction that the complete information about the executed programs need to be known statically. The timing analysis for preemptively scheduled systems lifts the–often unrealistic–assumption that safety-critical real-time applications run from start to completion without any interrupt. To this end, I have developed various methods to precisely bound the preemption cost and to integrate these costs in the timing verification process. In addition, I have worked on the support for dynamic memory allocation in hard real-time systems and probabilistic timing analysis for random cache replacement policies. With respect to optimizing real-time systems and selecting appropriate hardware, I have conducted studies comparing scheduling policies and memory architectures for real-time systems. In the course of this research, I have acquired in-depth expertise in the area of static timing analysis and real-time scheduling, which is necessary to successfully conduct the proposed project and to develop a timing verification framework for real-time multi-cores.

4g. International activities

National Representative of the ICT COST Action IC1202 Timing Analysis on Code-Level (TACLe) October 2013 - now

Visiting Researcher at the University of York December 2011
Visiting the Real Time Systems Group headed by Prof. Alan Burns

Visiting Researcher at Seoul National University October 2006
Visiting the Memory & Storage Architecture Lab headed by Prof. Sang Lyul Min

Visiting Researcher at the Mälardalen University January 2006
Visiting the WCET analysis research group headed by Prof. Björn Lisper

4h. Other academic activities

Program Committee Member of

- Real-Time Scheduling Open Problems Seminar (RTSOPS 2015)
- Conference on Languages, Compilers, Tools and Theory for Embedded Systems (LCTES 2014)
- International Conference on Real-Time Networks and Systems (RTNS 2014)
- International Workshop on Worst-Case Execution Time Analysis (WCET 2014)
- Real-Time Scheduling Open Problems Seminar (RTSOPS 2014)
- International Conference on Real-Time Networks and Systems (RTNS 2013)
- Real-Time Scheduling Open Problems Seminar (RTSOPS 2013)
- Junior Researcher Workshop on Real-Time Computing (JRWRTC 2011)
- Junior Researcher Workshop on Real-Time Computing (JRWRTC 2010)

Organization/Workshop Chair of

- Junior Researcher Workshop on Real-Time Computing (JRWRTC 2013)

Assistant Lecturer at the University of Applied Science Saarbrücken 2008 - 2013

Reviewer for the main international Journals on Real-Time/Embedded Systems
Journal of Real-Time Systems (RTSJ), ACM Transactions on Embedded Computing Systems (TECS), Journal of Systems Architecture (JSA), Leibniz Transactions on Embedded Systems (LITES), IEEE Transactions on Computers (TC)

4i. Scholarships, grants and prizes

Outstanding Paper Award ECRTS 2014
For the paper entitled *Evaluation of Cache Partitioning for Hard Real-Time Systems*

Outstanding Paper Award RTNS 2014
For the paper entitled
Accounting for Cache Related Pre-emption Delays in Hierarchical Scheduling

Best Paper Award LCTES 2011
For the paper entitled *Precise and Efficient Parametric Path Analysis*

Outstanding Achievements Award / FDSI Saarbrücken 2006
Distinction for outstanding achievements in graduate studies

Output

5a. Output indicators

The Real-Time Systems community has a strong focus on dedicated conferences, with the Real-Time Systems Symposium (RTSS) and Euromicro Conference on Real-Time Systems (ECRTS) being the flagship conferences with acceptance rates typically below 25%. The WCET Workshop is the main event for researchers working on Worst-case execution time analysis and is for the community as important as a full conference. The only journal dedicated to Real-Time Systems is the Springer Journal of Real-Time Systems, which is thus prefered the target for larger publications. Conferences with a wider focus, foremost Design Automation and Test in Europe (DATE) and Design Automation Conference (DAC), only play a secondary role for the real-time community, but have a strong outreach to the embedded design community and to the embedded industry. The acceptance rate of these conferences for full papers is typically around 20%.

5b. Output

Refereed articles

[1-S] Roeland Douma, Sebastian Altmeyer and Andy Pimentel. Fast and Precise Cache Performance Estimation for Out-of-order Processors In *Proceedings of Design, Automation, and Test in Europe (DATE'15) (accepted for publication)*, March 2015.

[2] Sebastian Altmeyer, Liliana Cucu Grosjean and Robert I. Davis. Static Probabilistic Timing Analysis for Real-Time Systems using Random Replacement Caches In *Journal of Real-Time Systems (RTSJ) (accepted for publication)*, 2015.

[3-S] Reinder J. Bril, Sebastian Altmeyer, Martijn van den Heuvel, Robert I. Davis, and Moris Behnam Integrating Cache-Related Pre-emption Delays into Analysis of Fixed Priority Scheduling with Pre-emption Thresholds In *Proceedings of the 35th IEEE Real-Time Systems Symposium (RTSS'14)*, December 2014.

[4-S] Will Lunniss, Sebastian Altmeyer and Robert I. Davis A comparison between fixed priority and Edf scheduling accounting for cache related pre-emption delays. *Leibniz Transactions on Embedded Systems*, 2014.

[5-S] Will Lunniss, Sebastian Altmeyer, Giuseppe Lipari, and Robert I. Davis Accounting for cache related pre-emption delays in hierarchical scheduling. In *Proceedings of the 22nd International Conference on Real-Time Networks and Systems (RTNS 2014)*, October 2014.

[6-S] Sebastian Altmeyer, Roeland Douma, Will Lunniss, and Robert I. Davis Evaluation of cache partitioning for hard real-time systems. In *Proceedings of the 26th Euromicro Conference on Real-Time Systems (ECRTS'14)*, July 2014.

[7-S] Will Lunniss, Sebastian Altmeyer, and Robert I.!Davis. Accounting for cache related pre-emption delays in hierarchical scheduling with local edf scheduler. In *Proceedings of the 8th Junior Researcher Workshop on Real-Time Computing (JRWRTC 2014)*, October 2014.

[8] Benjamin Lesage, David Griffin, Robert I. Davis, and Sebastian Altmeyer. On the application of static probabilistic timing analysis to memory hierarchiess. In *Proceedings Real-Time Scheduling Open Problems Seminar (RTSOPS'14)*, July 2014.

[9] Sebastian Altmeyer, Liliana Cucu-Grosjean, Robert I. Davis, and Benjamin Lesage. Progress on static probabilistic timing analysis for systems with random cache replacement policies. In *Proceedings Real-Time Scheduling Open Problems Seminar (RTSOPS'14)*, July 2014.

[10] Raphael Poss, Sebastian Altmeyer, Mark Thompson, and Rob Jelier. Academia 2.0: removing the publisher middle-man while retaining impact. In *Proceedings of the 1st Workshop on Reproducible Research Methodologies and New Publication Models in Computer Engineering (TRUST'14)*, June 2014.

[11-S] Jan Reineke, Sebastian Altmeyer, Daniel Grund, Sebastian Hahn, and Claire Maiza. Selfish-lru: Preemption-aware caching for predictability and performance. In *Proceedings of the 20th Real-Time and Embedded Technology and Applications Symposium (RTAS'14)*, April 2014.

[12-S] Sebastian Altmeyer and Robert I. Davis. On the correctness, optimality and precision of static probabilistic timing analysis. In *Proceedings of Design, Automation, and Test in Europe (DATE'14)*, March 2014.

[13-S] Robert I. Davis, Luca Santinelli, Sebastian Altmeyer, Claire Maiza, and Liliana Cucu-Grosjean. Analysis of probabilistic cache related pre-emption delays. In *Proceedings of the 25th Euromicro Conference on Real-Time Systems (ECRTS'13)*, July 2013.

[14-S] Will Lunniss, Sebastian Altmeyer, Claire Maiza, and Robert I. Davis. Integrating cache related pre-emption delay analysis into edf scheduling. In *Proceedings of the 19th Real-Time and Embedded Technology and Applications Symposium (RTAS'13)*, April 2013.

[15-S] Jack Whitham, Robert I. Davis, Neil C. Audsley, Sebastian Altmeyer, and Claire Maiza. Investigation of scratchpad memory for preemptive multitasking. In *Proceedings of the 33nd IEEE Real-Time Systems Symposium (RTSS'12)*, December 2012.

[16-S] Sebastian Altmeyer, Robert I. Davis, and Claire Maiza. Improved cache related pre-emption delay aware response time analysis for fixed priority pre-emptive systems. *Real-Time Systems*, 2012.

[17-S] Will Lunniss, Sebastian Altmeyer, and Robert I. Davis. Optimising task layout to increase schedulability via reduced cache related pre-emption delays. In *Proceedings of the 20th International Conference on Real-Time Networks and Systems (RTNS'12)*, October 2012.

[18-S] Ernst Althaus, Sebastian Altmeyer, and Rouven Naujoks. Precise and efficient parametric path analysis. In *Proceedings of the 2011 conference on Languages, compilers, and tools for embedded systems (LCTES'11)*, April 2011.

[19] Pascal Montag and Sebastian Altmeyer. Precise WCET calculation in highly variant real-time systems. In *Proceedings of Design, Automation, and Test in Europe (DATE'11)*, March 2011.

[20] Sebastian Altmeyer and Claire Maiza. Influence of the task model on the precision of scheduling analysis for preemptive systems – status report. In *Proceedings of the 2nd International Real-Time Scheduling Open Problems Seminar (RTSOPS'11)*, July 2011.

[21-S] Sebastian Altmeyer, Robert I. Davis, and Claire Maiza. Cache related pre-emption aware response time analysis for fixed priority pre-emptive systems. In *Proceedings of the 32nd IEEE Real-Time Systems Symposium (RTSS'11)*, December 2011.

[22] Ernst Althaus, Sebastian Altmeyer, and Rouven Naujoks. Symbolic worst case execution times. In *Proceedings of 8th International Colloquium on Theoretical Aspects of Computing (ICTAC'11)*, July 2011.

[23-S] Sebastian Altmeyer and Claire Maiza. Cache-related preemption delay via useful cache blocks: Survey and redefinition. *Journal of Systems Architecture*, 2010.

[24] Reinhard Wilhelm, Sebastian Altmeyer, Claire Burguière, Daniel Grund, Jörg Herter, Jan Reineke, Björn Wachter, and Stephan Wilhelm. Static timing analysis for hard real-time systems. In *Procedeedings of the 11th Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI'10)*, January 2010.

[25-S] Sebastian Altmeyer, Claire Maiza, and Jan Reineke. Resilience analysis: Tightening the crpd bound for set-associative caches. In *Proceedings of the 2010 conference on Languages, compilers, and tools for embedded systems (LCTES'10)*, April 2010.

[26] Jörg Herter and Sebastian Altmeyer. Precomputing memory locations for parametric allocations. In *Proceedings of 10th International Workshop on Worst-Case Execution Time Analysis (WCET'10)*, July 2010.

[27] Sebastian Altmeyer and Claire Burguière. Influence of the task model on the precision of scheduling analysis for preemptive systems. In *Proceedings of the 1st International Real-Time Scheduling Open Problems Seminar (RTSOPS'10)*, July 2010.

[28-S] Sebastian Altmeyer, Claire Burguière, and Reinhard Wilhelm. Computing the maximum blocking time for scheduling with deferred preemption. In *Proceedings of the 1st Workshop on Software Technologies for Future Dependable Distributed Systems (STFSSD'09)*, July 2009.

[29-S] Claire Burguière, Jan Reineke, and Sebastian Altmeyer. Cache-related preemption delay computation for set-associative caches—pitfalls and solutions. In *Proceedings of 9th International Workshop on Worst-Case Execution Time Analysis (WCET'09)*, June 2009.

[30-S] Sebastian Altmeyer and Claire Burguière. A new notion of useful cache block to improve the bounds of cache-related preemption delay. In *Proceedings of the 21st Euromicro Conference on Real-Time Systems (ECRTS'09)*, July 2009.

[31-S] Sebastian Altmeyer and Gernot Gebhard. WCET analysis for preemptive systems. In *Proceedings of the 8th International Workshop on Worst-Case Execution Time Analysis (WCET'08)*, 2008.

[32] Sebastian Altmeyer, Christian Hümbert, Björn Lisper, and Reinhard Wilhelm. Parametric timing analysis for complex architectures. In *Procedeedings of the 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'08)*, August 2008.

[33] Gernot Gebhard and Sebastian Altmeyer. Optimal task placement to improve cache performance. In *Proceedings of the 7th ACM Conference on Embedded Systems Software (EMSOFT'07)*, October 2007.

Books

[34-S] Sebastian Altmeyer. *Analysis of Preemptively Scheduled Hard Real-time Systems*. epubli GmbH, 2013.

5c. Top publications

[6-S] Sebastian Altmeyer, Roeland Douma, Will Lunniss, and Robert I. Davis Evaluation of cache partitioning for hard real-time systems. In *Proceedings of the 26th Euromicro Conference on Real-Time Systems (ECRTS'14)*, July 2014.

[12-S] Sebastian Altmeyer and Robert I. Davis. On the correctness, optimality and precision of static probabilistic timing analysis. In *Proceedings of Design, Automation, and Test in Europe (DATE'14)*, March 2014.

[16-S] Sebastian Altmeyer, Robert I. Davis, and Claire Maiza. Improved cache related pre-emption delay aware response time analysis for fixed priority pre-emptive systems. *Real-Time Systems*, 2012.

[18-S] Ernst Althaus, Sebastian Altmeyer, and Rouven Naujoks. Precise and efficient parametric path analysis. In *Proceedings of the 2011 conference on Languages, compilers, and tools for embedded systems (LCTES'11)*, April 2011.

[30-S] Sebastian Altmeyer and Claire Burguière. A new notion of useful cache block to improve the bounds of cache-related preemption delay. In *Proceedings of the 21st Euromicro Conference on Real-Time Systems (ECRTS'09)*, July 2009.

Statements by the applicant

Ethical aspects

	Not applicable	Not yet applied for	Applied for	Received
Approval from a recognised medical ethics review committee	X			
Approval from an animal experiments committee	X			
Permission for research with the population screening Act	X			

- I have completed this form truthfully.
- By submitting this document I declare that I satisfy the nationally and internationally accepted standards for scientific conduct as stated in the Netherlands Code of Conduct for Scientific Practice 20121 (Association of Universities in the Netherlands).
- I have submitted non-referees.

Name: **Sebastian J. Altmeyer**

Place: **Amsterdam, The Netherlands**

Date: **January 6. 2015**