ASSUME

ITEA3

Affordable Safe & Secure Mobility Evolution

Technical-oriented talk about the principles and benefits of the ASSUME-its approach and tooling

Deliverable D6.5.1

Deliverable Information			
Nature	Document	**Dissemination Level**	State of the art
Project	ASSUME	**Project Number**	14014
Deliverable ID	D6.5.1	**Date**	05.10.2016
Status	Initiated	**Version**	0.3
Contact Person	Moharram Challenger	**Organisation**	UNIT
Phone	+90 (232) 339 6633	**E-Mail**	moharram.challenger@unitbilisim.com

Author Table

Name	Company	Email
Moharram Challenger	UNIT	Moharram.challenger@unitbilisim.com
Alexander Viehl	FZI	Viehl@fzi.de

Change and Revision History

Version	Date	Reason for Change	Affected pages
0.1	03.10.2016	Initial version	
0.2	03.10.2016	Main content is added	
0.3	07.10.2016	Introduction added	4-5
0.4	10.10.2016	Document is reviewed	

Table of Contents

1. Executive Summary

In order to foster the ASSUME viewpoint and to use it in the new fields, we engage a campaign to develop a vibrant user community for all the Open Source technologies produced in the context of ASSUME. Work on this user community culminate in a set of open-source technical talks given during the project. The aim of these talks will be to create an independent, self-sustaining and long-lived user community around the ASSUME-ITEA project, and to evangelize the principles and benefits of the approach to industrial world.

2. Introduction

In the premium market’s segments, innovative functions are the most important factor to influence buying decisions. Future mobility solutions will increasingly rely on smart components that continuously monitor the environment and assume more and more responsibility for a convenient, safe and reliable operation in parallel to an additional facilitation of energy consumption optimization and emissions reduction. Public perception moves towards higher expectations with regard to the safety of highly autonomous systems. With hands-off systems, a failure rate clearly below the one of a human actor is expected. Consequently, novel design and verification methods for such highly automated systems are needed to satisfy future safety-relevant systems requirements.

During the design process of a complex distributed system, system level requirements are broken down into more fine-grained technical hard- and software requirements and mapped to subsystem parts. During this requirement break down, various models are created to represent distinct aspects of the developed system. Hence, traceability between different abstraction levels of requirements and sys-tem parts must be established. The designs models and hence design decisions have to be verified with respect to their associated functional and safety requirements and it has to be ensured that the implementation does not violate requirements.

One major limitation today is the unavailability of synthesis and verification tools for these specific models. While research prototypes show the general feasibility of formally proving the correctness of models or code with respect to given requirements, these tools have not been adopted by the industry widely.

Tools working on source code or implementation level that check for specific errors or design flaws are available and well applied today. However, higher-level requirements and the correctness of design decisions cannot be checked effortlessly and completely with high confidence.

In addition to that, further challenges arising with the shift to more autonomy are the increasing complexity and performance requirements of autonomous systems. On the way to realize this vision, the need for computing power will drastically increase far beyond what can be provided by conventional sequential single-core hardware. While the required efficiency and scalability compels future embedded micro-controllers (µC) to rely on multi- and many-core architectures, the change in hardware architecture also necessitates fundamental advances to software development methodology. Replacing today’s essentially sequential technology by interconnected cores and omnipresent communication poses the colossal challenge in software development to identify and exploit means for concurrency still guaranteeing reliable and predictable behavior. One problem here is that analysis techniques and flows have to be extended to support parallelism and system complexity on several design and implementation levels.

Current analysis techniques are severely limited by the size and complexity of the embedded systems to be analyzed. Tools using abstract interpretation to prove the absence of runtime defects usually become difficult to use for code sizes above 200,000-300.000 lines of code depending on the used programming features and the code complexity. Model checking techniques are currently limited to much smaller sizes of the programs state space since they enumerate over all possible program states without abstraction. Moreover, tools for the analysis of concurrent and multi-core software currently present a large number of false positives to the user. Consequently, an efficient assessment of analysis results for concurrent systems in an industrial environment is currently not feasible.

Combinations of different analysis methods and tools for concurrent systems are mostly in a premature state of research and not viable for industrial application yet. One reason for this situation is the absence of standardized interfaces between verification and modeling tools that support verification tasks. Formal verification tools typically only apply one technology and support one implementation language. A close collaboration of these tools is needed allowing the exchange of analysis findings and given assumptions across modeling languages and tooling borders.

The main goal of the ASSUME project is the affordable, standard-compliant development and verification of highly automated, safety relevant, and performance critical mobility systems. A strong focus is on development methods for concurrent systems and static verification techniques. The ASSUME algorithm portfolio will be the key technology to bring innovative solutions from sandboxes into consumers' daily lives. ASSUME provides a seamless engineering methodology to overcome this roadblock. The problem is addressed on the constructive and on the analytic side. For efficient construction and synthesis of embedded systems, the project provides new tools, standards and methodologies to cover most of the challenges by design. In addition, ASSUME provides a well-integrated sound static analysis solution that allows proving the absence of problems even in a multi-core environment. New algorithms will be integrated in exploitable tools. New interoperability standards and requirements formalization standards will facilitate cooperation between different market players.

In this context, research activities are pursued and an early dissemination of the targeted and achieved results will be forced. This report gives an overview on technical talks that have been given or are planned for the next phases of the project – partially on a regular basis.

3. Technical Talks

In this section, the technical presentations are elaborated. These talks are realized in conferences, workshops or symposiums.

3.1. TU/e and Thales

Name	Suggested by	Contributors	Date
TIPS’16 Workshop (1st International workshop on the Timing Performance in Safety Engineering)	TU/e and Thales	TU/e and Thales	20 September 2016, Trondheim

Title of the talk: Dataflow-based Verification of Temporal Properties for Virtualized Multiprocessor Systems

Event full name: 1st International workshop on the Timing Performance in Safety Engineering (TIPS)

Event location: Trondheim, Norway

Date of talk: Tuesday, 20.09.2016

Organizer of the event: Thales and TU/e

Summary of the talk:
Over the last decade we have witnessed ever increasing use of virtualized multiprocessor platforms in the design of advanced digital systems. This is due to the fact that virtual platforms, by means of virtual machines, facilitate the design of complex systems involving large numbers of applications by providing both spatial and temporal isolation between them.
In particular, each application is assigned with a fraction of the platform's (spatial and temporal) capacity and can be treated as if it were executing on a platform of its one.

This means that in cases where applications have stringent temporal constraints we can analyze their temporal behavior in isolation because the behavior of one is not affected by the other.
In this talk we reflect on the model-based design flow developed at Eindhoven University of Technology that by the use of aforementioned virtualization principles guarantees composability and predictability. In particular, we discuss how timed dataflow-based design flow implemented in the SDF3 tool enables real-time dataflow applications to be automatically mapped, verified and executed on the CompSOC temporally composable platform providing strongly temporally isolated virtual multiprocessor platforms.

Estimated number of audience: 30

3.2. TUM

Name	Suggested by	Contributors	Date
Static Analysis Symposium’16	TUM	TUM	08.09.2016

Title of the talk: Enforcing Termination of Interprocedural Analysis

Event full name: Static Analysis Symposium 2016

Event location: Edinburgh, UK

Date of talk: 08.09.2016

Organizer of the event: Xavier Rival

Summary of the talk:
Interprocedural analysis by means of partial tabulation may not terminate when the same procedure is analyzed for infinitely many abstract calling contexts or when the abstract domain has infinite strictly ascending chains. As a remedy, we present a novel local solver for general abstract equation systems, be they monotonic or not, and prove that this solver fails to terminate only when infinitely many variables are encountered. We clarify in which sense the computed results are sound. Moreover, we show that interprocedural analysis performed by this novel local solver, is guaranteed to terminate for all non-recursive programs — irrespective of whether the complete lattice is infinite or has infinite strictly ascending or descending chains.

Estimated number of audience: 50

3.3. Kiel University

Title of the talk: Importance Sampling for Stochastic Timed Automata

Event full name: RP (Workshop on Reachability Problems)

Event location: Aalborg, DK

Date of talk: Tuesday, 20.09.2016

Organizer of the event: Department of Computer Science, Aalborg University

Summary of the talk:
We present an importance sampling framework that combines symbolic analysis and simulation to estimate the probability of rare reachability properties in stochastic timed automata. By means of symbolic exploration, our framework first identifies states that cannot reach the goal. A state-wise change of measure is then applied on-the-fly during simulations, ensuring that dead ends are never reached. The change of measure is guaranteed by construction to reduce the variance of

the estimator with respect to crude Monte Carlo, while experimental results demonstrate that we can achieve substantial computational gains.

Estimated number of audience: 40

3.4. THALES

Name	Responsible	Contributors	Date
RTAS conference	THALES	THALES	Vienna, Austria, 11-14 April 2016

Title of the talk: TEMPO: Integrating Scheduling Analysis in Industrial Design practices

Event full name: 22nd IEEE Real-Time and Embedded Technology and Applications Symposium

Event location: Vienna, Austria.

Date of talk: 12 April 2016

Organizer of the event: James Anderson, University of North Carolina at Chapel Hill, USA

Summary of the talk: Usually, the industrial practices rely on the subjective judgment of experienced software architects and developers to predict how design decisions may impact the system timing behavior. This is however risky since eventual timing errors are only detected after implementation and integration, when the software execution can be tested on system level, under realistic conditions. At this stage, timing errors may be very costly and time consuming to correct. Therefore, to overcome this problem we need an efficient, reliable and automated timing estimation method applicable already at early design stages and continuing throughout the whole development cycle. Scheduling analysis appears to be the adequate candidate for this purpose. However, its use in the industry is conditioned by a seamless integration in the software development process. This is not always an easy task due to the semantic mismatches that usually exist between the design and the scheduling analysis models. At Thales Research & Technology, we have developed a timing framework called TEMPO that solves the semantic issues through appropriate model transformation rules, thus allowing the integration of scheduling analysis in the development process of real-time embedded software. In this demonstration paper, we present the basic building blocks and functionalities of the TEMPO framework and describe the main visible stages in the model transformations involved.

Estimated number of audience: 30 persons

4. Conclusions and Discussion

This deliverable presents the technical talk during the project. Therefore, this document will be updated until the end of the project to report the related activities.

References

[1]

Annex A: N/A