Bosch Case Study – Second Year Report

KUN, Bosch

May 1, 2004

AMETIST DELIVERABLE 3.3.3

Project acronym: AMETIST
Project full title: Advanced Methods for Timed Systems
Project no.: IST-2001-35304
Project Co-ordinator: Frits Vaandrager
Project Start Date: 1 April 02
Duration: 36 months
Project home page: http://ametist.cs.utwente.nl/

At the beginning of Year 2 of the AMETIST project, Bosch decided to put the further development of the Car Periphery Supervision (CPS) system, i.e. the case study that was proposed as the topic of study in Task 3.3, to a halt. Consequently, the PCC decided that also within AMETIST the work on the CPS system would be finished after completion of the paper in progress on this case study.

The search for an alternative case study was somewhat delayed by the fact that, starting November 1, 2003, Stefan Kowalewski, the local coordinator of AMETIST, left Bosch to become a professor for "Software for Embedded Systems" at the Technical University in Aachen. The responsible person at Bosch for AMETIST has been Marko Auerswald from that moment on. Last month, Bosch has proposed an alternative case study to the project concerning the development of an airbag ECU, which will be tackled by the project during its third and final year.

In the remainder of this document, we will first summarize the results from year 2 on the CPS case study, and then outline content and goals for the new airbag case study that we intend to carry out during year 3.

The CPS Case Study

Below, we briefly summarize the work carried out within the AMETIST project on the case study "Real-time service allocation for Car Periphery Supervision".

The term *Car Periphery Supervision (CPS)* refers to technology for obtaining information about the environment of a car. Such technology can serve as the basis for many driver assistance services, e.g., parking assistence, pre-crash detection, blind spot supervision, lane change assistance, etc., most of which are still under development. There are different sensor technologies available for CPS realizations, e.g., ultrasonic, radar, lidar, infrared and video. In the case study we concentrate on Short Range Radar (SRR) technology. In Deliverable 3.3.1 [5], a preliminary description of the CPS system was provided by Bosch, as well as a list of timing related problems.

An initial study by partners UT and KUN made it clear that the initial description in [5] contained insufficient information to make formal analysis possible. In particular, the system requirements, assumptions on timing behavior of the various system components, and a listing of the assumptions on the environment (e.g., number and relative speed of approaching objects) were unclear. Hence a visit by a delegation from UT and KUN to the Bosch facilities in Frankfurt was arranged on March 5-7, 2003. Based on the fruitful discussions at this meeting with the domain experts from Bosch, study of relevant literature, and use of used MatLab to visualize the sensor visibility areas and to analyse possible trajectories of approaching objects, we were able to supplement the preliminary description of [5] with the necessary information to permit formal modelling and analysis [2, 4]. Ou results show once more the importance of such an analysis: they allow us to make the right abstractions in the dynamical model of the system, and to interpret correctly the results obtained by analyzing this model. The next step we took was to actually construct a formal dynamical model using UPPAAL. In this model the environment, the sensors and the ECU are all modelled as timed automata. This was achieved by splitting the value domains of nonlinear continuous variables into a finite number of regions, and overapproximating (with the help of MatLab) the possible behavior of the continuous variables in the timed automata model. In Deliverable 3.3.2 [1], we reported on our modelling efforts.

The efforts during the second year of the project concentrated on verification. By encoding the various assumptions on the environment within our UPPAAL model, we were able to establish some key correctness properties of the system (for specific choices of the system parameters):

1. the ECU has a sufficiently accurate view on what is happening in the environment.
2. When an object reaches the pre-crash region, the ECU knows about this within a small (specific) number of time units.
3. The ECU avoids false alarms.

4. The system is deadlock free.

A paper describing these results will (hopefully) be presented at WODES'04 [3].

The original aim of the case study — as stated in the technical annex — was to use the timed automata framework to specify precisely the logical and timing requirements for the CPS system, and to design and verify (or alternatively, automatically synthesize) a dynamical resource allocation scheme on this basis. In the work reported, these original aims have been achieved, although the model of [3] is is probably somewhat more abstract than what was originally intended, and our efforts were restricted to verification (rather than design and/or synthesis). The project has decided that building more detailled models of the CPS system, and contributing to the design and synthesis of resource allocation schemes, makes no sense given that the work on CPS has been abandoned by Bosch.

The Airbag ECU Case Study

The goal of the new AMETIST case study that has been proposed by Bosch is to explore — in the course of a pilot project — the suitability of using timed automata-based modeling for increasing product quality in a cost effective way. The pilot project will be attached to a current platform development project within Bosch of an airbag ECU. Within the development life-cycle, four areas of application will be explored:

- **Specification of real-time behavior**. During ECU architecture definition, components and their behavior have to be specified. The benefit of using timed automata at this point (apart of their use in the next items) is the capability of simulation.
- **Validation of the architecture specification**. As described, executable models of the components can be used to simulate system behavior for specific scenarios. A scenario-based analysis of the architecture is a common approach to validate architecture decisions.
- **Verification of safety properties in a distributed real-time control system**. Safety critical functionality usually is implemented in a redundant way with diverse channels. These channels have a logical as well as temporal control of their own. In the example of an airbag ECU, several logical or physical plausibilization channels exist to avoid inadvertent deployment. Nevertheless, these shall not hinder correct deployment. Modeling the real-time behavior of the channels and the relevant behavior of the environment will allow to identify whether specific sequences in environment could lead to conditions, under which plausibilization channels hinder the correct deployment of restraint devices.
- **Validation of the implementation**. From formal specifications of the real-time behavior of components, test cases can be derived automatically. An appropriate coverage criteria based on the behavior model can be formulated. Testing the implementation with this generated test set ensures that the implementation really implements the specified behavior and thus also fulfills the verified safety properties.

The consortium believes that this new case study constitutes a challenging problem, on which many of our methods and tools can be tested: certainly, this case study will push UPPAAL to its limits, but possibly it will also be an opportunity to demonstrate the usefulness of the work on Scenario Based Programming using Live Sequence Charts at WIS, and the recent work on generation of test sequences from timed automataton specifications at UT and VERIMAG. The key criterion for success will be the appreciation by Bosch of the usefulness and cost effectiveness of our results.

References

[1] AMETIST. Bosch case study: First year report, jun 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.3.2.pdf. Deliverable 3.3.2 from the IST project AMETIST.

[2] B. Gebremichael, H. Hermanns, T. Krilavičius, and Y.S. Usenko. Hybrid modeling of a vehicle surveillance system with real-time data processing. In *Proc. Int. Conf. on Dynamical Systems Modeling and Stability Investigation*, page 419, Kyiv, Ukraine, May 2003. Available from World Wide Web: http://www.cs.kun.nl/ita/publications/papers/biniam/gkhu.html.

[3] B. Gebremichael, T. Krilavičius, and Y. Usenko. A formal analysis of a car periphery supervision system, 2004. Available from World Wide Web: http://www.cs.kun.nl/ita/publications/papers/biniam/gku.html. Submitted to WODES 2004. Also available as Technical Report NIII-R04xx, NIII, University of Nijmegen.

[4] B. Gebremichael, T. Krilavičius, and Y.S. Usenko. Real-time service allocation for car periphery supervision: Requirements and environment analysis, 2003. Available from World Wide Web: http://www.cs.kun.nl/ita/publications/papers/biniam/gku2.html. Internal Ametist document.

[5] S. Kowalewski and M. Rittel. Real-time service allocation for car periphery supervision, 2002. Available from World Wide Web: http://ametist.cs.utwente.nl/RESEARCH/AMETIST_CPSPrelimDescription_1_0.pdf. Deliverable 3.3.1 from the IST project AMETIST.