AVACS: Automatic Verification and Analysis of Complex Systems Highlights and Lessons Learned

Werner Damm(✉)

AVACS Coordinator,
Carl von Ossietzky Universität Oldenburg, Oldenburg, Germany
werner.damm@offis.de

This talk presents highlights and lessons learned from the Transregional Collaborative Research Center AVACS, funded by the German Science Foundation under contract SFB-TR 14 from January 1, 2004 to December 31, 2015 with a total funding of about 30 Million €, involving between 18 to 22 principal investigators at the three AVACS sites Freiburg, Oldenburg and Saarbrücken. Through this funding the German Science Foundation provided an excellent environment for foundational cross-site research in the highly relevant and challenging research are of Automatic Verification and Analysis of Complex Systems.

The AVACS project (see www.avacs.org) addresses the rigorous mathematical verification and analysis of models and realizations of complex safety-critical computerized systems, such as aircraft, trains, cars, or networked systems of these, whose failure can endanger human life. Our aim is to raise the state of the art in automatic verification and analysis techniques (V&A) from a level, where it is applicable only to isolated facets of the underlying space of mathematical models, to a level allowing a comprehensive and holistic verification of such systems:

1. We investigate the mathematical models and their interrelationship, as they arise at the various levels of design of safety-critical computerized systems. Behavioral models range from classical nondeterministic transition systems to probabilistic, real-time, and hybrid system models, to models reflecting the dynamic evolution of the system communication structure.
2. The investigated classes of models cover all typical system structures in this application domain, describing how to build models of complex systems hierarchically from such classes of models. These include distributed target architectures (such as hierarchical bus structures connecting multiple electronic control units), task models (task structures coming with communication and timing requirements), specification models of electronic control units (such as captured in Matlab/Simulink), system models (e.g., of vehicles), and models of systems of systems (e.g., for coordinated vehicle maneuvers).
3. The investigated classes of time models are expressive enough to cover all layers of the design space of such applications, including physical latencies of vehicles in performing coordinated maneuvers, system-level timing requirements such as response times to external events and timeliness requirements for protocols,

N. Bjørner and F. de Boer (Eds.): FM 2015, LNCS 9109, pp. 18–19, 2015.
DOI: 10.1007/978-3-319-19249-9_2

dense-time closed-loop models of controllers and plants, discrete-time design models of controllers, end-to-end deadlines on task chains, and worst-case execution times of tasks on modern processor architectures.

4. We provide largely automatic techniques to verify or falsify the compliance of models expressed in this rich model space against classes of requirements subsuming timeliness, safety, probabilistic reachability, stability and other classes of requirements, formalized in suitable logics.
5. We provide methods and tools for building such formal proofs for complete systems from guarantees of subsystems, ultimately striving to relate top-level requirements, such as for performing coordinated vehicle maneuvers to avoid collisions, to worst-case execution times of the tasks implementing control functions for such maneuvers.

http://www.springer.com/978-3-319-19248-2

FM 2015: Formal Methods
20th International Symposium, Oslo, Norway, June 24-26, 2015, Proceedings
Bjørner, N.; de Boer, F. (Eds.)
2015, XVI, 610 p. 156 illus., Softcover
ISBN: 978-3-319-19248-2