Verification of Hybrid Systems

(Invited Tutorial)

Martin Fränzle

Department Informatik
Carl von Ossietzky Universität Oldenburg
D-26111 Oldenburg, Germany
`fraenzle@informatik.uni-oldenburg.de`

Embedded digital systems have become ubiquitous in everyday life. Many such systems, including many of the safety-critical ones, operate within or comprise tightly coupled networks of both discrete-state and continuous-state components. The behavior of such *hybrid discrete-continuous systems* cannot be fully understood without explicitly modeling and analyzing the tight interaction of their discrete switching behavior and their continuous dynamics, as mutual feedback confines fully separate analysis to limited cases. Tools for building such integrated models and for simulating their approximate dynamics are commercially available, e.g. Simulink with the Stateflow extension¹. Simulation is, however, inherently incomplete and has to be complemented by *verification*, which amounts to showing that the coupled dynamics of the embedded system and its environment is well-behaved, regardless of the actual disturbance and the influences of the application context, as entering through the open inputs of the system under investigation. Basic notions of being well-behaved demand that the system under investigation may never reach an undesirable state (*safety*), that it will converge to a certain set of states (*stabilization*), or that it can be guaranteed to eventually reach a desirable state (*progress*).

Within this tutorial, we concentrate on automatic verification and analysis of hybrid systems, with a focus on fully symbolic methods manipulating both the discrete and the continuous state components symbolically. We provide an introduction to hybrid discrete-continuous systems, demonstrate the use of predicative encodings for compactly encoding operational high-level models, and continue to a number of methods for automatically analyzing safety, stability, and progress. These methods entail semi-decision and approximation methods for dealing with the general undecidability and for improving scalability, corresponding data structures and decision diagrams, the use of advanced arithmetic constraint solvers for manipulating large and complex-structured Boolean combinations of undecidable arithmetic constraints, and the automatic generation of Lyapunov-like witness functions for stability and progress.

The tutorial in particular provides an overview over recent results of the Transregional Collaborative Research Center “Automatic Verification and Analysis of Complex Systems” (AVACS)² pertaining to hybrid systems.

¹ <http://www.mathworks.com/products>

² Supported by the German Research Foundation (DFG) under contract SFB/TR 14 AVACS, see www.avacs.org