

Automated Testing

Rance Cleaveland

University of Maryland, USA

In model-based testing, (semi-)formal models of systems are used to drive the derivation of test cases to be applied to the system-under-test (SUT). The technology has long been a part of the traditional hardware-design workflows, and it is beginning to find application in embedded-software development processes also. In automotive and land-vehicle control-system design in particular, models in languages such as MATLAB(r) / Simulink(r) / Stateflow(r) are used to drive the testing of the software used to control vehicle behavior, with tools like Reactis(r), developed by a team including the speaker, providing automated test-case generation support for this endeavor.

This talk will discuss how test-case generation capabilities may also be used to help verify that models meet formal specifications of their behavior. The method we advocate, Instrumentation-Based Verification (IBV), involves the formalization of behavior specifications as models that are used to instrument the model to be verified, and the use of coverage testing of the instrumented model to search for specification violations. The presentation will discuss the foundations of IBV, the test-generation approach and other features in Reactis that are used to support IBV, and the results of several case studies involving the use of the methods.