The Future of Model-based Testing

Jan Peleska
University of Bremen and Verified Systems International GmbH, Bremen
peleska@uni-bremen.de
With contributions by
Felix Brüning, Mario Gleirscher, Wen-ling Huang, Niklas Krafczyk, and Robert Sachtleben

February 3, 2023

Extended Abstract

Since the 1970s, model-based testing (MBT) has been comprehensively investigated in the research communities, and its industrial application has been fostered for at least two decades. Though convincing success stories about MBT have been published, we have to admit that its acceptance in industry is still not as good as it should be. This is reflected, for example, by the facts that commercial MBT tools are not selling as well as had been originally envisaged by the "campaigners of MBT" (such as myself) and that MBT has not been integrated into the verification plans of the key players in the embedded systems world.

In the first part of this presentation, the reasons for this limited success are analysed: some of them are related to the observation that the more general approach of model-based systems engineering (MBSE) has not become the silver bullet for systems development as originally claimed by the advocates of MBSE and their tool vendors. Another reason is given by the fact that automated test generation tools – event though they may guarantee a certain amount of test strength in relation to a specific fault model – will not necessarily produce test cases that are intuitively convincing to verification engineers and certification authorities. Indeed, the automated creation of "meaningful" end-to-end system tests from models seems to be an open problem that has been insufficiently recognised in the research communities until today. Another reason lies in the fact that the current standards for the verification of safety-critical control systems for railways, avionics, and automotive demand a requirements-based verification approach that is well-aligned to property-oriented testing. The verification by showing conformance of a system under test to its model, however, is only accepted, if requirements can be comprehensively traced to the model and from there to the generated test cases. Consequently, MBT seems more like a detour from the perspective of these standards. Finally – to state this bluntly – programming experts in industry are often not very good at writing models.

While we currently do not see a "silver bullet solution" for the general problem of insufficient MBSE acceptance and insufficient modelling formalisms, we advocate a

novel combination of different existing test approaches in order to create a method of applying “MBT without writing models”. First, property-oriented test cases are created from the requirements and formalised in LTL in a semi-automated way. By translating the formulae into Omega automata, formula models that are suitable for testing a given property are created in an automated way. Second, these tests are executed and mutated, following the principles of coverage guided fuzz testing. The coverage guidance is not obtained from software code (since the approach should be applicable to systems testing), but from a model that is learnt in the background during execution of the property-oriented tests. When the generation of new test cases by fuzzing has refined the learnt model in a sufficient way, it can be shown that the test cases used so far have a well-defined strength. Therefore, it is no longer necessary to apply model checking on the created model, as has been advocated by most of the existing strategies combining testing and model learning. We are currently working on a theory showing that under certain hypotheses about the system under test, certain degrees of model refinement during the learning process guarantee that the system implements the test properties correctly.