Automated Environment-Model Based Testing with Intelligent Methods in an Industrial Setting

Annamária Szenkovits

Cluj-Napoca Faculty of Mathematics and Computer Science, Babeş-Bolyai University

szenkovitsa@cs.ubbcluj.ro

PhD Research Report. Model-based testing can facilitate the process of test automation, thus, it can significantly reduce the costs of testing. We present a testing framework where model-based testing is used in combination with search-based methods in order to automate test input generation. Our framework is optimized for the reactive scenario, where the System Under Test (SUT) is in continuous interaction with its environment. During its life-cycle, the SUT continuously reads the input data coming from the environment, updates its internal state, then writes the outputs to its actuators, by which it acts upon its environment. The main idea of our approach can be summarized as follows:

We create a non-deterministic, executable environment model based on the system specifications. This environment model is used for stimulating the system under test (SUT), but also to automatically generate test inputs. The language proposed for creating the environment model is Lutin, an automatic test generator for reactive programs, which enables us to perform guided random exploration of the environment's state space [**?**]. Guidance (in test case generation) can be accomplished by making use of certain in-built features of the Lutin language which effectively offer the possibility to parameterize the process which generates test inputs during testing. To achieve high model coverage and to target specific internal state regions of the SUT we fine-tuned the Lutin environment parameters so that the test cases generated and executed against the SUT would cover the structure of the SUT code as much as possible. In order to achieve this, we used search based techniques like Differential Evolution [**?**], adaptive Differential Evolution [**?**] and Reinforcement Learning.

In order to investigate the effectiveness of our method, we present a case study involving a real-life, industrial system from the domain of railway automation. Numerical experiments conducted on this system illustrate that our approach can be used efficiently to increase the structural coverage of the SUT.

References

[1] Raymond, P., Roux, Y., Jahier, E.: Lutin: A language for specifying and executing reactive scenarios. EURASIP J. Emb. Sys. **2008** (2008)

[2] Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization **11** (1997) 341–359

[3] J. Zhang and A. C. Sanderson, "Jade: adaptive differential evolution with optional external archive," *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 5, pp. 945–958, 2009.