Constructing Formal Models through Automata Learning

Authors: Fides Aarts, Faranak Heidarian, Frits Vaandrager

Abstract: Model-based system development is becoming an increasingly important driving force in the software and hardware industry. The construction of models typically requires specialized expertise, is time consuming and involves significant manual effort, implying that in practice often models are not available, or become outdated as the system evolves. In practice, 80% of software development involves legacy code, for which only poor documentation is available. Manual construction of models of legacy components is typically very labor intensive and often not cost effective. The solution is to infer models automatically through observations and test, that is, through black box reverse engineering.
State-of-the-art tools for active learning of state machines are able to learn state ma- chines with at most in the order of 10.000 states. This is not enough for learning models of realistic software components which, due to the presence of program variables and data parameters in events, typically have much larger state spaces. Abstraction is the key when learning behavioral models of realistic systems. Hence, in most practical applications where automata learning is used to construct models of software components, researchers manually define abstractions which, depending on the history, map a large set of concrete events to a small set of abstract events that can be handled by automata learning tools. We show how such abstractions can be constructed fully automatically for a class of extended finite state machines in which one can test for equality of data parameters, but no operations on data are allowed. Our approach uses counterexample-guided abstraction refinement: whenever the current abstraction is too coarse and induces nondeterministic behavior, the abstraction is refined automatically. Using a prototype implementation of our algorithm, we have succeeded to learn fully automatically models of several realistic software components, including the biometric passport and the SIP protocol.

References:
1. F. Aarts, B. Jonsson, and J. Uijen. Generating models of infinite-state communication protocols using regular inference with abstraction. In A. Petrenko, J.C. Maldonado, and A. Simao, editors, 22nd IFIP International Conference on Testing Software and Systems, Natal, Brazil, November 8-10, Proceedings. IFIP, 2010.
2. F. Aarts, J. Schmaltz, and F.W. Vaandrager. Inference and abstraction of the biometric passport. In Proceedings 4th International Symposium On Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2010), 18-20 October 2010
3. F. Aarts and F.W. Vaandrager. Learning i/o automata. In P. Gastin and F. Laroussinie, editors, 21st International Conference on Concurrency Theory (CONCUR), Paris, France, August 31st - September 3rd, 2010, Proceedings, volume 6269 of Lecture Notes in Com- puter Science, pages 7185. Springer, 2010.

Biographies of the authors:

1- Fides Aarts is a PhD student at the Radboud University in Nijmegen. After she has finished her Master thesis in Uppsala on inference and abstraction of communication protocols under supervision of Bengt Jonsson, she continued working on learning automata. Currently, she is involved in automatically generating abstractions of automata to extent inference to systems with large parameter domains.

2- Faranak Heidarian is a last year PhD student in Radboud University Nijmegen. She works in MBSD group and her research is about abstraction refinement. She a has bachelor degree in software engineering, and a Masters degree in Computer Science from Sharif University of Technology, Tehran, Iran.

3- Frits Vaandrager has a strong interest in the development and application of theory, (formal) methods and tools for the specification and analysis of computer based systems. In particular, he is interested in real-time embedded systems, distributed algorithms and protocols.
Together with Lynch, Segala, and Kaynar he developed the (timed, probabilistic and hybrid) input/output automata formalisms, which are basic mathematical frameworks to support description and analysis of computing systems. He has been and is involved in a large number of projects in which formal verification and model checking technology is applied to tackle practical problems from industrial partners. Within the OCTOPUS project with Oc\'{e} he is currently involved in the construction of the DSEIR toolset for model-driven design-space exploration for embedded systems. Recently, he has also become very interested in automata learning.