Conformance Testing of Cyber-Physical Systems

Houssam Abbas

In a typical Model-Based Design (MBD) process for Cyber-Physical Systems, an initial 'simple' *Model* is successively refined and made more accurate and complex; then it is implemented on a real-time computational platform, and further modified to yield an *Implementation*. The goal is to produce a system that satisfies a formal specification Φ. This successive refinement raises the question of how "close" are the "simple" Model and the "complex" Implementation. Answering this question is important because it is not always possible to verify formally that the Implementation satisfies the specification Φ. Moreover, even if the Implementation satisfies Φ, it will have unspecified behavior which might exhibit bugs. By quantifying the 'closeness' between Model and Implementation, our level of confidence in the Implementation derives from our confidence in the Model, and the fact that the Model satisfies Φ.

Because formal analysis of non-deterministic models is rarely utilized and supported by industry tools, language inclusion can not be used to answer this question. Thus, an appropriate notion of closeness, or *conformance*, between Model and Implementation is required. *Conformance testing* is the process of establishing whether behaviors exhibited by Model and Implementation are conformant. Existing works apply only to certain classes of systems and rely on the full knowledge of the mathematical representations of Model and Implementation, often not available for industrial CPS.

In this work, we give a rigorous mathematical definition of conformance between two output trajectories $\mathbf{y}_M$ and $\mathbf{y}_I$ of the Model and Implementation, resp., when driven from the same initial conditions, with the same control input. We term this conformance notion $(T, J, (\tau, \varepsilon))$-closeness. Its distinctive feature is that it measures the difference between $\mathbf{y}_M$ and $\mathbf{y}_I$ in both space and time. Coarsely, two output trajectories are conformant with degree $(\tau, \varepsilon) \in \mathbb{R}_+^2$, over a hybrid time-horizon $(T, J) \in \mathbb{R}_+ \times \mathbb{N}$, if every $\mathbf{y}_M$-point has a $\mathbf{y}_I$-point ε-close to it within a window of width 2τ, and vice-versa. Several application-dependent notions of system similarity can be shown to be implied by $(T, J, (\tau, \varepsilon))$-closeness.

Using $(T, J, (\tau, \varepsilon))$-closeness, it is possible to perform the following MBD tasks in a rigorous manner:

◦ Define conformance between a Model and Implementation, which are said to be conformant with degree $(T, J, (\tau, \varepsilon))$ iff given the same initial conditions, and the same input signal, they produce trajectories that are $(T, J, (\tau, \varepsilon))$-close.

◦ Given a tuple (τ, ε), determine whether the Model and Implementation are conformant by performing a search over the initial conditions and input signal space for two trajectories that are not (τ, ε)-close. If such a pair is found, then the Implementation needs to be revised to conform to the Model.

◦ Given T and J, determine a smallest pair (τ, ε) such that the two systems are (τ, ε)-close. Such a smallest value is termed the *degree of conformance* between the two systems.

We demonstrate the above tasks on an industrial automatic transmission, where the Model is in Simulink, and the Implementation is a high-fidelity engine model from Simuquest with a large number of black box components. Using our methods, we can reliably approximate the degree of conformance between the two systems using stochastic optimization techniques.

We then give automatic formula re-writing rules such that, if a Model trajectory $\mathbf{y}_M$ satisfies the formula φ, and the corresponding Implementation trajectory $\mathbf{y}_I$ is (τ, ε)-close to it, then $\mathbf{y}_I$ satisfies the re-written formula. This alleviates the testing requirement greatly, since the Implementation need not, in principle, be tested for those requirements.

Stochastic optimization heuristics are sometimes the only way to deal with complex optimizations, especially ones where the objective function is only available as a black-box. However, one downside of using stochastic optimization is that the guarantees they provide are probabilistic and usually asymptotic in nature. So future work will focus on developing methods for computing the conformance degree with non-probabilistic guarantees, which can be used in a formal verification framework. Another challenge is to fully incorporate conformance in system modeling: i.e., how to model systems with a view towards the Implementation satisfying a specification? Finally, it will be interesting to adapt monitoring algorithms to provide, not only satisfaction of the current trace, but also satisfaction of trajectories that are $(T, J, (\tau, \varepsilon))$-close to it.

Details of parts of this work are available online as report arXiv:1401.5200. This work benefited from the input of Raymond Turin at SimuQuest, and was partially funded under NSF awards CNS 1116136, CNS 1319560, IIP-0856090 and the NSF I/UCRC Center for Embedded Systems.

H. Abbas is with the School of Electrical, Energy and Computer Engineering, Arizona State University e-mail: hyabbas@asu.edu

Parts of this work are joint work with Bardh Hoxha and Georgios Fainekos (School of Computing, Informatics, and Decision Systems Engineering, ASU), and James Kapinski, Jyotirmoy Deshmukh and Koichi Ueda (Toyota Model-Based Design group)