Modeling Unpredictable or Random Environments

Jeannette M. Wing*
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213-3890

Context: Formal Specification and Verification

Given formal specifications of a system and a desired property of that system, the verification question is "Does this property hold of this system?" My long-standing interest in the specification and verification area has recently turned to autonomous and embedded systems. Typical autonomous systems are unmanned spacecraft and planetary exploration robots. Typical embedded systems are controllers found in aircraft and automobiles, though they are increasingly found in household appliances and consumer electronics.

There are some technically challenging formal modeling problems that arise from these kinds of applications because of their operating environments:

- The system's environment is *unknown* or *unpredictable*. For example, for robots exploring Mars, we cannot predict all environmental conditions or events. Any system which interacts with humans must deal with unpredictable behavior, not easily formalized.
- The system's environment is *random*. Even if we were able to identify all possible environmental events, their likelihoods of occurrence might not be uniformly distributed; more usually, there is a high variance between occurrences of normal and those of exceptional events. For example, while we might explicitly model the possibility of a power surge due to thunderstorms (or some other act of Mother Nature), we would expect such events to be rare.
- The system's environment is *continuous* and *dynamic*. For example, temperature, air pressure, and wind speed are continuous quantities that vary over time. A control system may be monitoring consumption of continuous resources such as fuel or regulating flow of continuous output such as water.

We use *hybrid systems* to handle the third set of environmental challenges: mode changes occur discretely and within each mode, different control laws defined over continuous variables model system behavior. To address the first two sets of environmental challenges, we are investigating the use of stochastic models, e.g., Markov Decision Processes.

Modeling Unpredictable Behavior

Let's consider two traditional techniques for modeling the environment, where its behavior is unknown or unpredictable.

1. *Open model approach* ("all bets are off"): Here, we carefully model the system by placing pre-conditions on each of its state transitions. Verification is relative to these pre-conditions holding; if they don't hold–because of unknown or unpredictable environmental behavior–all bets are off. At least we are not guaranteeing anything outside of that which we modeled.
2. *Closed model approach* (catchall "fault" states and "unexpected event" state transitions): Here, we include in the model of the system a catchall "fault" state, which can be reached from every other state nondeterministically. We similarly include for each state in the model of the environment, a nondeterministic catchall "unexpected event" state transition, which whenever taken causes the system correspondingly to go to its "fault" state. Verification is relative to the composition of the system and environment. Anything not modeled by either of the two components (system and environment) or by the composition of the two is outside the boundary of discourse.

   Aside: at this level of discussion, whether the system is modeled to have two kinds of nondeterministic choice is a detail; e.g., in CSP a system may nondeterministically make an internal choice or the environment may make the system move nondeterministically because of an external choice.

Both approaches are not completely satisfying. In the first case, by definition we will not be able to say any-

*This research is sponsored in part by the Defense Advanced Research Projects Agency and the Army Research Office (ARO) under contract no. DAAD19-01-1-0485. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the DOD, ARO, or the U.S. Government.

thing about unknown behavior. It's a conservative approach because it qualifies under what conditions we answer "yes" to the verification question (i.e., only when the pre-conditions hold). In the second case, we will almost always answer "no" because we can always take the environment's "unexpected event" transition and go to the system's "fault" state. Answering "no", however, is uninteresting. What we really want to be able to assert is that under the highly likely normal conditions, the system will behave correctly.

Modeling Stochastic Behavior

To address the second set of challenges in modeling a system's environment, specification and verification researchers have used probabilistic models, e.g., stochastic transition systems (de Alfaro *et al.* 2000) and probabilistic I/O automata (Segala & Lynch 1994; Wu, Smolka, & Stark 1997) and probabilistic logics, e.g., Probabilistic Computation Tree Logic (Bansson & Jonsson 1994) and Probabilistic Branching Time Logic (Baier & Kwiatkowska 1998). For example, PCTL formulae are interpreted over finite state discrete-time Markov chains. The logic is very expressive and has a model checking algorithm that can be implemented using techniques based on Multi-Terminal Binary Decision Diagrams (Bianco & de Alfaro 1995; Clarke *et al.* 1993). MTBDDs represent transition probability matrices and differ from Binary Decision Diagrams in that the leaves may have values other than 0 and 1; in particular, the leaves contain transition probabilities. Others (Bozga & Maler 1999) use Probabilistic Decision Graphs to represent probability functions; and Conditional PDGs, Markov transition functions. PDGs label all nodes, not just leaf nodes, with probabilities; CPDGs extend PDGs to handle undetermined variables.

Unfortunately, in my preliminary discussions with colleagues, some of whom invented the techniques and data structures discussed above, I get a very skeptical response toward the use of stochastic models to reason about uncertainty or randomness. I would like to understand why:

- Is it because the state spaces are so unreasonably large to be impractical? How impractical?
- Is it because no one knows where the numbers, i.e., probability distributions, come from?
- Is it because the mathematical models are inadequate in some ways? What ways? Or perhaps too unwieldy? Why?

What I Hope to Learn from the Workshop

What I would like to get from this workshop is a better sense of the foundational and practical limitations of models, in particular stochastic models, for reasoning about uncertainty. Of these current limitations, what kind of progress is reasonable to expect in the near future? What are the pitfalls I should avoid in trying to apply these models?

References

Baier, C., and Kwiatkowska, M. 1998. Model checking for a probabilistic branching time logic with fairness. *Distributed Computing* (11):125–155.

Bansson, H., and Jonsson, B. 1994. A logic for reasoning about time and probability. *Formal Aspects of Computing* 6(5):512–535.

Bianco, A., and de Alfaro, L. 1995. Model checking of probabilistic and nondeterministic systems. In *Foundations of Software Technology and Theoretical Computer Science*, LNCS 1026, 499–513. Springer Verlag.

Bozga, M., and Maler, O. 1999. On the representation of probabilities over structured domains. In *Proc. of CAV'99*, LNCS 1633. Springer-Verlag.

Clarke, E.; Fujita, M.; McGeer, P.; Yang, J.; and Zhao, X. 1993. Multi-terminal binary decision diagrams: An efficient data structure for matrix representation. In *IWLS 93: International Workshop on Logic Synthesis.*

de Alfaro, L.; Kwiatkowska, M.; Norman, G.; Parker, D.; and Segala, R. 2000. Symbolic model checking of probabilistic processes using mtbdds and the kronecker representation. In *TACAS 2000.* Springer Verlag.

Segala, R., and Lynch, N. 1994. Probabilistic simulations for probabilistic processes. In *CONCUR'94*, LNCS 836, 481–496. Springer Verlag.

Wu, S.-H.; Smolka, S.; and Stark, E. 1997. Composition and behaviors of probabilistic I/O automata. *Theoretical Computer Science* (176):1–38.