Automatic Refinement and Vacuity Detection for Symbolic Trajectory Evaluation

Orna Grumberg
The Technion, Haifa, Israel

Temporal logic model checking is an efficient procedure that receives a model of a system and a property written in temporal logic. It returns "yes", if the system satisfies the property and "no", otherwise.

Model checking procedures typically suffer from high space requirements. One of the most successful approach to reducing these requirements is abstraction, which produces a smaller abstract model by hiding (abstracting away) some of the system's details. The abstraction is guaranteed to be conservative in the sense that every property true of the abstract model is also true of the concrete, full, model of the system. For abstractions defined over 3-valued semantics it also holds that whenever the abstract model falsifies the property, the concrete model falsifies it as well. Only when the model checking result is inconclusive, the abstract model needs to be refined by adding more system details into the model.

In these lectures we will present a powerful model checking technique called Symbolic Trajectory Evaluation (STE), which is particularly suitable for hardware. STE is applied to a circuit M, described as a graph over *nodes* (gates and latches). The specification consists of assertions in a restricted temporal language. The assertions are of the form $A \Rightarrow C$, where the *antecedent* A expresses constraints on nodes n at different times t, and the *consequent* C expresses requirements that should hold on such nodes $(n,\ t)$. *Abstraction* in STE is derived from the specification by initializing all inputs not appearing in A to the X ("unknown") value. A fourth value, $\perp$, represents a contradiction between the constraint of A on some node $(n,\ t)$ and its actual behavior in M . A *refinement* amounts to changing the assertion in order to present node values more accurately.

STE has been in active use in industry, and has been very successful in verifying huge circu its containing large data paths.

We will start by giving the needed background on temporal logic model checking. We will then describe symbolic simulation and the specific type of abstraction, used in STE. We will propose a technique for automatic refinement of assertions in STE, in case the model checking results in X. We will also define the notion of hidden vacuity for STE and suggest several methods for detecting it.

If time allows we will extend the discussion to *Generalized* STE (GSTE) which can express all ω-regular properties.