Temporal Reasoning about Program Executions

Rajeev Alur

University of Pennsylvania

While temporal verification of programs is a topic with a long history, its traditional basis —semantics based on word languages— is ill-suited for modular reasoning about procedural programs. We address this inadequacy by defining the semantics of procedural (potentially recursive) programs in terms of *languages of nested words* and developing a framework for temporal reasoning around it. This generalization has two benefits. First, this style of reasoning naturally unifies Manna-Pnueli-style temporal reasoning with Hoare-style reasoning about structured programs. Second, it allows verification of "non-regular" properties of specific procedural contexts—for example, "if a procedure acquires a lock, then the same invocation releases it before returning." In this talk, we will first discuss the *Nested Word Temporal Logic*, a temporal logic for infinite nested words, and argue that it is the "right" logic for temporal reasoning about procedural programs based on theoretical results about decidability of the propositional fragment and first-order expressive-completeness. Then, we will present, sound and relatively complete, proof rules for a variety of classes of properties such as *local safety*, *local response*, *global safety*, and *staircase reactivity*.

This talk is based on joint work reported in the following publications:

1. Adding nesting structure to words, full version under journal review (with P. Madhusudan).
2. First-order and temporal logics for nested words, *Logical Methods in Computer Science* (LMCS) **4(4: 11)**, 2008 (with M. Arenas, P. Barcelo, K. Etessami, N. Immerman, and L. Libkin).
3. Temporal reasoning for procedural programs, Pennsylvania State University Technical Report CSE-08-015, 2008 (with S. Chaudhuri).

L. de Alfaro (Ed.): FOSSACS 2009, LNCS 5504, p. 15, 2009.