Combination of Isabelle/HOL with Automatic Tools

Sergey Tverdyshev *
Saarland University, Germany
deru@wjpserver.cs.uni-sb.de

Abstract. We describe results and status of a sub project of the Verisoft [1] project. While the Verisoft project aims at verification of a complete computer system starting with hardware and up to user applications, the goal of our sub project is an efficient hardware verification.

We use the Isabelle theorem prover [2] as the major tool for hardware design and verification. Since many hardware verification problems can be efficiently solved by automatic tools, we combine Isabelle with model checkers and SAT solvers. This combination of tools speeds up verification of hardware and simplifies sharing of the results with verification of the whole computer system. To increase the range of problems which can be solved by external tools we implemented in Isabelle several algorithms for handling uninterpreted functions and data abstraction.

The resulting combination was applied to verify many different hardware circuits, automata, and processors.

In our project we use open source tools that are free for academical and commercial purposes.

1 Introduction

In large verification projects such as verification of a complete computer system the linking of verification results from different parts plays a major role. Specifying and proving all theorems within one environment, e.g. a higher order logic (HOL) theorem prover, makes linking a lot easier. Such a combination is also much safer because verification gaps, due to a manual transfer of the results from one system into another, are excluded. This was one of the motivations for this work.

In a long-term project Verisoft we are currently working on verification of a computer system starting with hardware, going through compiler, operating system kernel, operating system and up to end user applications. The main verification tool for all parts of the project is the Isabelle theorem prover for higher order logic. Because many hardware verification problems can be efficiently solved by external automatic tools, we combined Isabelle with the NuSMV model checker [3] and SAT solvers. In this paper we describe the result of the combination and demonstrate applications of this combination for hardware verification.

* Supported by The Verisoft Project under grant 01 IS C38 of the German Ministry for Education and Research (BMBF)
2 Related Work

The most recent combination external tools into Isabelle was done through input-output automata [4]. In this work the user gives a model and manually defines its abstraction. Then an LTL model checker is used to prove temporal properties of the abstracted model and a $\mu$-calculus model checker is used to check forward simulations between these two models. A drawback is that defining a suitable abstraction for a big model can be a very hard task. On the contrary, our approach does not have this disadvantage because an abstracted model is derived fully automatically.

An interesting ongoing work of L. Paulson’s group [5] is the integration of the first order theorem prover SPASS [6] into Isabelle. A highlight of the integration is that SPASS proofs can be converted into Isabelle proofs and then rechecked by Isabelle. In this approach the user does not have to trust an external tool, and soundness of the translation can be guaranteed. However, at the moment the integration is experimental and has only a very basic functionality.

The UCLID system [7] is another interesting tool that can handle big problems with great automation. It also has a lot of built-in features, e.g. handling of uninterpreted functions, efficient algorithms for term reduction. The UCLID system is mostly used for verification of invariants of a system (safety properties) but liveness properties (directly) are missing [7]. Even though the UCLID system is more powerful than our tool, our approach allows verification of liveness properties directly. Furthermore we believe that complete automatic abstraction, as we implemented, is more suitable for an automatic proof tool. Integration with Isabelle increases the domain of application of our tool but we have to pay for that with user’s involvement in proof process. Last but not the least: UCLID is distributed as a close source system with a strict license.

The rest of the paper is organized as follows. In the Section 3 we present the used tools. Section 4 provides subset of the HOL we use to specify hardware. In Section 5 the main functionalities of the translation tool are presented. Section 6 reports results of applications of the resulting system to hardware verification.

3 Tools

The Theorem Prover. Isabelle [2] is a generic theorem prover that supports several object logics. We use Isabelle with its instantiation of HOL. We refer to it as Isabelle/HOL.

The Modelchecker. NuSMV [3] is a symbolic model checker for CTL and LTL properties. It can perform bounded model checking using an external SAT solver. We used NuSMV to verify temporal properties of our models and as an external BDD decision procedure.

SAT Solvers. We implemented an algorithm to convert given problems into propositional logic extended by uninterpreted functions and linear arithmetic. Using this algorithm we can easily bind almost any SAT solver.

We bind these tools to Isabelle/HOL through the Oracle interface. Translation of a problem from Isabelle/HOL into language of an external tool is done by the translation tool.