New Directions in Symbolic Model Checking for Real-Time Systems (Progress Report)

Parosh Aziz Abdulla and Julien D'Orso

Original Objectives In the original description of the project, the following objectives are mentioned:

1. Using SAT-solvers instead of BDDs for symbolic model checking.
2. Extending existing bounded model checking techniques to real-time systems.
3. Combining SAT-solvers with other techniques such as partial orders to achieve efficient verification.
4. Applications to PLC-programs.

Progress The work of the project has been run in parallel with the ASTEC project *Symbolic Model Checking Using Stålmarck's Method.*

Since the project started one year ago, We have made the following progress regarding the four objectives stated above:

1. In [ABE99], we describe a tool called FixIt which adapts standard algorithms for symbolic reachability analysis to work with SAT-solvers instead of the classical technique of BDDs. BDDs have been the dominating tool in symbolic model checking for the last 15 years. The key element of our contribution is the combination of an algorithm that removes quantifiers over propositional variables and a simple representation that allows reuse of subformulas. The result will in principle

allow many existing BDD-based algorithms to work with SAT-solvers. We show that even with our relatively simple techniques it is possible to verify systems that are known to be hard for BDD-based model checkers. The paper [ABE99] received the best paper award of EAPLS – the European Association for Programming Languages and Systems, at the ETAPS conferences in 2000. The ETAPS conferences receive a total of several hundred submissions.

2. The approach of bounded model checking is dependent on checking a certain formula called *diameter* of the system [BCC+99, BCCZ99]. Unfortunately this formula often turns out to be hard to *prove* unsatisfiable, which makes BMC incomplete in practice. In contrast, the method described in [ABE99] is complete, and therefore more suitable to be extended to real-time systems. This represents a slight change of approach compared to the original description, i.e., we shall use SAT-based model checking rather than bounded model checking. In [AIN00b, AIN00a], we take the first step towards extending the method of [ABE99] to the context of infinite-state systems (in which real-time systems is a special case). More precisely, we show how to use propositional SAT-solvers to check safety properties of unbounded (infinite) Petri nets. Furthermore, in [AN00, AN99] we show how we can extend techniques for verifying Petri nets to *timed Petri nets*. Timed petri nets are a powerful computation model for describing real-time systems. The next step will be to combine the approaches of [ABE99], [AIN00b, AIN00a], and [AN00, AN99] to invent efficient verification algorithms for timed petri nets.

3. While SAT-based model checkers seem to behave very well in the case of deterministic systems, BDDs still cope much better in the case of systems with a lot of nondeterminism [WBCG00]. Examples of such systems are asynchronous systems such as communication protocols and distributed systems. One the main challenges currently facing the SAT-based technique is therefore to limit the formula size explosion which occurs when analyzing non-deterministic system. We are currently modifying the tool described in [ABE99], integrating the approach of *partial methods*. Partial methods have been applied successfully to algorithms which use an explicit representation of the state space. Our contribution will be to combine SAT-based model checking

with partial methods to achieve efficient verification of systems with non-deterministic behaviour.

4. We have implemented a compiler which translates a subset of the standard PLC language to propositional formulas. We can then use our SAT-based techniques as a proof engine for verifying PLC programs.

Activities of PhD student Following the original project plan we have recruited a PhD student (Julien D'Orso). Also, according to the plan, most of the first year has been devoted by Julien to learn the latest techniques in symbolic model checking. Furthermore, Julien has performed the following activities:

- He has written a compiler to translate PLC programs into propositional formulas.
- Has has supervised two MSc students (working within the above mentioned ASTEC project) to compile SMV programs to the FIXIT format.
- He has made extensive experiments to compare the performances of FIXIT with BDDs. He has given a characterization of the class of systems (namely non-deterministic systems) for which FIXIT performs worse than BDDs.
- He has now undertaken a very important challenge, namely to combine SAT-based techniques with partial order approaches. This research will hopefully lead to several publication during the next year.

Cooperation with Industry The implementation of FIXIT was sponsored by Prover Technology AB. Prover Technology and our group have agreed to jointly recruit an industrial PhD student to work on symbolic model checking. One of the main goals of having a joint PhD student is to exchange research information between the project and the ongoing activities on model checking at Prover. Parosh Abdulla is employed as a consultant at Prover, where he participates in several projects related to verification of systems described in different industrially used formalisms, such as message sequence charts and sequence diagrams in UML. Almost all relevant problems in these projects can be reduced to some form of model checking. This implies that the output

of the project will be useful for Prover both during implementation of pure model checking tools, and when building tools to support designs in the above formalisms.

References

[ABE99] Parosh Aziz Abdulla, Per Bjesse, and Niklas Eén. Symbolic reachability analysis based on sat solvers, 1999. To appear in Proc. TACAS'2000.

[AIN00a] Parosh Aziz Abdulla, Purushothaman Iyer, and Aletta Nylén. Sat-solving the coverability problem for unbounded petri net. Technical report, Department of Information Technology, Uppsala University, 2000.

[AIN00b] Parosh Aziz Abdulla, Purushothaman Iyer, and Aletta Nylén. Unfoldings of unbounded petri nets. In *Proc.* 12^{th} *Int. Conf. on Computer Aided Verification*, volume 1855 of *Lecture Notes in Computer Science*, pages 495–507, 2000.

[AN99] Parosh Aziz Abdulla and Aletta Nylén. BQOs and timed Petri nets. Technical Report DoCS 99/102, Department of Computer Systems, Uppsala University, 1999.

[AN00] Parosh Aziz Abdulla and Aletta Nylén. Better is better than well: On efficient verification of infinite-state systems. In *Proc.* 15^{th} *IEEE Int. Symp. on Logic in Computer Science*, pages 132–140, 2000.

[BCC+99] A. Biere, A. Cimatti, E. M. Clarke, M. Fujita, and Y. Zhu. Symbolic model checking using sat procedures instead of BDDs. In *Design Automation Conference (DAC'99)*, 1999.

[BCCZ99] A. Biere, A. Cimatti, E. M. Clarke, and Y. Zhu. Symbolic model checking without BDDs. In *Proc. TACAS '99,* 5^{th} *Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, 1999.

[WBCG00] P. F. Williams, A. Biere, E. M. Clarke, and A. Gupta. Combining decision diagrams and sat procedures for efficient symbolic model checking. In *Proc.* 12^{th} *Int. Conf. on Computer Aided Verification*, volume 1855 of *Lecture Notes in Computer Science*, pages 124–138, 2000.