Detailed description of the ICAPS'13 tutorial on

Model Checking of Hybrid Systems via Satisfiability Modulo Theories

Alessandro Cimatti
cimatti@fbk.eu
http://es.fbk.eu/people/cimatti/
Fondazione Bruno Kessler, Trento, Italy

Motivations

Complex embedded systems are increasingly present in our daily lives, whenever a computer-based system interacts with some physical plant or environment. Some application domains of interest are industrial production, automotive, railways, and aerospace. The key feature of such complex system, often known as *hybrid systems*, is the combination of *discrete dynamics* (e.g. from the control logic) and the *continuous dynamics* (e.g. from the physical system). Discrete dynamics represent, for example, control states and operation modes, while continuous dynamics take into account the physical aspects such as duration of activities, speed and position of moving objects, and profiles for resource consumption.

The ability to reasoning about such systems is important in two complementary dimensions. In the *design phases*, there is a need to predict the behaviour of the control algorithms before they are put into operation. In the *operation phases*, the ability to reason about such dynamic systems is a backbone for plan generation, plan validation, plan execution and monitoring, fault detection/isolation/recovery (FDIR), and replanning.

The objective of the tutorial is to present a formal and comprehensive account for modeling hybrid systems, and a set of powerful techniques to reason about them. The tutorial will be grounded in the well-studied formalism of hybrid

automata [2, 1]. We will rely on a symbolic representation in form of Satisfiability Modulo Theories formulae [5], which can be thought of as (fragments of) first-order logic where mathematical symbols are interpreted according to suitable theories (e.g. linear arithmetic).

The (combinational) backends are SMT *solvers*, that can be seen as a tight integration of SAT (to deal with the boolean reasoning) with dedicated constraints solvers (to deal with theory reasoning).

The algorithms for reasoning about hybrid systems are able to carry out various forms of reachability analysis, and can be classified in two types. The basic ones, that lift to the case of SMT the SAT-base algorithms developed for the case of finite state model checking (including for example bounded model checking, induction, and interpolation based analysis). More advanced ones take into account the distinguishing features of networks of hybrid automata (see [19, 35]).

Structure of the Tutorial

The tutorial will be organized as follows:

Motivations. In the first, introductory part, we present a general description of the distinguishing nature of hybrid systems, the reasoning problems arising in the design and operation phases, and some motivating examples from practical domains.

Hybrid automata. In this part, we present the formalism of hybrid automata, showing how it extends finite state modeling to the case of dynamics with continuous variables. We discuss the aspects related to component-based modeling, and the notion of hybrid automata networks. By means of some running examples, we show how, within the formalism of hybrid automata, it is possible to model actions with durations, resource consumption, and the distinction between cooperative and adversarial choices. The formal properties of hybrid systems (including decidability and undecidability of various relevant subclasses) are discussed.

Satisfiability Modulo Theories. We provide a comprehensive background on the field of SMT, including the standard language of SMT-LIB, the lazy and eager approaches to solving, the functions provided by the solvers (e.g. incrementality, unsatisfiable core extraction, interpolation), and the relationship with SAT and constraint solving.

SMT-based representation. We show how to symbolically model hybrid automata, with a representation based on SMT formulae. We show the power of symbolic modeling with respect to an explicit, enumerative representation.

SMT-based reasoning. We discuss the various algorithms for reachability analysis and scenario-based verification, that provide the basis to reason about hybrid automata with Satisfiability Modulo Theories.

Other applications of SMT. We show how the proposed techniques, within the ESA-funded project COMPASS, are used for formal verification, safety assessment, diagnosability analysis and FDIR analysis. Then we show how SMT is used, within the ESA-funded IRONCAP project, to model the operation modes of an exploratory rover, and to deal with uncertainty in temporal problems.

Relevance to ICAPS

We expect this tutorial to be relevant to the ICAPS audience, for the following specific reasons.

First, the tutorial provides comprehensive background on hybrid automata, a comprehensive and well-founded formalism, that allows to represent complex planning domains, with durative actions and other timing aspects, resource consumption, and complex physical dynamics. Within this framework, it is possible to cast many problems that are of direct relevance, such as plan validation, partial observability, and temporal problems under uncertainty.

Second, the tutorial covers in depth the field of SMT, a novel technology that has demonstrated significant effectiveness in other fields, and has many potentials of application (and in fact has been already applied) also in planning.

Third, we intend to present the ongoing application of the proposed techniques in some industrial application domains (e.g. aerospace, pipe laying vessel) that are examples of real-world planning domains, and as such are of direct relevance to the audience.

Speaker profile

Alessandro Cimatti is the Head of the research unit in Embedded Systems at the Fondazione Bruno Kessler, Center for Information and Commmunication Technologies – formerly IRST – in Trento, Italy. The unit carries out research activities in various fields of automated reasoning, formal verification, monitoring and FDIR, planning, and diagnosis.

Cimatti has made contributions to many research fields, including Bounded Model Checking [9, 8], Satisfiability Modulo Theories [11, 22, 18, 25, 28, 29], planning in nondeterministic domains via symbolic model checking [43, 41, 44, 6, 42], software verification [7, 23, 31, 37, 27, 38, 39], diagnosability checking [40, 10], formal safety assessment [17, 14], formal verification of hybrid automata [4, 3, 24, 36, 34, 35, 48], temporal reasoning under uncertainty [33, 32].

Cimatti has published 26 journal papers and 98 conference papers, and has an H-index of 37 (details are available at `http://scholar.google.it/citations?user=lbZ6n5IAAAAJ`). He has been a member of the program committees of the major conferences in artificial intelligence and formal verification, and has been program chair of FMCAD'08 [30] and of SAT'12 [47].

Cimatti has extensive experience in technology transfer, in the application of formal verification and planning in various application domains, including hardware [49], railways interlocking [26, 24], requirements validation of the European Train Control System (competitive call of the European Railway Agency) [45, 20, 21, 46], and various space applications, in several projects funded by the European Space Agency under competitive calls, such as OMCARE [16], COMPASS [15, 13, 12], and FOREVER [48].

Since 2005, Cimatti has been teaching courses at the University of Trento and at the Free University of Bozen, and has delivered several tutorials on Satisifability Modulo Theories and on the applications of SMT-based verification.

References

[1] Rajeev Alur. Formal verification of hybrid systems. In Samarjit Chakraborty, Ahmed Jerraya, Sanjoy K. Baruah, and Sebastian Fischmeister, editors, *EMSOFT*, pages 273–278. ACM, 2011.

[2] Rajeev Alur, Costas Courcoubetis, Thomas A. Henzinger, and Pei-Hsin Ho. Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems. In Robert L. Grossman, Anil Nerode, Anders P. Ravn, and Hans Rischel, editors, *Hybrid Systems*, volume 736 of *Lecture Notes in Computer Science*, pages 209–229. Springer, 1992.

[3] G. Audemard, M. Bozzano, A. Cimatti, and R. Sebastiani. Verifying Industrial Hybrid Systems with MathSAT. In A. Biere and O. Strichman, editors, *Proceedings of the International Workshop on Bounded Model Checking (BMC'04)*, volume 119 of *Electronic Notes in Theoretical Computer Science*, pages 17–32, Boston, Massachussets, July 2005. Elsevier.

[4] G. Audemard, A. Cimatti, A. Korniłowicz, and R. Sebastiani. Bounded Model Checking for Timed Systems. In D. Peled and M. Vardi, editors, *Proceedings of the International Conference on Formal Techniques for Networked and Distributed Systems (FORTE'02)*, volume 2529 of *LNCS*, pages 243–259, Houston, Texas, 2002. Springer.

[5] Clark W. Barrett, Roberto Sebastiani, Sanjit A. Seshia, and Cesare Tinelli. Satisfiability modulo theories. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability*, volume 185 of *Frontiers in Artificial Intelligence and Applications*, pages 825–885. IOS Press, 2009.

[6] P. Bertoli, A. Cimatti, M. Roveri, and P. Traverso. Strong Planning under Partial Observability. *Artificial Intelligence*, 170(4-5):337–384, 2006.

[7] Dirk Beyer, A. Cimatti, A. Griggio, Erkan Keremoglu, and R. Sebastiani. Software model checking via large block encoding. In *Proceedings of FMCAD 2009*, 2009.

[8] A. Biere, A. Cimatti, E. Clarke, M. Fujita, and Y. Zhu. Symbolic Model Checking Using SAT Procedures instead of BDDs. In *Proceedings of the 36th Design Automation Conference (DAC'99)*, pages 317–320, New Orleans, LA, USA, June 1999. ACM Press.

[9] A. Biere, A. Cimatti, E. Clarke, and Y. Zhu. Symbolic model checking without BDDs. In R. Cleaveland, editor, *Proceedings of the Fifth International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'99)*, volume 1579 of *Lecture Notes in Computer Science*, pages 193–207. Springer Verlag, 1999.

[10] B. Bittner, M. Bozzano, A. Cimatti, and X. Olive. Symbolic synthesis of observability requirements for diagnosability. In Hoffmann and Selman [50].

[11] M. Bozzano, R. Bruttomesso, A. Cimatti, T. Junttila, S. Ranise, P. van Rossum, and R. Sebastiani. Efficient Theory Combination via Boolean Search. *Information and Computation*, 204(10):1493–1525, 2006. Special Issue on Combining Logical Systems.

[12] M. Bozzano, A. Cimatti, J.-P. Katoen, V. Y. Nguyen, T. Noll, and M. Roveri. Safety, dependability and performance analysis of extended aadl models. *Comput. J.*, 54(5):754–775, 2011.

[13] M. Bozzano, A. Cimatti, J.-P. Katoen, Viet Yen Nguyen, Thomas Noll, M. Roveri, and Ralf Wimmer. A model checker for aadl. In Tayssir Touili, Byron Cook, and Paul Jackson, editors, *Proceedings of the 22nd International Conference on Computer Aided Verification*, volume 6174 of *Lecture Notes in Computer Science*, pages 562–565, Edinburgh, UK, 2010. Springer.

[14] M. Bozzano, A. Cimatti, O. Lisagor, C. Mattarei, S. Mover, M. Roveri, and S. Tonetta. Symbolic model checking and safety assessment of altarica models. *ECEASST*, 46, 2011.

[15] M. Bozzano, A. Cimatti, M. Roveri, J.-P. Katoen, V. Y. Nguyen, and T. Noll. Codesign of dependable systems: a component-based modeling language. In *MEMOCODE'09: Proceedings of the 7th IEEE/ACM international conference on Formal Methods and Models for Codesign*, pages 121–130, Piscataway, NJ, USA, 2009. IEEE Press.

[16] M. Bozzano, A. Cimatti, M. Roveri, and A. Tchaltsev. A comprehensive approach to on-board autonomy verification and validation. In Toby Walsh, editor, *IJCAI*, pages 2398–2403. IJCAI/AAAI, 2011.

[17] M. Bozzano, A. Cimatti, and F. Tapparo. Symbolic Fault Tree Analysis for Reactive Systems. In *Proceedings of the 5th International Symposium*

on Automated Technology for Verification and Analysis (ATVA'07), October 2007.

[18] R. Bruttomesso, A. Cimatti, A. Franzén, A. Griggio, and R. Sebastiani. Delayed Theory Combination vs. Nelson-Oppen for Satisfiability Modulo Theories: a Comparative Analysis. *Annals of Mathematics and Artificial Intelligence*, 55(1-2):63–99, 2009.

[19] L. Bu, A. Cimatti, X. Li, S. Mover, and S. Tonetta. Model checking of hybrid systems using shallow synchronization. In John Hatcliff and Elena Zucca, editors, *FMOODS/FORTE*, volume 6117 of *Lecture Notes in Computer Science*, pages 155–169. Springer, 2010.

[20] R. Cavada, A. Cimatti, A. Mariotti, C. Mattarei, A. Micheli, S. Mover, M. Pensallorto, M. Roveri, A. Susi, and S. Tonetta. Supporting Requirements Validation: The EuRailCheck Tool. In *ASE*, pages 665–667. IEEE Computer Society, 2009.

[21] A. Chiappini, A. Cimatti, Luca Macchi, Oscar Rebollo, M. Roveri, A. Susi, S. Tonetta, and Berardino Vittorini. Formalization and validation of a subset of the european train control system. In Jeff Kramer, Judith Bishop, Premkumar T. Devanbu, and Sebastián Uchitel, editors, *ICSE (2)*, pages 109–118. ACM, 2010.

[22] A. Cimatti. Beyond Boolean SAT: Satisfiability Modulo Theories. In *Discrete Event Systems, 2008. 9th International Workshop on (WODES'08)*, pages 68–73, Goteborg, Sweden, May 2008. IEEE Press.

[23] A. Cimatti. Smt-based software model checking. In Jaco van de Pol and Michael Weber 0002, editors, *SPIN*, volume 6349 of *Lecture Notes in Computer Science*, pages 1–3. Springer, 2010.

[24] A. Cimatti, R. Corvino, A. Lazzaro, I. Narasamdya, T. Rizzo, M. Roveri, A. Sanseviero, and A. Tchaltsev. Formal verification and validation of ertms industrial railway train spacing system. In Madhusudan and Seshia [51], pages 378–393.

[25] A. Cimatti, A. Franzén, A. Griggio, R. Sebastiani, and C. Stenico. Satisfiability Modulo the Theory of Costs: Foundations and Applications. In *Proceedings of TACAS 2010*, volume 6015 of *LNCS*. Springer, 2010.

[26] A. Cimatti, F. Giunchiglia, G. Mongardi, D. Romano, F. Torielli, and P. Traverso. Formal Verification of a Railway Interlocking System using Model Checking. *Journal on Formal Aspects in Computing*, 10:361–380, 1998.

[27] A. Cimatti and A. Griggio. Software model checking via ic3. In Madhusudan and Seshia [51], pages 277–293.

[28] A. Cimatti, A. Griggio, and R. Sebastiani. Efficient Generation of Craig Interpolants in Satisfiability Modulo Theories. *ACM Transactions on Computational Logic (TOCL)*, 12(1), October 2010. In press.

[29] A. Cimatti, A. Griggio, and R. Sebastiani. Computing small unsatisfiable cores in satisfiability modulo theories. *J. Artif. Intell. Res. (JAIR)*, 40:701–728, 2011.

[30] A. Cimatti and Robert B. Jones, editors. *Formal Methods in Computer-Aided Design, FMCAD 2008, Portland, Oregon, USA, 17-20 November 2008*. IEEE, 2008.

[31] A. Cimatti, A. Micheli, I. Narasamndya, and M. Roveri. Verifying SystemC: a Software Model Checking Approach. In N. Sharigina and R. Bloem, editors, *FMCAD 2010*, Lugano, Switzerland, October 2010.

[32] A. Cimatti, A. Micheli, and M. Roveri. Solving temporal problems using smt: Strong controllability. In Michela Milano, editor, *CP*, volume 7514 of *Lecture Notes in Computer Science*, pages 248–264. Springer, 2012.

[33] A. Cimatti, A. Micheli, and M. Roveri. Solving temporal problems using smt: Weak controllability. In Hoffmann and Selman [50].

[34] A. Cimatti, S. Mover, and S. Tonetta. A quantifier-free smt encoding of non-linear hybrid automata. In *FMCAD*, 2012.

[35] A. Cimatti, S. Mover, and S. Tonetta. Smt-based scenario verification for hybrid systems. *Formal Methods in System Design*, May 2012.

[36] A. Cimatti, S. Mover, and S. Tonetta. Smt-based verification of hybrid systems. In Hoffmann and Selman [50].

[37] A. Cimatti, I. Narasamdya, and M. Roveri. Boosting lazy abstraction for systemc with partial order reduction. In Parosh Aziz Abdulla and K. Rustan M. Leino, editors, *TACAS*, volume 6605 of *Lecture Notes in Computer Science*, pages 341–356. Springer, 2011.

[38] A. Cimatti, I. Narasamdya, and M. Roveri. Software model checking with explicit scheduler and symbolic threads. *Logical Methods in Computer Science*, 8(2), 2012.

[39] A. Cimatti, I. Narasamdya, and M. Roveri. Verification of parametric system designs. In *FMCAD*, 2012.

[40] A. Cimatti, C. Pecheur, and R. Cavada. Formal Verification of Diagnosability via Symbolic Model Checking. In G. Gottlob and T. Walsh, editors, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 363–369, Acapulco, Mexico, 2003. Morgan Kaufmann.

[41] A. Cimatti, M. Pistore, M. Roveri, and P. Traverso. Weak, strong, and strong cyclic planning via symbolic model checking. *Artificial Intelligence*, 147(1–2):35–84, 2003.

[42] A. Cimatti, M. Pistore, and P. Traverso. Automated Planning. In F. van Harmelen, V. Lifschitz, and B. Porter, editors, *Handbook of Knowledge Representation*, chapter 22. Elsevier, 2008.

[43] A. Cimatti and M. Roveri. Conformant Planning via Symbolic Model Checking. *Journal of Artificial Intelligence Research (JAIR)*, 13:305–338, 2000.

[44] A. Cimatti, M. Roveri, and P. Bertoli. Conformant planning via symbolic model checking and heuristic search. *Artificial Intelligence*, 159(1-2):127–206, 2004.

[45] A. Cimatti, M. Roveri, A. Susi, and S. Tonetta. Object models with temporal constraints. In Antonio Cerone and Stefan Gruner, editors, *6th IEEE International Conferences on Software Engineering and Formal Methods*, Lecture Notes in Computer Science, Cape Town, South Africa, 2008. IEEE.

[46] A. Cimatti, M. Roveri, A. Susi, and S. Tonetta. Formalizing requirements with object models and temporal constraints. *Software and Systems Modeling*, pages 1–14, 2010. 10.1007/s10270-009-0130-7.

[47] A. Cimatti and R. Sebastiani, editors. *Theory and Applications of Satisfiability Testing - SAT 2012 - 15th International Conference, Trento, Italy, June 17-20, 2012. Proceedings*, volume 7317 of *Lecture Notes in Computer Science*. Springer, 2012.

[48] A. Cimatti and S. Tonetta. A property-based proof system for contract-based design. In Vittorio Cortellessa, Henry Muccini, and Onur Demirörs, editors, *EUROMICRO-SEAA*, pages 21–28. IEEE Computer Society, 2012.

[49] A. Franzen, A. Cimatti, Alexander Nadel, R. Sebastiani, and Jonathan Shalev. Applying SMT in Symbolic Execution of Microcode. In N. Sharigina and R. Bloem, editors, *FMCAD 2010*, Lugano, Switzerland, October 2010.

[50] Jörg Hoffmann and Bart Selman, editors. *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada*. AAAI Press, 2012.

[51] P. Madhusudan and Sanjit A. Seshia, editors. *Computer Aided Verification - 24th International Conference, CAV 2012, Berkeley, CA, USA, July 7-13, 2012 Proceedings*, volume 7358 of *Lecture Notes in Computer Science*. Springer, 2012.