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 Communication 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 Satisfiability Modulo Theories and on the applications of SMT-based verification.
References


5


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


