Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science
Electronic Notes in Theoretical Computer Science 231 (2009) 277–292
www.elsevier.com/locate/entcs

The Tractability of Model-checking for LTL: The Good, the Bad, and the Ugly Fragments

Michael Bauland[a] Martin Mundhenk[b] Thomas Schneider[c]
Henning Schnoor[d,1] Ilka Schnoor[d] Heribert Vollmer[e,2]

[a] *Knipp GmbH, Dortmund, Germany*
`Michael.BaulandATknipp.de`

[b] *Institut für Informatik, Friedrich-Schiller-Universität Jena, Germany*
`mundhenkATcs.uni-jena.de`

[c] *School of Computer Science, University of Manchester, UK*
`schneiderATcs.man.ac.uk`

[d] *Department of Computer Science, Rochester Institute of Technology, Rochester, NY, USA*
`{hs,is}ATcs.rit.edu`

[e] *Theoretische Informatik, Leibniz-Universität Hannover, Germany*
`vollmerATthi.uni-hannover.de`

Abstract

In a seminal paper from 1985, Sistla and Clarke showed that the model-checking problem for Linear Temporal Logic (LTL) is either NP-complete or PSPACE-complete, depending on the set of temporal operators used. If, in contrast, the set of propositional operators is restricted, the complexity may decrease. This paper systematically studies the model-checking problem for LTL formulae over restricted sets of propositional and temporal operators. For almost all combinations of temporal and propositional operators, we determine whether the model-checking problem is tractable (in P) or intractable (NP-hard). We then focus on the tractable cases, showing that they all are NL-complete or even logspace solvable. This leads to a surprising gap in complexity between tractable and intractable cases. It is worth noting that our analysis covers an infinite set of problems, since there are infinitely many sets of propositional operators.

Keywords: computational complexity, linear temporal logic, model checking

1 Introduction

Linear Temporal Logic (LTL) has been proposed by Pnueli [11] as a formalism to specify properties of parallel programs and concurrent systems, as well as to reason about their behaviour. Since then, it has been widely used for these purposes. Recent developments require reasoning tasks—such as deciding satisfiability, validity, or model checking—to be performed automatically. Therefore, decidability

[1] Supported by the Postdoc Programme of the German Academic Exchange Service (DAAD)
[2] Supported in part by DFG VO 630/6-1.

doi:10.1016/j.entcs.2009.02.041

and computational complexity of the corresponding decision problems are of great interest.

The earliest and fundamental source of complexity results for the satisfiability problem (SAT) and the model-checking problem (MC) of LTL is certainly Sistla and Clarke's paper [17]. They have established PSPACE-completeness of SAT and MC for LTL with the temporal operators F (eventually), G (invariantly), X (next-time), U (until), and S (since). They have also shown that these problems are NP-complete for certain restrictions of the set of temporal operators. This work was continued by Markey [8]. The results of Sistla, Clarke, and Markey imply that SAT and MC for LTL and a multitude of its fragments are intractable. In fact, they do not exhibit any tractable fragment.

The fragments they consider are obtained by restricting the set of temporal operators and the use of negations. What they do not consider are arbitrary fragments of temporal *and* Boolean operators. For propositional logic, a complete analysis has been achieved by Lewis [6]. He divides all infinitely many sets of Boolean operators into those with tractable (polynomial-time solvable) and intractable (NP-complete) SAT problems. A similar systematic classification has been obtained by Bauland et al. in [3] for LTL. They divide fragments of LTL—determined by arbitrary combinations of temporal and Boolean operators—into those with polynomial-time solvable, NP-complete, and PSPACE-complete SAT problems.

This paper continues the work on the MC problem for LTL. Similarly as in [3], the considered fragments are arbitrary combinations of temporal and Boolean operators. We will separate the MC problem for almost all LTL fragments into tractable (i.e., polynomial-time solvable) and intractable (i.e., NP-hard) cases. This extends the work of Sistla and Clarke, and Markey [17,8], but in contrast to their results, we will exhibit many tractable fragments and exactly determine their computational complexity. Surprisingly, we will see that tractable cases for model checking are even very easy—that is, NL-complete or even L-solvable. There is only one set of Boolean operators, consisting of the binary *xor*-operator, that we will have to leave open. This constellation has already proved difficult to handle in [3,1], the latter being a paper where SAT for basic modal logics has been classified in a similar way.

While the borderline between tractable and intractable fragments in [6] and [3] is quite easily recognisable (SAT for fragments containing the Boolean function $f(x,y) = x \wedge \overline{y}$ is intractable, almost all others are tractable), our results for MC will exhibit a rather diffuse borderline. This will become visible in the following overview and is addressed in the Conclusion. Our most surprising intractability result is the NP-hardness of the fragment that only allows the temporal operator U and no propositional operator at all. Our most surprising tractability result is the NL-completeness of MC for the fragment that only allows the temporal operators F, G, and the binary *or*-operator. Taking into account that MC for the fragment with only F plus *and* is already NP-hard (which is a consequence from [17]), we would have expected the same lower bound for the "dual" fragment with only G plus *or*, but in fact we show that even the fragment with F and G and *or* is tractable. In the presence of the X-operator, the expected duality occurs: The fragment with F,