Why Is Modal Logic So Robustly Decidable?

Author
Vardi, Moshe Y.

Date
1997-04-12

Citation

Abstract
In the last 20 years modal logic has been applied to numerous areas of computer science, including artificial intelligence, program verification, hardware verification, database theory, and distributed computing. There are two main computational problems associated with modal logic. The first problem is checking if a given formula is true in a given state of a given structure. This problem is known as the model-checking problem. The second problem is checking if a given formula is true in all states of all structures. This problem is known as the validity problem. Both problems are decidable. The model-checking problem can be solved in linear time, while the validity problem is PSPACE-complete. This is rather surprising when one considers the fact that modal logic, in spite of its apparent propositional syntax, is essentially a first-order logic, since the necessity and possibility modalities quantify over the set of possible worlds, and model checking and validity for first-order logic are computationally hard problems. Why, then, is modal logic so robustly decidable? To answer this question, we have to take a close look at modal logic as a fragment of first-order logic. A careful examination reveals that propositional modal logic can in fact be viewed as a fragment of 2-variable first-order logic. It turns out that this fragment is computationally much more tractable than full first-order logic, which provides some explanation for the tractability of modal logic. Upon a deeper examination, however, we discover that this explanation is not too satisfactory. The tractability of modal logic is quite and cannot be explained by the relationship to two-variable first-order logic. We argue that the robust decidability of modal logic can be explained by the so-called tree-model property, and we show how the tree-model property leads to automata-based decision procedures.
are also robustly decidable.