Artificial Intelligence 229 (2015) 1–32

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

The complexity of reasoning with FODD and GFODD ☆

Benjamin J. Hescott, Roni Khardon *

Department of Computer Science, Tufts University, Medford, MA 02155, USA

ARTICLE INFO

Article history:
Received 8 July 2014
Received in revised form 2 July 2015
Accepted 13 August 2015
Available online 18 August 2015

Keywords:
Complexity analysis
Polynomial hierarchy
Decision diagrams
First order logic

ABSTRACT

Recent work introduced Generalized First Order Decision Diagrams (GFODD) as a knowledge representation language that is useful in mechanizing decision theoretic planning in relational domains. GFODDs generalize function-free first order logic and include numerical values and numerical generalizations of existential and universal quantification. Previous work presented heuristic inference algorithms for GFODDs and implemented these heuristics in systems for decision theoretic planning. In this paper, we study the complexity of the computational problems addressed by such implementations. In particular, we study the evaluation problem, the satisfiability problem, and the equivalence problem for GFODDs under the assumption that the size of the intended model is given with the problem, a restriction that guarantees decidability. Our results provide a complete characterization placing these problems within the polynomial hierarchy. The same characterization applies to the corresponding restriction of problems in first order logic, giving an interesting new avenue for efficient inference when the number of objects is bounded. Our results show that for Σ_k formulas, and for corresponding GFODDs, evaluation and satisfiability are Σ_k^P complete, and equivalence is Π_{k+1}^P complete. For Π_k formulas evaluation is Π_k^P complete, satisfiability is one level higher and is Σ_{k+1}^P complete, and equivalence is Π_{k+1}^P complete.

1. Introduction

The complexity of inference in first order logic has been investigated intensively. It is well known that the problem is undecidable, and that this holds even with strong restrictions on the types and number of predicates allowed in the logical language. For example, the problem is undecidable for quantifier prefix $\forall^2\exists^*$ with a signature having a single binary predicate and equality [8]. Unfortunately, the problem is also undecidable if we restrict attention to satisfiability under finite structures [6,24]. Thus, in either case, one cannot quantify the relative difficulty of problems without further specialization or assumptions. On the other hand, algorithmic progress in AI has made it possible to reason efficiently in some cases. In this paper we study such problems under the additional restriction that an upper bound on the intended model size is given explicitly. This restriction is natural for many applications, where the number of objects is either known in advance or known to be bounded by some quantity. Since the inference problem is decidable under this restriction, we can provide a more detailed complexity analysis.

This paper is motivated by recent work on decision diagrams, known as FODDs and GFODDs, and the computational questions associated with them. Binary decision diagrams [3,1] provide a successful representation language capturing func-

☆ A preliminary version of this paper has appeared as [10]. This paper includes a broader exposition and a significant amount of additional details in proofs and constructions required to obtain the technical results.

* Corresponding author.
E-mail addresses: hescott@cs.tufts.edu (B.J. Hescott), roni@cs.tufts.edu (R. Khardon).

http://dx.doi.org/10.1016/j.artint.2015.08.005

tions over propositional variables, that allows for efficient manipulation and composition of functions, and diagrams have been used in various applications in program verification and AI [3,1,11]. Motivated by this success, several authors have attempted generalizations to handle relational structure and first order quantification [9,33,30,16]. In particular FODDs [33] and their generalization GFODDs [16] have been introduced and shown to be useful in the context of decision theoretic planning [2,20,12,13] for problems with relational structure [15,17].

GFODDs can be seen to generalize the function-free portion of first order logic (i.e., signatures with constants but without higher arity functions) to allow for non-binary numerical values generalizing truth values, and for numerical quantifiers generalizing existential and universal quantification in logic. Efficient heuristic inference algorithms for such diagrams have been developed focusing on the finite model case, and using the notion of “reasoning from examples” [22,23,21]. This paper analyzes the complexity of the evaluation, satisfiability, and equivalence problems for such diagrams, focusing on the GFODD subset with min and max aggregation that are defined in the next section. To avoid undecidability and get a more refined classification of complexity, we study a restricted form of the problem where the finite size of the intended model is given as part of the input to the problem. As we argue below, this is natural and relevant in the applications of GFODDs for solving decision theoretic control problems. The same restrictions can be used for the corresponding (evaluation, satisfiability and equivalence) problems in first order logic, but to our knowledge this has not been studied before. We provide a complete characterization of the complexity showing an interesting structure. Our results are developed for the GFODD representation and require detailed arguments about the graphical representation of formulas in that language. The same lines of argument (with simpler proof details) yield similar results for first order logic. To translate our results to the language of logic, consider the quantifier prefix of a first order logic formula using the standard notation using Σ_k, Π_k to denote alternation depth of quantifiers in the formula. With this translation, our results show that:

(1) Evaluation over finite structures spans the polynomial hierarchy, that is, evaluation of Σ_k formulas is Σ_k^P complete, and evaluation of Π_k formulas is Π_k^P complete.

(2) Satisfiability, with a given bound on model size, follows a different pattern: satisfiability of Σ_k formulas is Σ_k^P complete, and satisfiability of Π_k formulas is Σ_{k+1}^P complete.

(3) Equivalence, under the set of models bounded by a given size, depends only on quantifier depth: both the equivalence of Σ_k formulas and equivalence of Π_k formulas are Π_{k+1}^P complete.

The positive results allow for constants in the signature but the hardness results, except for satisfiability for Π_1 formulas, hold even without constants. For signatures without constants, satisfiability of Π_1 formulas is in NP; when constants are allowed, it is Σ_2^P complete as in the general template.

These results are useful in that they clearly characterize the complexity of the problems solved heuristically by implementations of GFODD systems [15,17] and can be used to partly motivate or justify the use of these heuristics. For example, the “model checking reductions” of [16] that simplify the structure of diagrams replace equivalence tests with model evaluation on a “representative” set of models. When this set is chosen heuristically, as in [15], this leads to inference that is correct with respect to these models but otherwise incomplete. Our results show that this indeed leads to a reduction of the complexity of the inference problem, so that the loss in accuracy is traded for improved worst case run time. Importantly, it shows that without compromising correctness, the complexity of equivalence tests that are used to compress the representation will be higher. These issues and further questions for future work are discussed in the concluding section of the paper.

The rest of the paper is organized as follows. The next section defines FODDs and GFODDs and provides a more detailed motivation for the technical questions. Section 3 then develops the results for FODDs. We treat the FODD case separately for three reasons. First, this serves for an easy introduction into the results that avoids some of the more involved arguments that are required for GFODDs. Second, as will become clear, for FODDs we do not need the additional assumption on model size, so that the results are in a sense stronger. Finally, some of the proofs for GFODDs require alternation depth of at least two so that separate proofs are needed for FODDs in any case. Section 4 develops the results for GFODDs. The final section concludes with a discussion and directions for future work.

2. FODDs and GFODDs and their computational problems

This section introduces the GFODD representation language and associated computational problems, and explains how they are motivated by prior work on applying GFODDs in decision theoretic planning. We assume familiarity with basic concepts and notation in predicate logic [25,29,4] as well as basic notions from complexity theory [14,32,26].

Decision diagrams are similar to expressions in first order logic (FOL). They are defined relative to a relational signature, with a finite set of predicates $p_1, p_2, \ldots, p_n$ each with an associated arity (number of arguments), a countable set of variables $x_1, x_2, \ldots$, and a set of constants $c_1, c_2, \ldots, c_m$. We do not allow function symbols other than constants (that is, functions with arity ≥ 1). In addition, we assume that the arity of predicates is bounded by some numerical constant. A term is a variable or constant and an atom is either an equality between two terms or a predicate with an appropriate list of terms as arguments. Intuitively, a term refers to an object in the world of interest and an atom is a property which is either true or false.

To motivate the diagram representation consider first a simpler language of generalized expressions which we illustrate informally by some examples. In FOL we can consider open formulas that have unbound variables. For example, the atom $color(x, y)$ is such a formula and its truth value depends on the assignment of x and y to objects in the world. To