ELSEVIER

Artificial Intelligence 87 (1996) 187-213

Artificial Intelligence

Reasoning with models ★

Roni Khardon *, Dan Roth [1]

Aiken Computation Laboratory, Harvard University, 33 Oxford Street, Cambridge, MA 02138, USA

Received August 1994; revised January 1996

Abstract

We develop a model-based approach to reasoning, in which the knowledge base is represented as a set of models (satisfying assignments) rather than a logical formula, and the set of queries is restricted. We show that for every propositional knowledge base (*KB*) there exists a set of *characteristic models* with the property that a query is true in *KB* if and only if it is satisfied by the models in this set. We characterize a set of functions for which the model-based representation is compact and provides efficient reasoning. These include cases where the formula-based representation does not support efficient reasoning. In addition, we consider the model-based approach to *abductive reasoning* and show that for any propositional *KB*, reasoning with its model-based representation yields an abductive explanation in time that is polynomial in its size. Some of our technical results make use of the *monotone theory*, a new characterization of Boolean functions recently introduced.

The notion of *restricted queries* is inherent in our approach. This is a wide class of queries for which reasoning is efficient and exact, even when the model-based representation *KB* provides only an approximate representation of the domain in question.

Moreover, we show that the theory developed here generalizes the model-based approach to reasoning with Horn expressions and captures even the notion of reasoning with Horn approximations.

Keywords: Knowledge representation; Common-sense reasoning; Automated reasoning

★ An earlier version of the paper appears in the Proceedings of the National Conference on Artificial Intelligence, AAAI-94.

* Corresponding author. Research supported by ARO grant DAAL03-92-G-0115 (Center for Intelligent Control Systems). E-mail: roni@das.harvard.edu.

[1] Research supported by NSF grant CCR-92-00884 and by DARPA AFOSR-F4962-92-J-0466. Current address: Department of Applied Mathematics & CS, Weizmann Institute of Science, Rehovot 76100, Israel. E-mail: danr@wisdom.weizmann.ac.il.

PII S0004-3702(96)00006-9

1. Introduction

A widely accepted framework for reasoning in intelligent systems is the knowledge-based system approach [26]. The idea is to keep the knowledge in some *representation language* with a well-defined meaning assigned to those sentences. The sentences are stored in a knowledge base (*KB*) which is combined with a reasoning mechanism, used to determine what can be inferred from the sentences in the *KB*. Deductive reasoning is usually abstracted as follows: Given the knowledge base *KB*, assumed to capture our knowledge about the domain in question (the "world"), and a sentence α, a query that is assumed to capture the situation at hand, decide whether *KB* implies α (denoted $W \models \alpha$). This can be understood as the question: "Is α consistent with the current state of knowledge?"

Solving the question $KB \models \alpha$, even in the propositional case, is co-NP-hard and under the current complexity theoretic beliefs requires exponential time. Many other forms of reasoning which have been developed at least partly to avoid these computational difficulties, were also shown to be hard to compute [30,31].

A significant amount of recent work on reasoning is influenced by convincing arguments of Levesque [23] who argued that common-sense reasoning is a distinct mode of reasoning and that we should give a computational theory that accounts for both its speed and flexibility. Most of the work in this direction still views reasoning as a kind of theorem proving process, and is based on the belief that a careful study of the sources of the computational difficulties may lead to a formalism expressive enough to capture common-sense knowledge, while still allowing for efficient reasoning. Thus, this line of research aims at identifying classes of limited expressiveness, with which one can perform theorem proving efficiently [3,24,30,31]. None of these works, however, meets the strong tractability requirements required for common-sense reasoning (e.g. see [35]), even though the limited expressiveness of classes discussed there has been argued to be implausible [7].

Levesque argues [23,24] that reasoning with a more direct representation is easier and better suits common-sense reasoning. He suggests to represent the knowledge base *KB* in a vivid form, which bears a strong and direct relationship to the real world. This might be just a model of *KB* [8,28] on which one can evaluate the truth value of the query α. It is not clear, however, how one might derive a vivid form of the knowledge base. Moreover, selecting a model which is the most likely model of the real world, under various reasonable criteria, is computationally hard [28,32]. Most importantly, in order to achieve an efficient solution to the reasoning problem this approach modifies the problem: reasoning with a vivid representation no longer solves the problem $KB \models \alpha$, but rather a different problem, whose exact relation to the original inference problem depends on the method selected to simplify the knowledge base.

A model-based approach to reasoning

In this work we embark on the development of a model-based approach to common-sense reasoning. It is not hard to motivate a model-based approach to reasoning from a cognitive point of view and indeed, many of the proponents of this approach to reasoning have been cognitive psychologists [12, 13, 22]. In the AI community this approach can be seen as an example of Levesque's notion of "vivid" reasoning and has already been studied in [14].

The deduction problem $KB \models \alpha$ can be approached using the following model-based strategy:

Test set: A set S of possible assignments.

Test: If there is an element $x \in S$ which satisfies KB, but does not satisfy α, deduce that $KB \not\models \alpha$; Otherwise, $KB \models \alpha$.

Since, by the model theoretic definition of implication, $KB \models \alpha$ if and only if every model of KB is a model of α, the suggested strategy solves the inference problem if S is the set of *all* models (satisfying assignments) of KB. But, this set might be too large. A model-based approach becomes useful if one can show that it is possible to use a fairly small set of models as the test set, and still perform reasonably good inference, under some criterion.

We define a set of models, the *characteristic models* of the knowledge base, and show that performing the model-based test with it suffices to deduce that $KB \models \alpha$, for a restricted set of queries. We prove that for a fairly wide class of representations, this set is sufficiently small, and thus the model-based approach is feasible. The notion of *restricted queries* is inherent in our approach; since we are interested in formalizing common-sense reasoning, we take the view that a reasoner need not answer efficiently *all* possible queries. For a wide class of queries we show that exact reasoning can be done efficiently, even when the reasoner keeps in KB an "approximate" representation of the "world".

We show that the theory developed here generalizes the model-based approach to reasoning with Horn expressions, suggested in [14], and captures even the notion of reasoning with theory approximations [33]. In particular, our results characterize the Horn expressions for which the approach suggested in [14] is useful and explain the phenomena observed there, regarding the relative sizes of the logical formula representation and the model-based representation of KB. We also give other examples of expressive families of propositional expressions, for which our approach is useful.

We note that characteristic models were studied independently in the relational database community (where they are called "generators") [2, 25], for the special case of definite Horn expressions. The results in this paper have immediate implications in this domain (e.g., bounding the size of Armstrong relations), which are described elsewhere [17].

In addition, we consider the problem of performing *abduction* using a model-based approach and show that for any propositional knowledge base, using a model-based representation yields an abductive explanation in time that is polynomial in the size of the model-based representation. Some of our technical results make use of a new characterization of Boolean functions, called the *monotone theory*, introduced recently by Bshouty [4].

Recently, some more results on reasoning with models have been derived, exhibiting the usefulness of this approach. These include algorithms that use model-based representations to handle some fragments of Reiter's default logic as well as some cases of circuit diagnosis [20]. A theory of reasoning with partial models and the learnability of such representations is studied in [21]. The question of translating between characteristic models and propositional expressions (which is relevant in database theory as well) has also been studied. Some results on the complexity of this and related questions are described in [17].

Most of the work on reasoning assumes that the knowledge base is given in some form, and the question of how this knowledge might be acquired is not considered. While in this paper we also take this point of view, we are interested in studying the entire process of *learning* a knowledge base representation and reasoning with it. In particular, Bshouty [4] gives an algorithm that learns the model-based representation we consider here when given access to a *membership oracle* and an *equivalence oracle*. In [19] we discuss the issue of "learning to reason" and illustrate the importance of the model-based approach for this problem.

Summary of results

We now briefly describe the main contributions of the model-based approach developed in this paper. Our results can be grouped into three categories that can be informally described as follows:

(1) We define the set of characteristic models of a propositional expression (with respect to a class of queries) and prove that reasoning with this set supports correct *deduction* and *abduction*.
(2) We characterize the least upper bound of a function, with respect to a class $\mathcal{L}$ of functions, using a set of characteristic models. We then show that reasoning with this approximation is correct for queries in the class $\mathcal{L}$.
(3) We characterize classes of propositional formulas for which the size of the set of characteristic models is polynomial in the number of variables in the domain.

As a result, we show that in many cases where reasoning with the traditional representation is NP-hard we can perform efficient reasoning with models. This includes reasoning with $\log n$-CNF queries, Horn queries, and quasi-Horn queries.

Clearly, our algorithms do not solve NP-complete problems. While most hardness results for reasoning assume that *KB* is given as a CNF formula, we can perform reasoning efficiently since we represent the knowledge in a more accessible form. The efficiency of model-based reasoning strongly depends on the size of the representation. In Section 7 we discuss this issue in detail, and in particular, compare model-based representations to CNF and DNF representations.

The rest of this paper is organized as follows: We start with some technical preliminary definitions and then, in Section 3, we give an example that motivates the discussion in the rest of the paper. Section 4 reviews the monotone theory, which is the main tool in developing our results. In Section 5 we consider the deduction problem. We introduce the set of characteristic models, and analyze the correctness and efficiency of model-based deduction with this set. In Section 6 we show that in the case of Horn expressions our theory reduces to the work in [14]. Section 7 discusses the size of model-based representations. Section 8 describes applications of our theory to particular propositional languages. In Section 9 we consider the abduction problem, and in Section 10 we conclude with some reference to future work.

2. Preliminaries

We consider problems of reasoning where the "world" (the domain in question) is modeled as a Boolean function $f : \{0,1\}^n \to \{0,1\}$. Similarly, the knowledge base KB would consist of some representation (either a propositional formula or a set of specially chosen models) for a Boolean function.

Let $X = \{x_1, \dots, x_n\}$ be a set of *variables*, where each variable is associated with a world's attribute and can take the value 1 or 0 to indicate whether the associated attribute is true or false in the world. *Assignments*, denoted by x, y, z, are mappings from X to $\{0,1\}$, and we treat them as elements in $\{0,1\}^n$ with the natural mapping.

An assignment $x \in \{0,1\}^n$ satisfies f if $f(x) = 1$. An assignment x which satisfies f is also called a *model* of f. If f is a theory of the "world", a satisfying assignment of f is sometimes called a *possible world.*

For $x \in \{0,1\}^n$, $weight(x)$ denotes the number of variables assigned 1 in the assignment x. A literal is either a variable x_i (called a positive literal) or its negation $\overline{x_i}$ (a negative literal). A clause is a disjunction of literals, and a CNF formula is a conjunction of clauses. For example $(x_1 \vee \overline{x_2}) \wedge (x_3 \vee \overline{x_1} \vee x_4)$ is a CNF formula with two clauses. A term is a conjunction of literals, and a DNF formula is a disjunction of terms. For example $(x_1 \wedge \overline{x_2}) \vee (x_3 \wedge \overline{x_1} \wedge x_4)$ is a DNF formula with two terms. A CNF formula is Horn if every clause in it has at most one positive literal. A CNF formula is k-quasi-Horn if every clause in it has at most k positive literals. A CNF formula is in $\log n$-CNF if every clause in it has at most $\log n$ literals (of arbitrary polarity).

Every Boolean function has many possible representations and, in particular, both a CNF representation and a DNF representation. By the DNF size of f, denoted $|DNF(f)|$, we mean the minimum possible number of terms in any DNF representation of f. We call a DNF expression for f which has $|DNF(f)|$ terms, a minimal DNF representation of f. Similarly, the CNF size of f, denoted $|CNF(f)|$, is the minimum possible number of clauses in any CNF representation of f, and a CNF expression is minimal for f if it has $|CNF(f)|$ clauses.

It is important to distinguish between a Boolean function, namely a mapping $f : \{0,1\}^n \to \{0,1\}$, and a representation for the function. Every Boolean function can be represented in many ways. (In particular, a truth table is one such representation.) A standard way to represent Boolean functions is using propositional expressions such as CNF and DNF expressions, as discussed above. A *propositional language* is a collection of propositional expressions.

Some of our results hold for any Boolean function, and are therefore stated in terms of such functions. Other results have restrictions that relate to the representation of the functions; namely, they hold for functions in a certain propositional language. When the representation is clear from the context we sometimes refer to a propositional language as a class of Boolean functions. (That is, all those functions that can be represented using expressions in the language.) We denote classes of Boolean functions by $\mathcal{F}$, $\mathcal{G}$, and functions by f, g.

By "f entails g", denoted $f \models g$, we mean that every model of f is also a model of g. We also refer to the connective $\models$ by its equivalent, proof theoretic name, "implies". Since "entailment" and "logical implication" are equivalent, we can treat f either as a Boolean function (usually, using a propositional expression that represents the function), or as the set of its models, namely $f^{-1}(1)$. Throughout the paper, when no confusion can arise, we identify the Boolean function f with the set of its models, $f^{-1}(1)$. Observe that the connective "implies" ($\models$) used between Boolean functions is equivalent to the connective "subset or equal" ($\subseteq$) used for subsets of $\{0,1\}^n$. That is, $f \models g$ if and only if $f \subseteq g$.

Throughout this paper we measure size and time complexity with respect to n, the number of variables in the domain. Naturally, we are interested in time and size complexity which are polynomial in n. Sometimes, we say that a representation is *small* or *short*, or an algorithm *efficient*, and mean that they are polynomial in n.

3. A motivating example

We start by giving an example which will serve to motivate the abstract ideas developed in the rest of the paper and explain their relation to the problem of reasoning.

Assume that the world is described as a set of rules

$$f = \{\overline{x_1} \wedge \overline{x_2} \rightarrow x_3, \overline{x_1} \wedge \overline{x_4} \rightarrow x_2, x_1 \wedge x_2 \wedge x_4 \rightarrow x_3\},$$

and that we would like to perform deductive reasoning with respect to this world. That is, given a class $\mathcal{Q}$ of queries, we would like to know, for each $\alpha \in \mathcal{Q}$, whether $f \models \alpha$. For example, assume that

$$\mathcal{Q} = \{\overline{x_2} \wedge \overline{x_3} \rightarrow x_4, \overline{x_1} \wedge \overline{x_2} \rightarrow x_3 \wedge x_4\}.$$

First, notice that since f is interpreted as the world in which all the above rules hold, it can be represented in a CNF representation as

$$f = (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3 \vee \overline{x_4}).$$

Similarly, $\mathcal{Q}$ can be represented as a set of disjunctions

$$\mathcal{Q} = \{x_2 \vee x_3 \vee x_4, x_1 \vee x_2 \vee (x_3 \wedge x_4)\}.$$

In order to simplify the example, we make the assumption that all the queries we care about are *monotone*; that is, when represented as CNF or DNF, they contain only positive literals (as in $\mathcal{Q}$). The function f has 12 (out of the 16 possible) satisfying assignments. These are: [2]

$$\{0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1110, 1111\}.$$

[2] An element of $\{0,1\}^n$ denotes an assignment to the variables $x_1, \ldots, x_n$ (i.e., 0011 means $x_1 = x_2 = 0$, and $x_3 = x_4 = 1$).

Given a query α, since $f \models \alpha$ if and only if $f \subseteq \alpha$, it is sufficient to check whether α satisfies all these assignments. However, as the following argument shows, since we consider here only monotone queries, it is sufficient to test only some of these models.

First notice that a query α, being a monotone Boolean function, has the following property: If $\alpha(y) = 1$, and $y \leqslant x$, then also $\alpha(x) = 1$. (Here $\leqslant$ denotes the usual bitwise order relation on $\{0,1\}^n$, that is, $y \leqslant x$ iff for every index i such that $y_i = 1$ we also have $x_i = 1$.)

Now, if $f \not\models \alpha$, then there is a model x of f such that $\alpha(x) = 0$. If y is another model of f and $y \leqslant x$, then since α is monotone, $\alpha(y) = 0$. This implies that f must have a *minimal* model z, which does not satisfy α. Therefore, when all the queries are monotone, there is no need to evaluate α on *all* the satisfying assignments of f. Instead, it is sufficient to consider only the minimal satisfying assignments of f.

In the current example, it is easy to see that there are only three minimal assignments for f: $\{1000, 0100, 0011\}$. When considering $\alpha_1 = x_2 \vee x_3 \vee x_4$, we can see that $\alpha_1(1000) = 0$ and therefore $f \not\models \alpha_1$. On the other hand, when considering $\alpha_2 = x_1 \vee x_2 \vee (x_3 \wedge x_4)$, it is easy to see to all three minimal models of f satisfy α_2 and therefore $f \models \alpha_2$.

The example shows that sometimes it is not necessary to consider all the models of f in order to answer deduction queries with respect to f. Moreover, the set of "special" models we have used, seems to be considerably smaller than the set of all models.

This example motivates at least two important questions that we consider in the rest of this paper. First, can one define a set of "special" models, like the minimal models used in the example, for a wider set of queries? Secondly, the method presented is advantageous when the set of "special" models is significantly smaller than the set of all models. Can one quantify the size of this set? Both questions are answered affirmatively later in this paper.

Finally, we note that in the example presented, no assumptions were made on the "world" f. The set of special models depends only on the class of queries we want to answer correctly. Notice also, that the traditional approach to the problem $f \models \alpha$, in which f is represented as a propositional CNF expression, remains co-NP-hard even when the set of queries is restricted to be a set of monotone functions.

4. Monotone theory

In this section we introduce the notation, definitions and results of the monotone theory of Boolean functions, introduced by Bshouty [4].

Definition 4.1 (*Order*). We denote by $\leqslant$ the *usual partial order* on the lattice $\{0,1\}^n$, the one induced by the order $0 < 1$. That is, for $x, y \in \{0,1\}^n$, $x \leqslant y$ if and only if $\forall i, x_i \leqslant y_i$. For an assignment $b \in \{0,1\}^n$ we define $x \leqslant_b y$ if and only if $x \oplus b \leqslant y \oplus b$, where $\oplus$ denotes the XOR operation (bitwise addition modulo 2). As with other order relations, $x \leqslant_b y$ can also be written as $y \geqslant_b x$, and if $x \leqslant_b y$ and $x \neq y$ we write $x <_b y$.

Intuitively, if $b_i = 0$ then the order relation on the ith bit is the normal order; if $b_i = 1$, the order relation is reversed and we have that $1 <_{b_i} 0$.

The *monotone extension of* $z \in \{0,1\}^n$ with respect to b is:

$$\mathcal{M}_b(z) = \{x \mid x \geqslant_b z\}.$$

The *monotone extension of* f with respect to b is:

$$\mathcal{M}_b(f) = \{x \mid x \geqslant_b z, \text{ for some } z \in f\}.$$

The set of *minimal assignments of* f with respect to b is:

$$\min{}_b(f) = \{z \mid z \in f, \text{ such that } \forall y \in f, z \not>_b y\}.$$

The following claim lists some properties of $\mathcal{M}_b$, all are immediate from the definitions:

Claim 4.2. *Let $f, g : \{0,1\}^n \to \{0,1\}$ be Boolean functions. The operator $\mathcal{M}_b$ satisfies the following properties*:

(1) *If $f \subseteq g$ then $\mathcal{M}_b(f) \subseteq \mathcal{M}_b(g)$.*
(2) $\mathcal{M}_b(f \wedge g) \subseteq \mathcal{M}_b(f) \wedge \mathcal{M}_b(g)$.
(3) $\mathcal{M}_b(f \vee g) = \mathcal{M}_b(f) \vee \mathcal{M}_b(g)$.
(4) $f \subseteq \mathcal{M}_b(f)$.

Claim 4.3. *Let $z \in f$. Then, for every $b \in \{0,1\}^n$, there exists $u \in \min_b(f)$ such that $\mathcal{M}_b(z) \subseteq \mathcal{M}_b(u)$.*

Proof. If $z \notin \min_b(f)$ then $\exists y \in f$ such that $y \leqslant_b z$. Let u be a minimal element in f with this property. Then, $u \in \min_b(f)$ and clearly $\{x \mid x \geqslant_b z\} \subseteq \{x \mid x \geqslant_b u\}$, as needed. □

Using Claims 4.3 and 4.2 we get a characterization of the monotone extension of f:

Claim 4.4. *The monotone extension of f with respect to b is*:

$$\mathcal{M}_b(f) = \bigvee_{z \in f} \mathcal{M}_b(z) = \bigvee_{z \in \min_b(f)} \mathcal{M}_b(z).$$

Clearly, for every assignment $b \in \{0,1\}^n$, $f \subseteq \mathcal{M}_b(f)$. Moreover, if $b \notin f$, then $b \notin \mathcal{M}_b(f)$ (since b is the smallest assignment with respect to the order $\leqslant_b$). Therefore:

$$f = \bigwedge_{b \in \{0,1\}^n} \mathcal{M}_b(f) = \bigwedge_{b \notin f} \mathcal{M}_b(f).$$

The question is whether we can find a small set of negative examples b, and use it to represent f as above.

Definition 4.5 (*Basis*). A set B is a *basis for* f if $f = \bigwedge_{b \in B} \mathcal{M}_b(f)$. B is a basis for a class of functions $\mathcal{F}$ if it is a basis for all the functions in $\mathcal{F}$.

Using this definition, the representation

$$f = \bigwedge_{b\in B} \mathcal{M}_b(f) = \bigwedge_{b\in B} \bigvee_{z\in\min_b(f)} \mathcal{M}_b(z) \tag{1}$$

yields the following necessary and sufficient condition describing when $x \in \{0,1\}^n$ satisfies f:

Corollary 4.6. *Let B be a basis for f, and let $x \in \{0,1\}^n$. Then, $f(x) = 1$ if and only if for every basis element $b \in B$ there exists $z \in \min_b(f)$ such that $x \geqslant_b z$.*

The following claim bounds the size of the basis of a function f:

Claim 4.7. *Let $f = C_1 \wedge C_2 \wedge \cdots \wedge C_k$ be a CNF representation for f and let B be a set of assignments in $\{0,1\}^n$. If every clause C_i is falsified by some $b \in B$ then B is a basis for f. In particular, f has a basis of size $\leqslant k$.*

Proof. Let $B = \{b^1, b^2, \ldots, b^k\}$ be a collection of assignments such that b^i falsifies C_i. We show that $f = \bigwedge_{b\in B} \mathcal{M}_b(f)$. First observe that using Claim 4.2(4) we get $f \subseteq \bigwedge_{b\in B} \mathcal{M}_b(f)$. In order to show $f \supseteq \bigwedge_{b\in B} \mathcal{M}_b(f)$ we show that for all $y \notin f$ there exists $b \in B$ such that $y \notin \mathcal{M}_b(f)$, and therefore $y \notin \bigwedge_{b\in B} \mathcal{M}_b(f)$.

Consider $y \in \{0,1\}^n$ such that $y \notin f$ and assume, w.l.o.g., that $C_1(y) = 0$. Let $b = b^1$ be the corresponding element in B, and assume, by way of contradiction that $\mathcal{M}_b(f)(y) = 1$. Then, there is an assignment $z \in \min_b(\mathcal{M}_b(f)) = \min_b(f)$ such that $z \leqslant_b y$. We therefore have that $b \leqslant_b z \leqslant_b y$. Let x_i be a variable the appears in the clause C_1. Since $C_1(y) = C_1(b) = 0$, we must have $y_i = z_i = b_i$. Since this holds for all variables that appear in C_1, it implies that $C_1(z) = 0$ and contradicts the assumption that $z \in f$. □

The set of *floor* assignments of an assignment x, with respect to the order relation b, denoted $\lfloor x \rfloor_b$, is the set of all elements $z <_b x$ such that there does not exist y for which $z <_b y <_b x$ (i.e., z is strictly smaller than x relative to b and is different from x in exactly one bit).

The set of *local minimal assignments of f* with respect to b is:

$$\min{}^*_b(f) = \{x \mid x \in f, \text{ and } \forall y \in \lfloor x \rfloor_b,\ y \notin f\}.$$

The following claims bound the size of $\min_b(f)$:

Claim 4.8. *Let $f = D_1 \vee D_2 \vee \cdots \vee D_k$ be a DNF representation for f. Then for every $b \in \{0,1\}^n$, $|\min^*_b(f)| \leqslant k$.*

Proof. Let D be one of the terms in the representation, and let p be the number of literals in D. That is $D = \bigwedge_{j=1}^{p} x_{i_j}^{c_{i_j}}$ (here $x^1 = x$ and $x^0 = \bar{x}$). Clearly, the set $\min_b(D) = \min^*_b(D)$ contains a single element, z, defined by $z_i = c_i$ if x_i appears in D and $z_i = b_i$ if x_i does not appear in D. Further, for any two functions g_1, g_2,

$$\min{}_b^*(g_1 \cup g_2) \subseteq \min{}_b^*(g_1) \cup \overset{*}{\underset{b}{\min}}(g_2)$$

and therefore

$$|\min{}_b^*(f)| \leqslant \left|\bigcup_{i=1}^{k} \min{}_b^*(D_i)\right| \leqslant k. \qquad \square$$

Corollary 4.9. *Let $f = D_1 \vee D_2 \vee \cdots \vee D_k$ be a DNF representation for f. Then for every $b \in \{0,1\}^n$, $|\min_b(f)| \leqslant k$.*

Proof. This follows from Claim 4.8, observing that by definition $\min_b(f) \subseteq \min_b^*(f)$. $\square$

Example (*continued*). We go back to the example introduced in Section 3. Recall that we want to reason with respect to the function f, which has the CNF representation

$$f = (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3 \vee \overline{x_4}).$$

The function f has 12 (out of the 16 possible) satisfying assignments. The non-satisfying assignments of f are $\{0000, 0001, 0010, 1101\}$. Using Claim 4.7 we get that the set $B = \{0000, 1101\}$ is a basis for f.

The sets of minimal assignments with respect to this basis are:

$$\min{}_{0000}(f) = \{1000, 0100, 0011\},$$
$$\min{}_{1101}(f) = \{1100, 1111, 1001, 0101\}.$$

These can be easily found by drawing the corresponding lattices and checking which of the satisfying assignments of f are minimal. It is also easy to check that f can be represented as in Eq. (1) using the minimal elements identified.

5. Deduction with models

We consider the deduction problem $KB \models \alpha$. KB is the knowledge base, which is taken to be a propositional expression (i.e., a representation for some Boolean function), and α is also a propositional expression. The assertion $KB \models \alpha$ means that if some model $x \in \{0,1\}^n$ satisfies KB, then it must also satisfy α.

Let $\Gamma \subseteq KB \subseteq \{0,1\}^n$ be a set of models. To decide whether $KB \models \alpha$ we consider the straightforward model-based approach to deduction: for all the models $z \in \Gamma$ check whether $\alpha(z) = 1$. If for some z, $\alpha(z) = 0$, say "no"; otherwise say "yes".

By definition, if $\Gamma = KB$ this approach yields correct deduction, but representing KB by explicitly holding *all* the possible models of KB is not plausible. A model-based approach becomes feasible if one can make correct inferences when working with a small subset of models.

In this section we define a special collection of models, called the *characteristic models* of KB, and denoted Γ_{KB}^{B} (or Γ for short). We show that performing the model-based test on Γ yields correct deduction, and that *all* the characteristic models are necessary in order to guarantee correct reasoning with models. Therefore, this is an optimal set with this property.

5.1. Exact deduction

Definition 5.1. Let $\mathcal{F}$ be a class of functions, and let B be a basis for $\mathcal{F}$. For a knowledge base $KB \in \mathcal{F}$ we define the set $\Gamma = \Gamma_{KB}^{B}$ of *characteristic models* to be the set of all minimal assignments of KB with respect to the basis B. Formally,

$$\Gamma_{KB}^{B} = \bigcup_{b \in B} \{z \in \min{}_b(KB)\}.$$

Theorem 5.2. *Let $KB, \alpha \in \mathcal{F}$ and let B be a basis for $\mathcal{F}$. Then $KB \models \alpha$ if and only if for every $u \in \Gamma_{KB}^{B}$, $\alpha(u) = 1$.*

Proof. Clearly, $\Gamma = \Gamma_{KB}^{B} \subseteq KB$ and therefore, if there is $z \in \Gamma$ such that $\alpha(z) = 0$ then $KB \not\models \alpha$. For the other direction assume that for all $u \in \Gamma$, $\alpha(u) = 1$. We will show that if $y \in KB$, then $\alpha(y) = 1$. From Corollary 4.6, since B is a basis for α, and for all $u \in \Gamma$, $\alpha(u) = 1$, we have that

$$\forall u \in \Gamma,\ \forall b \in B,\ \exists v_{u,b} \in \min{}_b(\alpha) \text{ such that } u \geqslant_b v_{u,b}. \tag{2}$$

Consider now a model $y \in KB$. Again, Corollary 4.6 implies that

$$\forall b \in B,\ \exists z \in \min{}_b(KB) \text{ such that } y \geqslant_b z. \tag{3}$$

By the assumption, since $\min_b(KB) \subseteq \Gamma$, all the elements z identified in Eq. (3) satisfy α and therefore, as in Eq. (2) we have that

$$\forall z \in \min{}_b(KB),\ \exists v_{z,b} \in \min{}_b(\alpha) \text{ such that } z \geqslant_b v_{z,b}. \tag{4}$$

Substituting Eq. (4) into Eq. (3) gives the required condition on $y \in KB$:

$$\forall b \in B,\ \exists v_{(z),b} \in \min{}_b(\alpha) \text{ such that } y \geqslant_b v_{(z),b}$$

which implies, by Corollary 4.6, that $\alpha(y) = 1$. □

The above theorem requires that KB and α can be described by the same basis B. This requirement is somewhat relaxed in the following theorem.

Theorem 5.3. *Let KB be a propositional expression with basis B and let α be a query with basis B'. Then $\Gamma_{KB}^{B \cup B'}$ is a model-based representation for the inference problem $KB \models \alpha$. That is, $KB \models \alpha$ if and only if for every $u \in \Gamma_{KB}^{B \cup B'}$, $\alpha(u) = 1$.*

Proof. It is clear, from Eq. (1) and Claim 4.2(4), that $B \cup B'$ is a basis both for KB and α. Therefore, Theorem 5.2 implies the result. □

Example (*continued*). The set Γ relative to $B = \{0000, 1101\}$ is: $\Gamma = \{1000, 0100, 0011, 1100, 1111, 1001, 0101\}$. Note that it includes only 7 out of the 12 satisfying assignments of f. Since model-based deduction does not make mistakes on queries that are implied by f, we concentrate in our examples on queries that are not implied by f.

To exemplify Theorem 5.2 consider the query $\alpha_1 = \overline{x_2} \wedge \overline{x_3} \to x_4$. This is equivalent to $x_2 \vee x_3 \vee x_4$ which is falsified by 0000 so B is a basis for α_1. Reasoning with Γ will find the counterexample 1000 and will therefore conclude $f \not\models \alpha_1$.

The query $\alpha_2 = x_1 \wedge x_3 \to x_2$ is equivalent to $\overline{x_1} \vee x_2 \vee \overline{x_3}$ which is not falsified by the basis B. Therefore B is not a basis for α_2 and model-based deduction might be wrong. Indeed reasoning with Γ will not find a counterexample and will conclude $f \models \alpha_2$ (it is wrong since the assignments 1010, 1011 satisfy f but not α_2).

Next, to exemplify Theorem 5.3 consider adding a basis element for α_2. This could be either 1010, or 1011. Choosing 1010, the set of additional minimal elements in Γ is $\{1010\}$, and reasoning with Γ would be correct on α_2.

5.2. *Exact deduction with approximate theories*

We have shown in the discussion above how to perform deduction with the set of characteristic models Γ^B_{KB}, where B is a basis for the knowledge base KB. In this section we consider the natural generalization to the case in which the set of characteristic models of KB is constructed with respect to a basis B that is *not* a basis for the knowledge base KB.

We show that even in this case we can perform exact deduction. As we show, reasoning with characteristic models of KB with respect to a basis B is equivalent to reasoning with the *least upper bound* (LUB) [33] of KB in the class of functions with basis B. The significance of this, as proved in Theorem 5.6, is that for queries with basis B reasoning with models yields correct deduction.

A theory of knowledge compilation using Horn approximation was developed by Selman and Kautz in a series of papers [15, 16, 33]. Their goal is to speed up inference by replacing the original expression by two Horn approximations of it, one that implies the original expression (a lower bound) and one that is implied by it (an upper bound). While reasoning with the approximations instead of the original expression does not always guarantee correct reasoning, it sometimes provides a quick answer to the inference problems. Of course, computing the approximations is a hard computational problem, and this is why it is suggested as a *compilation process*. The computational problems of computing Horn approximations and reasoning with them are studied also in [5, 10, 30].

To facilitate the presentation we first define the notion of an approximation of KB. We then show that representing KB with a set of characteristic models with respect to a basis B yields a function which is the LUB of KB (in the class of functions with basis B). We proceed to show the implication to reasoning. In this case, reasoning with models yields correct deduction for all queries in the approximation language. In particular, since there is a small fixed basis for all Horn expressions (see Claim 6.2) we can construct a Horn LUB and reason with it, generalizing the concept defined and discussed in [15, 16, 33].

Definition 5.4 (*Least upper bound*). Let $\mathcal{F}$, $\mathcal{G}$ be families of propositional languages. Given $f \in \mathcal{F}$ we say that $f_{lub} \in \mathcal{G}$ is a $\mathcal{G}$-least upper bound of f if and only if $f \subseteq f_{lub}$ and there is no $f' \in \mathcal{G}$ such that $f \subset f' \subset f_{lub}$.

These bounds are called *$\mathcal{G}$-approximations* of the original function f. The next theorem characterizes the $\mathcal{G}$-LUB of a function and shows that it is unique.

Theorem 5.5. *Let f be any propositional expression and $\mathcal{G}$ the class of all propositional expressions with basis B. Then*

$$f_{lub} = \bigwedge_{b\in B} \mathcal{M}_b(f).$$

Proof. Define $g = \bigwedge_{b\in B} \mathcal{M}_b(f)$. We need to prove that (1) $f \subseteq g$, (2) $g \in \mathcal{G}$ and (3) there is no $f' \in \mathcal{G}$ such that $f \subset f' \subset g$. (1) is immediate from Claim 4.2(4). To prove (2) we need to show that B is a basis for g. Indeed,

$$\begin{aligned}
\bigwedge_{b\in B} \mathcal{M}_b(g) &= \bigwedge_{b\in B} \mathcal{M}_b\left(\bigwedge_{b\in B} \mathcal{M}_b(f)\right) \\
&\subseteq \left(\bigwedge_{b\in B} \mathcal{M}_b(f)\right) \wedge \left(\bigwedge_{b_i,b_j\in B, b_i\neq b_j} \mathcal{M}_{b_i}\mathcal{M}_{b_j}(f)\right) \\
&= g \wedge \left(\bigwedge_{b_i,b_j\in B, b_i\neq b_j} \mathcal{M}_{b_i}\mathcal{M}_{b_j}(f)\right) \\
&\subseteq g.
\end{aligned}$$

Using Claim 4.2(4) again we get $g \subseteq \bigwedge \mathcal{M}_b(g)$ and therefore $\bigwedge_{b\in B} \mathcal{M}_b(g) = g$, that is $g \in \mathcal{G}$. Finally, to prove (3) assume that there exists $f' \in \mathcal{G}$ such that $f \subseteq f'$. Then,

$$g = \bigwedge_{b\in B} \mathcal{M}_b(f) \subseteq \bigwedge_{b\in B} \mathcal{M}_b(f') = f',$$

where the last equality results from the fact that $f' \in \mathcal{G}$. Therefore, $g = f_{lub}$. □

The following theorem can be seen as a generalization of Theorem 5.2, in which we do not require that the basis B is the basis of KB.

Theorem 5.6. *Let $KB \in \mathcal{F}$, $\alpha \in \mathcal{G}$ and let B be a basis for $\mathcal{G}$. Then $KB \models \alpha$ if and only if for every $u \in \Gamma^B_{KB}$, $\alpha(u) = 1$.*

Proof. We have shown in Theorem 5.5 that

$$KB_{lub} = \bigwedge_{b\in B} \mathcal{M}_b(KB) = \bigwedge_{b\in B} \bigvee_{z\in \min_b(KB)} \mathcal{M}_b(z).$$

Assume $\alpha(u) = 1$ for every $u \in \Gamma^B_{KB}$. By Theorem 5.2, we have that $KB_{lub} \models \alpha$ and therefore $KB \models \alpha$. On the other hand, since $\Gamma^B_{KB} \subseteq KB$, if for some $u \in \Gamma^B_{KB}$, $\alpha(u) = 0$, then $KB \not\models \alpha$. $\square$

A result similar to the corollary that follows, for the case in which $\mathcal{G}$ is the class of Horn expressions, is discussed in [5, 6, 15].

Corollary 5.7. *Model-based reasoning with KB_{lub} (with respect to the language $\mathcal{G}$) is correct for all queries in $\mathcal{G}$.*

Example (*continued*). The Horn basis for our example is:

$$B_H = \{1111, 1110, 1101, 1011, 0111\}$$

(see Claim 6.2). The minimal elements with respect to 1101 were given before. Each of the models 1111, 0111, 1011, 1110 satisfies f and therefore for each of these, $\min_b(f) = b$ and together we get that

$$\Gamma^{B_H}_f = \{1111, 0111, 1011, 1100, 1001, 0101, 1110\}.$$

For the query $\alpha_3 = x_1 \wedge x_3 \rightarrow (x_2 \vee x_4)$, which is not Horn, reasoning with $\Gamma^{B_H}_f$ will be wrong (since 1010 satisfies f but not α). For the Horn query $\alpha_2 = x_1 \wedge x_3 \rightarrow x_2$, reasoning with $\Gamma^{B_H}_f$ will find the counterexample 1011 and therefore be correct.

5.3. All models are necessary

So far we have seen that characteristic models can support correct reasoning. The question is whether one really needs all these models in order to guarantee such performance. We next show that this is indeed the case. Any set of models, which guarantees correct reasoning with models for all queries in a class $\mathcal{G}$, must include all the characteristic models (with respect to this class).

In the following theorem, we say that a set R *supports correct reasoning for* $\mathcal{G}$, if for all $\alpha \in \mathcal{G}$, $KB \models \alpha$ if and only if for every $u \in R$, $\alpha(u) = 1$.

Theorem 5.8. *Let $B \subset \{0,1\}^n$ be a set of assignments, and let $\mathcal{G}$ be the class of all Boolean functions that can be represented using B as a basis. Let $KB \in \mathcal{F}$ and $R \subseteq KB \subseteq \{0,1\}^n$. If R supports correct reasoning for $\mathcal{G}$ then $\Gamma^B_{KB} \subseteq R$.*

Proof. Suppose there exists a set R that satisfies the above property and such that $\Gamma^B_{KB} \not\subseteq R$, and consider $x \in \Gamma^B_{KB} \setminus R$. We show that there is a function $\alpha^* \in \mathcal{G}$ such that for all $u \in R$, $\alpha^*(u) = 1$, but still $KB \not\models \alpha^*$, yielding a contradiction.

Indeed, let $\alpha^* = R^B_{lub}$. (That is, the LUB with respect to B of the function whose satisfying assignments are exactly the elements of R.) Then, by definition, $\alpha^* \in \mathcal{G}$, and $R \subseteq \alpha^*$, that is, all the elements in R satisfy α^*. However, $KB \not\models \alpha^*$. To see that, notice that since $x \in \Gamma^B_{KB}$, x is a minimal model with respect to some $b \in B$. With respect to this element b, we get that for all $z \in KB$, and in particular for all $z \in R$, $x \not\geq_b z$, that is $x \notin \mathcal{M}_b(R)$. Using Theorem 5.5 we get that $x \notin \alpha^*$. $\square$

Note the difference in the premises of Theorem 5.8 and the previous two theorems, Theorems 5.2 and 5.6. Theorem 5.8 shows that every element of the set Γ is necessary in order to get correct deduction. What the proof shows is that there is a function α^* in the class represented by B, which necessitates the use of each element x in Γ. Note that, in general, if B is a basis for $\mathcal{G}$ it does not mean that all functions in the class represented by B are in the class $\mathcal{G}$, and therefore the premises of the previous theorems are not enough to yield this result. (We discuss this point further in Section 8.)

In the next section we discuss with some details the basis B_H of the class of Horn expressions. We note that in this case, as well as in the case of the basis B_{H_k} of k-quasi-Horn functions, the bases represent those classes exactly. That is, a function is k-quasi-Horn if and only if it can be represented using B_{H_k}. Therefore, Theorem 5.8 holds for these cases.

6. Horn expressions

In this section we consider in detail the case of Horn formulas and show that in this case our notion of *characteristic models* coincides with the notion introduced in [14]. (Characteristic models for Horn expressions also coincide with the notion of generators in relational database theory [2, 18].) We then discuss the issue of using a *fixed* model-based representation for answering unrestricted queries. We show that this extension, discussed in [14], relies on a special property of Horn formulas and does not generalize to other propositional languages. An example that explains this phenomenon is given. We start by showing that Horn formulas have a small basis.

Definition 6.1. Let $B_H = \{u \in \{0,1\}^n \mid weight(u) \geqslant n-1\}$. In B_H, let $b^{(i)}$ $(1 \leqslant i \leqslant n)$ denote the element with the ith bit set to zero and all the others set to one, and by $b^{(0)}$ the element with weight n.

Claim 6.2. *The set $B_H = \{u \in \{0,1\}^n \mid weight(u) \geqslant n-1\}$ is a basis for any function that can be represented using a Horn CNF expression.*

Proof. Let KB be any Horn function. By Claim 4.7 it is enough to show that if C is a clause in the CNF representation of KB then it is falsified by one of the basis elements in B. Indeed, if C is a clause in which all the literals are negative, then it is falsified by $b^{(0)}$. If x_k is the only variable that appears un-negated in C then C is falsified by $b^{(k)}$. □

6.1. Characteristic models

In order to relate to the results from [14] we need a few definitions presented there.

For $u, v \in \{0,1\}^n$, we define the *intersection* of u and v to be the assignment $z \in \{0,1\}^n$ such that $z_i = 1$ if and only if $u_i = 1$ and $v_i = 1$ (i.e., the bitwise logical-and of u and v). For any set $S \subseteq \{0,1\}^n$, we denote by *intersect*(S) the assignment resulting from bitwise intersection of all models in S.

The *closure* of $S \subseteq \{0,1\}^n$, denoted *closure*(S), is defined as the smallest set containing S that is closed under intersection.

Let *KB* be a Horn expression. The set of the *Horn characteristic models* of *KB*, denoted here $char_H(KB)$ is defined as the set of models of *KB* that are not the intersection of other models of *KB*. Formally,

$$char_H(KB) = \{u \in KB \mid u \notin closure(KB \setminus \{u\})\}. \tag{5}$$

The following claim is due to McKinsey [27], and has also been discussed by Horn [11]. The claim in [27] is given in the context of first order equational expressions, and the notation used there is substantially different. To facilitate the discussion, we give an adaption of the proof to the current terminology. (A different proof of this property, for the propositional domain, appears in [6].)

Claim 6.3 (see [27]). *A Boolean function can be represented using a Horn expression if and only if its set of models is closed under intersection.*

Proof. A proof that the models of Horn expressions are closed under intersection is given in [37]. For the other direction, let C be a CNF expression such that its models are closed under intersection. We claim that every clause c in C can be replaced with a Horn clause c_H such that $C \models c_H \models c$. Therefore, C can be re-written as a Horn expression. (Since $C \models \bigwedge c_H \models \bigwedge c = C$, where the intersection is over all the clauses c in C.)

Let c be any clause implied by C, and assume that it has m positive literals. Define m Horn strengthening [14] clauses of c, as follows: c_i includes all the negative literals of c, and the ith positive literal of c. Thus, each of the clauses c_i is a Horn disjunction, and $c_i \models c$. The claim is that one of the clauses c_i can serve as c_H above.

Assume, by way of contradiction, that this is not the case. Then, for all i, there is a model x^i such that $C(x^i) = 1$ and $c_i(x^i) = 0$. Let $S = \{x^i\}$, and let y be the assignment defined by $y = intersect(S)$. Then since x^i falsifies c_i, all the negative literals of c_i are falsified in x^i (that is, all the variables have value 1), and therefore are also falsified in y. Similarly, the positive literal in c_i is falsified in x^i (that is, the variable has value 0). Therefore, in the intersection y this positive literal is also falsified. We conclude that all the literals in c are falsified in y, and therefore $c(y) = 0$ and $C \not\models c$, a contradiction. □

Based on this characterization of Horn expressions, it is clear that if *KB* is a Horn expression and $M \subseteq KB$ any subset of its models, then $closure(M) \subseteq closure(KB) = KB$. In [14] it is shown that if we take $M = char_H(KB)$, then we get

$$closure(char_H(KB)) = closure(KB) = KB.$$

In particular, Eq. (5) implies that $char_H(KB)$ is the smallest subset of *KB* with that property. Based on this it is then shown that model-based deduction using $char_H(KB)$ yields correct deduction. In the following we show that with respect to the basis B_H from Claim 6.2, and for any Horn expression *KB*, $char_H(KB) = \Gamma_{KB}^{B_H}$. Therefore $char_H(KB)$ is an instance of the theory developed in Section 5, and we can reason with it according to Theorem 5.2.

Theorem 6.4. *Let KB be a Horn expression and*

$$B_H = \{u \in \{0,1\}^n \mid weight(u) \geqslant n-1\}.$$

Then, $char_H(KB) = \Gamma_{KB}^{B_H}$.

Proof. Denote $\Gamma = \Gamma_{KB}^{B_H}$. In order to show that $char_H(KB) \subseteq \Gamma$, it is sufficient to prove that $KB = closure(\Gamma)$. This is true since $char_H(KB)$ is the smallest subset of KB with that property.

Consider $x \in KB$, Corollary 4.6 implies that for all $b^{(i)} \in B$, there exists $u^{(i)} \in \min_{b^{(i)}}(f)$ such that $x \geqslant_{b^{(i)}} u^{(i)}$. We claim that

$$x = intersect(\{u^{(k)} \mid x_k = 0\}) \in closure(\Gamma).$$

To see that, consider first the zero bits of x. Let $x_j = 0$, this implies that $u^{(j)}$ is in the intersection and that it satisfies $x \geqslant_{b^{(j)}} u^{(j)}$. Since $x_j = 0$ and $b_j^{(j)} = 0$ the fact $x_j \geqslant_{b_j^{(j)}} u_j^{(j)}$ implies $u_j^{(j)} = 0$, and the intersection on this bit is also 0.

Consider now the case $x_j = 1$. Since all the $u^{(k)}$ in the intersection are such that $x_k = 0$, the order relation on the jth bit is always the reversed order, $\leqslant_1$. That is, all the $u^{(k)}$ in the intersection satisfy $1 = x_j \geqslant_1 u_j^{(k)}$. This implies that for all the $u^{(k)}$ in the intersection $u_j^{(k)} = 1$ and the intersection on this bit is also 1. This completes the proof of $char_H(KB) \subseteq \Gamma$.

To prove $\Gamma \subseteq char_H(KB)$, we show that if $x \in \Gamma$, x cannot be represented as $x = intersect(\{y, z\})$ where $y, z \in KB$ and $x \neq y, z$. Since $char_H(KB)$ is the collection of *all* those elements in KB (from Eq. (5)), we get the result.

Consider $x \in \min_{b^{(k)}}(KB) \subseteq \Gamma$, and suppose by way of contradiction that $\exists y, z \in KB$ such that $x = intersect(\{y, z\})$ and $x \neq y, z$. Fix the order relation $b^{(k)}$ and consider the indices of x. First consider an index $i \neq k$. Since $b_i^{(k)} = 1$ the order relation of the ith index is the reversed one. Now, if $y_i = z_i$ then $x_i = y_i = z_i$, and if $y_i \neq z_i$ then $x_i = 0$. Therefore, in both cases we get that for all $i \neq k$, $x_i \geqslant_{b_i^{(k)}} y_i$ and $x_i \geqslant_{b_i^{(k)}} z_i$. For the case $k = 0$, the indices $i \neq k$ include all the bits. This implies $x \geqslant_{b^{(k)}} y$ and $x \geqslant_{b^{(k)}} z$ and since $x \in \min_{b^{(k)}}(KB)$, this contradicts the assumption that $x \neq y, z$, and therefore proves the claim.

Otherwise, when $k \neq 0$ we consider also the order relation of the kth index, which is the usual order. Again, if $y_k = z_k$ then $x_k = y_k = z_k$ and if $y_i \neq z_i$ then $x_i = 0$. This implies that $x_k \geqslant_{b_k^{(k)}} y_k$ or $x_k \geqslant_{b_k^{(k)}} z_k$.

Together with the case $i \neq k$ we get that $x \geqslant_{b^{(k)}} z$ or $x \geqslant_{b^{(k)}} y$ (depends on whether $z_k = 0$ or $y_k = 0$). But since $x \in \min_{b^{(k)}}(KB)$, this contradicts the assumption that $x \neq y, z$, and completes the proof. □

6.2. General queries

In [14] it is shown that when the "world" can be described as a Horn expression one can answer any CNF query without re-computing the characteristic models. While we have shown that our general model-based representation coincides with that of [14] for the case of Horn expressions, it turns out that the ability to answer any query relies on a special property of Horn expressions, and does not generalize to other propositional languages. We next give a counterexample that exemplifies this.

The deduction scheme in [14] when α is a general CNF expression, utilizes the following observations:

(1) Every disjunction α can be represented as $\alpha = \beta_1 \vee \cdots \vee \beta_k$, where the β_i are Horn disjunctions.
(2) $KB \models \alpha_1\alpha_2$ if and only if $KB \models \alpha_1$ and $KB \models \alpha_2$.
(3) Let KB be a Horn expression and α any disjunction. If $KB \models \alpha$ then there is a Horn disjunction β such that $KB \models \beta$ and $\beta \models \alpha$.

Notice that observation (3) uses McKinsey's proof of Claim 6.3. (In [14] it is derived in a different way, using a completeness theorem for resolution given in [36].)

Observation (2) implies that it is enough to consider queries that are disjunctions. Given α, the deduction scheme in [14] decomposes it into the Horn disjunctions β_i and tests deduction against the β_i. By (3) at least one of the β_i is implied by KB. While observations (1) and (2) are true even for non-Horn expressions, a decomposition as in (3) does not hold in more general cases. In particular, even expressions in 2-quasi-Horn have minimal conclusions which are not 2-quasi-Horn.

Example. Let: $KB = (x_1 \vee x_2 \vee \overline{x_3} \vee \overline{x_4}) \wedge (x_3 \vee x_5 \vee \overline{x_6})$, $\alpha = x_1 \vee x_2 \vee \overline{x_4} \vee x_5 \vee \overline{x_6}$. The knowledge base is a 2-quasi-Horn expression, and it is easy to check that $KB \models \alpha$. However, there is no disjunction β such that $KB \models \beta \models \alpha$.

7. The size of model-based representations

The complexity of model-based reasoning is directly related to the number of models in the representation. It is therefore important to compare this size with the size of other representations of the same function. In the previous section we have shown that our model-based representation is the same as that in [14] when the function is Horn. In [14] examples are given for large Horn expressions with a small set of characteristic models and vice versa, but it was not yet understood when and why it happens. Our results imply that the set of characteristic models of a Horn expression is small if the size of a DNF description for the same function is small. As we show, the other direction is not true. That is, there are Horn expressions with a small set of characteristic models but an exponential size DNF. We start with a bound on the size of the model-based representation.

Lemma 7.1. *Let B be a basis for the knowledge base KB. Then, the size of the model-based representation of KB is*

$$|\Gamma_{KB}^{B}| \leqslant \sum_{b \in B} |\min{}_b(KB)| \leqslant |B| \cdot |DNF(KB)|.$$

Proof. The lemma follows from Corollary 4.9. □

As the following claim shows, this bound is actually achieved for some functions. For the next claim, we need the following terminology. A term t is an *implicant* of a function f, if $t \models f$. A term t is a *prime implicant* of a function f, if t is an implicant of f and the conjunction of any proper subset of the literals in t is not an implicant.

Claim 7.2. *For any b-monotone function f, $|\min_b(f)| = |DNF(f)|$.*

Proof. We first consider monotone functions (i.e., 0^n-monotone). It is well known that for a monotone function there is a unique DNF representation in which each term is a prime implicant. Let f be a monotone function and consider this representation for f. As in Claim 4.8 we can map every term in the representation to its corresponding minimal element. Moreover, since the terms are monotone and the order relation is 0^n, each of these minimal elements is indeed a minimal element of f (otherwise one of the terms in the representation is not a prime implicant). Therefore, there is a one to one correspondence between prime implicants and minimal assignments of f with respect to $b = 0^n$, and $|\min_{0^n}(f)| = |DNF(f)|$. The same arguments hold for any b-monotone function with respect to the order relation b (one can simply rename the variables) and therefore $|\min_b(f)| = |DNF(f)|$. $\square$

Claim 7.2 explains the two examples in [14]. Both examples are 1^n-monotone Horn functions, one has a small DNF and the other has an exponentially large DNF. In particular, consider the function

$$\begin{aligned} f &= \bigwedge_{x_i \in \{p_i, q_i\}} (\overline{x_1} \vee \overline{x_2} \vee \cdots \vee \overline{x_n}) \\ &= \bigvee_i (\overline{p_i} \wedge \overline{q_i}) \end{aligned}$$

defined on the $2n$ variables $\{p_i, q_i\}_{i=1}^n$ in [14]. The function f does not have a short propositional CNF expression (see details there), but its DNF size is $O(n)$ (observe that when multiplying out the CNF expression, every term contains $\overline{p_i} \wedge \overline{q_i}$ for some i) and the size of the set of characteristic models is $O(n^2)$. On the other hand [14], the function

$$\begin{aligned} f &= \bigwedge_i (\overline{p_i} \vee \overline{q_i}) \\ &= \bigvee_{x_i \in \{p_i, q_i\}} (\overline{x_1} \wedge \overline{x_2} \wedge \cdots \wedge \overline{x_n}) \end{aligned}$$

has a linear size CNF expression, but its DNF, and therefore the set of characteristic models are of size $O(2^n)$.

We note that exponential size model-based representations are not restricted to happen in b-monotone functions. One can easily construct such functions by using, for example, a conjunction of several functions, each b-monotone with a different b (of course the DNF size has to be exponential here too). The following claim shows that DNF size is not a lower bound on the size of the model-based representation. There are expressions for which the DNF size is exponential but the size of the model-based representation, and therefore, the complexity of model-based reasoning, is polynomial.

Claim 7.3. *There exist Horn formulas with an exponential size DNF and a set $\Gamma_f^{B_H}$ of linear size.*

Proof. For each n we exhibit a formula f with the required property. The function

$$\begin{aligned} f = & (\overline{x_1} \vee \overline{x_2} \vee \cdots \vee \overline{x_{\sqrt{n}-1}} \vee x_{\sqrt{n}}) \\ & \wedge (\overline{x_{\sqrt{n}+1}} \vee \overline{x_{\sqrt{n}+2}} \vee \cdots \vee \overline{x_{2\cdot\sqrt{n}-1}} \vee x_{2\cdot\sqrt{n}}) \\ & \wedge \cdots \wedge (\overline{x_{n-\sqrt{n}+1}} \vee \overline{x_{n-\sqrt{n}+2}} \vee \cdots \vee \overline{x_{n-1}} \vee x_n) \end{aligned}$$

is clearly in Horn form.

The size of its DNF representation is $\sqrt{n}^{\sqrt{n}}$. This is easy to observe by renaming each negative literal as its negation. This yields a monotone formula in which each term we get, by multiplying one variable from each clause, is a prime implicant.

The set Γ is of size $< 2n$. Recall that $b^{(i)} \in B_H$ denotes the basis element in which the variable x_i is assigned 0, and that $b^{(0)} = 1^n$. First observe that for $i \neq k\sqrt{n}$, $b^{(i)}$ is a satisfying assignment of f and therefore has only one minimal element (that is, itself). For $i = k\sqrt{n}$, $b^{(i)}$ is not a satisfying assignment of f. There is however only one clause, C^i, not satisfied by $b^{(i)}$, the clause which includes the variable x_i. Now, since each variable appears only once in f, each of the variables in C^i we flip yields a satisfying assignment which is minimal. This contributes $\sqrt{n}$ minimal assignments. (Flipping variables not in C^i does not contribute minimal assignments with respect to $b^{(i)}$.) One last note is that each of these $b^{(i)}$ would have 1^n as one of the minimal assignments, so we need to count it only once, and count $(\sqrt{n}-1)$ for each of the $b^{(i)}$. Altogether there are $(n-\sqrt{n})+\sqrt{n}(\sqrt{n}-1)+1 = 2(n-\sqrt{n})+1$ minimal elements. □

We note that while the previous examples were constructed as Horn functions, it is clear that the same can be done for any class of functions with known basis.

The previous discussion indicates that in some sense, the DNF representation of a Boolean function and the characteristic models representation introduced here are incomparable. While we do not consider here the question of how to get this knowledge representation, the following fact, shown in [19], points to one more advantage of model-based representations. Let f be a function with a polynomial size DNF representation. It is shown there that while we do not know how to efficiently learn a DNF representation of the function from data, it is possible to efficiently learn the set of characteristic models of this function. Therefore, a complete system that learns from data and then reasons using characteristic models can be built, while such a system cannot be built using DNF.

Considering model-based representations, Claim 7.2 implies that for every basis there is a function which has an exponential number of characteristic models. Nevertheless, one might hope that there is a basis for which the least upper bound will always have small representations in some (maybe other) form that admits fast reasoning. Kautz and

Selman [16] show that for Horn representations this is not the case. In particular, they show that unless NP $\subseteq$ non-uniform P there is a function whose Horn LUB does not have a short representation that allows for efficient reasoning. This can be generalized,[3] using essentially the same proof, to hold for every fixed basis and in particular, k-quasi-Horn, $\log n$-CNF, and monotone functions. We therefore have the following theorem:

Theorem 7.4. *Unless* NP $\subseteq$ non-uniform P, *for every fixed basis B there exists a function whose LUB does not have a short representation which admits efficient reasoning.*

8. Applications

In Section 5 we developed the general theory for model-based deduction. In this section we discuss applications of this theory to specific propositional languages.

Our basic result (Theorem 5.2) assumed that the knowledge base and the query share the same basis. We give such queries a special status.

Definition 8.1. Let B be a basis for *KB*. A query α is *relevant* to *KB* (and B), if B is a basis for α.

The notion of relevant queries depends on the particular choice of basis, and is therefore hard to characterize in general. However, relevant queries are useful in situations where *KB* has some special properties (e.g., all the rules are of bounded length). Moreover, the language used for representing *KB* may indicate which queries are more important in a particular domain.

Theorem 5.3 suggests one way in which to overcome the difficulty in the case where the basis B of *KB* is not a basis for the query α. This can be done by: (1) adding the basis B' of the query to the knowledge base basis, and (2) computing additional characteristic models based on the new basis. Claim 4.7 suggests a simple way for computing the basis for a given query, as required in (1). However, the problem of computing additional characteristic models is in general a hard problem that we do not address here. Neither do we consider computing additional models in an on-line process performed for each query. At this point we assume that the knowledge base is given in the form of its set of characteristic models.

A second and preferred way of using the model-based representation is suggested in Section 5.2. Theorem 5.6 shows that one can reason correctly with respect to an unrestricted *KB*, as long as it is represented as a set of characteristic models with respect to the basis of the query language. The power of this method results from the fact that wide classes of query languages have small bases. Some important classes of functions which have fixed polynomial size bases are: (1) Horn CNF formulas, (2) reversed Horn CNF formulas (CNF with clauses containing at most one *negative* literal), (3) k-quasi-Horn formulas (a generalization of Horn expressions in which there are at most k positive literals in each clause), (4) k-quasi-reversed Horn formulas and (5)

[3] This issue has been brought to our attention by Henry Kautz and Bart Selman.

$\log n$-CNF formulas (CNF in which the clauses contain at most $O(\log n)$ literals). This fact is captured by the following definition for common queries.

Definition 8.2. A function is *common* if every clause in its CNF representation is taken from one of the above classes. The union of the bases for these classes is a basis, B_C, for all common functions. We refer to this class as the class of common queries $\mathcal{L}_C$.

Notice that we could add to $\mathcal{L}_C$ any additional class with a fixed polynomial size basis which may fit a particular application. However, we can only add a polynomial number of such classes (since otherwise the cumulative basis size will be too large).

In Claim 6.2 we have shown that Horn formulas have a short basis. A similar construction yields a basis for reversed Horn formulas, k-quasi-Horn formulas, and k-quasi-reversed Horn formulas.

Claim 8.3. *There is a polynomial size basis for reversed Horn formulas, k-quasi-Horn formulas, and k-quasi-reversed Horn formulas.*

Proof. The analysis is very similar to the one in Claim 6.2. By flipping the polarity of all bits in B_H we can get a basis for reversed Horn. Similarly, using the set $B_{H_k} = \{u \in \{0,1\}^n \mid weight(u) \geqslant n-k\}$ we get a basis for k-quasi-Horn, and flipping the polarity of all bits in B_{H_k} we get a basis for k-quasi-reversed Horn formulas. □

Next we consider the expressive class of $\log n$-CNF formulas, in which there are up to $O(\log n)$ variables in a clause, and show that it has a polynomial size basis.

An (n,k)*-universal set* is a set of assignments $\{d_1, \ldots d_t\} \subseteq \{0,1\}^n$ such that every subset of k variables assumes all of its 2^k possible assignments in the d_i. It is known [1] that for $k = \log n$ one can construct (n,k)-universal sets of polynomial size.

Claim 8.4 (see [4]). *Let B be an (n,k)-universal set. Then B is a basis for any k-CNF KB.*

Proof. By Claim 4.7 it is enough to show that if C is a clause in the k-CNF representation of KB then it is falsified by one of the basis elements in B. Let $C = l_{i_1} \vee \cdots \vee l_{i_k}$ be a clause in the CNF representation of KB, where $l_{i_k} \in \{x_{i_k}, \overline{x_{i_k}}\}$. Let $a \in \{0,1\}^n$ be an assignment. Then the value $C(a)$ is determined only by $a_{i_1}, \ldots, a_{i_k}$ and since B is an (n,k)-universal set, there must be an element $b \in B$ for which $C(b) = 0$. □

We note that in general the fact that B is a basis for the class of functions $\mathcal{F}$ does not mean that all functions with basis B are in $\mathcal{F}$. For example, given a particular $(n, \log n)$-universal set B, many other Boolean functions, outside of $\log n$-CNF, have B as their basis. (That is, the class of common queries is in fact wider than stated.) Thus, f_{lub} with respect to B, an $(n, \log n)$ set, is not equivalent to the least upper bound in the class $\log n$-CNF but rather it is the least upper bound in the richer class of functions with basis B. It is easy to observe that this does not happen when using the bases B_H and B_{H_k}. In these cases, the classes of Boolean functions described by the bases are exactly the classes of Horn expressions and k-quasi-Horn expressions, respectively.

8.1. Main applications

Reasoning with common or relevant queries reduces to a simple evaluation of a propositional formula on a polynomial number of assignments. This is a very simple and easily parallelizable procedure. Moreover, Theorem 5.6 shows that in order to reason with common queries, we do not need to use the basis of KB at all, and it is enough to represent KB by the set of characteristic models with respect to the basis of the query language. Lemma 7.1 together with Theorems 5.2, 5.3, 5.5 and 5.6 imply the following general applications of our theory:

Theorem 8.5. *Any function $f : \{0,1\}^n \to \{0,1\}$ that has a polynomial size representation in both DNF and CNF form can be described with a polynomial size set of characteristic models.*

Theorem 8.6. *Any function $f : \{0,1\}^n \to \{0,1\}$ in $\mathcal{F} \in \mathcal{L}_C$ which has a polynomial size DNF representation can be described with a polynomial size set of characteristic models.*

Theorem 8.7. *Let KB be a knowledge base (on n variables) that can be described with a polynomial size set Γ of characteristic models. Then, for any relevant or common query, model-based deduction using Γ is both correct and efficient.*

Theorem 8.8. *Let KB be a knowledge base (on n variables) that can be described with a polynomial size DNF. Then there exists a fixed, polynomial size set of models Γ, such that for any common query, model-based deduction using Γ is both correct and efficient.*

The results in this paper are concerned mainly with propositional languages. While many AI formalizations use first order logic as their main tool, some applications do not need the full power of first order logic. It is quite easy to observe that any such formalization which is function free and has a fixed and finite number of objects can be mapped into a finite propositional language. This can be done by introducing a propositional variable for every possible instantiation of predicates and objects. For example, if the objects are $a, b, c, \ldots$, then a predicate $p(x, y)$ would be represented using $p(a,b), p(b,c), p(a,c), \ldots$ as propositional variables. The number of such variables is bounded by $n_p(n_o)^a$, where n_p denotes the number of predicate symbols, n_o denotes the number of objects, and a denotes the maximum arity of the predicates. Furthermore, a function free universally quantified sentence in Horn form (or any other language with a fixed basis) remains in Horn form[4] in the new propositional domain. These observations imply that our results hold for these restricted first order logic formalizations, where the polynomial bounds are relative to the number of variables in the propositional domain.

[4] We note that the CNF formula size grows exponentially with the number of quantifiers. This, however, does not affect our results which depend on the size of the basis and the size of the DNF formula.

9. Abduction with models

In this section we consider the question of performing abduction using a model-based representation. In [14] it is shown that for a Horn expression *KB*, abduction can be done in polynomial time using characteristic models. This is contrasted with the fact that using formula-based representation the problem is NP-hard [34]. In this section we show that the algorithm presented in [14] works for non-Horn expressions as well.

Abduction is the task of finding a minimal explanation for some observation. Formally [29], the reasoner is given a knowledge base *KB* (the *background theory*), a set of propositional letters[5] A (the *assumption set*), and a query letter q. An *explanation* of q is a minimal subset $E \subseteq A$ such that

(1) $KB \wedge (\bigwedge_{x\in E} x) \models q$ and

(2) $KB \wedge (\bigwedge_{x\in E} x) \neq \emptyset$.

Thus, abduction involves tests for entailment (1) and consistency (2), but also a search for an explanation that passes both tests. We now show how one can use the algorithm from [14] for any propositional expression *KB*.

Theorem 9.1. *Let KB be a background propositional theory with a basis B, let A be an assumption set and q be a query. Let* $B_H = \{x \in \{0,1\}^n \mid weight(x) \geqslant n-1\}$. *Then, using the set of characteristic models* $\Gamma = \Gamma_{KB}^{B\cup B_H}$ *one can find an abductive explanation of q in time polynomial in* $|\Gamma|$ *and* $|A|$.

Proof. We use the algorithm *Explain* suggested in [14] for the case of a Horn knowledge base. For a Horn expression *KB* the algorithm uses the set $char_H(KB) = \Gamma_{KB}^{B_H}$ defined in Section 6. We show that adding the Horn basis B_H and the additional characteristic models to a general model-based representation is sufficient for it to work in the general case.

The abduction algorithm *Explain* starts by enumerating all the characteristic models. When it finds a model in which the query holds (i.e., $q = 1$) it sets E to be the conjunction of all the variables in A that are set to 1 in that model. (This is the strongest set of assumptions that are valid in this model.)

The algorithm then performs the entailment test ((1) in the definition above) to check whether E is a valid explanation. This test is equivalent to testing the deduction $KB \models (q \vee (\bigvee_{x\in E} \bar{x}))$, that is, a deductive inference with a Horn clause as the query. According to Theorem 5.3, this can be done efficiently with $\Gamma_{KB}^{B\cup B_H}$.

If the test succeeds, the assumption set is minimized in a greedy fashion by eliminating variables from E and using the entailment test again. It is clear that if the algorithm outputs a *minimal* assumption set E (in the sense that no subset of E is a valid explanation, not necessarily of smallest cardinality) then it is correct. Minimality is

[5] The task of abduction is normally defined with arbitrary literals for explanations. For Horn expressions explanations turn out to be composed of positive literals (this can be concluded from [29, Corollary 4]). Here we restrict ourselves to explanations composed of positive literals (by allowing only positive literals in the assumption set) when using general expressions. One may therefore talk about "positive explanations" instead of explanations. We nevertheless continue with the term explanation.

guaranteed by the greedy algorithm, the requirement (1) by the deductive test, and the requirement (2) by the existence of the original model that produced the explanation.

It remains to show that if an explanation exists, the algorithm will find one. To prove this, it is sufficient to show that in such a case there is a model $x \in \Gamma$ in which both the bit q and a superset of E are set to 1.

The existence of x is a direct consequence of including the base assignment $b = 1^n$ in the basis. This is true as relative to b we have $1 <_b 0$ for each bit. Therefore if there is a model y which satisfies some explanation E, either it is a minimal assignment relative to b, or $\exists x \leqslant_b y$ and x is in Γ. In the first case $x = y$ is the required assignment, in the second case we observe that $y_i = 1$ implies $x_i = 1$ which is what we need. □

It is quite easy to see that the above theorem can be generalized in several ways. First, we can allow the assumption set A to have up to k negative literals for some constant k and use the basis for k-quasi-Horn instead of B_H. Secondly, we can allow the query q to be any Horn expression instead of just one positive literal.

10. Conclusions and further work

This paper develops a formal theory of model-based reasoning. We have shown that a simple model-based approach can support exact deduction and abduction even when an exponentially small portion of the model space is tested. Our approach builds on (1) identifying a set of characteristic models of the knowledge base that together capture all the information needed to reason with (2) a restricted set of queries. We prove that for a fairly large class of propositional expressions, including expressions that do not allow efficient formula-based reasoning, the model-based representation is compact and provides efficient reasoning.

The restricted set of queries, which we call *relevant queries* and *common queries*, can come from a wide class of propositional languages (and include, for example, quasi-Horn expressions and $\log n$-CNF), or from the same propositional language that represents the "world". We argue that this is a reasonable approach to take in the effort to give a computational theory that accounts for both the speed and flexibility of common-sense reasoning.

The usefulness of the approach developed here is exemplified by the fact that it explains, generalizes and unifies many previous investigations, and in particular the fundamental works on reasoning with Horn models [14] and Horn approximations [15, 16, 33].

Recently, some more positive results for reasoning with characteristic models have been obtained, exhibiting the usefulness of this approach. In particular, efficient algorithms for reasoning within context and for default reasoning have been developed [20]. An extension of the theory presented here, that applies in the case where only partial assignments are given in the knowledge base, is described in [21].

This work is part of a more general framework which views *learning* as an integral part of the reasoning process. We believe that some of the difficulties in constructing an adequate computational theory to reasoning result from the fact that these two tasks are viewed as separate. The "learning to reason" framework, which emphasizes this view, is developed and investigated in [19]. In particular, the results there illustrate the importance of the model-based approach to reasoning.

Several directions for future research are possible. As mentioned in the paper, our results hold for restricted cases of first order logic, where the number of objects, and therefore the size of models is bounded. However, in order to apply this, one has to lose all the structure embedded in the first order formalization. In the general case, though, even the size of the models may be infinite and it is not clear how one can overcome this problem. On the positive side, we note that Fagin [9] has shown that for a certain class of (Horn related) first order logic expressions, a single (infinite) model suffices to answer all queries in the language. Another line of research concerns the problem of planning. Since the original formalizations of planning were in the form of deduction queries, one can reduce a planning problem to several deduction queries. The question here is whether this reduction can be done in a way that the queries can be answered efficiently using a model-based approach. Some simple implications, for finite pre-fixed domains, are quite immediate. However, solving the general question is more demanding.

Acknowledgements

We wish to thank Les Valiant for helpful discussions and comments.

References

[1] N. Alon, J. Bruck, J. Naor, M. Naor and R. Roth, Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs, *IEEE Trans. Inform. Theory* **38** (1992) 509-516.

[2] C. Beeri, M. Dowd, R. Fagin and R. Statman, On the structure of Armstrong relations for functional dependencies, *J. ACM* **31** (1984) 30-46.

[3] R. Brachman and H. Levesque, The tractability of subsumption in frame based description languages, in: *Proceedings AAAI-84*, Austin, TX (1984) 34-37.

[4] N.H. Bshouty, Exact learning via the monotone theory, in: *Proceedings IEEE Symposium on Foundation of Computer Science*, Palo Alto, CA (1993) 302-311.

[5] M. Cadoli, Semantical and computational aspects of Horn approximations, in: *Proceedings IJCAI-93*, Chambery (1993) 39-44.

[6] R. Dechter and J. Pearl, Structure identification in relational data, *Artif. Intell.* **58** (1992) 237-270.

[7] J. Doyle and R. Patil, Two theses of knowledge representation: language restrictions, taxonomic classification, and the utility of representation services, *Artif. Intell.* **48** (1991) 261-297.

[8] D. Etherington, A. Borgida, R. Brachman and H. Kautz, Vivid knowledge and tractable reasoning: Preliminary report, in: *Proceedings IJCAI-89*, Detroit, MI (1989).

[9] R. Fagin, Horn clauses and database dependencies, *J. ACM* **29** (1982) 952-985.

[10] R. Greiner and D. Schuurmans, Learning useful Horn approximations, in: *Proceedings Third International Conference on Principles of Knowledge Representation and Reasoning*, Cambridge, MA (1992) 383-392.

[11] A. Horn, On sentences which are true on direct unions of algebras, *J. Symbolic Logic* **16** (1951) 14-21.

[12] P.N. Johnson-Laird, *Mental Models* (Harvard University Press, Cambridge, MA, 1983).

[13] P.N. Johnson-Laird and R.M.J. Byrne, *Deduction* (Lawrence Erlbaum, London, 1991).

[14] H. Kautz, M. Kearns and B. Selman, Reasoning with characteristic models, in: *Proceedings AAAI-93*, Washington, DC (1993) 34-39.

[15] H. Kautz and B. Selman, A general framework for knowledge compilation, in: *Proceedings International Workshop on Processing Declarative Knowledge*, Kaiserlautern (1991).
[16] H. Kautz and B. Selman, Forming concepts for fast inference, in: *Proceedings AAAI-92*, San Jose, CA (1992) 786–793.
[17] R. Khardon, Translating between Horn representations and their characteristic models, *J. Artif. Intell. Res.* **3** (1995) 349–372.
[18] R. Khardon, H. Mannila and D. Roth, Reasoning with examples: propositional formulae and database dependencies, Tech. Rept. TR-15-95, Aiken Computation Laboratory, Harvard University, Cambridge, MA (1995).
[19] R. Khardon and D. Roth, Learning to reason, in: *Proceedings AAAI-94*, Seattle, WA (1994) 682–687; full version: TR-02-94, Aiken Computation Laboratory, Harvard University, Cambridge, MA (1994).
[20] R. Khardon and D. Roth, Default-reasoning with models, in: *Proceedings IJCAI-95*, Montreal, Que. (1995) 319–325.
[21] R. Khardon and D. Roth, Learning to reason with a restricted view, in: *Proceedings Workshop on Computational Learning Theory* (1995) 301–310.
[22] S.M. Kosslyn, *Image and Mind* (Harvard University Press, Cambridge, MA, 1983).
[23] H. Levesque, Making believers out of computers, *Artif. Intell.* **30** (1986) 81–108.
[24] H. Levesque, Is reasoning too hard?, in: *Proceeding 3rd NEC Research Symposium* (1992).
[25] H. Mannila and K.J. Raiha, Design by example: an application of Armstrong relations, *J. Comput. Syst. Sci.* **33** (1986) 126–141.
[26] J. McCarthy, Programs with common sense, in: *Proceedings Symposium on the Mechanization of Thought Processes*, National Physical Laboratory (1958) Vol. 1, 77–84; reprinted in: M. Minsky, ed., *Semantic Information Processing* (MIT Press, Cambridge, MA, 1968) 403–409; also in: R. Brachman and H. Levesque, eds., *Readings in Knowledge Representation* (1985).
[27] J.C.C. McKinsey, The decision problem for some classes of sentences without quantifiers, *J. Symbolic Logic* **8** (1943) 61–76.
[28] C.H. Papadimitriou, On selecting a satisfying truth assignment, in: *Proceedings 32nd Annual IEEE Symposium on Foundations of Computer Science* (1991) 163–169.
[29] R. Reiter and J. De Kleer, Foundations of assumption-based truth maintenance systems, in: *Proceedings AAAI-87*, Seattle, WA (1987) 183–188.
[30] D. Roth, On the hardness of approximate reasoning, in: *Proceedings IJCAI-93*, Chambery (1993) 613–618; also: *Artif. Intell.* **82** (1996) 273–302.
[31] B. Selman, Tractable default reasoning, PhD thesis, Department of Computer Science, University of Toronto, Toronto, Ont. (1990).
[32] B. Selman and H. Kautz, Model-preference default theories, *Artif. Intell.* **45** (1990) 287–322.
[33] B. Selman and H. Kautz, Knowledge compilation using Horn approximations, in: *Proceedings AAAI-91*, Anaheim, CA (1991) 904–909.
[34] B. Selman and H. Levesque, Abductive and default reasoning: a computational core, in: *Proceedings AAAI-90*, Boston, MA (1990) 343–348.
[35] L. Shastri, A computational model of tractable reasoning—taking inspiration from cognition, in: *Proceedings IJCAI-93*, Chambery (1993) 202–207.
[36] J. Slage, C. Chang and R. Lee, Completeness theorems for semantic resolution in consequence finding, in: *Proceedings IJCAI-69*, Washington, DC (1969) 281–285.
[37] M.H. Van Emden and R.A. Kowalski, The semantics of predicate logic as a programming language, *J. ACM* **23** (1976) 733–742.