Semantical and Computational Aspects of Horn Approximations"

Marco Cadoli
Dipartimento di Informatica e Sistemistica
Universita di Roma "La Sapienza"
via Salaria 113, 1-00198 Roma, Italia
cadoliCassi.ing.uniromal.it

Abstract

In a recent study Selman and Kautz proposed a method, called *Horn approximation,* for speeding up inference in propositional Knowledge Bases. Their technique is based on the *compilation* of a propositional formula into a pair of Horn formulae: a Horn Greatest Lower Bound (GLB) and a Horn Least Upper Bound (LUB). In this paper we address two questions that have been only marginally addressed so far: 1) what is the semantics of the Horn approximations? 2) what is the exact complexity of finding Horn approximations? We obtain semantical as well as computational results. The major results of the former kind are: Horn GLBs are closely related to models of the circumscription; reasoning wrt the Horn LUB can be mapped into classical reasoning. The major results of the latter kind are: finding a Horn GLB is "mildly" harder than solving the original inference problem; finding the Horn LUB is a search problem that cannot be parallelized. We believe that our results provide useful criteria that may help finding a knowledge compilation policy.

1 Introduction

In a recent study [Selman and Kautz, 1991; Kautz and Selman, 1992] Selman and Kautz proposed a method, called *Horn approximation,* for speeding up inference in propositional Knowledge Bases. Propositional inference is the problem of checking whether $\Sigma \models \alpha$ holds, where Σ and α are propositional formulae. The starting point of their technique stems from the fact that inference for general propositional formulae is co-NP-complete —hence polynomially unfeasible— while it is doable in polynomial time when Σ is a Horn formula. The fascinating question they address is the following: is it possible to *compile* a propositional formula Σ into a Horn one Σ' so that a significant amount of the inferences that are performed under Σ can be performed under Σ' in polynomial time?

Selman and Kautz notice that there exist two different ways of doing such a compilation. In the first case the compiled formula satisfies the relation $\Sigma' \models \Sigma$, or equivalently $\mathcal{M}(\Sigma') \subseteq \mathcal{M}(\Sigma)$ —where $\mathcal{M}(\Phi)$ denotes the set of models of the formula Φ. For this reason Σ' is called a *Horn lower bound* —or LB— of Σ. As an example — taken from [Selman and Kautz, 1991]— let Φ be the formula $(man \longrightarrow person) \wedge (woman \longrightarrow person) \wedge (man \vee woman)$. The formula $\Phi_{lb} = man \wedge woman \wedge person$ is a Horn LB of Φ.

The second form of compilation is dual. The compiled version of Σ is a Horn formula Σ' that satisfies the relation $\Sigma \models \Sigma'$, or equivalently $\mathcal{M}(\Sigma) \subseteq \mathcal{M}(\Sigma')$. Σ' is called a *Horn upper bound* —or UB— of Σ. Returning to the previous example, the formula $\Phi_{ub} = (man \longrightarrow person) \wedge (woman \longrightarrow person)$ is a Horn UB of Φ.

The importance of having compiled forms of a Knowledge Base is in that sometimes we can use them for providing a quick answer to an inference problem. As an example, if we are faced with the problem of checking $\Sigma \models \alpha$, we may benefit from the fact that for any Horn LB Σ_{lb} of Σ, $\Sigma_{lb} \not\models \alpha$ implies $\Sigma \not\models \alpha$. Σ_{lb} is therefore a *complete approximation* of Σ. Dually, a Horn UB Σ_{ub} is a *sound approximation* of Σ, since $\Sigma_{ub} \models \alpha$ implies $\Sigma \models \alpha$.

Selman and Kautz notice that some complete approximations are better than others. In the previous example, both $\Phi_{lb1} = man \wedge woman \wedge person$ and $\Phi_{lb2} = man \wedge person$ are Horn LBs of Φ. Φ_{lb2} seems to be a better approximation than Φ_{lb1}, since $\mathcal{M}(\Phi_{lb1}) \subset \mathcal{M}(\Phi_{lb2}) \subset \mathcal{M}(\Phi)$, hence the former is in some precise sense "closer" to Φ than the latter. This consideration leads to the notion of *Horn greatest lower bound* —or GLB— of a formula Σ, which is a Horn formula Σ_{glb} such that $\mathcal{M}(\Sigma_{glb}) \subseteq \mathcal{M}(\Sigma)$ and for no Horn formula Σ' it holds that $\mathcal{M}(\Sigma_{glb}) \subset \mathcal{M}(\Sigma') \subseteq \mathcal{M}(\Sigma)$. In the previous example Φ_{lb2} is a Horn GLB of Φ.

The same argument can be done for Horn upper bounds: in our example both $\Phi_{ub1} = (man \longrightarrow person) \wedge (woman \longrightarrow person)$ and $\Phi_{ub2} = person$ are Horn UBs of Φ, but $\mathcal{M}(\Phi) \subset \mathcal{M}(\Phi_{ub2}) \subset \mathcal{M}(\Phi_{ub1})$, hence Φ_{ub2} is a better approximation of Φ. A *Horn least upper bound* —or LUB— of a formula Σ is a Horn for-

•Work supported by the ESPRIT Basic Research Action N.3012-COMPULOG and by the Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo of the CNR (Italian Research Council).

mula Σ_{lub} such that $\mathcal{M}(\Sigma) \subseteq \mathcal{M}(\Sigma_{lub})$ and for no Horn formula Σ' it holds that $\mathcal{M}(\Sigma) \subseteq \mathcal{M}(\Sigma') \subset \mathcal{M}(\Sigma_{lub})$. Φ_{ub2} is a Horn LUB of Φ.

Selman and Kautz's proposal is to approximate inference wrt a propositional formula Σ by using its Horn GLBs and LUBs. In this way inference could be unsound or incomplete, but it is anyway possible to spend more time and use a general inference procedure to determine the answer directly from the original formula. The general inference procedure could still use the approximations to prune its search space (see [Selman and Kautz, 1991, page 905]). It is also important to notice that Horn GLBs and LUBs can be computed off-line, hence this form of approximate reasoning is actually a *compilation.*

Table 1 summarizes the major properties of Horn GLBs and LUBs stated in [Selman and Kautz, 1991; Kautz and Selman, 1992]. The four columns refer respectively to:

- logical relation wrt Σ (i. e. what kind of inference can be performed using this approximation?);
- size of the formula wrt the size $|\Sigma|$ of Σ;
- number of possible approximations of this kind;
- computational complexity of the search problem of finding the approximation.

	INFER.	SIZE	NUMBER	COMPLEX.
Σ_{glb}	compl. unsound	linear	many	NP-hard
Σ_{lub}	sound incompl.	in general expon.	one	NP-hard

Table 1: Some properties of Horn GLBs and LUBs.

Horn approximations have two computational problems: 1) computing them is an NP-hard task and 2) due to its exponential size, it may be impossible to store the Horn LUB. About the first aspect Selman and Kautz notice that since approximations could be computed off-line, the computational cost of finding them will be amortized over the total set of subsequent queries to the Knowledge Base. With respect to the second aspect, they propose in [Kautz and Selman, 1992] a technique for "compressing" the Horn LUB into a (quasi-)equivalent formula. Due to reasons related to circuit complexity theory, it is not possible to apply the technique in general (see [Kautz and Selman, 1992] for further details).

Other computational properties of Horn approximations are studied in [Greiner and Schuurmans, 1992; Roth, 1993].

In this paper we address two important questions that have not been addressed so far:

1. is it possible to describe Horn approximations with a <u>semantics</u> that does not rely on the syntactic notion of Horn clause?
2. what is the <u>exact</u> complexity of finding Horn approximations?

An answer to the first question shows the exact meaning of the approximate answers. An answer to the second question tells in which cases it is reasonable —from the computational point of view— to use Horn approximations.

We obtain two different kinds of results:

semantical

- Horn GLBs of Σ are closely related to models of the circumscription of Σ;
- reasoning wrt Horn GLBs is the same as reasoning by counterexamples using only minimal models;
- while *skeptical* reasoning wrt the Horn GLBs of a formula Σ is the same as ordinary reasoning wrt Σ, *brave* reasoning wrt the Horn GLBs of Σ is the same as reasoning wrt *CIRC*(Σ);
- compiling more knowledge does not always give better Horn GLBs;
- reasoning wrt the Horn LUB can be mapped into classical reasoning;
- the Horn LUB of Σ is related to *CWA*(Σ).

computational

- finding a Horn GLB is "mildly" harder than solving the original inference problem;
- reasoning wrt the Horn LUB is exactly as hard as solving the original inference problem;
- finding a Horn UB is a search problem that cannot be parallelized.

We believe that our results provide useful criteria that may help finding a knowledge compilation policy. In particular, we show that an interesting tradeoff seems to emerge between the computation done during the compilation time and the computation done during the query answering time.

The structure of the paper is as follows: in Section 2 and 3 we study Horn GLBs and LUBs, respectively; we discuss our results in Section 4.

2 Horn GLBs

In this section Σ denotes a propositional formula and Σ_{glb} denotes one of its Horn GLBs. We assume that both of them are in CNF. We start with some considerations on the syntactic form of Horn GLBs.

In [Selman and Kautz, 1991] a *Horn-strengthening* of a clause γ is a Horn clause γ_H such that $\gamma_H \subseteq \gamma$ and there is no Horn clause γ'_H such that $\gamma_H \subset \gamma'_H \subseteq \gamma$. As noticed in [Selman and Kautz, 1991, Lemma 2], every clause in Σ_{glb} is a Horn-strengthening (i. e. a "witness") of a clause in Σ. On the other hand we can easily prove that for each clause in Σ there is at least one Horn-strengthening ("witness") in Σ_{glb}.

In the following two subsections we prove that Horn GLBs of a formula Σ are closely related to the *minimal* models of Σ. Minimal models of a propositional formula have the property that the set of atoms that they map into 1 is minimal. More formally (see [Lifschitz, 1985]), given two models M, N of a formula, we write $M \leq N$ iff

$\{x \mid M(x) = 1\} \subseteq \{x \mid N(x) = 1\}$ and we write $M < N$ iff the containment is strict. The models of a formula Φ that are minimal in this preorder are called the minimal models of Φ. Minimal models are important in the theory of non-monotonic reasoning, since they are the semantical counterpart of circumscription [McCarthy, 1980; Lifschitz, 1985]: The models of $CIRC(\Sigma)$ are exactly the minimal models of Σ.

We recall that Horn formulae have a unique minimal model (the *minimum* model).

2.1 From GLBs to minimal models

Let Σ be a propositional formula and Σ_{glb} a Horn GLB of Σ. We prove that the minimum model M of Σ_{glb} is minimal for Σ, thus proving that if a Horn GLB of Σ is known, then it is possible to obtain in linear time a minimal model of Σ (see [Dowling and Gallier, 1984]).

First of all we notice that M is also a model of Σ. Now, let's assume that M is not minimal, and let N be a model of Σ such that $N < M$. We prove that we can build a Horn formula U such that $\mathcal{M}(\Sigma_{glb}) \subset \mathcal{M}(U) \subseteq \mathcal{M}(\Sigma)$, thus contradicting the assumption that Σ_{glb} is a Horn GLB of Σ.

The Horn formula U is built as follows:

```
begin
   unmark all the clauses of Σ;
   U := true;
   for each clause γ = a1 ∨ ··· ∨ am ∨ ¬b1 ∨ ··· ∨ ¬bn
   of Σ do
         for i := 1 to m do
            if N(ai) = 1
            then begin
                  (* add a Horn-strengthening of γ *)
                  U := U ∧ (ai ∨ ¬b1 ∨ ··· ∨ ¬bn);
                  mark γ
               end;
   for each unmarked clause γ of Σ
   do begin
         let γ' be (one of) the witness(es) of γ in Σglb;
         U := U ∧ γ';
      end;
end.
```

Since U is a collection of Horn-strengthenings of Σ, $\mathcal{M}(U) \subseteq \mathcal{M}(\Sigma)$ holds. It is easy to prove that N is a model of U: 1) N clearly satisfies all the clauses in U that come from marked clauses of Σ; 2) N must satisfy at least one negative literal of each clause γ' in U that comes from an unmarked clause γ of Σ, otherwise γ would have been marked.

Now we prove that $\mathcal{M}(\Sigma_{glb}) \subset \mathcal{M}(U)$ holds. Since $N \in \mathcal{M}(U)$ and $N \notin \mathcal{M}(\Sigma_{glb})$, it is sufficient to prove that $\mathcal{M}(\Sigma_{glb}) \subseteq \mathcal{M}(U)$. Let's take a generic model P of Σ_{glb}; we prove that it is also a model of U. Since P is a model of Σ_{glb}, $M < P$ must hold, hence $N < P$ holds too. As a consequence P satisfies all the clauses in U that come form marked clauses of Σ. As far as the other clauses of U are concerned, they are clauses of Σ_{glb} as well, therefore P satisfies all of them.

The following theorem summarises the above result.

Theorem 1 Let Σ be a propositional formula and Σ_{glb} a Horn GLB of Σ. The minimum model of Σ_{glb} is minimal for Σ.

Theorem 1 implies that if we have a Horn GLB of Σ, then we can obtain in linear time (see [Dowling and Gallier, 1984]) a minimal model of Σ. More technically, the theorem shows a polynomial reduction from the search problem of finding a minimal model of Σ to the search problem of finding a Horn GLB of Σ. The present author analyzed in [Cadoli, 1992] the computational complexity of the search problem of finding a minimal model of a propositional formula. One of the results of that paper is that finding a minimal model of a formula Σ is hard (using many-one reductions) with respect to the class $P^{NP[O(\log n)]}$. [1] It is important to remark that $P^{NP[O(\log n)]}$-hard problems are in a precise sense computationally harder than NP-complete or co-NP-complete problems [2]. We recall that the problem of deciding whether $\Sigma \models \alpha$ holds is co-NP-complete.

As shown in [Cadoli, 1992], $P^{NP[O(\log n)]}$-hardness of finding a minimal model holds even if a model of Σ is known. This fact can be compared with a consideration in [Selman and Kautz, 1991, Theorem l]: Σ_{glb} is satisfiable iff Σ is satisfiable, hence finding a Horn GLB is NP-hard. We can now say that even if we know that Σ is satisfiable and have one of its models in hand, finding a Horn GLB is still $P^{NP[O(\log n)]}$-hard. We recall that finding a model (not necessarily minimal) of a propositional formula is per se an NP-hard task.

Corollary 2 Finding a Horn GLB of a propositional formula Σ is $P^{NP[O(\log n)]}$-hard. This holds even if a model of E is already known.

We notice that the above corollary gives us just a lower bound. It is reasonable to ask how easy is to find a Horn GLB, i. e. to give an upper bound to the problem. In [Selman and Kautz, 199l] an algorithm for computing a Horn GLB of a formula Σ is shown. The algorithm performs an exponential number of polynomial steps. It is possible to show that a Horn GLB can be found in polynomial time by a deterministic Turing machine with access to an NP oracle, i. e. to prove that the problem is in the class P^{NP}. This means that we only need a polynomial number of queries to the GLB in order to "pay off" the overhead of the knowledge compilation.

2.2 From minimal models to GLBs

We now show that if we have a minimal model M of a formula Σ, then we can easily build a very good approximation of a Horn GLB of Σ. In particular we show that we can build in linear time a Horn LB of Σ whose minimum model is M. This result allows us to perform, in

[1] $P^{NP[O(\log n)]}$ is the class of decision problems that can be computed by a polynomial-time deterministic machine which can use for free an oracle (or subroutine) that answers a set of NP-complete queries (e. g. satisfiability checks) whose cardinality is bound by a logarithmic function. We refer the reader to [Johnson, 1990] for a thorough description of all the complexity classes that are cited in this paper.

[2] Both NP-complete and co-NP-complete problems can be solved with a single call to an oracle in NP.

the following subsection, some interesting considerations on the semantics of Horn GLBs.

We build a Horn LB W of Σ, whose minimum model is M. Moreover, we prove that there is a Horn GLB of Σ whose minimum model is also M. W is built as follows:

begin
  unmark all the clauses of Σ;
  $W :=$ **true**;
  for each clause $\gamma = a_1 \vee \cdots \vee a_m \vee \neg b_1 \vee \cdots \vee \neg b_n$ of Σ **do**
    for $i := 1$ **to** m **do**
      if $M(a_i) = 1$
      then begin
        $W := W \wedge (a_i \vee \neg b_1 \vee \cdots \vee \neg b_n)$;
        mark γ
      end;
  for each unmarked clause $\gamma = a_1 \vee \cdots \vee a_m \vee \neg b_1 \vee \cdots \vee \neg b_n$ of Σ **do**
    $W := W \wedge (\neg b_1 \vee \cdots \vee \neg b_n)$;
end.

M is clearly a model of W, since 1) M satisfies all of the clauses in W that come from marked clauses of Σ, 2) M must satisfy at least one negative literal of each clause γ' in W that comes from an unmarked clause γ of Σ, otherwise γ would have been marked. Since W is a collection of Horn-strengthenings of Σ, $\mathcal{M}(W) \subseteq \mathcal{M}(\Sigma)$ holds. As a consequence M is the minimum model of W.

W is a LB of Σ, but it might be not a Horn GLB. As an example if $\Sigma = \neg a \vee b$, $M(a) = M(b) = 0$, then $W = \neg a$. Let V be a Horn GLB of Σ such that $\mathcal{M}(W) \subset \mathcal{M}(V)$. It is easy to prove that M is the minimum model of V: all of the models of V must be greater or equal than its minimum model; since M is a model of V and no model of Σ is smaller than M, M is the minimum model of V.

Theorem 3 *Let M be a minimal model of Σ. There is a Horn GLB of Σ whose minimum model is M.*

2.3 Semantical consequences

Theorems 1 and 3 can be summarized as follows.

Theorem 4 *The set of minimal models of a formula Σ and the set of minimum models of the Horn GLBs of Σ are the same.*

We now address some interesting semantical consequences of the above results.

As noticed in [Selman and Kautz, 1991] a traditional AI approach is *reasoning by counterexamples*, which consists in refuting a possible consequence of a theory by means of a suitable model that contradicts it (an example of this technique is in the early work [Gelernter, 1959]). This approach is based on the well-known property $M \not\models \alpha \Longrightarrow \Sigma \not\models \alpha$, that holds for any pair of formulae α, Σ and any model M of Σ. Selman and Kautz indicate that reasoning under a specific Horn GLB is an improved version of such a reasoning schema, since a single Horn GLB captures a *set* of models of the original theory. Using Theorem 4 and the well-known fact that the minimum model of a Horn formula completely characterizes the set of its positive consequences, we can say that, as far as positive theorems are concerned, reasoning under Horn GLBs is the same as reasoning by counterexamples using only minimal models. This does not hold for negative theorems.

Selman and Kautz address briefly the issue of how reasoning with respect to a set of Horn GLBs looks like, proving [Selman and Kautz, 1991, Theorem 3] that a formula is equivalent to the disjunction of all its Horn GLBs. The notions of *skeptical* and *brave* reasoning are frequently used in the literature. We say that a formula α skeptically follows from the Horn GLBs of a formula Σ (written $skep\text{-}glb(\Sigma) \vdash \alpha$) if for each of its Horn GLBs Σ_{glb} it holds $\Sigma_{glb} \models \alpha$. We also say that α bravely follows (written $brave\text{-}glb(\Sigma) \vdash \alpha$) if there exists a Horn GLB Σ_{glb} such that $\Sigma_{glb} \models \alpha$.

The result by Selman and Kautz can be rephrased in the following way: $skep\text{-}glb(\Sigma) \vdash \alpha$ iff $\Sigma \models \alpha$. Using Theorems 1 and 3 we can say that — as far as positive theorems are concerned — brave reasoning wrt Horn GLBs is the same as brave reasoning wrt minimal models. More precisely let α be a positive clause, $brave\text{-}glb(\Sigma) \vdash \alpha$ iff there exists a minimal model M of Σ s. t. $M \models \alpha$, i. e. iff $CIRC(\Sigma) \not\models \neg\alpha$ (see [Lifschitz, 1985]). Using a result shown in [Eiter and Gottlob, 1991] we can say that brave reasoning wrt Horn GLBs is a decision problem which is hard wrt the class Σ_2^p of the polynomial hierarchy.

Let us see how the relation with non-monotonicity just shown affects approximate inference under Horn GLBs. We consider two knowledge bases Σ, Σ^+ such that $\mathcal{M}(\Sigma^+) \subset \mathcal{M}(\Sigma)$, i. e. $\Sigma^+ \models \Sigma$ and $\Sigma \not\models \Sigma^+$. It is well known that for a generic formula α, $CIRC(\Sigma) \models \neg\alpha$ does not imply $CIRC(\Sigma^+) \models \neg\alpha$, even if $\Sigma^+ \not\models \alpha$ holds. Using the relation $(CIRC(\Sigma) \models \neg\alpha)$ iff $(brave\text{-}glb(\Sigma) \not\vdash \alpha)$ we prove the following result.

Proposition 5 *Let Σ, Σ^+ be two formulae such that $\Sigma^+ \models \Sigma$ and $\Sigma \not\models \Sigma^+$; let α be a positive clause. If $\Sigma^+ \not\models \alpha$ and $brave\text{-}glb(\Sigma) \not\vdash \alpha$ then it might be the case that $brave\text{-}glb(\Sigma^+) \vdash \alpha$.*

We recall that reasoning using a generic Horn GLB is complete and unsound wrt reasoning using the original formula. In other words if $brave\text{-}glb(\Sigma) \not\vdash \alpha$ then we know that $\Sigma \not\models \alpha$, i. e. α is disproved. Proposition 5 says that if we are able to disprove a formula α using any complete compilation of a "small" formula Σ, then we are not guaranteed that we are able to disprove α using a generic complete compilation of a "bigger" formula Σ^+, even if $\Sigma^+ \not\models \alpha$. In other words it is not true that compiling more knowledge we always have better complete approximations. As an example, let $\Sigma = \neg a \vee \neg b$ and $\Sigma^+ = \Sigma \wedge (a \vee b)$. Clearly $\Sigma^+ \not\models a$ and $brave\text{-}glb(\Sigma) \not\vdash a$. Moreover $brave\text{-}glb(\Sigma^+) \vdash a$, as Σ^+ has two different Horn GLBs: $(a \wedge \neg b)$ and $(\neg a \wedge b)$.

For the sake of completeness, we notice that $brave\text{-}glb(\Sigma) \vdash \alpha$ does not imply $brave\text{-}glb(\Sigma^+) \vdash \alpha$: this can be seen if $\Sigma = a \vee b$, $\Sigma^+ = \Sigma \wedge b$ and $\alpha = a$.

3 Horn LUB

In this section Σ denotes a propositional formula and Σ_{lub} denotes its Horn LUB. As shown in [Kautz and Selman, 1992] in general it is not possible to store efficiently the Horn LUB of Σ. In particular the size of Σ_{lub} can be exponential in the size of Σ, and this seems to be independent on the representation used for Σ_{lub} (see [Kautz and Selman, 1992] for further details). As a consequence any method for efficiently representing the Horn LUB is incomplete. In [Selman and Kautz, 1991, page 908] the authors propose to *approximate* the Horn LUB with Horn upper bounds oflimited length. This idea is used in [Greiner and Schuurmans, 1992], where Horn UBs with a limited number of Horn clauses are studied. In Subsection 3.1 we investigate about this idea and analyze its computational properties. Other computational properties of Horn LUBs are addressed in Subsection 3.2. In Subsection 3.3 we make a brief semantical remark.

3.1 Horn UBs with a limited number of clauses

Σ_{lub} is logically equivalent to the conjunction of all the Horn prime implicates of Σ [Selman and Kautz, 1991, Theorem 4]. Σ_{lub} therefore guarantees sound and complete reasoning wrt Σ as far as inference of Horn formulae is concerned: for all Horn formulae α, $(\Sigma_{lub} \models \alpha)$ iff $(\Sigma \models \alpha)$. One natural choice is to approximate Σ_{lub} with a formula that guarantees sound and complete reasoning wrt Σ as far as inference of *short* Horn formulae is concerned. As an example of this kind of approximation, we define the formula Σ^1_{ub} to be the conjunction of the formulae in the set $\{x \mid x \text{ is a positive literal and } \Sigma \models x\}$. Notice that Σ^1_{ub} is a Horn UB of Σ. This formula is a reasonable approximation of Σ, since 1) at least all the positive atomic queries are answered correctly and 2) it has a nice short representation.

An interesting question is the following: how difficult is to obtain Σ^1_{ub}? We notice that finding Σ^1_{ub} is the search problem that amounts to decide for each propositional variable x occurring in Σ whether $\Sigma \models x$ holds. It is well known that just deciding $\Sigma \models x$ for a single propositional variable is co-NP-complete, but it is important to understand if the task of deciding $\Sigma \models x$ for *many* propositional variables can be parallelized. In other words we are interested in the following practical problem: is it possible to obtain Σ^1_{ub} with one —or few— queries to a propositional theorem prover, or is it the case that the best strategy is just to ask separately for each propositional variable x of Σ whether $\Sigma \models x$ holds? Clearly if it is possible to parallelise the process of building Σ^1_{ub} —or any other approximation of Σ_{lub}— then we have better chances to obtain good approximations of Σ_{lub}.

Several authors (see for example [Beigel, 1988; Krentel, 1988]) studied the computational complexity of search problems of the kind we are addressing here. The goal of the research in this field is to understand "how much NP-hardness" does an NP-hard problem contain. The problem QUERY, which is a generalization of the standard satisfiability problem SAT, is defined in [Krentel, 1988]. The input of QUERY are k propositional formulae $T_1, \ldots, T_k$ and its output are k bits $b_1, \ldots, b_k$, where for any i $(1 \leq i \leq k)$, $b_i = 1$ if T_i is satisfiable, and $b_i = 0$ if T_i is not satisfiable. Beigel shows in [Beigel, 1988] that it is very unlikely that QUERY can be solved with less than k queries to a SAT oracle. In other words it seems that any strategy for solving QUERY cannot be better than solving independently the k corresponding SAT problems. A general property of this kind of NP-hard problems (see [Beigel, 1988] for further details) is that it is not possible to gain efficiency via parallelization.

QUERY can be immediately mapped into the problem of finding the approximation Σ^1_{ub} of a given formula Σ. Moreover the proof can be immediately adapted to the problem of finding any set of Horn prime implicates of Σ.

This result can be interpreted in the following way: the task of finding short approximations of Σ_{lub} —like for example Σ^1_{ub}— contains "a lot of NP-hardness". There seems to be a direct correspondence between the size of the approximation and the computational effort that we need to obtain it. As a consequence there is little hope to obtain good approximations of Σ_{lub} by performing few calls to a theorem prover.

3.2 How hard is to decide $\Sigma_{lub} \models \alpha$?

In the previous subsection we addressed the issue of how hard is to compile Σ, and in particular how hard is to obtain an approximation of Σ_{lub}. In this subsection we are interested in another computational property of Σ_{lub}: we want to know how hard is to reason wrt Σ_{lub}, regardless of the representation of this formula that we are currently storing in our memory. In other words we want to understand what is the exact complexity of deciding $\Sigma_{lub} \models \alpha$, assuming that the inputs are Σ and α.

We assume that the formula α is in CNF. In particular, let α be a clause $\neg b_1 \vee \cdots \vee \neg b_m \vee a_1 \vee \cdots \vee a_n$, that we denote as $\beta \longrightarrow a_1 \vee \cdots \vee a_n$, where β is a shorthand for the conjunction $b_1 \wedge \cdots \wedge b_m$. It is not hard to prove that inference wrt Σ_{lub} can be mapped into classical inference. Let $\mathcal{L}$ be the set of letters that occur in Σ, $\mathcal{L}_1, \ldots, \mathcal{L}_n$ be n disjoint sets of letters of the same arity of $\mathcal{L}$ and $\Sigma^1, \ldots, \Sigma^n$ be n duplicates of Σ built on $\mathcal{L}_1, \ldots, \mathcal{L}_n$, respectively. In an analogous way we define $\beta^1, \ldots, \beta^n$. $\Sigma_{lub} \models \alpha$ holds iff

$$\Sigma^1 \wedge \cdots \wedge \Sigma^n \models (\beta^1 \wedge \cdots \wedge \beta^n) \longrightarrow (a^1_1 \vee \cdots \vee a^n_n) \quad (1)$$

This proves that the problem of deciding $\Sigma_{lub} \models \alpha$ can be solved by means of a single call to a propositional theorem prover. More formally, $\Sigma_{lub} \models \alpha$ is a co-NP-complete problem, i. e. it has exactly the same complexity as the original problem of deciding $\Sigma \models \alpha$.

Summarizing all the results presented so far we can say that:

1. it is not possible to represent Σ_{lub} explicitly in the memory;
2. we can make Σ_{lub} partially explicit; this is doable off-line but it is more difficult than the original task of reasoning wrt Σ;
3. if we keep Σ_{lub} completely implicit then reasoning wrt it is exactly as hard as reasoning (on-line) wrt Σ.

An interesting tradeoff therefore exists between the amount of compilation that we want to perform off-line and the amount of reasoning that we want to do on-line.

3.3 A semantical remark

Equation (1) gives a sound and complete characterization of inference wrt Σ_{lub} in terms of classical propositional inference. In this subsection we make a brief remark about the relation existing between Horn LUBs and closed-world assumption [Reiter, 1978].

Observation 6 Let M be the minimum model of Σ_{lub}. M is the intersection of all the minimal models of Σ. Therefore M is a model of Σ iff the closed-world assumption $CWA(\Sigma)$ of Σ is consistent.

We notice that $CWA(\Sigma)$ may be consistent even i Σi s non-Horn: The CWA of $\neg a \vee b \vee c$ is consistent. Relations between Horn LUBs and closed-world reasoning are implicit in the works [Borgida and Etherington, 1989; Selman and Kautz, 199l].

4 Discussion

The computational results that we have seen in Sections 2 and 3 show that when we deal with knowledge compilation there exists an interesting tradeoff between computation during compile time (off-line) and computation during query-answering time (on-line).

In Section 2 we have seen that the computational effort of finding a Horn GLB is justified only if a significant number of queries to it will be done. In particular we have seen that the compilation is more expensive than a set of query answering tasks. The size of such a set has a lower bound which is a function logarithmic in the size of the input and an upper bound which is a function polynomial in the size of the input.

In Section 3 we have obtained similar results, showing that high-quality Horn UBs need a significant computational effort.

Since compilation causes anyway loss of information (either soundness or completeness), the computational effort spent in the compilation must be compared to the quality of the inference obtained. It is an open issue to find an adequate formal framework for comparing the two aspects.

Acknowledgments

I am grateful to H. Kautz and B. Selman for fruitful discussions on their method. I had interesting discussions also with L. C. Aiello, M. Lenzerini, E. Omodeo and M. Schaerf, who also read previous versions of the paper. Thanks to D. Schuurmans for sending me a preliminary version of his paper.

References

[Beigel, 1988] R. Beigel. NP-hard sets are p-terse unless R=NP. Technical Report 88-04, The Johns Hopkins University, Department of Computer Science, August 1988.

[Borgida and Etherington, 1989] A. Borgida and D. W. Etherington. Hierarchical knowledge bases and efficient disjunctive reasoning. In Proceedings of the First International Conference on the Principles of Knowledge Representation and Reasoning (KR-89), pages 33-43, 1989.

[Cadoli, 1992] M. Cadoli. On the complexity of model finding for nonmonotonic propositional logics. In Proceedings of the Fourth Italian Conference on Theoretical Computer Science, pages 125-139. World Scientific Publishing Co., October 1992.

[Dowling and Gallier, 1984] W. P. Dowling and J. H. Gallier. Linear-time algorithms for testing the satisfiability of propositional Horn formulae. Journal of Logic Programming, 1:267-284, 1984.

[Eiter and Gottlob, 1991] T. Eiter and G. Gottlob. Propositional circumscription and extended closed world reasoning are Π^p_2-complete. Technical Report CD-TR 91/20, Technische Universitat Wien, Austria, Christian Doppler Labor fur Expertensysteme, 1991. To appear in Theoretical Computer Science.

[Gelernter, 1959] H. Gelernter. Realization of a geometry theorem-proving machine. In Proceedings of the International Conference on Information Processing, pages 273-282, Paris, 1959. UNESCO House.

[Greiner and Schuurmans, 1992] R. Greiner and D. Schuurmans. Learning useful Horn approximations. In Proceedings of the Third International Conference on the Principles of Knowledge Representation and Reasoning (KR-92), 1992.

[Johnson, 1990] D. S. Johnson. A catalog of complexity classes. In J. van Leeuwen, editor, Handbook of Theoretical Computer Science, volume A, chapter 2. Elsevier Science Publishers B. V. (North Holland), 1990.

[Kautz and Selman, 1992] H. A. Kautz and B. Selman. Forming concepts for fast inference. In Proceedings of the Tenth National Conference on Artificial Intelligence (AAAI-92), pages 786-793, 1992.

[Krentel, 1988] M. V. Krentel. The complexity of optimization problems. Journal of Computer and System Sciences, 36:490-509, 1988.

[Lifschitz, 1985] V. Lifschitz. Computing circumscription. In Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85), pages 121-127, 1985.

[McCarthy, 1980] J. McCarthy. Circumscription - a form of non-monotonic reasoning. Artificial Intelligence Journal, 13:27-39, 1980.

[Reiter, 1978] R. Reiter. On closed world data bases. In H. Gallaire and J. Minker, editors, Logic and Data Bases, pages 119-140. Plenum, 1978.

[Roth, 1993] D. Roth. On the hardness of approximate reasoning. In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93), 1993.

[Selman and Kautz, 1991] B. Selman and H. A. Kautz. Knowledge compilation using Horn approximations. In Proceedings of the Ninth National Conference on Artificial Intelligence (AAA 1-91), pages 904-909, 1991.