A Logic of Limited Belief for Reasoning with Disjunctive Information

Yongmei Liu
Dept. of Computer Science
University Of Toronto
Toronto, ON, M5S 3G4, Canada
yliu@cs.toronto.edu

Gerhard Lakemeyer
Dept. of Computer Science
RWTH Aachen
52056 Aachen, Germany
gerhard@cs.rwth-aachen.de

Hector J. Levesque
Dept. of Computer Science
University Of Toronto
Toronto, ON, M5S 3G4, Canada
hector@cs.toronto.edu

Abstract

The goal of producing a general purpose, semantically motivated, and computationally tractable deductive reasoning service remains surprisingly elusive. By and large, approaches that come equipped with a perspicuous model theory either result in reasoners that are too limited from a practical point of view or fall off the computational cliff.

In this paper, we propose a new logic of belief called $\mathcal{SL}$ which lies between the two extremes. We show that query evaluation based on $\mathcal{SL}$ for a certain form of knowledge bases with disjunctive information is tractable in the propositional case and decidable in the first-order case. Also, we present a sound and complete axiomatization for propositional $\mathcal{SL}$.

Introduction

One of the most important yet elusive goals in the whole area of Knowledge Representation is to devise a semantically coherent yet computationally well-behaved reasoning service that could be used as a black box by a wide variety of systems in a wide variety of applications. There are two obvious limit points we might consider: at one extreme, we imagine a service based on classical logical entailment, perhaps augmented nonmonotonically; at the other extreme, we imagine a service based only on retrieval, perhaps augmented by some syntactic normalization. In between these two limits, however, there is controversy: for some, any divergence from classical logical entailment is semantically problematic, and all talk about computational tractability is taken as obsession with worst cases; for others, any attempt to go beyond retrieval in a domain-independent way is misguided, as it fails to use whatever structure is provided by the application domain. Be that as it may, in this paper we will propose a new reasoning service that does lie between the extremes mentioned.

There are many ways of specifying what a reasoning service should do. One idea that has proved quite fruitful in the last twenty years or so has been to think of the desired service in terms of a *logic of belief* [1] (Levesque 1984; Konolige 1986; Vardi 1986; Fagin & Halpern 1988; Fagin, Halpern, & Vardi 1990; Lakemeyer 1990; Cadoli & Schaerf 1992; Delgrande 1995). The idea is this: instead of considering

what the service must be like in terms of the inferences it can or must draw (given sentences $\phi_1, \ldots, \phi_n$, must it infer the sentence ψ?), we consider the *beliefs* of the system overall, and what properties the set of beliefs must satisfy. The logic of belief serves to provide a precise theoretical framework for analyzing these properties. There are sentences in this language of the form $B\phi$, saying that sentence ϕ is believed, and the semantic interpretations of the logic tell us under what conditions such a sentence will be true, and therefore what follows. Questions about the reasoning process now become questions about the *closure* properties of belief: if $B\phi_1, \ldots, B\phi_n$ are all true, does it follow in the logic that $B\psi$ is also true? We can think of the ϕ_i as the stipulated or *explicit* beliefs of the system, and the question is whether or not ψ is a derived or *implicit* belief. In a logic of belief we can also ask other sorts of questions that are difficult or impossible to formulate otherwise. For example, we can ask if the system has various forms of introspection (if $\neg B\psi$ is true, does it follow that $B\neg B\psi$ is true?) or *de re* beliefs (if the $B\phi_i$ are all true, does it follow that $\exists x B\psi$ is also true?).

Of course using a logic of belief in this way would be a lot less interesting if the reasoning service coincided exactly with classical logical entailment, that is, if for every ϕ and ψ in our representation language, $B\phi$ logically entailed $B\psi$ in the logic of belief iff ϕ classically entailed ψ. This is the case, for example, with the standard possible-world logics of belief, originated by Hintikka (Hintikka 1962; Halpern & Moses 1992), and which suffer from what Hintikka called *logical omniscience*. It would also be less interesting if the reasoning service coincided with retrieval, that is, if $B\phi$ were true iff ϕ were an element of some given list of sentences.

In between these two extremes, two broad approaches have emerged in the specification of a logic of tractable belief.[2] First, there are the *syntactic* approaches exemplified in (Konolige 1986; Vardi 1986; Fagin & Halpern 1988), where the logical interpretations either include sets of sentences (beyond the atomic ones) or mark them in some way (*e.g.* the sentences that the reasoner is aware of). In this case, a reasoner can believe ϕ and $(\phi \supset \psi)$ but fail to believe ψ because ψ is not syntactically blessed in the interpretation. Second, there are the *semantic* approaches ex-

---

[1] As is the custom, in this paper we do not distinguish between knowledge and belief.

[2] This oversimplifies the situation considerably.

emplified in (Levesque 1984; Lakemeyer 1990; Cadoli & Schaerf 1992), and deriving originally from work on tautological entailment (Anderson & Belnap 1975; Dunn 1976; Patel-Schneider 1985), where the logical interpretations assign truth values to atoms, but allow them to receive fewer or more than one. In this case, a reasoner can believe ϕ and $(\phi \supset \psi)$ but fail to believe ψ because both ϕ and $\neg\phi$ are somehow taken as true.

In this paper, we follow the tradition of the semantic approach to logics of tractable belief, but diverging from the multiple truth values and tautological entailment. Most of the criticism to date about that approach has had to do with its semantics: what is the intuitive understanding of a sentence receiving two truth values (Fagin & Halpern 1988)? Here our criticism in the next section is different: we argue that despite its apparent tractability in certain cases (Levesque 1984), a reasoning service based on tautological entailment is required to handle disjunctions in a way that does too much in some contexts and not enough in others.

In the sequel, we first revisit disjunctions, and motivate why we need to consider two possible forms of disjunctions separately within the logic. Then we present a new logic of belief, which we call the subjective logic $\mathcal{SL}$, and discuss the resulting properties of beliefs. Next, we consider the computational property of a reasoning service based on $\mathcal{SL}$ for a form of knowledge bases (KBs) with disjunctive information, the so-called proper$^+$ KBs proposed in (Lakemeyer & Levesque 2002): we show that it is tractable in the propositional case and decidable in the first-order case. Also, we give a sound and complete axiomatization for propositional $\mathcal{SL}$. Finally, we discuss related work and conclude with future work.

Disjunctions Reconsidered

As observed in (Lakemeyer & Levesque 2002), although disjunctions can be used in many ways in a commonsense KB, it has two major applications: (1) to represent *rules* such as in Horn clauses, where we may need to perform chaining in the reasoning; and (2) to represent *incomplete knowledge* about some individual(s), where we may need to split cases. We believe that (2) is the computational problem. To see why, consider the two example KBs in Figure 1. The reader

KB1	KB2
$(P(a) \vee P(e) \vee P(f))$	$(P(a) \vee Q(e) \vee Q(c))$
$(P(a) \vee P(e) \vee Q(f))$	$(Q(d) \vee P(b) \vee Q(a))$
$(P(a) \vee Q(e) \vee P(c))$	$(P(a) \vee P(e) \vee P(f))$
$(P(a) \vee Q(e) \vee Q(c))$	$(P(c) \vee Q(e) \vee P(a))$
$(Q(a) \vee P(b) \vee P(d))$	$(Q(a) \vee Q(b) \vee Q(g))$
$(Q(a) \vee P(b) \vee Q(c))$	$(P(a) \vee P(e) \vee Q(f))$
$(Q(a) \vee Q(b) \vee P(g))$	$(Q(b) \vee Q(a) \vee P(g))$
$(Q(a) \vee Q(b) \vee Q(g))$	$(Q(a) \vee P(d) \vee P(b))$

Figure 1: Two puzzles

is invited to confirm that one and only one of these logically entails $\exists x.(P(x) \wedge Q(x))$. So being required to handle (2) is also being required to solve combinatorial puzzles like this

automatically as part of the basic operation of the system. If we accept that this is perhaps asking too much of a reasoning service, then we need to rule out a service based simply on logical entailment, since KB2 does logically entail $\exists x.(P(x) \wedge Q(x))$. In fact, we also need to rule out *tautological entailment*, since KB2 also tautologically entails the sentence. Indeed, tautological entailment agrees with logical entailment in the absence of negation. So restricting a reasoning service to tautological entailment would still require it to be able to determine that $\exists x.(P(x) \wedge Q(x))$ is true for KB2 but not KB1.

It is significant that previous work proposing a limited form of reasoning based on tautological entailment (Levesque 1984; Frisch 1987) only worked in the propositional case and when the query was in CNF or the KB was in DNF (Cadoli & Schaerf 1996). The first-order case later studied in (Patel-Schneider 1985) and (Lakemeyer 1990) required considerable machinery beyond tautological entailment. Moreover, this additional effort only resulted in fewer inferences compared to tautological entailment and thus perhaps even less general applicability.

Our approach here will be to follow (Lakemeyer & Levesque 2002) and preserve (1), but to deal with (2) in a more controlled way. To handle (1) without also reasoning by cases, we will propose a logic of belief where clauses that are explicitly believed are closed under *unit propagation*. This means that disjunctions that express simple rules as material conditionals will be fully utilized. Because unit propagation does not result in an explosion of clauses, we believe that this very common form of reasoning can also be kept tractable. Other approaches based on unit propagation include (McAllester 1990; Dalal 1996; Crawford & Etherington 1998), but they are all restricted to the propositional case.

As for (2), we *do* want systems that can split cases and deal with incomplete knowledge that is disjunctive, but we need to do so in a controlled way. In fact what we will propose is a logic with a family of belief operators B_0, B_1, B_2, ..., where the difference concerns how much case splitting is tolerated in deriving implicit beliefs. For the above puzzle with KB2, it will turn out that $B_8 \exists x.(P(x) \wedge Q(x))$ will be true but $B_7 \exists x.(P(x) \wedge Q(x))$ will be false. Of course, the higher the level k, the more resources are required to determine what is an implicit belief at that level. Each of these belief operators will be closed under various forms of obvious reasoning. For example, if we believe ϕ at some level, we also believe $(\phi \vee \psi)$ at the same level, *i.e.* weakening. In addition, the beliefs at level 0 will be closed under unit propagation.

Some bad news: One tried-and-true and well-loved form of reasoning that we will need to give up on is the distribution of $\wedge$ over $\vee$, that is, that believing $(p \wedge (q \vee r))$ should always imply believing $(p \wedge q) \vee (p \wedge r)$. We can get this behavior by going up to higher levels (*e.g.* splitting the second clause here), but to require it at every level would force us to do too much reasoning. For example, after repeatedly distributing $\wedge$ over $\vee$ in KB2 above, it then becomes obvious that $\exists x.(P(x) \wedge Q(x))$ must be true.

The Subjective Logic $\mathcal{SL}$

The syntax

The language $\mathcal{L}$ is a standard first-order logic with equality. The language $\mathcal{SL}$ is a first-order logic with equality whose atomic formulas are belief atoms of the form $\boldsymbol{B}_k\phi$ where ϕ is a formula of the language $\mathcal{L}$ and $\boldsymbol{B}_k$ is a modal operator for any $k \geq 0$. $\boldsymbol{B}_k\phi$ is read as "ϕ is a belief at level k". We call $\mathcal{SL}$ a *subjective* logic because all predicates other than equality appear in the scope of a belief operator.

More precisely, we have the following inductive definitions. We have countably infinite sets of variables and constant symbols, which make up the *terms* of the language. The *atoms* are expressions of the form $P(t_1, \ldots, t_m)$ where P is a predicate symbol (excluding equality) and the t_i are terms. The *literals* are atoms or their negations. We use ρ to range over literals, and we use $\overline{\rho}$ to denote the complement of ρ.

The language $\mathcal{L}$ is the least set of expressions such that

1. if ρ is an atom, then $\rho \in \mathcal{L}$;

2. if t and t' are terms, then $(t = t') \in \mathcal{L}$;

3. if $\phi, \psi \in \mathcal{L}$ and x is a variable, then $\neg\phi$, $(\phi \vee \psi)$, and $\exists x.\phi \in \mathcal{L}$.

Clauses, which play an important role in our semantic definition, are inductively defined as follows:

1. a literal is a clause, and is called a unit clause;

2. if c and c' are clauses, then $(c \vee c')$ is a clause.

We identify a clause with the set of literals it contains. Only non-empty clauses appear in $\mathcal{L}$. The empty clause, however, which we denote by $\square$, can appear in $\mathcal{SL}$ and is needed in the definition of UP to follow.

The language $\mathcal{SL}$ is the least set of expressions such that

1. if $\phi \in \mathcal{L}$ or ϕ is $\square$, and $k \geq 0$, then $\boldsymbol{B}_k\phi \in \mathcal{SL}$, and is called a *belief atom* of level k;

2. if t and t' are terms, then $(t = t') \in \mathcal{SL}$;

3. if $\alpha, \beta \in \mathcal{SL}$ and x is a variable, then $\neg\alpha$, $(\alpha \vee \beta)$, and $\exists x.\alpha \in \mathcal{SL}$.

So, in short, the formulas of $\mathcal{SL}$ are such that all predicates other than equality must occur within a modal operator and the modalities are non-nested. As usual, we use $(\alpha \wedge \beta)$, $(\alpha \supset \beta)$, and $\forall x.\alpha$ as abbreviations. We write α^x_d to denote α with all free occurrences of x replaced by constant d.

The semantics

Sentences of $\mathcal{SL}$ are interpreted via a *setup*, which is a set of non-empty *ground clauses*, and which specifies which sentences of $\mathcal{L}$ are believed, and consequently which sentences of $\mathcal{SL}$ are true. Intuitively, one may think of a setup as indicating what is *explicitly* believed as a possibly infinite set of ground clauses. The semantics below then tells us the *implicit* beliefs that follow. We begin with some preparatory concepts.

Let s be a set of ground clauses. The closure of s under *unit propagation*, denoted by $\mathsf{UP}(s)$, is the least set s' satisfying: 1. $s \subseteq s'$; and 2. if $\rho \in s'$ and $\{\overline{\rho}\} \cup c \in s'$, then

$c \in s'$. We define $\mathsf{VP}(s)$ as the set $\{c \mid c$ is a ground clause and there exists $c' \in \mathsf{UP}(s)$ such that $c' \subseteq c\}$.

Next, observe that in classical logic we have the following patterns of obvious inference:

1. from ϕ, infer $\neg\neg\phi$;

2. from ϕ or ψ, infer $(\phi \vee \psi)$;

3. from ϕ and ψ, infer $(\phi \wedge \psi)$.

These patterns relate the inference of a formula to that of its subformulas. As a characterization of these patterns of obvious inference, we define the concept of belief reduction. Roughly, $(\boldsymbol{B}_k\phi)\downarrow$ denotes the $\mathcal{SL}$ formula resulting from pushing the belief operator into ϕ. Intuitively, we take the conclusion from $(\boldsymbol{B}_k\phi)\downarrow$ to $\boldsymbol{B}_k\phi$ to be an obvious one. For any $\phi \in \mathcal{L}$, the $\mathcal{SL}$ formula $(\boldsymbol{B}_k\phi)\downarrow$ is defined as follows:

1. $(\boldsymbol{B}_k c)\downarrow = \boldsymbol{B}_k c$, where c is a clause;

2. $(\boldsymbol{B}_k(t = t'))\downarrow = (t = t')$;

3. $(\boldsymbol{B}_k\neg(t = t'))\downarrow = \neg(t = t')$;

4. $(\boldsymbol{B}_k\neg\neg\phi)\downarrow = \boldsymbol{B}_k\phi$;

5. $(\boldsymbol{B}_k(\phi \vee \psi))\downarrow = (\boldsymbol{B}_k\phi \vee \boldsymbol{B}_k\psi)$, where ϕ or ψ is not a clause;

6. $(\boldsymbol{B}_k\neg(\phi \vee \psi))\downarrow = (\boldsymbol{B}_k\neg\phi \wedge \boldsymbol{B}_k\neg\psi)$;

7. $(\boldsymbol{B}_k\exists x.\phi)\downarrow = \exists x.\boldsymbol{B}_k\phi$;

8. $(\boldsymbol{B}_k\neg\exists x.\phi)\downarrow = \forall x.\boldsymbol{B}_k\neg\phi$.

In logic, we usually define concepts and prove properties about formulas by induction on the structure of formulas. The principle can be stated as follows. We first define a complexity measure $\|\cdot\|$ which maps formulas into natural numbers. Usually, the complexity measure is the length of the formula or the number of logical operators in the formula. Now let α be an arbitrary formula. Assuming that we have defined a concept C or proved a property P for all formulas β such that $\|\beta\| < \|\alpha\|$, we proceed to define C or prove P for α. In $\mathcal{SL}$, the complexity measure is more complicated, because we need to take into account both the length and the level of belief atoms. For example, we would like $\|\boldsymbol{B}_2\phi\| < \|\boldsymbol{B}_3\phi\|$. For any $\alpha \in \mathcal{SL}$, $\|\alpha\|$ is defined as follows:

1. $\|(t = t')\| = 1$;

2. $\|\neg\alpha\| = 1 + \|\alpha\|$;

3. $\|\exists x.\alpha\| = 3 + \|\alpha\|$;

4. $\|(\alpha \vee \beta)\| = 3 + \|\alpha\| + \|\beta\|$;

5. $\|\boldsymbol{B}_k\phi\| = 2^{k+m}$, where m is the length of ϕ, but where all atoms and equalities are considered to have length 1.

It is easy to prove the following property about $\|\cdot\|$:

Proposition 1 *1. For any ϕ, $\|\boldsymbol{B}_k\phi\| < \|\boldsymbol{B}_{k+1}\phi\|$;*

2. For any ϕ that is not a clause, $\|(\boldsymbol{B}_k\phi)\downarrow\| < \|\boldsymbol{B}_k\phi\|$.

Now we are ready to define truth in $\mathcal{SL}$. Let s be a setup. Then for any sentence $\alpha \in \mathcal{SL}$, $s \models \alpha$ (read "s satisfies α") is defined inductively on $\|\alpha\|$ as follows:

1. $s \models (d = d')$ iff d and d' are the same constant;

2. $s \models \neg\alpha$ iff $s \not\models \alpha$;

3. $s \models \alpha \vee \beta$ iff $s \models \alpha$ or $s \models \beta$;

4. $s \models \exists x.\alpha$ iff for some constant d, $s \models \alpha_d^x$;

5. $s \models \boldsymbol{B}_k \phi$ iff one of the following holds:

  (a) *subsume*: $k = 0$, ϕ is a clause c, and $c \in \mathsf{VP}(s)$;

  (b) *reduce*: ϕ is not a clause and $s \models (\boldsymbol{B}_k \phi)\!\downarrow$;

  (c) *split*: $k > 0$ and there is some $c \in s$ such that for all $\rho \in c$, $s \cup \{\rho\} \models \boldsymbol{B}_{k-1}\phi$.

By the above proposition, this semantics is well-defined. As usual, we say that a sentence $\alpha \in \mathcal{SL}$ is *valid* ($\models \alpha$) if for every setup s, we have that $s \models \alpha$.

Before discussing properties of the logic as a whole, we observe that the semantics above proposes three different justifications for believing a sentence ϕ (at level k):

1. ϕ is a clause, $k = 0$, and after doing unit propagation on the ground clauses that are explicitly believed, we end up with a subclause of ϕ;

2. we already have appropriate beliefs about the subformulas of ϕ, for example, believing both conjuncts of a conjunction, or some instance of an existential;

3. there is a clause in our explicit beliefs that if we were to split, that is, if we were to augment our beliefs by a literal in that clause, then in all cases we would end up believing ϕ at level $k - 1$.

Note that all three of these rules deal with disjunction but in quite different ways.

The reader should note the assumptions made with respect to the universe of discourse and, as a result, the treatment of equality. For one, all setups use the same universe of discourse, which is identical to the infinite set of constants in the language. Moreover, distinct constants stand for distinct individuals, which fixes the meaning of the equality predicate. All this allows giving quantifiers a substitutional interpretation and, previous criticism of substitutional interpretations notwithstanding (Kripke 1976), greatly simplifies the technical treatment.

Monotonicity of beliefs

We now prove the monotonicity of beliefs, that is, that new clauses can be added to any setup without revoking previously supported beliefs. This is a basic property used throughout the paper.

Let s and s' be two setups. We write $s \preceq s'$ iff for any $\boldsymbol{B}_k \phi$, if $s \models \boldsymbol{B}_k \phi$, then $s' \models \boldsymbol{B}_k \phi$.

Proposition 2 *For any $c \in \mathsf{UP}(s)$, there exists $c' \in s$ such that $c \subseteq c'$ and for all $\rho \in c' - c$, $\overline{\rho} \in \mathsf{UP}(s)$.*

Proposition 3 *If $s \subseteq \mathsf{VP}(s')$, then $\mathsf{VP}(s) \subseteq \mathsf{VP}(s')$.*

Proposition 4 *[Monotonicity]*
If $\mathsf{VP}(s) \subseteq \mathsf{VP}(s')$, then $s \preceq s'$.

Proof: We prove by induction on $\|\boldsymbol{B}_k \phi\|$.

1. $s \models_{B_0} c$ by subsumption. Since $\mathsf{VP}(s) \subseteq \mathsf{VP}(s')$, $s' \models_{B_0} c$.

2. $s \models B_k \phi$ by reduction. For each case of ϕ, it is easy to prove by induction that $s' \models B_k \phi$ too.

3. $s \models \boldsymbol{B}_k \phi$ by splitting on $c \in s$. Then for all $\rho \in c$, $s \cup \{\rho\} \models \boldsymbol{B}_{k-1}\phi$. Since $c \in \mathsf{VP}(s')$, by Proposition 2, there exist $c' \subseteq c$ and $c'' \in s'$ such that $c' \subseteq c''$ and for all $\rho \in c'' - c'$, $\overline{\rho} \in \mathsf{UP}(s')$. We prove that for all $\rho \in c''$, $s' \cup \{\rho\} \models \boldsymbol{B}_{k-1}\phi$, and hence $s' \models \boldsymbol{B}_k \phi$.

  (a) $\rho \in c'' - c'$. Then $\overline{\rho} \in \mathsf{UP}(s')$, and so $\square \in \mathsf{UP}(s' \cup \{\rho\})$. Pick any $\rho' \in c$, then $\mathsf{VP}(s \cup \{\rho'\}) \subseteq \mathsf{VP}(s' \cup \{\rho\})$ and $s \cup \{\rho'\} \models \boldsymbol{B}_{k-1}\phi$. By induction, $s' \cup \{\rho\} \models \boldsymbol{B}_{k-1}\phi$.

  (b) $\rho \in c'$. Then $\rho \in c$. Thus $s \cup \{\rho\} \models \boldsymbol{B}_{k-1}\phi$. Since $\mathsf{VP}(s \cup \{\rho\}) \subseteq \mathsf{VP}(s' \cup \{\rho\})$, by induction, we get that $s' \cup \{\rho\} \models \boldsymbol{B}_{k-1}\phi$. ∎

As an easy corollary, if $s \subseteq s'$, then $s \preceq s'$.

Properties of beliefs

We now consider the properties of beliefs, both at the same and across different levels. What interests us most are questions like when does a belief at a certain level entail another belief and when is this not the case. We will see that many properties agree with those which one finds in classical approaches to modeling belief such as possible-world semantics (Kripke 1959; Hintikka 1962). But there will also be a number of differences, which sets our model apart from existing approaches. We only include a few proofs.

Equality: Due to our treatment of equality, we have that at all levels, exactly the true equality sentences are believed:

$$\models \boldsymbol{B}_k e \equiv e, \tag{1}$$

where e contains no predicate symbols.

Belief Reductions: Obviously, we have

$$\models (\boldsymbol{B}_k \phi)\!\downarrow \supset \boldsymbol{B}_k \phi \tag{2}$$
$$\models \boldsymbol{B}_0 \phi \equiv (\boldsymbol{B}_0 \phi)\!\downarrow \tag{3}$$

Also, we have

$$\models \boldsymbol{B}_k \neg\neg\phi \equiv \boldsymbol{B}_k \phi \tag{4}$$
$$\models \boldsymbol{B}_k (\phi \wedge \psi) \equiv \boldsymbol{B}_k \phi \wedge \boldsymbol{B}_k \psi \tag{5}$$
$$\models \boldsymbol{B}_k \forall x.\phi \equiv \forall x.\boldsymbol{B}_k \phi \tag{6}$$

Proof: Since the proofs are all very similar, we only prove (4) here. It suffices to prove that $\models \boldsymbol{B}_k \neg\neg\phi \supset \boldsymbol{B}_k \phi$, since the other direction follows from (2). We prove this by induction on k. Basis: $k = 0$. Trivial. Induction step: Let $s \models \boldsymbol{B}_{k+1} \neg\neg\phi$. If this holds by reduction, then $s \models \boldsymbol{B}_{k+1}\phi$. Otherwise, there is some $c \in s$ such that for all $\rho \in c$, $s \cup \{\rho\} \models \boldsymbol{B}_k \neg\neg\phi$. By induction, $s \cup \{\rho\} \models \boldsymbol{B}_k \phi$. Thus $s \models \boldsymbol{B}_{k+1}\phi$. ∎

However, we have

$$\not\models \boldsymbol{B}_k (\phi \vee \psi) \supset \boldsymbol{B}_k \phi \vee \boldsymbol{B}_k \psi \tag{7}$$
$$\not\models \boldsymbol{B}_k \exists x.\phi \supset \exists x.\boldsymbol{B}_k \phi, \text{ for } k > 0 \tag{8}$$

We give two counter-examples for (7). Let $s_1 = \{(p \vee q)\}$. Then $s_1 \models \boldsymbol{B}_0 (p \vee q)$, but $s_1 \not\models \boldsymbol{B}_0 p$ and $s_1 \not\models \boldsymbol{B}_0 q$. Let $s_2 = \{(x \vee y), (\overline{x} \vee p), (\overline{y} \vee q)\}$. Then $s_2 \models \boldsymbol{B}_1 (p \vee q)$, but $s_2 \not\models \boldsymbol{B}_1 p$ and $s_2 \not\models \boldsymbol{B}_1 q$.

Distribution: Unfortunately, only one direction of each of the normal distribution laws goes through, as shown in the

following:

$$\models \boldsymbol{B}_k[(\phi \wedge \psi) \vee (\phi \wedge \eta)] \supset \boldsymbol{B}_k[\phi \wedge (\psi \vee \eta)] \qquad (9)$$

$$\not\models \boldsymbol{B}_1[p \wedge (q \vee r)] \supset \boldsymbol{B}_1[(p \wedge q) \vee (p \wedge r)] \qquad (10)$$

$$\models \boldsymbol{B}_k[\phi \vee (\psi \wedge \eta)] \supset \boldsymbol{B}_k[(\phi \vee \psi) \wedge (\phi \vee \eta)] \qquad (11)$$

$$\not\models \boldsymbol{B}_1[(p \vee q) \wedge (p \vee r)] \supset \boldsymbol{B}_1[p \vee (q \wedge r)] \qquad (12)$$

(10) and (12) hold for the same reason as the failure of Modus Ponens in (15) discussed below.

Thus normal form conversions generally do not preserve equivalence for beliefs at a fixed level k. Those who may find this troubling should recall our previous discussion where we pointed out that it is the distribution of $\wedge$ over $\vee$ (and not, say, closure under resolution) which would force us into solving puzzles like those in Figure 1, since no negations are involved.

Level Change: As expected, we have the following:

$$\models \boldsymbol{B}_k\phi \supset \boldsymbol{B}_{k+1}\phi \qquad (13)$$

Proof: Let s_0 be the empty setup. It is easy to see that $s_0 \not\models \boldsymbol{B}_k c$ for any k and clause c. Also, $s_0 \models \boldsymbol{B}_k e$ iff $s_0 \models e$ for any k and equality or inequality e. Thus $s_0 \models \boldsymbol{B}_k\phi$ iff $s_0 \models \boldsymbol{B}_{k+1}\phi$ for any ϕ.

Now let $s \models \boldsymbol{B}_k\phi$. If s is empty, then $s \models \boldsymbol{B}_{k+1}\phi$. Otherwise, pick any $c \in s$. By monotonicity, for all $\rho \in c$, $s \cup \{\rho\} \models \boldsymbol{B}_k\phi$. Thus $s \models \boldsymbol{B}_{k+1}\phi$. ∎

Modus Ponens: Finally, we consider the closure of beliefs under Modus Ponens. As expected, $\boldsymbol{B}_0$-beliefs are closed under unit propagation, while $\boldsymbol{B}_k$-beliefs are not for $k > 0$. However, we do have a generalized form of closure under unit propagation. Let ρ be a literal and c a clause. Then

$$\models \boldsymbol{B}_0\rho \wedge \boldsymbol{B}_0(\overline{\rho} \vee c) \supset \boldsymbol{B}_0 c \qquad (14)$$

$$\not\models \boldsymbol{B}_1 p \wedge \boldsymbol{B}_1(\overline{p} \vee q)] \supset \boldsymbol{B}_1 q \qquad (15)$$

$$\models \boldsymbol{B}_i\rho \wedge \boldsymbol{B}_j(\overline{\rho} \vee c) \supset \boldsymbol{B}_{i+j} c \qquad (16)$$

Proof: (15): Intuitively, this is because you may need one split for p and another for $(p \supset q)$, but one split may not get you q. To see why, let $s = \{(x \vee p),\ (\overline{x} \vee p),\ (y \vee \overline{p} \vee q),\ (\overline{y} \vee \overline{p} \vee q)\}$. Then $s \models \boldsymbol{B}_1 p$ by splitting on the first clause, and $s \models \boldsymbol{B}_1(p \supset q)$ by splitting on the third clause. But $s \not\models \boldsymbol{B}_1 q$.

(16): The proof is by induction on $i + j$. Basis: $i + j = 0$. This is simply (14). Induction step: $i + j > 0$. Suppose that $i > 0$. By induction, $\models \boldsymbol{B}_{i-1}\rho \wedge \boldsymbol{B}_j(\overline{\rho} \vee c) \supset \boldsymbol{B}_{i+j-1} c$. Now let $s \models \boldsymbol{B}_i\rho \wedge \boldsymbol{B}_j(\overline{\rho} \vee c)$. Then there is some $c \in s$ such that for all $l \in c$, $s \cup \{l\} \models \boldsymbol{B}_{i-1}\rho$. By monotonicity, $s \cup \{l\} \models \boldsymbol{B}_j(\overline{\rho} \vee c)$. Hence $s \cup \{l\} \models \boldsymbol{B}_{i+j-1} c$. Therefore $s \models \boldsymbol{B}_{i+j} c$. The case when $j > 0$ is similar. ∎

(15) shows that Modus Ponens is not a valid form of inference at a fixed level k. However, we do get a generalized form of Modus Ponens under a certain condition. In what follows, let $i, j \geq 0$, and let $\phi, \psi \in \mathcal{L}$ such that ψ does not contain equalities. Then we have

$$\models \boldsymbol{B}_i\phi \wedge \boldsymbol{B}_j(\phi \supset \psi) \supset \boldsymbol{B}_k\psi,\ \text{for some } k \qquad (17)$$

The proof needs the following:

$$\models \boldsymbol{B}_k\square \supset \boldsymbol{B}_k\psi \qquad (18)$$

$$\models \boldsymbol{B}_i\phi \wedge \boldsymbol{B}_j\neg\phi \supset \boldsymbol{B}_k\square,\ \text{for some } k \qquad (19)$$

Proof: (18): The proof is by induction on k. The base case is proved by induction on ψ. Note that ψ does not contain equalities.

(19): The proof is by induction on $\| \boldsymbol{B}_i\phi \| + \| \boldsymbol{B}_j\neg\phi \|$. Since $\models \boldsymbol{B}_j\neg\neg\phi \equiv \boldsymbol{B}_j\phi$, we only need to consider the cases when ϕ is a clause, an equality, a double negation, a disjunction, or an existential. Here we only prove the cases when ϕ is a clause or a disjunction. Other cases are either trivial or can be similarly proved. Case 1: ϕ is a clause c. Let n be the number of literals in c. Since $\models \boldsymbol{B}_j\neg c \equiv \bigwedge_{\rho \in c} \boldsymbol{B}_j\overline{\rho}$, by repeatedly applying (16), we have $\models \boldsymbol{B}_i c \wedge \boldsymbol{B}_j\neg c \supset \boldsymbol{B}_{i+nj}\square$.

Case 2: ϕ is $\phi_1 \vee \phi_2$ such that ϕ_1 or ϕ_2 is not a clause. By induction, there exist k_1, k_2, k_3, and k_4 such that

$$\models \boldsymbol{B}_i\phi_h \wedge \boldsymbol{B}_j\neg\phi_h \supset \boldsymbol{B}_{k_h}\square, h = 1, 2$$

$$\models \boldsymbol{B}_{i-1}\phi \wedge \boldsymbol{B}_j\neg\phi \supset \boldsymbol{B}_{k_3}\square,\ \text{if } i > 0$$

$$\models \boldsymbol{B}_i\phi \wedge \boldsymbol{B}_{j-1}\neg\phi \supset \boldsymbol{B}_{k_4}\square,\ \text{if } j > 0$$

If $i = 0$, let $k_3 = -1$; if $j = 0$, let $k_4 = -1$. We let $k = \max\{k_1, k_2, k_3 + 1, k_4 + 1\}$. Now let $s \models \boldsymbol{B}_i\phi \wedge \boldsymbol{B}_j\neg\phi$. If $s \models \boldsymbol{B}_i\phi$ by splitting, then $s \models \boldsymbol{B}_{k_3+1}\square$. If $s \models \boldsymbol{B}_j\neg\phi$ by splitting, then $s \models \boldsymbol{B}_{k_4+1}\square$. Otherwise, we have that $s \models \boldsymbol{B}_i\phi_h \wedge \boldsymbol{B}_j\neg\phi_h$ for some $h = 1, 2$. So $s \models \boldsymbol{B}_{k_h}\square$. ∎

Now we can prove (17).

Proof: The proof is by induction on j. There are two cases. Case 1: $\neg\phi \vee \psi$ is a clause. Then ϕ is an atom, say ρ; and ψ is a clause, say c. By (16), $\models \boldsymbol{B}_i\rho \wedge \boldsymbol{B}_j(\overline{\rho} \vee c) \supset \boldsymbol{B}_{i+j} c$. Case 2: $\neg\phi \vee \psi$ is not a clause. By (19), there exists a k_1 such that $\models \boldsymbol{B}_i\phi \wedge \boldsymbol{B}_j\neg\phi \supset \boldsymbol{B}_{k_1}\square$. By (18), $\models \boldsymbol{B}_i\phi \wedge \boldsymbol{B}_j\neg\phi \supset \boldsymbol{B}_{k_1}\psi$. If $j = 0$, let $k_2 = -1$; otherwise, by induction, there exists a k_2 such that $\models \boldsymbol{B}_i\phi \wedge \boldsymbol{B}_{j-1}(\phi \supset \psi) \supset \boldsymbol{B}_{k_2}\psi$. Then $k = \max\{j, k_1, k_2 + 1\}$ is the value we want. ∎

A Reasoning Service Based on $\mathcal{SL}$

As we mentioned in the introduction, $\mathcal{SL}$ is intended to serve as a foundation for limited but decidable (and even tractable) reasoning services. The idea is to model the reasoning service as belief implication, *i.e.* validity of formulas of the form $(\boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi)$, where *KB* is a knowledge base, and ϕ is a query. More precisely, we have

Definition 1 The *query evaluation problem* based on $\mathcal{SL}$ for a fixed value k (the QESL problem in short) is as follows: Given a knowledge base KB in $\mathcal{L}$ and a formula ϕ in $\mathcal{L}$, decide whether the $\mathcal{SL}$ formula $(\boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi)$ is valid.

Intuitively, if a KB is thought as providing the explicit beliefs of the system, formulated not as a possibly infinite set of ground clauses, but as a finite set of sentences of $\mathcal{L}$ using quantification, then the implicit beliefs at level k are those sentences ϕ such that $(\boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi)$ is valid.

Example 1 Consider KB1 and KB2 in Figure 1, and the query $\phi = \exists x.(P(x) \wedge Q(x))$. Then we have:

1. $\not\models (\boldsymbol{B}_0 KB1 \supset \boldsymbol{B}_k\phi)$, for any k;
2. $\not\models (\boldsymbol{B}_0 KB2 \supset \boldsymbol{B}_k\phi)$, for any $k < 8$;
3. $\models (\boldsymbol{B}_0 KB2 \supset \boldsymbol{B}_k\phi)$, for every $k \geq 8$.

Example 2 Consider the following KB with only one predicate $C(p_1, p_2)$ saying that the two persons are compatible.

1. $\forall x \forall y.C(x,y) \supset C(y,x)$;
2. $\forall x.C(x,ann) \vee C(x,bob)$;
3. $\neg C(bob,fred)$;
4. $C(carol,eve) \vee C(carol,fred)$;
5. $\forall x.x \neq bob \wedge x \neq carol \supset C(dan,x)$;
6. $\neg C(eve,ann) \vee \neg C(eve,fred)$.

We have the following queries:

1. $\phi_1 = C(fred,ann)$;
2. $\phi_2 = \forall x \exists y C(x,y)$;
3. $\phi_3 = \exists x \exists y \exists z[C(x,y) \wedge C(x,z) \wedge \neg C(y,z)]$;
4. $\phi_4 = \exists x \exists y[x \neq y \wedge C(x,carol) \wedge C(y,carol)]$.

Then we have:

1. $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0 \phi_1$,
   since $C(fred,ann)$ can be obtained by unit propagation from $\neg C(bob,fred)$, $\neg C(fred,bob) \vee C(bob,fred)$, and $C(fred,ann) \vee C(fred,bob)$.

2. $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_1 \phi_2$,
   since for each constant d, we obtain $\exists y C(d,y)$ by case analysis over $C(d,ann) \vee C(d,bob)$.

3. $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_1 \phi_3$,
   since we have $C(dan,fred)$, $C(dan,ann)$, and $C(dan,eve)$, hence we obtain ϕ_3 by case analysis over $\neg C(eve,ann) \vee \neg C(eve,fred)$.

4. $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_2 \phi_4$, but $\not\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_1 \phi_4$,
   since we obtain ϕ_4 by case analysis over $C(carol,ann) \vee C(carol,bob)$ and $C(carol,eve) \vee C(carol,fred)$;
   but we cannot get ϕ_4 by one case analysis only.

We have given informal explanations here. Formal proofs can be obtained by resorting to Theorem 5 below.

Logical correctness

A basic concern of a reasoning service is its logical correctness, that is, just how closely it aligns with classical logical entailment. We now show that query evaluation based on $\mathcal{SL}$ is classically sound, that is, if $(\boldsymbol{B}_0 KB \supset \boldsymbol{B}_k \phi)$ is valid, then $\mathcal{E} \cup KB$ classically entails ϕ, where $\mathcal{E}$ consists of the axioms of equality and the infinite set $\{(d \neq d') \mid d \text{ and } d' \text{ are distinct constants}\}$.

As we have noted earlier, it is part of the semantics of $\mathcal{SL}$ that the domain of discourse is essentially the set of constants and equality is identity. Levesque (1998) calls first-order interpretations *standard* if they make the same assumption. As the following theorem shows, the restriction to standard interpretations can be captured by $\mathcal{E}$.

Theorem *[from (Levesque 1998)]*
Suppose S is any set of closed wffs, and that there is an infinite set of constants that do not appear in S. Then $\mathcal{E} \cup S$ is satisfiable iff it has a standard model.

Obviously, a standard interpretation can be represented as an (infinite) set s of ground literals such that for each ground atom a, exactly one of a and $\neg a$ is in s. We use $\models_{\text{FOL}}$ to denote the support and entailment relations in classical first-order logic.

Lemma 1 *Let s be a standard interpretation. Then $s \models_{\text{FOL}} \phi$ iff $s \models \boldsymbol{B}_k \phi$.*

Proof: It is easy to prove by induction that $s \models_{\text{FOL}} \phi$ iff $s \models \boldsymbol{B}_0 \phi$. It is also easy to prove by induction that when s is a set of literals, $s \models \boldsymbol{B}_{k+1} \phi$ iff $s \models \boldsymbol{B}_k \phi$. ■

Theorem 1 *If $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k \phi$, then $\mathcal{E} \cup KB \models_{\text{FOL}} \phi$.*

Proof: By the above theorem, it suffices to prove that every standard model s of KB is also a model of ϕ. Since we have that $s \models_{\text{FOL}} KB$, by lemma 1, $s \models \boldsymbol{B}_0 KB$, and therefore $s \models \boldsymbol{B}_k \phi$. Again, by lemma 1, $s \models_{\text{FOL}} \phi$. ■

We now consider the issue of classical completeness of query evaluation based on $\mathcal{SL}$. Of course, in general, this reasoning is classically incomplete, which is necessary for the sake of tractability. But there do exist a few simple cases where it is classically complete.

In previous work, Levesque (1998) proposed a generalization of databases called *proper KBs*, which allow a limited form of incomplete knowledge, equivalent to a consistent set of ground literals. The classical entailment problem for proper KBs is not decidable. So Levesque proposed a sound but incomplete reasoning procedure V for proper KBs which was classically complete for queries in a normal form called $\mathcal{NF}$. On the other hand, the expressiveness of proper KBs is still quite limited. So Lakemeyer and Levesque (2002) proposed an extension to proper KBs called proper[+] KBs, which allow simple forms of disjunctive information. We now define these precisely.

In what follows, we use θ to range over substitutions of all variables by constants, and write $\alpha\theta$ as the result of applying the substitution to α. We use $\forall\alpha$ to mean the universal closure of α. We let e range over ewffs, *i.e.* quantifier-free formulas containing no predicate symbols.

Definition 2 Let e be an ewff and c a clause. Then a formula of the form $\forall(e \supset c)$ is called a $\forall$-clause. A *KB* is called *proper[+]* if it is a finite non-empty set of $\forall$-clauses. Given a proper[+] *KB*, we define $\text{gnd}(KB)$ as the infinite setup $\{c\theta \mid \forall(e \supset c) \in KB \text{ and } \models e\theta\}$. A *KB* is called *proper* if it is proper[+] and $\text{gnd}(KB)$ is a consistent set of ground literals.

Our first result is that reasoning based on $\mathcal{SL}$ is classically complete for proper KBs when the query is in $\mathcal{NF}$:

Theorem 2 *Let KB be proper, and let $\phi \in \mathcal{NF}$. If $\mathcal{E} \cup KB \models_{\text{FOL}} \phi$, then $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0 \phi$.*

Proof: By Levesque's result that V is complete for queries in $\mathcal{NF}$, if $\mathcal{E} \cup KB \models_{\text{FOL}} \phi$, then $V[\phi] = 1$. By Corollary 1 below, we have that $V[\phi] = 1$ iff $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0 \phi$. ■

In the propositional case, when the KB is proper[+] and the query is again in $\mathcal{NF}$, we get a form of "eventual completeness", which is to say that for each query that is a logical entailment, there is a k for which the query is an implicit belief at level k:

Theorem 3 *In the propositional case, if KB is proper[+], $\phi \in \mathcal{NF}$, and KB $\models_{\text{FOL}} \phi$, then there exists a k such that $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k \phi$.*

Proof: Let k be the number of non-unit clauses in *KB*. We prove that $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k \phi$. By Theorem 5 below, it is equivalent to proving that $\mathrm{gnd}(KB) \models \boldsymbol{B}_k \phi$. Note that $\mathrm{gnd}(KB)$ is *KB* itself. We prove by induction on k. Basis: $k = 0$. Then *KB* is proper. By Theorem 2, $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0 \phi$. Induction step: $k > 0$. Then there exists a non-unit clause c in *KB*. Since $KB \models_{\mathrm{FOL}} \phi$, we have that $KB - \{c\} \cup \{\rho\} \models_{\mathrm{FOL}} \phi$ for all $\rho \in c$. By induction, $\mathrm{gnd}(KB - \{c\} \cup \{\rho\}) \models \boldsymbol{B}_{k-1} \phi$. Thus $\mathrm{gnd}(KB) \models \boldsymbol{B}_k \phi$. ∎

Computing implicit beliefs

The other important concern of a reasoning service is its computational property. In this section, we show that for proper[+] KBs, query evaluation based on $\mathcal{SL}$ is tractable in the propositional case and decidable in the first-order case.

We begin by considering the simple case of proper KBs. We show that Levesque's reasoning procedure V is actually a decision procedure for the QESL problem over proper KBs. This results from an observation that relates $\mathcal{SL}$ to tautological entailment. Here is the definition of tautological entailment for standard interpretations from (Lakemeyer & Levesque 2002).

A literal setup is a set of ground literals. The support relation $\models^t$ between literal setups and sentences is defined as follows:

1. $s \models^t l$ iff $l \in s$, where l is a literal;
2. $s \models^t (t = t')$ iff t is identical to t';
3. $s \models^t \neg(t = t')$ iff t is not identical to t';
4. $s \models^t \neg\neg\phi$ iff $s \models^t \phi$;
5. $s \models^t (\phi \vee \psi)$ iff $s \models^t \phi$ or $s \models^t \psi$;
6. $s \models^t \neg(\phi \vee \psi)$ iff $s \models^t \neg\phi$ and $s \models^t \neg\psi$;
7. $s \models^t \exists x.\phi$ iff $s \models^t \phi_d^x$ for some constant d;
8. $s \models^t \neg\exists x.\phi$ iff $s \models^t \neg\phi_d^x$ for all constant d.

A set of sentences Σ tautologically entails a sentence ϕ ($\Sigma \longrightarrow \phi$) iff for all literal setup s, if $s \models^t \psi$ for all $\psi \in \Sigma$, then $s \models^t \phi$.

Lemma 2 *Let s be a consistent literal setup.*
Then $s \models^t \phi$ iff $s \models \boldsymbol{B}_0 \phi$.

Proof: Easy by induction. Note that $\mathsf{UP}(s)$ is s itself. ∎

Theorem 4 *Let KB be proper, and let $\phi \in \mathcal{L}$.*
Then $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0 \phi$ iff $KB \longrightarrow \phi$.

Proof: Since *KB* is proper, $\mathrm{gnd}(KB)$ $\mathrm{gnd}(KB)$ does not contain complementary literals. We have $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0 \phi$ iff (by Theorem 5 below) $\mathrm{gnd}(KB) \models \boldsymbol{B}_0 \phi$ iff (by Lemma 2) $\mathrm{gnd}(KB) \models^t \phi$ iff $KB \longrightarrow \phi$, by Lemma 4 in (Lakemeyer & Levesque 2002). ∎

Note that this theorem does not conflict with our goal of avoiding the difficulties with tautological entailment, because it only holds for proper KBs. In the presence of disjunctive information, $\mathcal{SL}$ and tautological entailment will behave differently.

In (Lakemeyer & Levesque 2002), it was shown that $KB \longrightarrow \phi$ iff $V[\phi] = 1$. Thus we have

Corollary 1 *V is a decision procedure for the QESL problem for proper KBs.*

Levesque (1998) claimed without proof that V can be implemented efficiently using database techniques. Liu and Levesque (2003) substantiated this claim by obtaining a tractability result for V.

Now let us consider the general case of proper[+] KBs. In the rest of this section, we assume that KB is proper[+] and $\phi \in \mathcal{L}$. We first present a theorem which reduces the QESL problem for proper[+] KBs to a model checking problem (for an infinite model).

Lemma 3
(1) $\mathrm{gnd}(KB) \models \boldsymbol{B}_0 KB$.
(2) If $s \models \boldsymbol{B}_0 KB$, then $\mathsf{VP}(\mathrm{gnd}(KB)) \subseteq \mathsf{VP}(s)$.

Proof: It is easy to see that $s \models \boldsymbol{B}_0 KB$ iff for any $c \in \mathrm{gnd}(KB)$, $s \models \boldsymbol{B}_0 c$. Thus (1) $\mathrm{gnd}(KB) \models \boldsymbol{B}_0 KB$; and (2) if $s \models \boldsymbol{B}_0 KB$, then $\mathrm{gnd}(KB) \subseteq \mathsf{VP}(s)$, by Proposition 3, $\mathsf{VP}(\mathrm{gnd}(KB)) \subseteq \mathsf{VP}(s)$. ∎

So in a sense, $\mathrm{gnd}(KB)$ is the minimal model of *KB*.

Theorem 5 $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k \phi$ *iff* $\mathrm{gnd}(KB) \models \boldsymbol{B}_k \phi$.

Proof: The only-if direction follows from $\mathrm{gnd}(KB) \models \boldsymbol{B}_0 KB$. Suppose that $\mathrm{gnd}(KB) \models \boldsymbol{B}_k \phi$. Let $s \models \boldsymbol{B}_0 KB$. Then $\mathsf{VP}(\mathrm{gnd}(KB)) \subseteq \mathsf{VP}(s)$. By monotonicity, $s \models \boldsymbol{B}_k \phi$. ∎

We then get the following result about propositional reasoning using $\mathcal{SL}$:

Theorem 6 *In the propositional case, determining whether $\models (\boldsymbol{B}_0 KB \supset \boldsymbol{B}_k \phi)$ can be done in time $O((ln)^{k+1})$, where l is the size of ϕ, and n is the size of KB.*

Proof: We resort to Theorem 5. Note that in the propositional case, *KB* is simply a set of clauses, and $\mathrm{gnd}(KB)$ is *KB* itself. Let $f(k)$ denote the time complexity of deciding if $\mathrm{gnd}(KB) \models \boldsymbol{B}_k \phi$. Then we have: (1) $f(0) = O(ln)$, since unit propagation can be done in linear time; and (2) $f(k) = O(ln \cdot f(k-1))$, where $k > 0$, since each splitting operation is associated with a logical operator or clause. Solving the recurrence, we get that $f(k)$ is $O((ln)^{k+1})$. ∎

Corollary 2 *The QESL problem for proper[+] KBs is tractable (for small, fixed k) in the propositional case.*

Next, we will show that in the first-order case, the QESL problem for proper[+] KBs is decidable by presenting a procedure called W for deciding whether $\mathrm{gnd}(KB) \models \boldsymbol{B}_k \phi$. W is a slight variant of the reasoning procedure X proposed for proper[+] KBs by Lakemeyer and Levesque (2002). The main idea behind W is that to decide $\mathrm{gnd}(KB) \models \boldsymbol{B}_k \phi$, it suffices to consider (1) a finite set of constants when evaluating quantifications, and (2) a finite subset of $\mathrm{gnd}(KB)$ when performing unit propagation or splitting. The argument for this is essentially the same as for X. The intuition is that constants not mentioned in *KB* or ϕ behave the same, so we only need to pick a certain number of representatives.

Let $m \geq 0$. We use H_m^+ to denote the union of the constants in *KB*, those mentioned in the query ϕ, and m new constants appearing nowhere in *KB* and ϕ. Let n be the maximum number of variables in a $\forall$-clause of *KB*. We use

gnd$(KB)|H_n^+$ to denote the set $\{c\theta \mid \forall(e \supset c) \in KB, \theta \in H_n^+,$ and $\models e\theta\}$, where by $\theta \in H_n^+$ we mean that θ only takes constants from H_n^+.

$$W[KB, k, \phi] = \begin{cases} 1 & \text{if one of the following} \\ & \text{conditions (1)-(9) holds} \\ 0 & \text{otherwise} \end{cases}$$

1. $k = 0$, ϕ is a clause c, and there exists $c' \in \mathsf{UP}(\text{gnd}(KB)|H_n^+)$ such that $c' \subseteq c$.

2. $\phi = (d = d')$, and d is identical to d'.

3. $\phi = \neg(d = d')$, and d is distinct from d'.

4. $\phi = \neg\neg\psi$, and $W[KB, k, \psi] = 1$.

5. $\phi = (\psi \vee \eta)$, ψ or η is not a clause, and $W[KB, k, \psi] = 1$ or $W[KB, k, \eta] = 1$.

6. $\phi = \neg(\psi \vee \eta)$, $W[KB, k, \neg\psi] = 1$, and $W[KB, k, \neg\eta] = 1$.

7. $\phi = \exists x.\psi$, and $W[KB, k, \psi_d^x] = 1$ for some $d \in H_1^+$.

8. $\phi = \neg\exists x.\psi$, and $W[KB, k, \neg\psi_d^x] = 1$ for all $d \in H_1^+$.

9. $k > 0$, ϕ is a clause, a disjunction, or an existential, and there is a $\forall(e \supset c) \in KB$ and a $\theta \in H_n^+$ such that $\models e\theta$ and for all $\rho \in c\theta$, $W[KB \cup \{\rho\}, k-1, \phi] = 1$.

Let $*$ be any bijection from constants to constants. We use α^* to denote α with every constant d replaced by d^*. We let Σ^* denote $\{\alpha^* \mid \alpha \in \Sigma\}$. We use θ^* to denote the substitution which assigns variable x the value d^* if θ assigns x the value d. It is easy to prove the following:

Proposition 5 *1.* $\models e$ *iff* $\models e^*$*, where e is an ewff.*
2. $c \in \mathsf{UP}(s)$ *iff* $c^* \in \mathsf{UP}(s^*)$*.*
3. $s \models \boldsymbol{B}_k\phi$ *iff* $s^* \models \boldsymbol{B}_k\phi^*$*.*
4. gnd$(KB)^* = $ gnd(KB^*)*.*

Let $ec_1, \ldots, ec_n$ be the list of constants appearing in H_n^+ but not KB or the query ϕ. Let L be a list of constants $d_1, \ldots, d_k$ $(k \leq n)$ not appearing in H_n^+. We let $id(L)$ represent the bijection that swaps d_i and ec_i, $i = 1, \ldots, k$, and leaves the rest constants unchanged. Note that for any $c \in \mathsf{UP}(\text{gnd}(KB))$, c mentions at most n constants not appearing in H_n^+.

Lemma 4 *Let* $c \in \mathsf{UP}(gnd(KB))$*. Let* $*$ *be* $id(L)$ *where* L *is the list of constants appearing in* c *but not* H_n^+*. Then* $c^* \in \mathsf{UP}(gnd(KB)|H_n^+)$*.*

Proof: We prove by induction on the length of a derivation. Basis: $c \in$ gnd(KB). Then there exist $\forall(e \supset d) \in KB$ and θ s.t. $\models e\theta$ and $c = d\theta$. Since $\models e\theta$ iff $\models e^*\theta^*$, *i.e.* $e\theta^*$, we have that $d\theta^*$, *i.e.* c^*, is in gnd(KB) too. Thus $c^* \in \mathsf{UP}(\text{gnd}(KB)|H_n^+)$. Induction step: c is obtained from ρ and $c \vee \overline{\rho}$. Let $\star$ be $id(L')$ where L' is the list of constants appearing in $c \vee \overline{\rho}$ but not H_n^+. By induction, both $\rho^\star$ and $(c \vee \overline{\rho})^\star$ are in $\mathsf{UP}(\text{gnd}(KB)|H_n^+)$. Thus $c^\star$, *i.e.* c^*, is in $\mathsf{UP}(\text{gnd}(KB)|H_n^+)$ too. ∎

Lemma 5 *Let* ϕ *be a formula in* $\mathcal{L}$ *with a single free variable* x*. Let* b *and* d *be two constants that do not appear in* KB *or* ϕ*. Then* gnd$(KB) \models \boldsymbol{B}_k\phi_b^x$ *iff* gnd$(KB) \models \boldsymbol{B}_k\phi_d^x$*.*

Proof: Let $*$ be the bijection that swaps b and d and leaves the rest constants unchanged. Then gnd$(KB) \models \boldsymbol{B}_k\phi_b^x$ iff (by Proposition 5 (3)) gnd$(KB)^* \models \boldsymbol{B}_k(\phi_b^x)^*$ iff (by Proposition 5 (4)) gnd$(KB^*) \models \boldsymbol{B}_k(\phi_b^x)^*$ iff gnd$(KB) \models \boldsymbol{B}_k\phi_d^x$, since $*$ leaves constants in KB or ϕ unchanged. ∎

Lemma 6 *Suppose that* gnd$(KB) \models \boldsymbol{B}_k\phi$ *by splitting on* $c \in gnd(KB)$*. Then* gnd$(KB) \models \boldsymbol{B}_k\phi$ *by splitting on some* $c' \in gnd(KB)|H_n^+$*.*

Proof: Let $*$ be $id(L)$ where L is the list of constants appearing in c but not H_n^+. Then $c^* \in$ gnd$(KB)|H_n^+$. Let $\rho \in c$. Then gnd$(KB) \cup \{\rho\} \models \boldsymbol{B}_{k-1}\phi$, that is, gnd$(KB \cup \{\rho\}) \models \boldsymbol{B}_{k-1}\phi$. Thus gnd$(KB \cup \{\rho\})^* \models \boldsymbol{B}_{k-1}\phi^*$, that is, gnd$(KB \cup \{\rho^*\}) \models \boldsymbol{B}_{k-1}\phi$. Therefore gnd$(KB) \models \boldsymbol{B}_k\phi$ by splitting on c^*. ∎

Theorem 7 gnd$(KB) \models B_k\phi$ *iff* $W[KB, k, \phi] = 1$*.*

Proof: We prove by induction on $\| B_k\phi \|$. Here we only prove the cases of clauses, disjunctions, and quantifications. The other cases follow easily from properties of beliefs.

1. By Lemma 4, when H_n^+ contains constants appearing in c, $c \in \mathsf{UP}(\text{gnd}(KB))$ iff $c \in \mathsf{UP}(\text{gnd}(KB)|H_n^+)$. Thus gnd$(KB) \models \boldsymbol{B}_k c$ iff $k = 0$ and there exists $c' \in \mathsf{UP}(\text{gnd}(KB)|H_n^+)$ such that $c' \subseteq c$, or $k > 0$ and gnd$(KB) \models \boldsymbol{B}_k c$ by splitting.

2. gnd$(KB) \models \boldsymbol{B}_k(\psi \vee \eta)$, where ψ or η is not a clause, iff gnd$(KB) \models \boldsymbol{B}_k\psi$ or gnd$(KB) \models \boldsymbol{B}_k\eta$ or gnd$(KB) \models \boldsymbol{B}_k(\psi \vee \eta)$ by splitting.

3. By Lemma 5, gnd$(KB) \models \boldsymbol{B}_k\neg\exists x.\psi$ iff gnd$(KB) \models \boldsymbol{B}_k\neg\psi_d^x$ for all $d \in H_1^+$.

4. gnd$(KB) \models \boldsymbol{B}_k\exists x.\psi$ iff gnd$(KB) \models \boldsymbol{B}_k\psi_d^x$ for some $d \in H_1^+$ or gnd$(KB) \models \boldsymbol{B}_k\exists x.\psi$ by splitting.

5. By Lemma 6, gnd$(KB) \models \boldsymbol{B}_k\phi$ by splitting iff this holds by splitting on some $c \in$ gnd$(KB)|H_n^+$. ∎

Corollary 3 *The QESL problem for proper$^+$ KBs is decidable in the first-order case.*

A Complete Axiomatization for Propositional $\mathcal{SL}$

In this section, we present a sound and complete axiomatization for propositional $\mathcal{SL}$, *i.e.* a set of axioms and inference rules that generate all and only the valid sentences. Although it is not intended as a step towards "automating" the logic, it does provide another useful perspective on the valid sentences. As far as we can tell, due to the peculiarity of the semantics of $\mathcal{SL}$, the general techniques for obtaining complete axiomatizations for classical logics of knowledge and belief (Halpern & Moses 1992) do not apply to $\mathcal{SL}$. The key to our complete axiomatization lies in the construction of sets of representative models, called RM-sets, for belief atoms $\boldsymbol{B}_k\phi$. Since the definition of RM-sets is non-trivial, we leave it to the end of this section. For now, it is sufficient to know that a RM-set of $\boldsymbol{B}_k\phi$ is a finite set Δ of finite setups, and atoms appearing in Δ but not ϕ are called helping atoms. In what follows, we identify a finite setup t with the conjunction of the clauses in t.

Our proof system is as follows:

Axioms:

A1 All instances of propositional tautologies

A2 Unit Resolution: $\boldsymbol{B}_0\rho \wedge \boldsymbol{B}_0(\overline{\rho} \vee c) \supset \boldsymbol{B}_0 c$, where ρ is a literal and c is a clause

A3 Subsumption: $\boldsymbol{B}_0 c \supset \boldsymbol{B}_0 c'$, where c and c' are clauses, and $c \subseteq c'$

A4 Belief Reduction for $\boldsymbol{B}_0$: $\boldsymbol{B}_0\phi \supset (\boldsymbol{B}_0\phi)\!\downarrow$

A5 Belief Reduction for $\boldsymbol{B}_k$: $(\boldsymbol{B}_k\phi)\!\downarrow \supset \boldsymbol{B}_k\phi$

Inference rules:

R1 Modus Ponens: from α and $\alpha \supset \beta$ infer β

R2 Case Analysis: from $(\bigvee_{\rho \in c} \boldsymbol{B}_0\rho) \wedge \boldsymbol{B}_j\psi \supset \boldsymbol{B}_k\phi$, infer $\boldsymbol{B}_0 c \wedge \boldsymbol{B}_j\psi \supset \boldsymbol{B}_{k+1}\phi$, where c is a clause

R3 Representative Model: from $(\bigvee_{t \in \Delta} \boldsymbol{B}_0 t) \supset \alpha$, infer $\boldsymbol{B}_k\phi \supset \alpha$, where Δ is a RM-set of $\boldsymbol{B}_k\phi$ such that its helping atoms do not appear in α.

Theorem 8 *The axiom system is sound and complete.*

The proof is presented in (Liu 2004). The soundness part is a typical proof by induction on the length of a derivation, where the main complication is the soundness of R2 and R3.

The completeness part is more involved but here are the main ideas: A belief literal is a belief atom or its negation; a belief clause is a finite set of belief literals; a $\mathcal{SL}$ formula is in CNF if it is a conjunction of belief clauses. Clearly, any $\mathcal{SL}$ formula α can be put into an equivalent formula in CNF. To prove a valid $\mathcal{SL}$ formula α, we first prove its CNF form and then prove α from it by using A1 and R1. Now consider a valid belief clause

$$\beta = \boldsymbol{B}_{j_1}\phi_1 \wedge \ldots \wedge \boldsymbol{B}_{j_m}\phi_m \supset \boldsymbol{B}_{k_1}\psi_1 \vee \ldots \vee \boldsymbol{B}_{k_n}\psi_n.$$

Let Δ_i be a RM-set of $\boldsymbol{B}_{j_i}\phi_i$, $i = 1, \ldots, m$ such that helping atoms of $\Delta_1, \ldots, \Delta_m$ are pairwise disjoint. To prove β, we first prove $(\bigvee_{t_1 \in \Delta_1} \boldsymbol{B}_0 t_1) \wedge \ldots \wedge (\bigvee_{t_m \in \Delta_m} \boldsymbol{B}_0 t_m) \supset \boldsymbol{B}_{k_1}\psi_1 \vee \ldots \vee \boldsymbol{B}_{k_n}\psi_n$ and then prove β from this formula by repeatedly applying R3. Now consider a valid belief clause

$$\gamma = \boldsymbol{B}_0 t \supset \boldsymbol{B}_{k_1}\psi_1 \vee \ldots \vee \boldsymbol{B}_{k_n}\psi_n,$$

where t is a finite setup. We claim that $\boldsymbol{B}_0 t \supset \boldsymbol{B}_{k_i}\psi_i$ is valid for some $i = 1, \ldots, n$. Since $t \models \boldsymbol{B}_0 t$, $t \models \boldsymbol{B}_{k_i}\psi_i$ for some i. Now let $s \models \boldsymbol{B}_0 t$. Then $t \subseteq \mathsf{VP}(s)$. By monotonicity, $s \models \boldsymbol{B}_{k_i}\psi_i$. Thus to prove γ, we prove $\boldsymbol{B}_0 t \supset \boldsymbol{B}_{k_i}\psi_i$ for some i. Finally, valid formulas of the form $\boldsymbol{B}_0 t \supset \boldsymbol{B}_k\phi$ can be proved by using axioms and proof rules other than R3.

We now present the definition of RM-sets, beginning with the definition of splitting models. Intuitively, a RM-set Δ of a belief atom $\boldsymbol{B}_k\phi$ is a finite set of finite models of $\boldsymbol{B}_k\phi$ such that each model of $\boldsymbol{B}_k\phi$ has a representative in Δ, in a sense we will explain soon. Moreover, if $s \models \boldsymbol{B}_k\phi$ by splitting, then its representative in Δ is a splitting model.

Let c be a clause and s a setup. We use $c\tilde{\vee}s$ to denote the setup $\{(c \vee d) \mid d \in s\}$.

Definition 3 Let $\Delta = \{t_1, \ldots, t_n\}$ be a finite set of finite setups. Let $x_i, y_i, z_i, i = 1, \ldots, n$, be distinct atoms not

appearing in Δ. We call them helping atoms. The following is a type-1 splitting model wrt Δ:

$$\{\bigvee_i x_i \vee y_i\} \cup \bigcup_i \neg x_i \tilde{\vee} t_i \cup \bigcup_i \neg y_i \tilde{\vee} t_i.$$

The following is a type-2 splitting model wrt Δ:

$$\{\bigvee_i x_i \vee y_i\} \cup \bigcup_i \{\neg x_i \vee z_i, \neg y_i \vee z_i\} \cup$$
$$\bigcup_i (\neg x_i \vee \neg z_i)\tilde{\vee} t_i \cup \bigcup_i (\neg y_i \vee \neg z_i)\tilde{\vee} t_i.$$

Definition 4 The RM-sets of $\boldsymbol{B}_k\phi$ are inductively defined on $\|\boldsymbol{B}_k\phi\|$ as follows:

1. The only RM-set of $\boldsymbol{B}_0 c$ is $\{\{c\}\}$.
2. If Δ is a RM-set of $\boldsymbol{B}_k c$, and t is a type-i splitting model wrt Δ (if $k > 0$ then $i = 1$ else $i = 2$), then $\{t\}$ is a RM-set of $\boldsymbol{B}_{k+1} c$.
3. A RM-set of $\boldsymbol{B}_k\psi$ is a RM-set of $\boldsymbol{B}_k\neg\neg\psi$.
4. If Δ_i is a RM-set of $\boldsymbol{B}_k\neg\psi_i$, $i = 1, 2$, and the helping atoms of Δ_1 and Δ_2 are disjoint, then $\{t_1 \cup t_2 \mid t_i \in \Delta_i, i = 1, 2\}$ is a RM-set of $\boldsymbol{B}_k\neg(\psi_1 \vee \psi_2)$.
5. If Δ_i is a RM-set of $\boldsymbol{B}_0\psi_i$, $i = 1, 2$, then $\Delta_1 \cup \Delta_2$ is a RM-set of $\boldsymbol{B}_0(\psi_1 \vee \psi_2)$.
6. If Δ_i is a RM-set of $\boldsymbol{B}_{k+1}\psi_i$, $i = 1, 2$, Δ is a RM-set of $\boldsymbol{B}_k(\psi_1 \vee \psi_2)$, and t is a type-i splitting model wrt Δ (if $k > 0$ then $i = 1$ else $i = 2$), then $\Delta_1 \cup \Delta_2 \cup \{t\}$ is a RM-set of $\boldsymbol{B}_{k+1}(\psi_1 \vee \psi_2)$.

The following theorem characterizes RM-sets:

Theorem 9 *Let Δ be a RM-set of $\boldsymbol{B}_k\phi$, and let H be its helping atoms. Then*

1. *Δ is a finite set of finite setups.*
2. *For all $t \in \Delta$, $t \models \boldsymbol{B}_k\phi$.*
3. *For any setup s such that $s \models \boldsymbol{B}_k\phi$ and s does not mention atoms in H, there exists $t \in \Delta$ such that for any α not mentioning atoms in H, $s \cup t \models \alpha$ iff $s \models \alpha$.*

The proof is presented in (Liu 2004).

The above Property 3 says that each model s of $\boldsymbol{B}_k\phi$ such that s does not mention atoms in H has a representative t in Δ in the sense that $s \cup t$ and s agree on all $\mathcal{SL}$ formulas not mentioning atoms in H. Now we are in a good position to explain the motivation behind defining two types of splitting models. Consider the belief atom $\boldsymbol{B}_1 p$. Assume that our definition was: if t is a type-1 splitting model wrt $\{p\}$, then $\{t\}$ is a RM-set of $\boldsymbol{B}_1 p$. Now let $t = \{x \vee y, \overline{x} \vee p, \overline{y} \vee p\}$, and let $s = \{u \vee v, \overline{u} \vee w, \overline{v} \vee w, \overline{u} \vee \overline{w} \vee p, \overline{v} \vee \overline{w} \vee p, \overline{p}\}$. Then $s \models \boldsymbol{B}_1 p$, $s \cup t \models \boldsymbol{B}_0 p \wedge \boldsymbol{B}_0\overline{p}$, but $s \not\models \boldsymbol{B}_0 p \wedge \boldsymbol{B}_0\overline{p}$. Thus Property 3 would not hold.

Example 3

1. $\Delta = \{\{p, q\}, \{r\}\}$ is a RM-set of $\boldsymbol{B}_0[p \wedge q \vee r]$;
2. t_1 is a type-2 splitting model wrt Δ, where $t_1 = \{u \vee v \vee x \vee y, \overline{u} \vee w, \overline{v} \vee w, \overline{x} \vee z, \overline{y} \vee z, \overline{u} \vee \overline{w} \vee p, \overline{v} \vee \overline{w} \vee p, \overline{u} \vee \overline{w} \vee q, \overline{v} \vee \overline{w} \vee q, \overline{x} \vee \overline{z} \vee r, \overline{y} \vee \overline{z} \vee r\}$;
3. $\{t_2\}$ is RM-set of $\boldsymbol{B}_1 r$, where $t_2 = \{u \vee v, \overline{u} \vee w, \overline{v} \vee w, \overline{u} \vee \overline{w} \vee r, \overline{v} \vee \overline{w} \vee r\}$;

4. $\{t_3\}$ is RM-set of $\boldsymbol{B}_1(p \wedge q)$, where $t_3 = \{u \vee v, \overline{u} \vee w, \overline{v} \vee w, \overline{u} \vee \overline{w} \vee p, \overline{v} \vee \overline{w} \vee p, x \vee y, \overline{x} \vee z, \overline{y} \vee z, \overline{x} \vee \overline{z} \vee q, \overline{y} \vee \overline{z} \vee q\}$;

5. $\{t_1, t_2, t_3\}$ is a RM-set of $\boldsymbol{B}_1[p \wedge q \vee r]$.

Related Work

Our work on $\mathcal{SL}$ grew out of our attempts to semantically characterize the reasoning procedure X proposed for proper$^+$ KBs in (Lakemeyer & Levesque 2002). The main difference between X and the procedure W presented in this paper is: in X, the depth of case splitting allowed depends on the form of the query, while in W, this number is supplied explicitly as an extra parameter k.

Existing semantic approaches to limited reasoning can be put into two categories. Early work (Levesque 1984; Frisch 1987; Schaerf & Cadoli 1995; Patel-Schneider 1985; Lakemeyer 1990) was based on tautological entailment. Later work (Dalal 1996; Crawford & Etherington 1998) was based on unit propagation, but restricted to the propositional case. The last two grew out of attempts to semantically characterize the concept of Socratic completeness, which was first introduced in (Crawford & Kuipers 1989) and later generalized to the notion of Socratic proof system (McAllester & Givan 1993). Dalal's work is limited to a propositional clausal language. Crawford and Etherington (1998) attempted to extend this work to the full propositional language. They proposed a non-deterministic semantics. However, their notion of models is so loosely defined that almost none of the normal Boolean laws holds in their logic. In the following, we first compare $\mathcal{SL}$ with tautological entailment, and then with Dalal's logic.

In some cases, $\mathcal{SL}$ is stronger than tautological entailment. For example, we have that $\models \boldsymbol{B}_0[p \wedge (\overline{p} \vee r)] \supset \boldsymbol{B}_0 r$ and $\models \boldsymbol{B}_0[(p \vee q) \wedge (\overline{p} \vee r) \wedge (\overline{q} \vee r)] \supset \boldsymbol{B}_1 r$, but $p \wedge (\overline{p} \vee r) \not\longrightarrow r$ and $(p \vee q) \wedge (\overline{p} \vee r) \wedge (\overline{q} \vee r) \not\longrightarrow r$. However, in some other cases, $\mathcal{SL}$ is weaker than tautological entailment. Consider KB2 in Figure 1. We have that $KB2 \longrightarrow \exists x.(P(x) \wedge Q(x))$, but $\not\models \boldsymbol{B}_0 KB2 \supset \boldsymbol{B}_k \exists x.(P(x) \wedge Q(x))$ for $k < 8$. Also, there are cases where $\mathcal{SL}$ coincides with tautological entailment. For example, as shown by Theorem 4, the two coincide on proper KBs. As to the computational property, consider proper$^+$ KBs. We know that deciding whether $KB \longrightarrow \phi$ is co-NP-hard in the propositional case and undecidable in the first-order case, while deciding whether $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k \phi$ is tractable in the propositional case and decidable in the first-order case.

Dalal (1996) considers a propositional clausal language, and provides a model-theoretic semantics for Boolean Constraint Propagation (BCP), a variant of unit propagation. More precisely, he defines an entailment relation $\approx$ between clausal theories and clauses, and shows that a refutation variant of BCP is sound and complete for $\approx$, that is, for any clausal theory Σ and any clause c, $\Sigma \approx c$ iff the empty clause can be obtained by BCP from $\Sigma \cup \overline{c}$, where $\overline{c} = \{\overline{\rho} \mid \rho \in c\}$. Moreover, Dalal extends the inference relation $\vdash_{\mathrm{BCP}}$ to a family of inference relations $\vdash_k^{\mathrm{BCP}}$, $k \geq 0$, by allowing Modus Ponens on clauses of restricted size. This family of inference relations is eventually complete.

Now we restrict ourselves to the propositional clausal language, and compare $\mathcal{SL}$ with Dalal's logic. Let Σ be a clausal theory and c a clause. We write $\Sigma \models_k^{\mathrm{SL}} c$ if $\models (\boldsymbol{B}_0 \Sigma \supset \boldsymbol{B}_k c)$. First, note that tautologous clauses are handled differently in the two approaches. We have that $p \vdash_{\mathrm{BCP}} (q \vee \overline{q})$ but $\not\models \boldsymbol{B}_0 p \supset \boldsymbol{B}_k (q \vee \overline{q})$ for any k. Secondly, $\vdash_{\mathrm{BCP}}$ is strictly stronger than $\models_0^{\mathrm{SL}}$. For example, we have that $(p \vee q) \wedge (\overline{p} \vee q) \vdash_{\mathrm{BCP}} q$ but $\not\models \boldsymbol{B}_0[(p \vee q) \wedge (\overline{p} \vee q)] \supset \boldsymbol{B}_0 q$. However, in general, $\vdash_k^{\mathrm{BCP}}$ and $\models_k^{\mathrm{SL}}$ are incomparable. For example, let $\Sigma_1 = \{(u \vee v), (\overline{u} \vee v), (\overline{v} \vee p \vee q), (\overline{v} \vee \overline{p} \vee q)\}$, then $\Sigma_1 \vdash_1^{\mathrm{BCP}} q$ but $\not\models \boldsymbol{B}_0 \Sigma_1 \supset \boldsymbol{B}_1 q$; let $\Sigma_2 = \{(u \vee v), (x \vee y), (\overline{u} \vee \overline{x} \vee p), (\overline{u} \vee \overline{y} \vee q), (\overline{v} \vee \overline{x} \vee q), (\overline{v} \vee \overline{y} \vee p)\}$, then $\models \boldsymbol{B}_0 \Sigma_2 \supset \boldsymbol{B}_2(p \vee q)$ but $\Sigma_2 \not\vdash_2^{\mathrm{BCP}} (p \vee q)$. Finally, similar to $\vdash_k^{\mathrm{BCP}}$, $\models_k^{\mathrm{SL}}$ is eventually complete for nontautologous clauses, which are examples of queries in the normal form $\mathcal{NF}$.

Conclusions

In this paper, we have proposed a new logic of limited belief called $\mathcal{SL}$, with the goal of providing a semantically coherent and computationally attractive reasoning service for knowledge bases with disjunctive information. Reasoning based on $\mathcal{SL}$ is always classically sound, and in some simple cases, is also classically complete. Given disjunctive facts, it performs unit propagation, but only does case analysis in a limited way, under user control. While the reasoning service is well-defined for any first-order KB, we have considered its computational property for two special cases. For proper KBs, which represent incomplete knowledge without disjunction, the reasoning service can be realized using the efficient database procedure discussed in (Liu & Levesque 2003). For proper$^+$ KBs, which represent incomplete knowledge including disjunction, we have proved that the reasoning service is tractable in the propositional case and decidable in the first-order case. Also, we have presented a sound and complete axiomatization for propositional $\mathcal{SL}$.

There are a number of topics for future research. First of all, the Representative Model inference rule in our axiomatization is obscure and unintuitive. It would be desirable to find a more natural axiom system. Moreover, we would like to generalize our axiomatization to the first-order case. It would also be interesting to analyze the complexity of the satisfiability problem of propositional $\mathcal{SL}$. Also, we would like to explore in the first-order case, under what restrictions on proper$^+$ KBs and queries, reasoning based on $\mathcal{SL}$ is eventually complete. But the more pressing problem is this: while query evaluation based on $\mathcal{SL}$ for proper$^+$ KBs is decidable, it is crucial to identify "islands of tractability" by applying restrictions on proper$^+$ KBs and queries. This can be seen as an extension of the work presented in (Liu & Levesque 2003), where a tractable case of the reasoning procedure V was identified. We expect that the graphical notion of tree-width will again play an important role in this research.

Acknowledgments

References

Anderson, A., and Belnap, N. 1975. *Entailment: The Logic of Relevance and Necessity.* Princeton University Press.

Cadoli, M., and Schaerf, M. 1992. Approximate reasoning and non-omniscient agents. In *Proc. of the 4th Conference on Theoretical Aspects of Reasoning about Knowledge (TARK-92)*, 159–183.

Cadoli, M., and Schaerf, M. 1996. On the complexity of entailment in propositional multivalued logics. *Annals of Mathematics and Artificial Intelligence* 18(1):29–50.

Crawford, J., and Etherington, D. 1998. A nondeterministic semantics for tractable inference. In *Proc. of AAAI-98*, 286–291.

Crawford, J., and Kuipers, B. 1989. Towards a theory of access limited reasoning. In *Proc. of KR-89*, 67–78.

Dalal, M. 1996. Semantics of an anytime family of reasoners. In *Proc. of the 12th European Conference on Artificial Intelligence (ECAI-96)*, 360–364.

Delgrande, J. 1995. A framework for logics of explicit belief. *Computational Intelligence* 11(1):47–88.

Dunn, J. 1976. Intuitive semantics for first-degree entailments and coupled trees. *Philosophical Studies* 29:149–168.

Fagin, R., and Halpern, J. 1988. Belief, awareness, and limited reasoning. *Artificial Intelligence* 34(1):39–76.

Fagin, R.; Halpern, J.; and Vardi, M. 1990. A nonstandard approach to the logical omniscience problem. In *Proc. of the 3rd Conference on Theoretical Aspects of Reasoning about Knowledge (TARK-90)*, 41–55.

Frisch, A. 1987. Inference without chaining. In *Proc. of IJCAI-87*, 515–519.

Halpern, J., and Moses, Y. 1992. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence* 54(3):319–379.

Hintikka, J. 1962. *Knowledge and Belief.* Cornell University Press.

Konolige, K. 1986. *A Deduction Model of Belief.* Brown University Press.

Kripke, S. 1959. A completeness theorem in modal logic. *Journal of Symbolic Logic* 24:1–14.

Kripke, S. 1976. Is there a problem with substitutional quantification? In Evans, G., and McDowell, J., eds., *Truth and Meaning*. Clarendon Press, Oxford. 325–419.

Lakemeyer, G., and Levesque, H. 2002. Evaluation-based reasoning with disjunctive information in first-order knowledge bases. In *Proc. of KR-02*, 73–81.

Lakemeyer, G. 1990. *Models of Belief for Decidable Reasoning in Incomplete Knowledge Bases*. Ph.D. Dissertation, Department of Computer Science, University of Torono.

Levesque, H. 1984. A logic of implicit and explicit belief. In *Proc. of AAAI-84*, 198–202.

Levesque, H. 1998. A completeness result for reasoning with incomplete first-order knowledge bases. In *Proc. KR-98*, 14–23.

Liu, Y., and Levesque, H. 2003. A tractability result for reasoning with incomplete first-order knowledge bases. In *Proc. of IJCAI-03*, 83–88.

Liu, Y. 2004. Ph.D. Dissertation, Department of Computer Science, University of Torono. In preparation.

McAllester, D., and Givan, B. 1993. Taxonomic syntax for first order inference. *Journal of the ACM* 40(2):246–283.

McAllester, D. 1990. Truth maintenance. In *Proc. of AAAI-90*, 1109–1116.

Patel-Schneider, P. 1985. A decidable first-order logic for knowledge representation. In *Proc. of IJCAI-85*, 455–458.

Schaerf, M., and Cadoli, M. 1995. Tractable reasoning via approximation. *Artificial Intelligence* 74:249–310.

Vardi, M. 1986. On epistemic logic and logical omniscience. In *Proc. of the 1st Conference on Theoretical Aspects of Reasoning about Knowledge (TARK-86)*, 293–305.