Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Document
D-93-25

Modeling Epistemic Propositions

Hans-Jürgen Bürckert, Werner Nutt (Eds.)

December 1993

Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH

Postfach 20 80
67608 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
66123 Saarbrücken, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern and Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Atlas Elektronik, Daimler-Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Mannesmann-Kienzle, SEMA Group, and Siemens. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- ❑ Intelligent Engineering Systems
- ❑ Intelligent User Interfaces
- ❑ Computer Linguistics
- ❑ Programming Systems
- ❑ Deduction and Multiagent Systems
- ❑ Document Analysis and Office Automation.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Friedrich J. Wendl
Director

Modeling Epistemic Propositions

Hans-Jürgen Bürckert, Werner Nutt (Eds.)

DFKI-D-93-25

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-9201).

Modeling Epistemic Propositions

Workshop during the
17th German Conference on Artificial Intelligence (KI'93)
Humboldt University
Berlin, September 13-14, 1993

Organizers:
Hans-Jürgen Bürckert and Werner Nutt

Preface

This is a collection of papers presented at a workshop on "Modeling Epistemic Propositions" held on September 13 and 14 at the Humboldt University in Berlin during the 17th German Conference on Artificial Intelligence (KI'93). The workshop was the first of its kind in Germany and its goal was to provide a forum for researchers that are interested in problems of epistemic modeling.

Modeling epistemic and doxastic propositions, *i.e.*, knowledge and belief, is one of the main themes of research in knowledge representation. One of the reasons is in the fundamental goal of Artificial Intelligence: the ability of a system to reflect its own knowledge, to consider what information is held by other systems, and to take into account the knowledge and the intentions of human users is a necessary precondition for intelligent behavior.

There are, however, also pragmatic reasons that suggest to describe mental states differently from the physical world. For instance, in natural language systems special operators are used to represent beliefs, intentions, and other mental modes. When specifying distributed systems or multi-agent systems, epistemic operators are employed to distinguish between descriptions of the state of the world and representations of an agent's information about it. In most cases, such specifications are only descriptive in the sense that they are used to infer properties of the specified system. Recent work, however, also aims at executing them. In user modeling, knowledge of the system is separated from the user's knowledge in a similar fashion. Several authors like Levesque and Reiter proposed to conceive integrity constraints as descriptions of the legal epistemic states of a knowledge base and therefore to formalize them in logics with epistemic operators. Finally, several approaches to formalize non-monotonic reasoning are based on epistemic operators.

During the workshop, eight papers were presented, David Pearce gave an invited talk, where he compared recent approaches in Philosophy and Artificial Intelligence, and Christoph Lingenfelder and Rolf Socher-Ambrosius held a tutorial on the semantics and proof theory of modal logics.

The focus of the workshop was on the ***logic based*** formalization of knowledge and belief. Since the work by Moore in the early 80'ies, modal logics have been generally accepted for this purpose. Not surprisingly, most contributions to the workshop followed this approach. The papers centered around three main topics:

- different axiomatizations of epistemic modalities, their advantages and drawbacks, and their adequacy (Pearce, Thijsse and Wansing, Wagner, Rollinger)
- inference procedures for epistemic logics (Hustadt, Laux)
- applications of epistemic logics to Artificial Intelligence (Becker and Lakemeyer, Blok, Donini *et al.*).

In the call for papers, we proposed as a further topic the combination of epistemic logics with other special purpose logics, like logics for temporal or non-monotonics reasoning. We also would have liked to see contributions following a paradigm other than modal logics or even criticizing the logic-based approach per se. Unfortunately, we did not receive submissions of this kind.

We would like to thank the authors for their contributions and all the participants for the lively discussion during the sessions. We feel that the workshop has met its goal to establish contacts between researchers from different sites in Germany that would not have come together otherwise. At the end of the workshop it has been suggested to hold a second event of this kind during the German Conference on Artificial Intelligence in 1994. We hope that we will see the participants again at "Modeling Epistemic Propositions" during KI'94 in Saarbrücken.

December 1993 Hans-Jürgen Bürckert and Werner Nutt

Contents

Talks

Appendix

Epistemic Operators and Knowledge-Based Reasoning: A Survey and Critical Comparison of some Recent Approaches in Philosophy and Artificial Intelligence

David Pearce
Gruppe Logik, Wissenstheorie und Information
Institut für Philosphie, Freie Universität Berlin
Habelschwerdter Allee 30, 1000 Berlin 33, Germany
pearce@inf.fu-berlin.de

Abstract

We discuss several recent approaches to the modelling of knowledge operators in Philosophical Logic and Artificial Intelligence. Particular attention will be given to the following topics:

1. The representation of several different modal operators within the same system (e.g., knowledge, belief, memory)
2. Epistemic logics based on nonclassical concepts of inference (e.g., partial and constructive inference)
3. Epistemic operators in the area of nonmonotonic knowledge representation systems.

Our main aim is to point out some connections between these areas, in particular between (2) and (3).

An Epistemic Extension of CLASSIC as Interaction Language for CLASSIC Knowledge Bases

Andreas Becker and Gerhard Lakemeyer

Universität Bonn
Institut für Informatik III
Römerstraße 164, 53117 Bonn
e-mail: becker@gmdzi.gmd.de, gerhard@cs.uni-bonn.de

As shown by Levesque, the interaction with a knowledge base can be viewed in functional terms considering the operations ASK and TELL, by which queries can be put to the system and new information can be added to the system, respectively. In particular, Levesque emphasizes the fact that the expressiveness is enhanced considerably by an epistemic interaction language, even when the knowledge base consists only of first order logic (FOL) formulas and does not contain any epistemic operators. Interestingly enough, epistemic ASK- and TELL-operations can be reduced to ordinary non-epistemic operations.

The interaction language for knowledge bases written in the concept language Classic, which can be considered a first order logic fragment, so far consists of Classic expressions in combination with metalogical concepts to increase expressiveness. The project presented in the paper aims at an epistemic extension of Classic (K-Classic) in order to get a more powerful interaction language for operations like ASK and TELL. Essentially, the increase in power results from the ability to talk not only about the application domain when interacting with the knowledge base, but also about its knowledge about the domain. Apart from defining the extended language, we investigate whether reduction procedures similar to Levesque's approach can be applied. Moreover, integrity constraints are dealt with, which can be formulated as special epistemic queries that always must return the answer 'yes' for the particular integrity constraint to be fulfilled and which can therefore be viewed as specific ASK-operations. Finally, epistemic assertions as arguments for TELL-operations can be used, among other things, to introduce the Closed World Assumption locally for single concepts or roles.

On the contribution of contextual information to the semantics and pragmatics of focus

Peter I. Blok
University of Leuven
University of Groningen

0. Introduction

Focus is one of the most troublesome concepts of linguistic theory. The main source of the confusion is the fact that the *phenomena* and their theoretical description are being mixed up. E.g., a constituent is assumed to be in focus when it is intonationally stressed as well when it is prominent in a discourse. In this paper we will try to avoid this kind of confusions and limit our definition of focus to the phenomenal side:

For a constituent to be in focus it is a necessary condition that it either contain a sub-constituent with contrastive intonational stress or is syntactically marked by means of a cleft or left dislocation construction.

Hence, we will consider focus to be a *syntactic* phenomenon. The problem is, of course, what contribution to the meaning of a sentence the focus has.
The phenomena we will deal with are the following three paradigmatic cases of focus:

I Stress in sentences with a so-called *focus adverb*, like *only*.
II Stress in *negated* sentences.
III Sentences with stressed constituents which do not fall under I or II.

In the examples that fall under I, the position of the stress influences the truth conditions of the sentence, as is illustrated in (1):

(1) a John only introduced *Bill* to Sue
b John only introduced Bill to *Sue*
c John only *introduced* Bill to Sue

(1)a and (1)b are not necessarily true in the same models.[1]
In negated sentences, case II, the truth conditions are not affected by the position of the stress:

(2) a John didn't introduce *Bill* to Sue
b John didn't introduce Bill to *Sue*

(2)a and (2)b are true in the same models. However, their use is different: a speaker uttering (2)a seems to indicate that John introduced someone else than Bill to Sue, whereas a speaker of (2)b indicates that John introduced Bill to someone else.
Case III encompasses sentences like (3)a and (3)b:

(3) a John introduced *Bill* to Sue
b John introduced Bill to *Sue*

Clearly, (3)a and (3)b have the same truth conditions. Moreover, a speaker of (3)a or (3)b does not seem to have to say more than what (s)he said. This distinguishes case III from case II. We will call case III *bare focus* (BF).
To be sure, the typology just introduced remains at the observational level, but it mirrors the complexity of the phenomena involved. It also illustrates the fact that focus is not a coherent phenomenon in traditional semiotics: in case A the stress influences the *semantics*, whereas in cases II and III it has repercussions for the *pragmatics* only. Many linguists find this quite puzzling. Since the syntactic phenomenon (i.e. stress) is the same in I, II, and III, they argue, the interpretation should have something in common as well. Most contemporary focus theorists (Rooth, von Stechow, Krifka) try to give a *semantic* description of what focus constructions have in common. In order to do so, they have to postulate a second level of semantic interpretation, besides the normal denotation, which is allegedly determined by the focus structure of the sentence. In the next section we will investigate this so called *alternative semantics* (AS).

[1] It is not quite clear what the truth conditions of (1)c are; but they will be different from those of (1)a and (1)b.

2. Alternative Semantics

We will now take a brief look at the most prominent paradigm in focus theory, so called Alternative Semantics (AS), as proposed by Rooth (1985). The central idea is this. In the examples in (1), *only* is in pre-VP position. Because of this syntactic reason, *only* is assumed to take the VP for an argument. But then, *only* denotes the superset relation. It is the other argument of *only* which makes the interpretations of the sentences in (1) differ. This argument is the so called P-set. Intuitively, P-sets are defined recursively in the following way: if a basic expression b is unfocused, its P-set, $||b||$, is the singleton set containing its denotation. Hence, the P-set of "Sue" in (1)a is {Sue}, where Sue $\in D_e$. If a basic expression is focused, its P-set is the whole domain of the type of the focused expression. Therefore, the P-set of "Bill" in (1)a is D_e. P-sets of complex expressions are built by taking the Cartesian product of the P-sets of their sub-constituents. Less intuitive, but formally more correct, P-sets are sets of objects corresponding to what Montague (1974) calls *meanings*, i.e. functions from indices and assignments into Extensions. It is important to notice that P-sets are *more* than meanings only: meanings correspond to words in the language, whereas P-sets consist of all functions of the appropriate type in the model. For (1) we get the following P-sets:

<u>P-set</u>

||(1)a|| = ||introduce *Bill* to Sue|| = $\{P \mid \exists y\ P = [\![{}^\wedge\text{introduce y to Sue}]\!]^M\}$

||(1)b)|| = ||introduce Bill to *Sue*|| = $\{P \mid \exists y\ P = [\![{}^\wedge\text{introduce Bill to y}]\!]^M\}$

||(1)c|| = ||*introduce* Bill to Sue|| = $\{P \mid \exists Y\ P = [\![{}^\wedge Y(\text{Bill to Sue})]\!]^M\}$

The meaning of (1)a then becomes: "for all functions in ||(1)a|| there is just one true of John, i.e. the one in which y is *Bill*", and analogously for (1)b and (1)c. The second order translations which resemble the proposed analysis most are, respectively,

(4)a $\forall P([P\{j\} \wedge \exists y\ P = {}^\wedge\lambda x\ \text{introduce}(x,y,s)] \rightarrow P = {}^\wedge\lambda x\ \text{introduce}'(x,b,s))$
(4)b $\forall P([P\{j\} \wedge \exists y\ P = {}^\wedge\lambda x\ \text{introduce}(x,b,y)] \rightarrow P = {}^\wedge\lambda x\ \text{introduce}'(x,b,s))$
(4)c $\forall P([P\{j\} \wedge \exists Y\ P = {}^\wedge\lambda x\ Y(x,b,s)] \rightarrow P = {}^\wedge\lambda x\ \text{introduce}'(x,b,s))$

These representations serve to make it clear that the only difference between (1)a through c is the domain of quantification. Note, however, that (4)c is nonsensical: if there is only one three place relation true of John, Bill and Sue, it means that John, Bill and Sue are the only members of the domain. We will come back to this important point below.
In (2), the P-set plays a slightly different role; the implicature is defined as the proposition that the intersection of the properties of John and the P-set is not empty:

(5) "(2)a" $\twoheadrightarrow$ $\exists P \exists y(P = {}^\wedge\lambda x\ \text{introduce}(x,y,s) \wedge P\{j\})$

The function of the focus in (3) is quite puzzling: in Rooth (1992) it is described as the expression that the focused item was chosen from the P-set. However, couldn't that be said of the unfocused counterpart as well? If *Bill* wasn't chosen in *John introduced Bill to Sue*, it would have been another sentence. In the traditional presupposition analysis of (3) we are stuck with the problem why (the utterance of) the sentence would presuppose what is being said. The existential presupposition is entailed by (3)a, but by its unfocused counterpart as well. It may be clear that these kinds of "explanations" are not satisfactory at all. We will come back to this further on.
The basic idea of AS then is that focused sentences are interpreted relatively to the P-set. The P-set is a set of semantic entities of the type of the argument of the focus operator (like *only*), and varying over the domain of the focused element.

3. Discussion of AS

AS claims to be a general theory of focus. However, as already pointed out above, it has little to say about the *pragmatics* of BF constructions. Moreover, despite its intentions, it turns out that semantically the force of AS is limited to focused proper names only. The problems that arise with sentences like

(6) John only *swims*

are typical for the whole approach. We will, therefore, discuss this point more extensively here.

The intuitive, but clearly incorrect translation of (6) is (7):

(7) $\forall P\ ([P \in ||\hat{}swim'||] \wedge P\{j\}] \rightarrow P = \hat{}swim)$[2]

The P-set of (6) are all functions $\{f | f(w,g) \in D_{(s,(e,t))}\}$[3]
This set contains, among others, trivial properties like $[\hat{}(\text{to be oneself})]^M$, $[\hat{}(\text{to be John})]^M$.... In order ot make sense of (6), we should ban these from the P-set. Let us define trivial properties as

(8) $T = \{f | \text{for all } (w,g): f(w,g)(w)(j) = 1\}$

Moreover, if John swims, it is also true that he swims or plays tennis. We can easily avoid this problem by demanding that every property John has is a logical consequence of swim. We get, then,

(9) (6) iff $\forall P([P\{j\} \wedge P \in (||\hat{}swim'|| - T)] \rightarrow \Box(swim'(j) \rightarrow P\{j\}))$

This, however, cannot be correct either. There will always be properties in the P-set of ^swim' which are neither logical consequences of ^swim' nor necessary properties, but which intuitively do not influence the truth conditions of (6). The kind of properties I am referring to are contingently present in the model, and may even not be the denotation of a word in the language. E.g.: Let D_e be {j,b,s} and let f be a function such that

for any w,g, $f(w,g)(w_0) = \{j,b\}$
$f(w,g)(w_1) = \{b\}$
.
.
.

Certainly, $f \in ||\hat{}swim||$. But the existence of such an f would intuitively not

---

2 This formula is a mixture of syntax and semantics, but its intention will be clear. For a proper syntactization of the concept P-set, *vide* de Mey (1993) or Blok (1993).

3. Cf. note 2.

make (6) false, as long as f is not the denotation of properties like ^walk, ^play_tennis etc. Therefore, under this analysis, (6) is true iff D_e = {j}. And that is clearly not what we want.
Note that similar problems arise when we try to compute the generalized conversational implicatures and presuppositions of (10)a and b respectively:

(10) a John doesn't *swim*
b John *swims*

∃PP{j} is not very spectacular as implicature, nor as presupposition.
In order to avoid these and related problems, Rooth (1985,1992) proposes to restrict the P-set by some *context* set, that is, a set of contextually given alternatives. The proposed interpretation of (6) then becomes

(11) $\forall P([P \in (||\hat{}swim|| \cap C) \wedge P\{j\}] \rightarrow P = \hat{}swim)$

But then, C will always be a (proper) subset of the P-set. Hence, $||\hat{}swim|| \cap C$ = C. Therefore, there is no point computing the P-set at all. If we are to give a semantics of sentences like (6) or a pragmatics for (10), we will have to concentrate on context sets, not on P-sets. An even more striking point is that the proper alternative of a focused word may be of a different *type* than the word itself, as in

(12) John didn't *introduce* Bill to Sue, he just *talked* to them!

Introduce is a three place relation, whereas *talk_to* is a two place relation. Nevertheless, the example (12) is very well possible and shows that in its context, *talk_to* is among the alternatives of *introduce*.

4. Awareness sets

From the above, it follows that there is no way to control the semantically defined set of alternatives. We will to have abandon the approach and look for a more *syntactic* alternative. The problem at hand has strong resemblances with the so called problem of logical omniscience. In epistemic logic, one often wants to conclude that an agent knows or believes a proposition without his knowing or believing all of its logical consequences. Here, the modalities are restricted to given *awareness sets* of "relevant" propositions. These sets are syntactic in the sense that A(p) is true just if p $\in A$. Hence, A is a set of syntactic objects (cf. Fagin and Halpern 1988). Moreover, if $\models p \rightarrow q$, and $\models A(p)$, it is not necessarily the case that A(q), etc. The awareness sets we need have slightly different properties. Since we have dealt with the problem of logical consequence by the adjustment of the consequent, as in (9), we can admit anything logical derivable from the "core" elements. The problem is, however, how these "core" elements enter the awareness set. Notice that this is indeed a logical problem like the classical problem of (un)awareness: alternatives are defined as objects of a certain semantic type. These can be computed automatically from the model. Given one alternative, all the others follow immediately. This automatic procedure should somehow be limited, just as the "automatic generation" of logical consequences in the classical case. Let us call this problem A. Problem B is the following. Let a speaker utter (6) with the background assumption that the relevant set of propositions is {John swims, John bikes, John walks}. Let a hearer interpret this sentence with the background assumption that the relevant set is {John swims, John bikes, John works}. And let it actually be true that John works, besides that he swims. What about the truth of the sentence in such a model? The sentence is true with respect to the awareness set of the speaker, whereas it is false with respect to the awareness set of the hearer. One could say, of course, that the awareness set is a variable of the truth relation. But this would not provide us with a more general description of the use of sentences like (6) in discourse. That is, it is not very satisfactory to neglect two party situations by the argument that they cannot straightforwardly be described as a special case of one party situations. As we will argue below, the situation is just the other way around.

A.

In the literature, proposals have been made to define the *syntactic* property of "alternative" in a lexical fashion. Gabbay and Moravcsik (1978) thereto introduced a lexical semantics based on the notions "core meaning" and "shades of meaning". Intuitively, the core meaning of a word consists of its formal properties like logical type together with more lexical semantic properties like the sort of arguments it takes, the meaning postulates it is subject to etc. The shades of meaning make a word to what it is: *swim* and *walk* have the same core meaning,

but they do not have the same shades of meaning. A *range of incompatibles* is a set of words which have the same core meaning, but a different shades of meaning. It seems as if this range of incompatibles is exactly what we are after for our awareness set. The disappointment is that the construction of ranges of incompatibles is not compositional. Take for example the intransitive verb "to eat apples" in *John only eats apples*. What is its range of incompatibles? It is not lexically given, since it is a complex expression. The range of incompatibles of *eat* will be something like {drink, chew...} and the range of incompatibles of *apple* will be {banana, lime,...}. But then, it may be clear that the awareness set we are looking for does not contain *drink a banana*. The relevant word may be *work* or *kiss a woman*, who knows. But it certainly has nothing to do with the subconstituents of the focused expression. As such, this problem seems unsolvable to me from a logical or linguistic point of view. It is a problem of AI.
But there is more. Compare the sentences (13) and (14):

(13) John only *walks and talks*
(14) John only *walks or talks*

As pointed out above, there is no problem admitting *John walks* and *John talks* to the awareness set in (13). If John walks and talks, he necessarily walks and he equally talks. But what is the meaning of (14)? If John walks, does it make the sentence false? Not so, in my opinion. On the other hand, if the speaker *knows* that John walks, (14) is a pragmatically awkward expression. Hence, what a hearer of (14) pragmatically infers is (among other things) the following[4];

(15) $\forall p\ [(L_s p \wedge A_s p) \rightarrow (p = \text{John walks or talks})]$

that is, *John walks or talks* is the only *explicit* knowledge of the speaker. And here, *John walks* and *John talks* are certainly members of A. Pragmatically, we may then assume that awareness sets are closed under sub-formulas. But we should keep semantics and pragmatics strictly apart.

B.

---

[4] L should be taken here KD45; cf. Blok 1993.

As we have seen, (6) should be interpreted relative to an awareness set. A speaker will, of course, think that it is clear which set (s)he is aware of. This boils down to the assumption

(16) $A_s = A_h$

(s stands for speaker, h for hearer). One could say that if the speaker is not entertaining this assumption, the locution is not felicitous. On the other hand, the speaker may err with respect to this. What about the truth conditions in such a case? Although the locution is more or less a misfire, this does not imply it should not have a meaning. In my view, the situation should be modeled as follows: if the sentence is true in $A_s \cup A_h$, the sentence is true. If there is a counter example in $A_s \cap A_h$, it is false. If there is a counter example in A_h which is not in A_s or the other way around[5], the sentence is true nor false.

Standard partial logic provides us with the following possibility: given the truth table

φ	ψ	$\varphi \wedge \psi$	φ xx ψ	$\varphi \rightarrow \psi$
1	1	1	1	1
1	*	*	*	*
1	0	0	*	0
*	1	*	*	1
*	*	*	*	*
*	0	0	*	*
0	1	0	*	1
0	*	0	*	1
0	0	0	0	1

(Blamey 1986, p.7)

and "$\forall$" also in the usual partial way:

[5] This last situation is pragmatically unlikely because of the quality maxim which says: try to make your contribution one that is true!

$M,g \models \forall p\ \varphi$ iff $M,g_{p/a} \models \varphi$ for any $a \in D(M)$
$M,g \dashv \forall p\ \varphi$ iff for some $a \in D(M)$, $M,g_{p/a} \dashv \varphi$

we can translate the sentence as

(17) $\forall p\ ([(A_s(p) \text{ xx } A_h(p)) \wedge {}^{\vee}p] \rightarrow p = {}^{\wedge}\text{John swims})$

For instance, let $A_s(p')$ be true, i.e. $p' \in A_s$, and let $A_h(p')$ be true also, i.e $p' \in A_h$. Moreover, let $[\![{}^{\vee}p]\!] = 1$. Then, the antecedent of (17) is true for p'. If $p' \neq {}^{\wedge}$John swims, (17) is false, as required.
Now, let $A_s(p')$ be true and $A_h(p')$ be false; this is the situation in which the awareness sets of speaker and hearer do not match. The first conjunct of the antecedent of (17) is *. If $[\![{}^{\vee}p]\!] = 1$, the whole antecedent is *. If $p' = {}^{\wedge}$John swims, it confirms (17). If $p' \neq {}^{\wedge}$John swims, (17) is *. If $[\![{}^{\vee}p]\!] = 0$, the antecedent becomes false, and hence the implication true.
What we see, then, is that for (17) to be true, it has to be true for all $p \in A_s \cup A_h$, i.e. for all $p \in A_s \cup A_h$, either ${}^{\vee}p$ must be false or $p = {}^{\wedge}$John swims. However, for (17) to be false, the counter example should be in $A_s \cap A_h$.
It is important to observe that if the meaning assignment function itself is total, the approach collapses into a 2 valued system if the assumption (16) turns out to be true: if $A_s = A_h$, $A_s(p)$ xx $A_h(p)$ equals $A_s(p)$ for all assignments. This is, of course, exactly what we want: the speaker assumes that (s)he is "relevant", and that the sentence is unambiguously true. The one agent case is, then, a special case of two (or more) agent case.
There may be an unsatisfactory aspect to this analysis: the syntactic representation of the sentence differs with the number of participant in the discourse. It may be clear that if there are two hearers (17) would be altered into $\forall p\ ([((A_s(p) \text{ xx } A_{h1}(p)) \text{ xx } A_{h2}(p)) \wedge {}^{\vee}p] \rightarrow p = {}^{\wedge}\text{John swims})$, etc. What we may prefer is a *unique* representation which interpretation differs along the number of awareness sets involved. A way to achieve this is the following simple semantics, amended with *restrictive quantification*:

Let M be <D,W,V>

Let p,q be constants of L. Then

$M,w,g \models p$ iff $V(p,w) = 1$

$M,w,g \models \neg p$ iff $M,w,g \not\models p$

$M.w.g \models p \wedge q$ iff $M,w,g \models p$ and $M,w,g \models q$

Let p be a variable of L

$M,w,g \models p$	iff $g(p)(w) = 1$
$M,w,g \models \forall p\ q$	iff for all g' which differ from g at most with respect to p, $M,w,g' \models q$

Restrictive quantification

Let p be an unquantified formula of L

$M,w,g \models^{\psi} p$ iff $M,w,g \models p$

$M,w,g \models^{\psi} \forall p\ q$ iff for all assignments g' which differ from g at most with respect to p such that there is a $r \in \psi$ such that for all w, $V(r,w) = g'(p)(w)$, $M,w,g' \models^{\psi} q$

hence,

$M,w,g \dashv^{\psi} \forall p\ q$ iff there is an assignment g' which differs from g at most with respect to p such that there is a $r \in \psi$ such that for all w, $V(r,w) = g'(p)(w)$, $M,w,g' \dashv^{\psi} q$

Two agents:

$M,w,g \models^{\psi,\varphi} \forall p\ q$ iff for all assignments g' which differ from g at most with respect to p such that there is a $r \in (\psi \cup \varphi)$ such that for all w, $V(r,w) = g'(p)(w)$, $M,w,g' \models^{\psi} q$

and

$M,w,g \dashv^{\psi,\varphi} \forall p\, q$ iff there is an assignment g' which differs from g at most with respect to p such that there is a $r \in (\psi \cap \varphi)$ such that for all w, $V(r,w) = g'(p)(w)$, $M,w,g' \dashv^{\psi} q$

<u>Box</u>

$M,w,g, \vDash^{\psi} \Box(p)$ iff for all w', $M,w',g \vDash p$

Now we will check whether we get the intended truth conditions. First for the one agent case:

$M,w,g \vDash^{\psi} \forall p\, [p \rightarrow \Box(swim(j) \rightarrow p)]$

iff for all assignments g' such that $g'(p) \in \psi$[6], $g'(p)(w) = 0$ or for all w', $V(swim(j),w') = 0$ or $g'(p)(w') = 1$

Second for the two agent case

$M,w,g \vDash^{\psi,\varphi} \forall p\; p \rightarrow \Box(swim(j) \rightarrow p)$

iff ..... such that $g'(p) \in \psi \cup \varphi$[7] ...

$M,w,g \dashv^{\psi,\varphi} \forall p\; p \rightarrow \Box(swim(j) \rightarrow p)$

iff there is an assignment g' such that $g' \in \psi \cap \varphi$ and $g'(p)(w) = 1$ and there is a w' such that $V(swim(j),w') = 1$ and $g'(p)(w') = 1$

This is indeed what we had in mind.
Note that the two agent case still collapses into the (2 valued) one agent case if

6. This abbreviates the more correct formulation in the definitions above.

7 cf. note 6.

assumption (16) turns out to be true.

References

Blamey, S (1986) "Partial Logic" in: D.Gabbay and F.Guentner (eds) *Handbook of Philosophical Logic* vol. III, 1-70

Blok, P.I. (1993) *The interpretation of focus; an epistemic approach to pragmatics*, doct.diss. University of Groningen

Fagin, R.F and J.Y.Halpern (1988), "Belief, Awareness and Limited Reasoning", *Artificial Intelligence* 34, 39-76

Gabbay, D.M. & J.M. Moravcsik (1978), "Negation and Denial", in: Guenthner,F and C.Rohrer (eds), *Studies in Formal Semantics*, Amsterdam: North Holland (1978), 251-265

Mey, J. de (1993), "P-sets revisited", in: K.Bimbó and A.Maté (eds) *Proceedings of the 4th symposium on logic and language*, Budapest: Áron Publishers, 179-198

Rooth, M.E. (1985), *Association with Focus*, PhD. diss. University of Massachusetts at Amherst

Rooth, M.E. (1992), "A Theory of Focus Interpretation", *Natural Language Semantics*, 1,1 , 75-117

Thijsse, E.G.C. (1992), *Partial Logic and Knowledge Representation*, Delft: Eubron Publishers

Queries, Rules and Definitions as Epistemic Sentences in Concept Languages

Francesco M. Donini* Maurizio Lenzerini*
Daniele Nardi* Werner Nutt† Andrea Schaerf*

Abstract

Concept languages have been studied in order to give a formal account of the basic features of frame-based languages. The focus of research in concept languages was initially on the semantical reconstruction of frame-based systems and the computational complexity of reasoning. More recently, attention has been paid to the formalization of other aspects of frame-based languages, such as non-monotonic reasoning and procedural rules, which are necessary in order to bring concept languages closer to implemented systems. In this paper we discuss the above issues in the framework of concept languages enriched with an epistemic operator. In particular, we show that the epistemic operator both introduces novel features in the language, such as sophisticated query formulation and closed world reasoning, and makes it possible to provide a formal account for some aspects of the existing systems, such as rules and definitions, that cannot be characterized in a standard first-order framework.

1 Introduction

Structured or taxonomical representations of knowledge have been studied in Artificial Intelligence with the aim of providing for both a compact representation and efficient reasoning methods. Semantic networks and frames are well known examples of this kind of knowledge representation languages.

Concept languages (also called terminological languages or description logics) have been studied for several years in order to provide a formalization of structured knowledge representation languages and to analyze the computational properties of the associated reasoning tasks [3, 5, 6, 19, 28]. However, concept languages are given a set-theoretic first-order semantics and leave out several aspects of practical systems. Therefore, it seems now appropriate to enrich such languages both to explore novel language features and to account for some of those aspects that cannot be easily described in a standard first-order framework.

This need is discussed in the literature (see for example [11, 29]) and can be easily recognized by looking at recent knowledge representation systems based on concept languages

*Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", Via Salaria 113, I-00198 Roma, Italy (email: {donini, lenzerini, nardi, aschaerf}@assi.ing.uniroma1.it)
†German Research Center for Artificial Intelligence (DFKI), Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany (email: nutt@dfki.uni-sb.de)

such as [2, 30]. Work in this direction has already begun with proposals of extending concept languages to deal with different forms of non-monotonic reasoning (see for example [1, 23]).

We proposed in [7] to enrich concept languages with an epistemic operator defined in the style of [14, 15, 25]. While the main emphasis of that paper was to show that answering queries formulated in the epistemic concept languages can be done by extending the calculus for instance checking developed in [10], here we aim at discussing in more detail the advantages provided by such an extension both for enhancing the capabilities of concept languages, and for formalizing non-standard features of existing systems.

In particular, we focus our attention on the use of the epistemic operator in order (1) to define a more powerful query language; (2) to be able to formulate queries requiring some forms of closed world reasoning; (3) to formalize the notion of procedural rule; (4) to precisely characterize weak forms of concept definition. All these aspects show that the epistemic operator turns out to be flexible enough to account for several different notions in an elegant and uniform way.

With regard to Point (1), we provide several examples that show how the new query language allows one to address both aspects of the external world as represented in the knowledge base, and aspects of what the knowledge base knows about the external world. It is worth noting that one advantage of the extended query language is the formalization of integrity constraints, which are viewed as sentences referring to what the knowledge base knows about the world (see [7]). This aspect, however, is not further discussed in the present paper.

With regard to Point (2), we show that a careful usage of the epistemic operator allows one to express queries whose processing forces the system to assume complete knowledge about (part of) the knowledge base. Note that this approach is different from assigning a closed world semantics to the knwoledge base itself. In fact, the nonmonotonicity is not in the semantics of the knowledge base, but a form of nonmonotonic reasoning is achieved by the system when answering special kinds of queries.

Points (3) and (4) are concerned with the formalization of two important features of some existing systems. In particular, systems like [2, 12, 22, 30] include suitable structures for the representation of procedural rules, enabling both behavioral models of objects and expertise in an application domain to be expressed. We propose to express procedural rules as special epistemic sentences in the knowledge base. While procedural rules are usually defined informally in existing systems, we show that a nice formalization of these features can be achieved in our framework, thus clarifying both their semantics, and their interaction with the other parts of the knowledge base. Moreover, we show that epistemic sentences provide an account for weak forms of concept definitions similar to those found in LOOM [16] and other systems. This formalization makes it clear that weak definitions provide a form of incomplete reasoning that is both computationally advantageous, and semantically well founded.

The paper is organized as follows. Section 2 recalls the basic notions about the concept language $\mathcal{ALC}$, which is a powerful concept language (including concept conjunction, disjunction, negation, as well as existential and universal quantification of roles), together with its usage in the definition of knowledge bases. Section 3 presents the epistemic concept language $\mathcal{ALCK}$, obtained by adding an epistemic operator to $\mathcal{ALC}$. Section 4 elaborates on the features of $\mathcal{ALCK}$ when used as a query language over knowledge bases expressed in $\mathcal{ALC}$. Section 5 focuses on some forms of closed world reasoning that can be expressed with the epistemic operator. Section 6 proposes a formalization of procedural rules as special classes of epistemic sentences, while Section 7 discusses the use of epistemic sentences in expressing weak forms of concept inclusions and definitions. Finally, conclusions are drawn in Section 8.

2 Concept Knowledge Bases

We make use of the concept language $\mathcal{ALC}$ (see [5, 28]) to define a knowledge base.[1] Like any concept language, $\mathcal{ALC}$ allows one to express the knowledge about the classes of interest in a particular application through the notions of *concept* and *role*. Intuitively, concepts represent the classes of objects in the domain to be modeled, while roles represent relationships between objects. Starting with primitive concepts and roles, one can construct complex expressions by means of various concept forming operators.

The syntax and semantics of $\mathcal{ALC}$ are as follows. We assume that two alphabets of symbols, one for *primitive concepts*, and one for *primitive roles*, are given. The letter A will always denote a primitive concept, and the letter P will denote a role, which in $\mathcal{ALC}$ is always primitive. The *concepts* (denoted by the letters C and D) of the language $\mathcal{ALC}$ are built out of primitive concepts and primitive roles according to the syntax rule:

$$
\begin{array}{rcll}
C, D & \longrightarrow & A \mid & \text{(primitive concept)} \\
 & & \top \mid & \text{(top)} \\
 & & \bot \mid & \text{(bottom)} \\
 & & C \sqcap D \mid & \text{(intersection)} \\
 & & C \sqcup D \mid & \text{(union)} \\
 & & \neg C \mid & \text{(complement)} \\
 & & \forall P.C \mid & \text{(universal quantification)} \\
 & & \exists P.C & \text{(existential quantification).}
\end{array}
$$

We use parentheses whenever we have to disambiguate concept expressions. For example, we write $(\exists P.D) \sqcap E$ to indicate that the concept E is not in the scope of $\exists P$.

An *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a nonempty set $\Delta^{\mathcal{I}}$ (the *domain* of $\mathcal{I}$) and a function $\cdot^{\mathcal{I}}$ (the *interpretation function* of $\mathcal{I}$) that maps every concept to a subset of $\Delta^{\mathcal{I}}$ and every role to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ such that the following equations are satisfied:

$$
\begin{array}{rcl}
\top^{\mathcal{I}} & = & \Delta^{\mathcal{I}} \\
\bot^{\mathcal{I}} & = & \emptyset \\
(C \sqcap D)^{\mathcal{I}} & = & C^{\mathcal{I}} \cap D^{\mathcal{I}} \\
(C \sqcup D)^{\mathcal{I}} & = & C^{\mathcal{I}} \cup D^{\mathcal{I}} \\
(\neg C)^{\mathcal{I}} & = & \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\
(\forall P.C)^{\mathcal{I}} & = & \{d_1 \in \Delta^{\mathcal{I}} \mid \forall d_2 : (d_1, d_2) \in P^{\mathcal{I}} \rightarrow d_2 \in C^{\mathcal{I}}\} \\
(\exists P.C)^{\mathcal{I}} & = & \{d_1 \in \Delta^{\mathcal{I}} \mid \exists d_2 : (d_1, d_2) \in P^{\mathcal{I}} \wedge d_2 \in C^{\mathcal{I}}\}.
\end{array}
$$

[1] Although we restrict our attention to $\mathcal{ALC}$, our framework can be applied to other languages as well.

An interpretation $\mathcal{I}$ is a *model* for a concept C if $C^{\mathcal{I}}$ is nonempty. A concept is *satisfiable* if it has a model and *unsatisfiable* otherwise. We say that C is *subsumed* by D if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every interpretation $\mathcal{I}$.

In terminological systems, the knowledge base includes both an intensional part, called terminology or simply *TBox*, and an extensional part, called assertional box or simply *ABox*. The TBox is constituted by a set of *inclusion statements* of the form

$$C \sqsubseteq D$$

where C, D are concepts. Inclusion statements are interpreted in terms of set inclusion: an interpretation $\mathcal{I}$ satisfies $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. An interpretation $\mathcal{I}$ is a *model* for a TBox if it satisfies all of its inclusions. As pointed out in [4], inclusions are more general than definitions, since definitions like $A \doteq C$ can be expressed as $A \sqsubseteq C$ and $C \sqsubseteq A$. Moreover, cyclic definitions are admitted and interpreted by the descriptive semantics [19].

The ABox is constituted by a set of assertions that specify either that an individual is instance of a concept or that a pair of individuals is instance of a role. Let $\mathcal{O}$ be an alphabet of symbols, called *individuals*. Syntactically, assertions are expressed in terms of *membership statements*, of the form

$$C(a)$$
$$P(a, b)$$

where a and b are individuals, C is a concept, and P is a role. $C(a)$ means that a is an instance of C, while $P(a, b)$ means that a is related to b by means of P. In order to give a formal semantics to assertions,the interpretation must be enriched with an injective function from $\mathcal{O}$ to $\Delta^{\mathcal{I}}$, i.e. each individual is associated with a unique domain element (Unique Name Assumption). Therefore an interpretation is now a triple $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, \gamma^{\mathcal{I}})$, and an assertion $C(a)$ is satisfied by $\mathcal{I}$ if $\gamma^{\mathcal{I}}(a) \in C^{\mathcal{I}}$. Similarly, an assertion $P(a, b)$ is satisfied by $\mathcal{I}$ if $(\gamma^{\mathcal{I}}(a), \gamma^{\mathcal{I}}(b)) \in P^{\mathcal{I}}$.

To summarize we define an $\mathcal{ALC}$-knowledge base as follows:

Definition 1 *An $\mathcal{ALC}$-knowledge base is a pair $\Sigma = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T}$ is a set of inclusion statements, and $\mathcal{A}$ is a set of membership assertions, whose concepts and roles belong to the language $\mathcal{ALC}$. An interpretation $\mathcal{I}$ is a* model *for $\Sigma = \langle \mathcal{T}, \mathcal{A} \rangle$ if it is a model for both $\mathcal{T}$ and $\mathcal{A}$.*

We say that Σ is *satisfiable* if it has a model. The set of models of Σ is denoted as $\mathcal{M}(\Sigma)$. Σ *logically implies* σ (written $\Sigma \models \sigma$), where σ is either an inclusion statement or a membership assertion, if every model in $\mathcal{M}(\Sigma)$ satisfies σ.

The most common kind of query to a knowledge base Σ is asking whether $C(a)$ (or $P(a, b)$) is logically implied by Σ. Notice that the semantics associated with concept languages is an open world semantics: the answer to a query Q will be `YES` if Q is true in every model for Σ, `NO` if Q is false in every model, and `UNKNOWN` otherwise.

It is well known (see for example [4]) that query answering in $\mathcal{ALC}$-knowledge bases is reducible to satisfiability. A calculus for knowledge base satisfiability in $\mathcal{ALC}$ is presented in [9] and shown to be complete and terminating.

3 An Epistemic Concept Language

In this section we present the epistemic concept language $\mathcal{ALCK}$, previously introduced in [7], which is an extension of $\mathcal{ALC}$ with an epistemic operator. Generally speaking, we follow [25], and use $\mathbf{K}C$ to denote the set of individuals *known* to be instances of the concept C in every model for the knowledge base. The syntax of $\mathcal{ALCK}$ is the following (where C, D denote concepts, R denotes a role, A denotes a primitive concept and P a primitive role):

$$
\begin{array}{rcll}
C, D & \longrightarrow & A \mid & \text{(primitive concept)} \\
 & & \top \mid & \text{(top)} \\
 & & \bot \mid & \text{(bottom)} \\
 & & C \sqcap D \mid & \text{(intersection)} \\
 & & C \sqcup D \mid & \text{(union)} \\
 & & \neg C \mid & \text{(complement)} \\
 & & \forall R.C \mid & \text{(universal quantification)} \\
 & & \exists R.C \mid & \text{(existential quantification)} \\
 & & \mathbf{K}C & \text{(epistemic concept)} \\
R & \longrightarrow & P \mid & \text{(primitive role)} \\
 & & \mathbf{K}P & \text{(epistemic role).}
\end{array}
$$

The semantics of $\mathcal{ALCK}$ is an adaptation to the framework of concept languages of the one proposed in [14, 15, 25]. As in the cited papers, some issues typical of first-order modal systems arise. Such issues concern the interpretation structures and are dealt with by the following assumptions:

- every interpretation is defined over a fixed domain, called Δ (Common Domain Assumption);
- for every interpretation the mapping from the individuals into the domain elements, called γ, is fixed (Rigid Term Assumption).

An *epistemic interpretation* is a pair $(\mathcal{I}, \mathcal{W})$ where $\mathcal{I}$ is an interpretation and $\mathcal{W}$ is a set of interpretations such that the following equations are satisfied:

$$
\begin{array}{rcl}
\top^{\mathcal{I},\mathcal{W}} & = & \Delta \\
\bot^{\mathcal{I},\mathcal{W}} & = & \emptyset \\
A^{\mathcal{I},\mathcal{W}} & = & A^{\mathcal{I}} \\
P^{\mathcal{I},\mathcal{W}} & = & P^{\mathcal{I}} \\
(C \sqcap D)^{\mathcal{I},\mathcal{W}} & = & C^{\mathcal{I},\mathcal{W}} \cap D^{\mathcal{I},\mathcal{W}} \\
(C \sqcup D)^{\mathcal{I},\mathcal{W}} & = & C^{\mathcal{I},\mathcal{W}} \cup D^{\mathcal{I},\mathcal{W}} \\
(\neg C)^{\mathcal{I},\mathcal{W}} & = & \Delta \setminus C^{\mathcal{I},\mathcal{W}} \\
(\forall R.C)^{\mathcal{I},\mathcal{W}} & = & \{d_1 \in \Delta \mid \forall d_2. (d_1, d_2) \in R^{\mathcal{I},\mathcal{W}} \rightarrow d_2 \in C^{\mathcal{I},\mathcal{W}}\} \\
(\exists R.C)^{\mathcal{I},\mathcal{W}} & = & \{d_1 \in \Delta \mid \exists d_2. (d_1, d_2) \in R^{\mathcal{I},\mathcal{W}} \wedge d_2 \in C^{\mathcal{I},\mathcal{W}}\} \\
(\mathbf{K}C)^{\mathcal{I},\mathcal{W}} & = & \bigcap_{\mathcal{J} \in \mathcal{W}} (C^{\mathcal{J},\mathcal{W}})
\end{array}
$$

$$(\mathbf{K}P)^{\mathcal{I},\mathcal{W}} = \bigcap_{\mathcal{J}\in\mathcal{W}} (P^{\mathcal{J},\mathcal{W}}).$$

Notice that, since the domain is fixed independently of the interpretation, it is meaningful to refer to the intersection of the extensions of a concept in different interpretations. It follows that $\mathbf{K}C$ is interpreted in $\mathcal{W}$ as the set of objects that are instances of C in every interpretation belonging to $\mathcal{W}$. In this sense, $\mathbf{K}C$ represents those objects *known* to be instances of C in $\mathcal{W}$. Notice also that if one discards $\mathbf{K}$ and $\mathcal{W}$ in the equations, one obtains the standard semantics of $\mathcal{ALC}$.

An $\mathcal{ALCK}$-knowledge base Ψ is a pair $\Psi = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T}$ is a set of inclusion statements and $\mathcal{A}$ is a set of membership assertions, whose concepts and roles belong to the language $\mathcal{ALCK}$. The truth of inclusion statements and membership assertions in an epistemic interpretation is defined in a straightforward way. An *epistemic model* for Ψ is a pair $(\mathcal{I}, \mathcal{W})$, where $\mathcal{I} \in \mathcal{W}$ and $\mathcal{W}$ is any maximal set of interpretations such that for each $\mathcal{J} \in \mathcal{W}$, every sentence (inclusion or membership assertion) of Ψ is true in $(\mathcal{J}, \mathcal{W})$.

Notice that the semantics of an $\mathcal{ALCK}$-knowledge base could be equivalently defined in terms of an accessibility relation on a set of possible worlds. More specifically, the constraints posed by the semantic equations on $\mathbf{K}C$ and $\mathbf{K}P$, correspond to a structure of possible worlds each one connected with all the others. Therefore, the accessibility relation would be an equivalence relation, as in the modal system S5. However, the epistemic models of a knowledge base correspond to S5 models with a maximal set of worlds. In particular, if Σ is an $\mathcal{ALC}$-knowledge base, i.e. it does not contain epistemic operators, then its epistemic models are all the pairs $(\mathcal{I}, \mathcal{M}(\Sigma))$ for every $\mathcal{I} \in \mathcal{M}(\Sigma)$.

An $\mathcal{ALCK}$-knowledge base Ψ is said to be *satisfiable* if there exists an epistemic model for Ψ, *unsatisfiable* otherwise. Ψ logically implies an assertion σ, written $\Psi \models C(a)$, if σ is true in every epistemic model for Ψ.

4 $\mathcal{ALCK}$ as a Query Language

In this section, we use $\mathcal{ALCK}$ as a query language to $\mathcal{ALC}$-knowledge bases. First of all we introduce the notion of epistemic query.

Definition 2 *Given an $\mathcal{ALC}$-knowledge base Σ, an $\mathcal{ALCK}$-concept C, and an individual a, the answer to the query $C(a)$ posed to Σ is* YES *if $\Sigma \models C(a)$,* NO *if $\Sigma \models \neg C(a)$, and* UNKNOWN *otherwise. Moreover, the* answer set *of C w.r.t. Σ is the set $\{a \in \mathcal{O}_\Sigma \mid \Sigma \models C(a)\}$, where $\mathcal{O}_\Sigma$ is the set of individuals appearing in Σ.*

To answer epistemic queries posed to an $\mathcal{ALC}$-knowledge base Σ one can check whether Σ plus the negation of the query is unsatisfiable. In [7], we defined a sound and complete calculus to answer epistemic queries to an $\mathcal{ALC}$-knowledge base consisting of an ABox only. Although such a calculus does not consider the TBox, it can be suitably extended in order to treat inclusion statements in the spirit of [4, 9]. We do not present the extended calculus in this paper. It is reported in [8], where the decidability of the problem of answering epistemic queries to an $\mathcal{ALC}$-knowledge base is proved and its computational properties are discussed.

Our goal here is to show that the use of epistemic operators in queries allows for a more sophisticated interaction with the a knowledge representation system. For this purpose we

AdvCourse $\doteq$ Course $\sqcap$ $\forall$ENROLLED.Grad,
BasCourse $\doteq$ Course $\sqcap$ $\forall$ENROLLED.Undergrad,
IntCourse $\doteq$ Course $\sqcap$ $\exists$ENROLLED.Grad $\sqcap$ $\exists$ENROLLED.Undergrad,
$\exists$TEACHES.Course $\sqsubseteq$ Grad $\sqcup$ Professor,
Grad $\doteq$ Student $\sqcap$ $\exists$DEGREE.Bachelor,
Undergrad $\doteq$ Student $\sqcap$ $\neg$Grad

The TBox $\mathcal{T}_1$

Professor(bob), TEACHES(bob, ee282), TEACHES(john, cs324),
TEACHES(john, cs221), Course(cs221), Course(cs324),
IntCourse(ee282), ENROLLED(ee282, peter), ENROLLED(cs221, mary),
ENROLLED(cs221, susan), ENROLLED(cs324, susan),
ENROLLED(cs324, peter), Undergrad(peter), Student(susan),
Student(mary), DEGREE(mary, bs), Bachelor(bs)

The ABox $\mathcal{A}_1$

Figure 1: The $\mathcal{ALC}$-knowledge base Σ_1

provide an example of an $\mathcal{ALC}$-knowledge base and discuss various kinds of queries that can be posed to it using the language $\mathcal{ALCK}$.

In Figure 1 we show an $\mathcal{ALC}$-knowledge base $\Sigma_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ describing information about a university. The TBox $\mathcal{T}_1$ contains information about the various classes of persons working in the university and the courses supplied by the university. We use $D \doteq C$ as a shorthand for $C \sqsubseteq D$ and $D \sqsubseteq C$. The ABox $\mathcal{A}_1$ keeps track of the actual persons and courses involved in the university, together with the relations between them. The ABox $\mathcal{A}_1$ is also shown in graph form in Figure 2.

It can be easily shown that Σ_1 is satisfiable and that it has several different models. In fact, it does not have complete knowledge about the represented world. For example, since EE282 is an intermediary course, Σ_1 knows that at least one graduate student is enrolled in EE282, but it doesn't know who she/he actually is. Similarly, Σ_1 knows that Susan is either a graduate or an undergraduate, without knowing which one.

Notice that the information in $\mathcal{T}_1$ plays a role in the deduction of properties of individuals in $\mathcal{A}_1$. For example, Σ_1 knows that Mary is a graduate student, because she has a bachelor's degree and thus, according to $\mathcal{T}_1$, she falls under the description of graduate student.

We consider now various $\mathcal{ALCK}$ queries directed to Σ_1. In particular, in order to understand the role of the epistemic operator **K**, we consider both $\mathcal{ALC}$ queries and modified versions of them including **K**. The comparison between their respective semantics highlights the role of **K** in the query language.

We start with a pair of queries involving one single existential quantifier:

- Query 1a: $\Sigma_1 \models \exists$ENROLLED.Grad(ee282) ? Answer: YES.

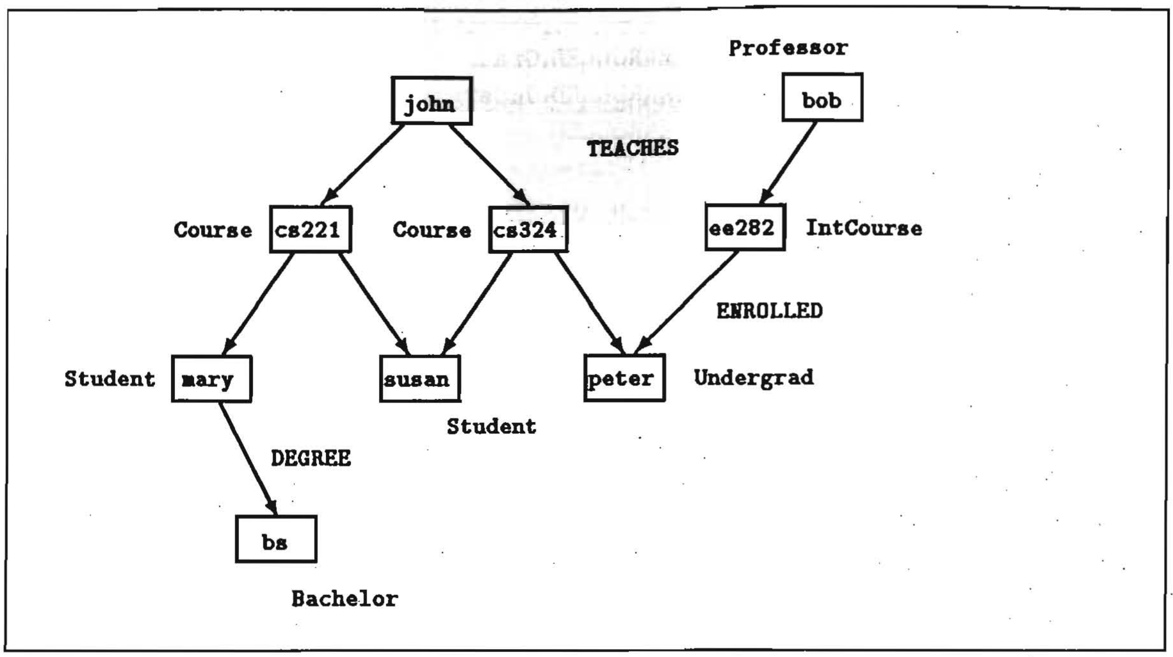


Figure 2: A pictorial representation of the ABox

- Query 1b: $\Sigma_1 \models \exists$KENROLLED.KGrad(ee282) ? Answer: NO.

Query 1a asks whether there is a graduate student enrolled in EE282. The answer is YES because EE282 is an intermediary course and therefore, according to $\mathcal{T}_1$, there is at least one graduate student enrolled in it. However, as we already mentioned, the name of the enrolled student is unknown. It might either be one of the individuals named in Σ_1 or a different one about whom no information is given. Moreover, it is not even ensured that it is the same one in all models.

On the other hand, Query 1b asks whether there exists an individual who is known both to be enrolled in EE282 and to be a graduate student. In other words, it asks for an individual, say fred, such that both the assertions ENROLLED(ee282, fred) and Grad(fred) hold in every model for Σ_1. Such an individual does not exist, thus the answer to the query is NO.

The next pair of queries shows the interaction of the epistemic operator with the disjunction constructor:

- Query 2a: $\Sigma_1 \models$ Grad $\sqcup$ Professor(john) ? Answer: YES.

- Query 2b: $\Sigma_1 \models$ KGrad $\sqcup$ KProfessor(john) ? Answer: NO.

Query 2a asks whether John is either a graduate student or a professor. The answer is YES, and it can be derived by the fact that it is stated in the ABox that he teaches two courses, and, according to the TBox, everybody who teaches at least one course is either a graduate student or a professor.

Query 2b, instead, asks whether he is either known to be a graduate student or known to be a professor. It is easy to verify that none of them is true and therefore the answer to this query is NO.

We consider now queries that involve also universal quantifiers:

- Query 3a: $\Sigma_1 \models \forall$TEACHES.(IntCourse $\sqcup \neg$Course)?(bob) Answer: UNKNOWN.
- Query 3b $\Sigma_1 \models \forall$KTEACHES.?K(IntCourse $\sqcup \neg$Course)(bob) Answer: YES.

Query 3a asks whether every course taught by Bob is an intermediary one. The answer is UNKNOWN because there are models for Σ_1 in which Bob teaches only intermediary courses as well as models in which he teaches also courses that are not intermediary.

Query 3b, instead, asks whether everything that is known to be taught by Bob is also known to be either an intermediary course or not to be a course. Since the only thing taught by Bob is EE282, and it is indeed an intermediary course, the answer to Query 3b is YES.

In the above example the addition of **K** has changed the answer from UNKNOWN to YES. Notice that it is also possible that Query 3a could be answered NO and Query 3b still be answered YES: Suppose that the assertion $\exists$TEACHES.AdvCourse (bob) is added to Σ_1 and then the same queries are asked. Query 3a now gets the answer NO, because AdvCourse and IntCourse are disjoint concepts. However, the set of known courses taught by Bob is not changed, and therefore the answer to Query 3b is still YES.

We now consider some queries involving nested quantifiers: Queries 4a and 4b involve two levels of existential quantification. The innermost quantifier is carried by the concept IntCourse, which has existential quantifiers in its definition in $\mathcal{T}_1$.

- Query 4a: $\Sigma_1 \models \exists$TEACHES.IntCourse(john) ? Answer: YES.
- Query 4b: $\Sigma_1 \models \exists$KTEACHES.KIntCourse(john) ? Answer: NO.

Query 4a asks whether John teaches an intermediary course. At a superficial reading of the query, it might seem that the answer should be NO. The answer NO is supported by the fact that none of the courses taught by John is known to be an intermediary course, i.e. neither IntCourse(cs221) nor IntCourse(cs324) is a logical consequence of Σ_1. Nevertheless, the correct answer is YES, and in order to get it, one must reason by *case analysis*: As we have already remarked, the knowledge base does not provide the information as to whether Susan is a graduate or an undergraduate; however, since she is a student, according to $\mathcal{T}_1$, she must either be one or the other. This fact ensures that in every model for Σ_1 either Grad(susan) or Undergrad(susan) holds. Consider now the set of models for Σ_1 in which Grad(susan) holds. In each of these models, the course CS324 is taken by both a graduate (Susan) and an undergraduate (Peter), thus it is an intermediary course. Similarly, consider the set of the remaining models for Σ_1, i.e. the ones in which Undergrad(susan) holds. It is easy to see that in every model for this set the course CS221, this time, is taken by both a graduate (Mary) and an undergraduate (Susan), and therefore it is an intermediary course.

In conclusion, in every model for Σ_1 either CS324 or CS221 is an intermediary course. It follows that in every model for Σ_1 John teaches an intermediary course, proving that the correct answer to Query 4a is YES.

On the other hand, Query 4b asks whether John is known to teach a course that is known to be an intermediary course. The courses known to be taught by John are CS221 and CS324 and the only known intermediate course is EE282, therefore none of them is within the conditions required by the query.

Query 4a shows how, in some cases, the first order semantics might not agree with the intuitive reading of a query. In fact, most people tend to read Query 4a as requiring the reasoning pattern that is actually associated with the semantics of Query 4b. In other words, they tend to rule out the case analysis from the computation. One good reason to do so is that case analysis generally makes reasoning harder. In fact, as proved in [26], the problem of answering queries with existential quantification under the first order semantics, is in general coNP-hard. Whereas, as shown in [7], queries involving existential quantification only of the form $\exists \mathbf{K}P.\mathbf{K}C$ can be answered in polynomial time w.r.t. the size of the knowledge base. However, there are also cases in which the intuition agrees with the first order interpretation. For this reason, in our opinion, it is important to have the operator K, which gives the possibility to choose between the two alternative readings of the query.

Regarding the interaction between the epistemic operator and the quantifiers, notice that we have always considered queries of the form $\exists \mathbf{K}P.\mathbf{K}C$ and $\forall \mathbf{K}P.\mathbf{K}C$, i.e. queries in which the K operator is placed in front of both the concept and the role. Such queries usually have an easy intuitive interpretation and therefore are the most interesting. Nevertheless, it might be worthwhile to consider even other possible variations of them, for example queries like $\exists \mathbf{K}P.C$ or $\forall P.\mathbf{K}C$. Such queries are perfectly legal in $\mathcal{ALCK}$, however, in some cases, they may lack an intuitive meaning. The reason is that they amalgamate $\mathcal{ALC}$-concepts with epistemic ones, resulting in something to which it is usually hard to give an intuitive meaning.

In other cases, though, they can play a useful role. As an example consider the following queries:

- Query 4c: $\Sigma_1 \models \exists$KTEACHES.IntCourse(john) ? Answer: YES.
- Query 4d: $\Sigma_1 \models \exists$TEACHES.KIntCourse(john) ? Answer: UNKNOWN.

Notice that Query 4c gets the same answer (YES) as Query 4a. In fact, since TEACHES(john, cs221) and TEACHES(john, cs324) are known, the addition of K in front of TEACHES does not change the answer to the query. Query 4d, instead, is answered UNKNOWN because the only known intermediate course is EE282 and we can neither prove nor exclude that John teaches it.

The fact that Query 4c gets the answer YES and Query 4d the answer UNKNOWN may help us understand the answers to Query 4a and 4b. In particular, it clarifies which is the actual reason that makes Query 4a and 4b different: It tells us that the incompleteness of the knowledge base is related to the concept IntCourse and not to the role TEACHES. In fact, TEACHES(john, cs324) and TEACHES(john, cs221) are both true in Σ_1, while IntCourse(cs324) and IntCourse(cs221) are not—only their disjunction is true.

5 Closed World Reasoning

The reason for the open world semantics of concept languages is that they are generally used in applications where one has to account for incomplete information. For example, even if all the known courses taught by Bob are intermediary, one does not want to conclude that all possible courses that Bob teaches are intermediary.

On the other hand, there are situations where it is natural to query a knowledge base under the Closed World Assumption. Referring to the knowledge base Σ_1 of Figure 1, consider the following examples:

- Query 5a: $\Sigma_1 \models \forall\mathtt{TEACHES}.\exists\mathtt{ENROLLED}.\top(\mathtt{john})$? Answer: UNKNOWN
- Query 5b: $\Sigma_1 \models \forall\mathbf{K}\mathtt{TEACHES}.\exists\mathtt{ENROLLED}.\top(\mathtt{john})$? Answer: YES

Query 5a gets the answer UNKNOWN because there is a model for Σ_1 where John teaches a course z, but there are no students enrolled in z, i.e. z is not an instance of the concept $\exists\mathtt{ENROLLED}.\top$. On the other hand, the correct reading of Query 5b is as follows: Is it true that for every course z that John is known to teach, there is at least one student enrolled in z? It is easy to see the answer to the query is YES.

The above example shows that the use of K allows one to pose queries to a knowledge base Σ asking the system to assume complete knowledge on a certain individual a and a certain role P in Σ (john and TEACHES in the example). In particular, assuming complete knowledge on a and P here means assuming that for every pair (a, b) such that $\Sigma \not\models P(a, b)$, the assertion $P(a, b)$ is false in Σ. It is clear that this kind of reasoning is a form closed world reasoning.

We show here that under certain restrictions, our query language allows us to achieve at least the expressive power of the (naive) Closed World Assumption (CWA) (see [24]). The restrictions affect both the content and the language of the knowledge base. We say a knowledge base is *simple* if it does not contain inclusion statements. In the following we consider simple knowledge bases where the ABox is expressed in the language $\mathcal{AL}_0$, whose concepts are formed according to the rule:

$$C, D \longrightarrow A \mid \neg A \mid C \sqcap D \mid \forall R.C.$$

More complex languages and knowledge bases and more powerful forms of closed world reasoning (e.g. Generalized CWA [18]) require a more sophisticated treatment, which is outside the scope of this paper.

We briefly reformulate the CWA in the setting of a simple $\mathcal{AL}_0$ knowledge base Σ. Let Σ^{CWA} be the knowledge base obtained from Σ by adding $\neg A(a)$ or $\neg P(a, b)$, respectively, for every assertion $A(a)$ or $P(a, b)$ that is not entailed by Σ. Now, for any concept C the statement $C(a)$ follows from Σ under the CWA, written $\Sigma \models_{\mathrm{CWA}} C(a)$, if $C(a)$ follows from Σ^{CWA}.

In the following, we assume that the $\mathcal{ALC}$-concepts used for querying a knowledge base are in *negation normal form*, i.e. negations signs are pushed down until they only occur in front of primitive concepts (see [28]). Given an $\mathcal{ALC}$-concept C in negation normal form, we define the $\mathcal{ALCK}$-concept $\overline{C}$ as follows:

$$\begin{aligned} \overline{A} &= \mathbf{K}A \\ \overline{\neg A} &= \neg\mathbf{K}A \\ \overline{C \sqcap D} &= \overline{C} \sqcap \overline{D} \\ \overline{C \sqcup D} &= \overline{C} \sqcup \overline{D} \end{aligned}$$

$$\overline{\exists P.C} = \exists \mathbf{K}P.\overline{C}$$
$$\overline{\forall P.C} = \forall \mathbf{K}P.\overline{C}.$$

The above transformation puts an epistemic operator in front of every primitive concept and primitive role. Now, it is possible to show that, if Σ is a simple $\mathcal{AL}_0$-knowledge base, C is an $\mathcal{ALC}$-concept, and a is an individual, then $\Sigma \models_{\mathrm{CWA}} C(a)$ if and only if $\Sigma \models \overline{C}(a)$. Moreover, checking whether $\Sigma \models \overline{C}(a)$ can be done in time polynomial in the size of both the query and the knowledge base. This is in sharp contrast to answering queries that are formulated with arbitrary $\mathcal{ALC}$-concepts, which is a PSPACE-hard problem even for a fixed $\mathcal{AL}_0$-knowledge base.

Intuitively, the reason for the above result is that for an $\mathcal{AL}_0$-concept C the assertion $C(a)$ is logically equivalent to a finite set of Horn clauses and, therefore, simple $\mathcal{AL}_0$ knowledge bases are equivalent to sets of Horn clauses. As a consequence, if such a knowledge base is satisfiable, it always has one minimum model, say $\mathcal{I}_0$. Hence, evaluating a query under the CWA amounts to evaluating it in $\mathcal{I}_0$. Now, putting a K in front of every primitive concept A and role P has the effect that A and P are taken as the intersection of their interpretations in all models of Σ, i.e., they are interpreted in $\mathcal{I}_0$. This explains why closed world reasoning can be enforced through the use of K. That queries can be answered in polynomial time is due to the fact that on the one hand the Horn clauses corresponding to a simple $\mathcal{AL}_0$-knowledge base do not contain function symbols and on the other hand that concepts have a hierarchical structure that makes them suitable for efficient bottom up evaluation.

Notice that transforming a query C into $\overline{C}$ implies answering the query under the assumption that the knowledge about every role is complete, like for example in [19, p. 113]. On the other hand, as noted in [13], there are situations where we would like to apply the closed world assumption only to some of the concepts and the roles of the knowledge base.

We argue that the use of epistemic operators as described in the previous sections is a natural way to achieve such a flexible way of interacting with the knowledge base. Indeed, the careful introduction of the epistemic operator into the query induces the system to answer queries under the assumption that *part* of the knowledge base is complete, in contrast to assigning a closed world semantics to the knowledge base itself.

Consider the following query to the knowledge base Σ_1 given in Section 4:

- Query 4e: $\Sigma_1 \models$ ∃KTEACHES.K(Course ⊓ ∃ENROLLED.Grad ⊓ ∃ENROLLED.(Student ⊓ ¬KGrad))(john) ? Answer: YES.

Notice that Query 4e is syntactically equal to Query 4b, except that the concept IntCourse is replaced by the $\mathcal{ALCK}$-concept

Course ⊓ ∃ENROLLED.Grad ⊓ ∃ENROLLED.(Student ⊓ ¬KGrad). (1)

Concept (1) differs from the definition of IntCourse in the fact that Undergrad is replaced by (Student ⊓ ¬KGrad). Concept (1) should be interpreted as the concept describing the courses that are intermediary under the assumption that every student is an undergraduate, unless the contrary is known. In fact, a course belongs to such a concept if both a graduate and a student not known to be a graduate are enrolled in it. It is easy to see that the course CS221 is an instance of Concept (1), and therefore the answer to Query 4e is YES.

Notice that asking queries like Query 4e is completely different from giving some kind of closed world semantics to the knowledge base. In fact, in our framework the knowledge base is perfectly monotonic, whereas using the epistemic operator the queries can be formulated in such a way that the reasoning which is required to compute the answers is nonmonotonic.

6 Rules as Epistemic Statements

In the previous sections we considered knowledge bases constituted by inclusions and membership assertions in $\mathcal{ALC}$. We now consider the case where epistemic sentences of a special kind are introduced into the knowledge base, and show that this extension formalizes the usage of procedural rules (or simply rules), as provided in many practical systems based on concept languages. In fact, systems such as CLASSIC [2] and LOOM [17], in addition to inclusions and membership assertions provide another mechanism for expressing knowledge, by means of so-called *rules*. Such rules are sentences the form

$$C \Rightarrow D$$

where C, D are concepts. The meaning of a rule is "if an individual is proved to be an instance of C, then derive that it is also an instance of D" (see [2]), and its behavior of rules is usually described in terms of a forward reasoning process that adds to the knowledge base the assertion $D(a)$ whenever $C(a)$ is proved to hold. We call *procedural extension* of a knowledge base Σ w.r.t. a set of rules the knowledge base resulting from such a forward reasoning process.

Rules in the context of frame-based systems are often defined informally. Attempts to precisely capture the meaning of such rules are based either on viewing them as knowledge base updates (see for example the *TELL* operation of [14]), or on ad hoc semantics (see [27]). Our aim in this section is to show that rules can be nicely formalized as particular epistemic sentences.

In the following we consider $\mathcal{ALCK}$-knowledge bases of the form $\langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T} = \mathcal{T}' \cup \mathcal{R}$ with $\mathcal{T}'$ being a set of $\mathcal{ALC}$-inclusion statements, and $\mathcal{R}$ a set of epistemic sentences, each one of the form[2]

$$\mathbf{K}C \sqsubseteq D$$

where C and D are $\mathcal{ALC}$-concepts. We call these sentences *trigger rules*, since they are our formal counterpart of the rules $C \Rightarrow D$. We also call C the antecedent and D the consequent of the trigger rule. As a notational convenience we write the $\mathcal{ALCK}$-knowledge base $\langle \mathcal{T}, \mathcal{A} \rangle$ as $\langle \Sigma, \mathcal{R} \rangle$, where $\Sigma = \langle \mathcal{T}', \mathcal{A} \rangle$.

From the definition of the semantics of $\mathcal{ALCK}$-knowledge bases it follows that an epistemic interpretation $(\mathcal{I}, \mathcal{W})$ satisfies the trigger rule $\mathbf{K}C \sqsubseteq D$ if $(\mathbf{K}C)^{\mathcal{I},\mathcal{W}} \subseteq D^{\mathcal{I},\mathcal{W}}$. Intuitively, the set of epistemic sentences $\mathcal{R}$ restricts the set of models for Σ to the maximal subsets that satisfy every trigger rule in $\mathcal{R}$. More precisely, it can be shown that if $(\mathcal{I}, \mathcal{W})$ is an epistemic model for $\Phi = \langle \Sigma, \mathcal{R} \rangle$, then $\mathcal{W}$ is a maximal subset of $\mathcal{M}(\Sigma)$ such that for

[2] In [7] we used the notation $\mathbf{K}C \Rightarrow \mathbf{K}D$. The two notations are equivalent in the semantics we give.

each $\mathcal{J} \in \mathcal{W}$, $(\mathcal{J}, \mathcal{W})$ satisfies every sentence in $\mathcal{R}$. Because of the form of such sentences, it can also be shown that there exists only one maximal subset $\mathcal{W}$ of $\mathcal{M}(\Sigma)$ such that for all $\mathcal{J} \in \mathcal{W}$, $(\mathcal{J}, \mathcal{W})$ satisfies every sentence in Φ.

Observe that when a concept C is equivalent to $\top$, i.e. $C^{\mathcal{I}} = \Delta$ for every interpretation $\mathcal{I}$, a trigger rule $\mathbf{K}C \sqsubseteq D$ is equivalent to the inclusion $\top \sqsubseteq D$. Besides this case, however, trigger rules are not expressible by inclusions. Indeed, the main difference between rules and inclusions is that the formers are intended to provide a reasoning mechanism which applies them in one direction only, namely from the antecedent to the consequent. Our formalization of rules with the epistemic operator correctly captures this property, as shown in the following example.

Consider the knowledge base $\Phi = \langle\langle\emptyset, \{\neg B(a)\}\rangle, \{\mathbf{K}A \sqsubseteq B\}\rangle$, and observe that there exists an epistemic model $(\mathcal{I}, \mathcal{W})$ of Φ such that $\gamma(a) \notin \neg A^{\mathcal{I}}$. Therefore, $\neg A(a)$ is not a logical consequence of Φ.

In order to characterize the notion of procedural extension we now introduce the concept of *first-order extension* of an $\mathcal{ALCK}$-knowledge base $\langle\Sigma, \mathcal{R}\rangle$. The first-order extension of $\Phi = \langle\Sigma, \mathcal{R}\rangle$, where $\Sigma = \langle\mathcal{T}, \mathcal{A}\rangle$, is the $\mathcal{ALC}$-knowledge base $\Sigma_{\mathcal{R}}$, which is the least solution (w.r.t. to set inclusion) of the following equations:

$$X = \langle\mathcal{T}', \mathcal{A}'\rangle$$

where

$$\begin{aligned} \mathcal{T}' &= \mathcal{T} \cup \{\top \sqsubseteq D \mid \mathbf{K}C \sqsubseteq D \in \mathcal{R} \text{ and } X \models \top \sqsubseteq C\} \\ \mathcal{A}' &= \mathcal{A} \cup \{D(a) \mid \mathbf{K}C \sqsubseteq D \in \mathcal{R} \text{ and } X \models C(a)\}. \end{aligned}$$

We do not delve into the details of the computation of the first-order extension. We simply remark that the solution of the above equations is unique and can be incrementally constructed starting from Σ in a number of steps which is polynomial w.r.t. the size of Φ.

First-order extensions are linked to the semantics by the following property. Let $\Phi = \langle\Sigma, \mathcal{R}\rangle$ be an $\mathcal{ALCK}$-knowledge base, let $(\mathcal{I}, \mathcal{W})$ be an epistemic model for Φ, and let $\Sigma_{\mathcal{R}}$ be the first-order extension of Φ. Then $\mathcal{W}$ coincides with the set of models for the $\mathcal{ALC}$-knowledge base $\Sigma_{\mathcal{R}}$. In other words, the result of the forward reasoning process on a knowledge base and set of trigger rules, which is represented by the least solution of the above equations, is correctly captured by the semantics of the $\mathcal{ALCK}$-knowledge base Φ, where the trigger rules are expressed as epistemic sentences.

We now show an example of the usage of rules in our framework. Consider the $\mathcal{ALCK}$-knowledge base $\Phi = \langle\Sigma, \mathcal{R}\rangle$:

$$\begin{aligned} \Sigma &= \langle\emptyset, \{\texttt{TEACHES(bill, cs248)}, \texttt{Grad(bill)}\}\rangle \\ \mathcal{R} &= \{\mathbf{K}\texttt{Grad} \sqsubseteq \forall\texttt{TEACHES.BasCourse}\}. \end{aligned}$$

The first-order extension of Φ is

$$\Sigma_{\mathcal{R}} = \langle\emptyset, \{\texttt{TEACHES(bill, cs248)}, \texttt{Grad(bill)}, \forall\texttt{TEACHES.BasCourse(bill)}\}\rangle.$$

Obviously, $\Sigma_{\mathcal{R}} \models \texttt{BasCourse(cs248)}$. From the semantics, one can verify that for every epistemic model $(\mathcal{I}, \mathcal{W})$ for Φ, we have $\gamma(\texttt{bill}) \in (\forall\texttt{TEACHES.BasCourse})^{\mathcal{I},\mathcal{W}}$ and

γ(cs248) $\in$ BasCourse$^{\mathcal{I},\mathcal{W}}$, i.e., both the assertion $\forall$TEACHES.BasCourse (bill) and BasCourse(cs248) are logical consequences of Φ, as one would expect.

It is worth noting that the calculus for answering epistemic queries, mentioned in Section 4, can be effectively used in the computation of the first-order extension of an $\mathcal{ALCK}$-knowledge base. In fact, the application of a trigger rule $\mathbf{K}C \sqsubseteq D$ requires to compute the answer set of the query $\mathbf{K}C$, which can be done by means of that calculus.

7 Weak Inclusions as Epistemic Statements

Recent studies on the formal properties of concept languages [4, 20, 21] show that one of the critical aspects of the implementation of knowledge representation systems based on concept languages is the treatment of inclusions. This problem is addressed for example in LOOM [16] by adopting a weak form of inclusion, which applies only to known individuals and disregards many inferences based on the use of contrapositives.

In this section we argue that the class of epistemic sentences used in the formalization of trigger rules can be regarded as a form of weak inclusion which may lead to significant computational advantages in comparison to inclusion statements as defined in Section 2.

To this purpose we introduce the notion of *weakening* of an $\mathcal{ALCK}$-knowledge base, which is the $\mathcal{ALCK}$-knowledge base obtained by replacing every inclusion statement $C \sqsubseteq D$ by the epistemic statement $\mathbf{K}C \sqsubseteq D$. More formally, let $\Phi = \langle\langle\mathcal{T},\mathcal{A}\rangle,\mathcal{R}\rangle$ be an $\mathcal{ALCK}$-knowledge base as defined in the previous section. The weakening of Φ is the $\mathcal{ALCK}$-knowledge base

$$\Phi^- = \langle\Sigma',\mathcal{R}'\rangle$$

where

$$\Sigma' = \langle\emptyset,\mathcal{A}\rangle$$

and

$$\mathcal{R}' = \mathcal{R} \cup \{\mathbf{K}C \sqsubseteq D \mid (C \sqsubseteq D) \in \mathcal{T}\}.$$

Intuitively, every inference we can make in Φ^- can be done in Φ as well, while the converse of course is not true. Hence, Φ^- can be regarded as a sound approximation of Φ, where the lost inferences are traded with a gain in the efficiency of reasoning. Before addressing in more detail this computational aspect, we present an example of the weakening transformation.

Consider the knowledge base $\Phi_1 = \langle\Sigma_1,\emptyset\rangle$, where $\Sigma_1 = \langle\mathcal{T}_1,\mathcal{A}_1\rangle$ is the knowledge base used in Section 4. The weakening Φ_1^- will be $\langle\langle\emptyset,\mathcal{A}_1\rangle,\mathcal{R}_1\rangle$, where $\mathcal{R}_1$ is shown in Figure 3. Recall that all definitions of the form $C \doteq D$ are a shorthand for $C \sqsubseteq D$ and $D \sqsubseteq C$.

It can be verified that all queries asked to Σ_1 in Section 4 have the same answer in Φ_1^-, except for queries 4a and 4d, reported here for the sake of clarity.

- Query 4a: $\Sigma_1 \models \exists$TEACHES.IntCourse(john) ? Answer: YES.
- Query 4d: $\Sigma_1 \models \exists\mathbf{K}$TEACHES.IntCourse(john) ? Answer: YES.

These queries receive the answer YES in Σ_1 because of a case analysis on Susan. Recall that, according to $\mathcal{T}_1$, the TBox of Σ_1, the two concepts Grad and Undergrad partition the concept Student. Being a student, Susan can be either a graduate or an undergraduate. In

$\mathbf{K}$AdvCourse $\sqsubseteq$ (Course $\sqcap$ $\forall$ENROLLED.Grad),
$\mathbf{K}$(Course $\sqcap$ $\forall$ENROLLED.Grad) $\sqsubseteq$ AdvCourse,
$\mathbf{K}$BasCourse $\sqsubseteq$ (Course $\sqcap$ $\forall$ENROLLED.Undergrad),
$\mathbf{K}$(Course $\sqcap$ $\forall$ENROLLED.Undergrad) $\sqsubseteq$ BasCourse,
$\mathbf{K}$IntCourse $\sqsubseteq$ (Course $\sqcap$ $\exists$ENROLLED.Grad $\sqcap$ $\exists$ENROLLED.Undergrad),
$\mathbf{K}$(Course $\sqcap$ $\exists$ENROLLED.Grad $\sqcap$ $\exists$ENROLLED.Undergrad) $\sqsubseteq$ IntCourse,
$\mathbf{K}\exists$TEACHES.Course $\sqsubseteq$ Grad $\sqcup$ Professor,
$\mathbf{K}$Grad $\sqsubseteq$ (Student $\sqcap$ $\exists$DEGREE.Bachelor),
$\mathbf{K}$(Student $\sqcap$ $\exists$DEGREE.Bachelor) $\sqsubseteq$ Grad,
$\mathbf{K}$Undergrad $\sqsubseteq$ (Student $\sqcap$ $\neg$Grad),
$\mathbf{K}$(Student $\sqcap$ $\neg$Grad) $\sqsubseteq$ Undergrad

Figure 3: The trigger rules of Φ_1^-, obtained by weakening the inclusions of $\mathcal{T}_1$

the first case, the course CS221 is an inetrmediary course, while in the second case CS324 is an inetrmediary course. Hence, in both cases John teaches an inetrmediary course.

On the contrary, it is easy to see that this does not happen in Φ_1^-, as shown by the following queries.

- Query 4f: $\Phi_1^- \models \exists$TEACHES.IntCourse(john) ? Answer: UNKNOWN.
- Query 4g: $\Phi_1^- \models \exists\mathbf{K}$TEACHES.IntCourse(john) ? Answer: UNKNOWN.

This is because in Φ_1^- the two concepts **Grad** and **Undergrad** do not partition the concept Student. What we just know is that individuals known to be undergraduates are inferred to be students and nongraduates, and vice versa, that individuals known to be students and nongraduates are inferred to be undergraduates. Since Susan is in neither of the two conditions, we cannot infer anything about her. In fact, there are now epistemic models for Φ_1^- where Susan is neither a graduate nor an undergraduate. Therefore, the two queries 4f and 4g receive the answer UNKNOWN.

One can also verify that contrapositives are not applicable in Φ_1^-. Compare the answer to $\neg\exists$DEGREE.Bachelor(peter) in the two knowledge bases:

- Query 5a: $\Sigma_1 \models \neg\exists$DEGREE.Bachelor(peter) ? Answer: YES.
- Query 5b: $\Phi_1^- \models \neg\exists$DEGREE.Bachelor(peter) ? Answer: UNKNOWN.

In fact, in Σ_1 Peter is known to be an undergraduate, hence a student who is a nongraduate. Since graduates are defined as students with a bachelor's degree, we can infer that Peter has none by using the contrapositive of the inclusion (Student $\sqcap$ $\exists$DEGREE.Bachelor) $\sqsubseteq$ Grad. Instead, in Φ_1^- we only can infer that Peter is a student and a nongraduate. This does not activate the contrapositive of the trigger rule $\mathbf{K}$(Student $\sqcap$ $\exists$DEGREE.Bachelor) $\sqsubseteq$ Grad.

Let us now go back to the computational advantages of weakening an $\mathcal{ALCK}$-knowledge base. In order to show such advantages, consider an $\mathcal{ALCK}$-knowledge base $\Phi = \langle\Sigma, \mathcal{R}\rangle$, where $\Sigma = \langle \mathcal{T}, \mathcal{A} \rangle$, and let $\Phi^- = \langle \Sigma', \mathcal{R}' \rangle$, where $\Sigma' = \langle \emptyset, \mathcal{A} \rangle$, be its weakening. Furthermore, assume that no rule in $\mathcal{R}'$ has an antecedent which is equivalent to $\top$.

Extending the results of the complexity analysis carried out in [4, 9], one can show that query answering in Φ can be solved in exponential space and double exponential time [8]. Since query answering in $\mathcal{ALC}$-knowledge bases with inclusions is known to be EXPTIME-hard [4], we do not expect to find any algorithm working in polynomial space, unless EXPTIME = PSPACE. On the other hand query answering in Φ^- amounts to solving the same problems in $\Sigma'_{\mathcal{R}'}$, which is the first-order extension of $\Sigma' = \langle \emptyset, \mathcal{A} \rangle$ w.r.t. $\mathcal{R}'$. Observing that $\Sigma'_{\mathcal{R}'}$ is a knowledge base constituted by an ABox only, we know from [7] that this problem can be solved in polynomial space. Since the size of $\Sigma'_{\mathcal{R}'}$ is polynomially related to the size of Φ^-, and therefore of Φ too, the above observation shows that weakening the inclusions of an $\mathcal{ALCK}$-knowledge base leads to an exponential decrease of the space required for query answering.

We can conclude that the notion of weakening proposed here provides a form of incomplete reasoning that is both computationally advantageous and semantically well-founded.

8 Conclusion

In this paper, we discussed the advantages of using an epistemic operator both for enhancing the capabilities of concept languages, and for formalizing non-standard features of existing knowledge representation systems based on concept languages. We have shown that the epistemic operator is flexible enough to account for several different notions in an elegant and uniform way, namely epistemic queries, closed world reasoning, procedural rules and weak forms of concept definition.

At the same time, we believe that our investigation on the epistemic operator raises a number of interesting issues related to the use of concept languages in practical systems, which we intend to address in future work. First of all, it is worth analyzing whether the class of epistemic sentences proposed for formalizing rules and definitions can be extended so as to capture more aspects, while retaining the nice computational properties. Moreover, it would be interesting to analyze whether epistemic sentences are powerful enough to express some form of default reasoning.

Acknowledgements

This work has been supported by the Esprit Basic Research Action 6810 (Compulog 2), by the German Ministry for Research and Technology (BMFT) under grant ITW 9201 as part of the TACOS project, and by the Italian National Research Council as part of the Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo, Sottoprogetto 7, LDR Ibridi.

References

[1] Baader, F. and Hollunder, B. Embedding defaults into terminological knowledge representation formalisms. In *Proc. of the 3nd Int. Conf. on Principles of Knowledge Representation and Reasoning KR-92*, pages 306–317. Morgan Kaufmann, 1992.

[2] Brachman, R. J., Borgida, A., McGuinness, D. L., and Alperin Resnick, L. The CLASSIC knowledge representation system, or, KL-ONE: the next generation. Preprints of the Workshop on Formal Aspects of Semantic Networks, Two Harbors, Cal., 1989.

[3] Brachman, R. J. and Levesque, H. J. The tractability of subsumption in frame-based description languages. In *Proc. of the 4th Nat. Conf. on Artificial Intelligence AAAI-84*, pages 34–37, 1984.

[4] Buchheit, M., Donini, F. M., and Schaerf, A. Decidable reasoning in terminological knowledge representation systems. In *Proc. of the 13th Int. Joint Conf. on Artificial Intelligence IJCAI-93*, 1993. In press.

[5] Donini, F. M., Lenzerini, M., Nardi, D., and Nutt, W. The complexity of concept languages. In Allen, J., Fikes, R., and Sandewall, E., editors, *Proc. of the 2nd Int. Conf. on Principles of Knowledge Representation and Reasoning KR-91*, pages 151–162. Morgan Kaufmann, 1991.

[6] Donini, F. M., Lenzerini, M., Nardi, D., and Nutt, W. Tractable concept languages. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence IJCAI-91*, pages 458–463, Sidney, 1991.

[7] Donini, F. M., Lenzerini, M., Nardi, D., Nutt, W., and Schaerf, A. Adding epistemic operators to concept languages. In *Proc. of the 3nd Int. Conf. on Principles of Knowledge Representation and Reasoning KR-92*, pages 342–353, 1992.

[8] Donini, F. M., Lenzerini, M., Nardi, D., Nutt, W., and Schaerf, A. Adding epistemic operators to concept languages. Technical report, Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", 1993. Forthcoming.

[9] Donini, F. M., Lenzerini, M., Nardi, D., and Schaerf, A. A hybrid system integrating datalog and concept languages. In *Proc. of the 2nd Italian Conf. on Artificial Intelligence*, volume 549 in Lecture Notes in Artificial Intelligence. Springer-Verlag, 1991. An extended version appeared also in the Working Notes of the AAAI Fall Symposium "Principles of Hybrid Reasoning", 1991.

[10] Donini, F. M., Lenzerini, M., Nardi, D., and Schaerf, A. From subsumption to instance checking. Technical Report 15.92, Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", 1992.

[11] Doyle, J. and Patil, R. S. Two thesis of knowledge representation: Language restrictions, taxonomic classification, and the utility of representation services. *Artificial Intelligence*, 48:261–297, 1991.

[12] Fikes, R. and Kehler, T. The role of frame-based representation in reasoning. *Communications of the ACM*, 28(9):904–920, 1985.

[13] Gelfond, M. and Przymusinska, H. Negation as failure: Careful closure procedure. *Artificial Intelligence*, 30:273–287, 1986.

[14] Levesque, H. J. Foundations of a functional approach to knowledge representation. *Artificial Intelligence*, 23:155–212, 1984.

[15] Lifschitz, V. Nonmonotonic databases and epistemic queries. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence IJCAI-91*, Sidney, 1991.

[16] MacGregor, R. A deductive pattern matcher. In *Proc. of the 6th Nat. Conf. on Artificial Intelligence AAAI-88*, pages 403–408, 1988.

[17] MacGregor, R. and Bates, R. The Loom knowledge representation language. Technical Report ISI/RS-87-188, University of Southern California, Information Science Institute, Marina del Rey, Cal., 1987.

[18] Minker, J. On indefinite data bases and the closed world assumption. In *Conf. on Automated Deduction, LNCS 138*, 1982.

[19] Nebel, B. *Reasoning and Revision in Hybrid Representation Systems.* Lecture Notes in Artificial Intelligence. Springer-Verlag, 1990.

[20] Nebel, B. Terminological reasoning is inherently intractable. *Artificial Intelligence*, 43:235–249, 1990.

[21] Nebel, B. Terminological cycles: Semantics and computational properties. In Sowa, J. F., editor, *Principles of Semantic Networks*, pages 331–361. Morgan Kaufmann, 1991.

[22] Quantz, J. and Kindermann, C. Implementation of the BACK system version 4. Technical Report KIT-Report 78, FB Informatik, Technische Universität Berlin, Berlin, Germany, 1990.

[23] Quantz, J. and Royer, V. A preference semantics for defaults in terminological logics. In *Proc. of the 3nd Int. Conf. on Principles of Knowledge Representation and Reasoning KR-92*, pages 294–305, 1992.

[24] Reiter, R. On closed world data bases. In Gallaire, H. and Minker, J., editors, *Logic and Databases*, pages 119–140. Plenum, 1978.

[25] Reiter, R. On asking what a database knows. In Lloyd, J. W., editor, *Symposium on computational logics*, pages 96–113. Springer-Verlag, ESPRIT Basic Research Action Series, 1990.

[26] Schaerf, A. On the complexity of the instance checking problem in concept languages with existential quantification. In *Proc. of the 8th Int. Symp. on Methodologies for Intelligent Systems ISMIS-93*, 1993. In press. Extended version to appear in *Journal of Intelligent Information Systems.*

[27] Schild, K. Towards a theory of frames and rules. Technical report, FB Informatik, Technische Universität Berlin, Berlin, Germany, 1989.

[28] Schmidt-Schauß, M. and Smolka, G. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991.

[29] Woods, W. A. Understanding subsumption and taxomony: A framework for progress. In Sowa, J., editor, *Principles of Semantic Networks*, pages 45–94. Morgan Kaufmann, 1991.

[30] Yen, J., Neches, R., and MacGregor, R. CLASP: Integrating term subsumption sstems and production systems. *IEEE trans. on Knowledge and Data Engineering*, 3(1):25–31, 1991.

Automated Support for the Development of Non-Classical Logics

Ullrich Hustadt
Max-Planck-Institut für Informatik,
Im Stadtwald, W-6600 Saarbrücken, Germany
Phone: (+49 681) 302-5431, Fax: (+49 681) 302-5430,
E-mail: `Ullrich.Hustadt@mpi-sb.mpg.de`

Abstract

The most natural means for specifying a non-classical logic is by means of a Hilbert calculus. Usually, the semantics of a non-classical logic is given in terms of possible worlds. Given an axiomatization of a non-classical logics, the *correspondence problem* in these logics is to find for every given Hilbert axiom an equivalent property of the accessibility relation (van Benthem (1984)). For mechanizing deduction in non-classical logics it is very important to find these correspondences Ohlbach (1991). So far the method for finding the correspondences was mostly by intuition and the verification required complex proofs van Benthem (1984).

SCAN is an algorithm which offers a method for computing the correspondences fully automatically. Moreover, since SCAN preserves equivalences, the computed correspondence axioms are *guaranteed to be complete* in the sense that a formula is derivable in the Hilbert calculus if and only if it is valid in the frames which are models of the computed correspondence axiom.

In this paper we present the SCAN algorithm and an application of it to the problem of collapsing modal operators in multi-modal logics: Given a Hilbert calculus for modal operators $\Box_{m_1}$ and $\Box_{m_2}$ we have to ensure that

$$\Box_{m_1} P \Leftrightarrow \Box_{m_2} P$$

doesn't hold for all formulae P, because this is in general an unwanted consequence of the given axiomatization.

1 Introduction

Except for time where we have a precise mathematical model, all the other notions used to describe mental attitudes of agents are fuzzy. What is 'belief', 'knowledge', 'intention' etc. in mathematical terms? Nobody knows, therefore the only way to use these notions in a computer system is by means of approximations – formulations which are mathematically precise, but which do not necessarily reflect all aspects of the reality. Since the application defines the necessary degree of the approximation, a method is needed for specifying the approximation on a very abstract level and compiling it as automatically as possible into a mechanizable deduction system. In Gabbay and Ohlbach (1992), Szałas (1992), and Simmons (1992) a method for transformation a Hilbert calculus into predicate logic was proposed, which can be applied to this problem.

As an example, suppose, *Want*, *Believe* and *because*–operators are to be defined. Nobody has an idea what these notions mean in precise mathematical terms. But everybody would agree to some basic correlations between them, for example if you want R because you believe that R implies Q then in reality you want Q, or formally

$$\textit{Want } R \textit{ because } \textit{Believe}(R \rightarrow Q) \rightarrow \textit{Want } Q$$

This is an Hilbert axiom correlating the *Want*–Operator with *Believe*, *because* and $\rightarrow$. With a growing number of modal operators like *Want* or *Believe*, it will be hard to tell wether the correlations are still consistent with each other. Furthermore, if we have chosen multiple modal operators to describe different mental attitudes, these modal operators shouldn't become equivalent. An equivalence like

$$\forall R\colon \textit{Want } R \leftrightarrow \textit{Believe } R$$

is usually an unwanted correlation, although the set of axioms can still be consistent. In such a situation, we say that *Want* and *Believe collapse.*

In the sequel we will introduce a method for second-order quantifier elimination. Gabbay and Ohlbach (1992) and Allgayer, Ohlbach, and Reddig (1992) have shown the usefulness of second-order quantifier elimination for the mechanization of non-classical logics. In this paper we will focus on the application of second-order quantifier elimination to the problem of axiomatizing consistent modal logics with non-collapsing modal operators.

2 Hilbert Calculi for Non-Classical Logics

The most natural means for specifying a logic for fuzzy notions like *Want* or *Believe* is by means of a Hilbert calculus. A Hilbert calculus consists of some axioms, i.e. formulae or formulae schemata respectively which are apriori considered true, together with some production rules for generating from the axioms other true formulae (tautologies). The axiomatization of normal modal logic (Chellas 1980) which may be used as a basis for axiomatizing 'belief' etc. is an example for a Hilbert calculus:

Example 1 (Hilbert Calculus for Normal Modal Logic)

Axioms:	all axioms of propositional logic	
	$\Box(P \to Q) \to (\Box P \to \Box Q)$	(K–Axiom)
Rules:	$\dfrac{P, \quad P \to Q}{Q}$	(Modus Ponens)
	$\dfrac{P}{\Box P}$	(Necessitation Rule)

Implicitly there is always a substitution rule allowing to consider the propositional variables $(P, Q, \ldots)$ as placeholders for arbitrary formulae. Different variants of the logic manifest themselves by additional Hilbert axioms.

Usually, the semantics of a modal logic is given in terms of possible worlds. Hilbert axioms like the K-Axiom above correspond to particular properties of the possible worlds structure. For example, the Hilbert axiom

$$\forall P \; \Box P \to P \tag{1}$$

corresponds to the reflexivity of the accessibility relation. The *correspondence problem* in these logics is to find for a given Hilbert axiom an equivalent semantic property of the accessibility relation (van Benthem 1984). For mechanizing deduction in non-classical logics it is very important to find these correspondences (Ohlbach 1991). So far the method for finding the semantic properties of the accessibility relation was mostly by intuition and the verification required complex proofs (van Benthem 1984).

In the following section, we will introduce an algorithm for second-order quantifier elimination, which can be used to mechanize the finding of semantic properties.

3 The SCAN Algorithm

SCAN[1] is the first algorithm which offers a method for computing the correspondences fully automatically. Moreover, since SCAN preserves equivalences, the computed semantic properties are *guaranteed to be complete* in the sense that a formula is derivable in the Hilbert calculus if and only if it is valid in the frames which are models of the computed corresponding semantic properties.

Definition 2 (The SCAN Algorithm)
Input to SCAN is a formula $\alpha = \exists P_1, \ldots, P_n \; \psi$ with predicate variables $P_1, \ldots, P_n$ and an arbitrary first–order formula ψ.
Output of SCAN — if it terminates — is a formula φ_α which is logically equivalent to α, but not containing the predicate variables $P_1, \ldots, P_n$.

SCAN performs the following three steps:

[1]SCAN means 'Synthesizing Correspondence Axioms for Normal logics.' The name has been chosen before we realized that is applicable in a general context.

1. ψ is transformed into clause form using second order skolemization. That means the resulting formula has the form: $\exists P_1, \ldots, P_n \exists f_1, \ldots, f_n \psi'$ where the f_i are the Skolem functions and ψ' is a set of clauses. From the algorithm's point of view, the quantifier prefix can be ignored. Therefore ψ' is treated as an ordinary clause set with the usual Skolem constants and functions.

2. All C–resolvents and C–factors with the predicate variables $P_1, \ldots, P_n$ have to be generated. C–resolution ('C' for constraint) is defined as follows:

$$\frac{\begin{array}{ll} P(s_1, \ldots, s_n) \vee C & P(\ldots) \text{ and } \neg P(\ldots) \\ \neg P(t_1, \ldots, t_n) \vee D & \text{are the } \textit{resolution literals} \end{array}}{C \vee D \vee s_1 \neq t_1 \vee \ldots \vee s_n \neq t_n}$$

and the C-factorization rule is defined analogously:

$$\frac{P(s_1, \ldots, s_n) \vee P(t_1, \ldots, t_n) \vee C}{P(s_1, \ldots, s_n) \vee C \vee s_1 \neq t_1 \vee \ldots \vee s_n \neq t_n}.$$

Notice that only C-resolutions between different clauses are allowed (no self resolution). A C-resolution or C-factorization can be optimized by destructively resolving literals $x \neq t$ where the variable x does not occur in t with the reflexivity equation. C–resolution and C–factorization takes into account that second order quantifiers may well impose conditions on the interpretations which must be formulated in terms of equations and inequations.

As soon as *all* resolvents and factors between a particular literal and the rest of the clause set have been generated (the literal is 'resolved away'), the clause containing this literal must be deleted (purity deletion). If all clauses are deleted this way, this means that α is a tautology.

All equivalence preserving simplifications may be applied freely. These are for example:

- Tautologous resolvents can be deleted.
- Subsumed clauses can be deleted.
- Subsumption factoring can be performed. Subsumption factoring means that a factor subsumes its parent clause. This may be realized by just deleting some literals. For example $Q(x), Q(a)$, where x is a variable, can be simplified to $Q(a)$.
- Subsumption resolution can also be performed. Subsumption resolution means that a resolvent subsumes its parent clause, and this again may be realized by deleting some literals Ohlbach and Siekmann (1991). For example the resolvent between $P \vee Q$ and $\neg P \vee Q \vee R$ is just $Q \vee R$ such that $\neg P$ can be deleted from the clause. (An instance of this operation is realized as so called 'unit_deletion' in the OTTER theorem prover.)

If an empty clause is generated, this means that α is contradictory.

3. If the previous step terminates and there are still clauses left then reverse the skolemization. A method for reversing the skolemization in a set F of clauses is (1) to abstract all arguments of all occurrences of Skolem functions by variables, i.e. $f(s_1, \ldots, s_n)$ is replaced with $f(x_1, \ldots, x_n)$ and additional literals $x_i \neq s_i$ are added to the clause where the x_i are fresh variables and (2) to consistently rename all variables such that the arguments of all occurrences of the Skolem function are the same. If this is possible and $F[f(x_1, \ldots, x_n)]$ is the result then $\forall x_1, \ldots, x_n \exists y\ F[y]$ is the solution. This process is repeated for all Skolem functions.

   If it is not possible to rename the variables consistently, the only chance is to take parallel Henkin quantifiers (Henkin 1961) or leave the second-order quantification.

△

The step from the Hilbert axioms to the second-order formulae is explained in detail in Gabbay and Ohlbach (1992). For our purposes it is enough to assume that we have semantic definitions for the modal operators and other connectives in the Hilbert axioms, which can be used to rewrite the axioms to second-order formulae.

We illustrate how the SCAN algorithm can be applied by computing the reflexivity of the accessibility relation from the axiom $\Box P \to P$.

Example 3 We start with the formula

$$\alpha = \forall P{:}\, \Box P \to P$$

and semantic definitions

$$\forall P{:}\, w \models \Box P \Leftrightarrow \forall w'{:}\, \mathcal{R}(w, w') \Rightarrow w' \models P$$

and

$$\forall P{:}\, w \models P \to R \Leftrightarrow w \models \neg P \vee R,$$

where $\mathcal{R}$ is the accessibility relation on worlds.

Since SCAN accepts only existentially quantified predicate variables, α is negated and then the semantic definitions are applied as rewrite rules until $\Box$ and $\to$ disappear. The result is put into clause form.

We obtain the two clauses:

$$\neg\mathcal{R}(w, w') \vee P(w'),$$
$$\neg P(w)$$

where w is a Skolem constant. Resolution with P-literals — only these resolutions are allowed — yields

$$\frac{\neg\mathcal{R}(w, w') \vee P(w'), \quad \neg P(w)}{\neg\mathcal{R}(w, w)}$$

The two parent clauses are pure (no further resolutions are possible) and can be deleted. The single remaining clause $\neg\mathcal{R}(w,w)$ is unskolemized to $\exists\, w{:}\, \neg\mathcal{R}(w,w)$ and negated yielding $\forall\, w{:}\, \mathcal{R}(w,w)$ which is the desired result[2].

4 Detecting Collapsing Modal Operators

The need for a tool like SCAN becomes apparent if we consider a set of Hilbert axioms specifying not only a single modal operator, but multiple, interacting modal connectives. For the standard Hilbert axioms describing a single modal operator the corresponding properties of the accessibility relation are well-known. For multiple, interacting modal operators the corresponding results are less accessible. With SCAN we can compute the corresponding properties of the accessibility relation automatically.

But we can do even more. The first step after specifying a set of Hilbert axioms for multiple, interacting modal operators is to exhibit their consistency. In a second step, we have to show that the modal operators do not collapse, i.e.

$$\Box_{m_1} P \leftrightarrow \Box_{m_2} P$$

shouldn't hold for distinct modal operators $\Box_{m_1}$ and $\Box_{m_2}$.

The first step can be done with the aid of SCAN and any first-order refutational theorem prover. We use SCAN to find the correspondence axioms of the given Hilbert axioms. If SCAN terminates and produces a set of first-order formulae for every axiom, we feed the first-order refutational theorem prover with all these formulae to find an inconsistency. Of course, the problem of proving the inconsistency of a set of first-order formulae is only semi-decidable. In case our set of formulae is consistent, the theorem prover may not terminate.

In the second step, we want to show that

$$\neg\forall\, P{:}\, \Box_{m_1} P \leftrightarrow \Box m_2\, P.$$

holds for modal operators $\Box_{m_1}$ and $\Box_{m_2}$. We use SCAN to eliminate the second order quantifier $\forall P$. We will get a corresponding first-order semantic property (if the process terminates). This first-order semantic property can be proven using a first-order refutational theorem prover.

Example 4 van der Hoek (1992) considers the following Hilbert axioms for a multimodal logic of knowledge and belief:

$$\neg\Box_{believe}\, \mathit{false} \tag{2}$$

$$\neg\Box_{know} P \rightarrow \Box_{know} \neg\Box_{know} P \tag{3}$$

$$\Box_{know} P \rightarrow \Box_{believe} P \tag{4}$$

$$\Box_{believe} P \rightarrow \Box_{believe} \Box_{know} P \tag{5}$$

[2] In this particular example we get the "relational" translation of the $\Box$-Operator (Moore 1980, Ohlbach 1991). A slight modification of the semantics of $\Box$ yields the more compact and computationally more efficient "functional" translation (Ohlbach 1991).

Axiom (2) says that we don't believe anything that is false. Axiom (3) is the axiom of negative knowledge introspection, i.e. if we don't know P, then we know that we don't know P. Axiom (4) defines that knowledge is 'stronger' than believe. Axiom (5) is intended to model the attitude of an agent who thinks that he is very critical in adopting believes: he only believes P if he believes that he knows P.

SCAN produces the following corresponding semantic properties in clause form

$$\mathcal{R}_{believe}(w_1, f(w_1)) \tag{6}$$
$$\neg\mathcal{R}_{know}(w_1, w_2) \vee \neg\mathcal{R}_{know}(w_1, w_3) \vee \mathcal{R}_{know}(w_2, w_3) \tag{7}$$
$$\neg\mathcal{R}_{believe}(w_1, w_2) \vee \neg\mathcal{R}_{know}(w_2, w_3) \vee \mathcal{R}_{believe}(w_1, w_3) \tag{8}$$
$$\neg\mathcal{R}_{believe}(w_1, w_2) \vee \mathcal{R}_{know}(w_1, w_2), \tag{9}$$

where w_1, w_2, and w_3 denote (universally quantified) variables and f is a Skolem function. (6)–(9) say that $\mathcal{R}_{believe}$ is serial, contained in $\mathcal{R}_{know}$, transitive over $(\mathcal{R}_{believe}, \mathcal{R}_{kow})$, and $\mathcal{R}_{know}$ is euclidean.

Now is there actually a difference between knowledge and believe provided the given Hilbert axioms hold? We already know that for all P

$$\Box_{know} P \rightarrow \Box_{believe} P$$

holds. Now we want to check if the converse is also true. That is, we check if

$$\Box_{believe} P \rightarrow \Box_{know} P \tag{10}$$

is an additional consequence of the Hilbert axioms. The corresponding semantic property of (10) is

$$\forall w_1, w_2 : \mathcal{R}_{know}(w_1, w_2) \rightarrow \mathcal{R}_{believe}(w_1, w_2).$$

Negating this property yields the clauses

$$\mathcal{R}_{know}(v, w) \tag{11}$$
$$\neg\mathcal{R}_{believe}(v, w), \tag{12}$$

where v and w are Skolem constants. Resolution of clause (8) with (12) and (6) yields

$$\neg\mathcal{R}_{know}(f(v), v) \tag{13}$$

Now clause (7) can be resolved with clause (13) and (11) giving

$$\neg\mathcal{R}_{know}(v, f(v)). \tag{14}$$

The resolvent of clause (14) and clause (9) is

$$\neg\mathcal{R}_{believe}(v, f(v)) \tag{15}$$

Resolution of clause (15) and (6) yields the empty clause, i.e. we have proven the inconsistency of this set $\{(6), \ldots, (12)\}$ of clauses.

So,

$$\Box_{know} P \leftrightarrow \Box_{believe} P$$

actually holds for all P. Every clause of the semantic properties corresponding to the considered Hilbert axioms has been used in the proof above. Therefore, all four Hilbert axioms together force the collapse of knowledge and belief. We have to abandon at least one of them to get back to a set of axioms where knowledge and belief are distinct operators.[3]

5 Limitations of SCAN

The SCAN algorithm can produce first-order semantic properties only for those second-order Hilbert axioms where such corresponding properties actually exist. From modal logic we know cases where a Hilbert axiom has a semantic property which is only second-order axiomatizable. The most well-known example is the McKinsey axiom

$$\forall P{:}\, \Box\Diamond P \rightarrow \Diamond\Box P.$$

SCAN produces the following property of the accessibility relation for the McKinsey axiom:

$$\forall a \begin{pmatrix} \forall f\ \exists x \\ \forall g\ \exists y \end{pmatrix} \begin{array}{l} ((\mathcal{R}(a,x) \rightarrow \mathcal{R}(x,f(x)))\wedge \\ (\mathcal{R}(a,y) \rightarrow \mathcal{R}(y,g(y)))) \rightarrow \\ (\mathcal{R}(a,x) \wedge \mathcal{R}(a,y) \wedge f(x) = g(y)). \end{array}$$

Because of the quantification over functions, this is still second-order. In such a case, we will not be able to make further investigations.

Furthermore, SCAN cannot invent a good axiomatization of a non-classical logic. For example, the McKinsey axiom combined with the transitivity axiom

$$\Box P \rightarrow \Box\Box P,$$

corresponds to a first-order axiomatizable property of the accessibility relation, i.e. atomicity of the accessibility relation

$$\forall x{:}\, \exists y{:}\, (\mathcal{R}(x,y) \wedge \forall z{:}\, \mathcal{R}(y,z) \rightarrow z = y))$$

(van Benthem 1984, page 203). Therefore, the combination of these two axioms is suitable for a mechanization using a first-order theorem prover. Of course, there is no way for SCAN to give a hint that the McKinsey axiom should be combined with the transitivity axiom to get a usable set of semantic properties.

[3] van der Hoek (1992) also proved the collapse of knowledge and belief. The advantage we get using SCAN is the ability to perform such checks automatically.

6 Conclusion

Cohen (1991) claims that AI is suffering a methodological malaise, because AI is dominated by two largely distinct camps: model-oriented researchers and system-oriented researchers. One of the problems is that the results of model-oriented researchers migrate only slowly to the system-oriented researchers and on the other hand, the real problems which system-oriented researchers are faced with, arouse only slowly the interest of model-oriented researchers.

SCAN can be seen as a tool developed by model-oriented researchers to help system-oriented researchers to solve their problems in their own way. SCAN is a tool for playing with different axiomatizations of non-classical logics. It can be used, as we have done in this note, to examine the consistency of an axiomatization or detect the collapse of modal operators in a modal logic. Another important application of SCAN is the development of deductive systems for non-classical logics. See Ohlbach (1991).

This illustrates, how SCAN reduces the routine work of researchers, yielding more room for the art of inventing axiomatizations of non-classical logics.

References

ALLGAYER, J., OHLBACH, H., AND REDDIG, C., 1992. Modelling Agents with Logic. In Andre, E., Cohen, R., Graf, W., Kass, B., Paris, C., and Wahlster, W., editors, *UM92 — Proceedings of the Third International Workshop on User Modeling, DFKI Document D-92-17.*

CHELLAS, B. F., 1980. *Modal logic: an introduction.* Cambridge University Press, New York.

COHEN, P., 1991. A Suvey of the Eight National Conference on AI: Pulling Together or Pulling Apart. *AI Magazine* **12**(1):16–41.

GABBAY, D. M. AND OHLBACH, H. J., 1992. Quantifier elimination in second-order predicate logic. *South African Computer Journal* **7**:35–43. appeared also in Proc. of KR92, Morgan Kaufmann, 1992, pp 425–436.

HENKIN, L., 1961. Some remarks on infinitely long formulas. In *Infinitistic Methods*, pp. 167–183. Pergamon Press, Oxford.

MOORE, R., 1980. Reasoning about Knowledge and Action. Technical Note 191, SRI International, Menlo Park, CA.

OHLBACH, H. J., 1991. Semantics Based Translation Methods for Modal Logics. *Journal of Logic and Computation* **1**(5):691–746.

OHLBACH, H. J. AND SIEKMANN, J. H., 1991. The Markgraf Karl Refutation Procedure. In Lassez, J. L. and Plotkin, G., editors, *Computational Logic, Essays in Honor of Alan Robinson*, pp. 41–112. MIT Press.

SIMMONS, H., 1992. An Algorithm for Eliminating Predicate Variables from Π_1^1 Sentences (with special reference to modal correspondence theory). Department of Computer Science, The University of Manchester, Draft.

SZAŁAS, A., 1992. On Correspondence Between Modal and Classical Logic: Automated Approach. Technical Report MPI–I–92–209, Max-Planck-Institute for Computer Science.

VAN BENTHEM, J., 1984. Correspondence Theory. In Gabbay, D. M. and Guenthner, F., editors, *Handbook of Philosophical Logic, Vol. II, Extensions of Classical Logic, Synthese Library Vo. 165*, pp. 167–248. D. Reidel Publishing Company, Dordrecht.

VAN DER HOEK, W., 1992. *Modalities for Reasoning about Knowledge and Quantities.* PhD thesis, Vrije Universiteit van Amsterdam, Elinkwijk, Utrecht, Netherlands.

Representing Belief in Multi-Agent Worlds via Terminological Logics

Armin Laux
German Research Center for Artificial Intelligence (DFKI)
Stuhlsatzenhausweg 3
66123 Saarbrücken, Germany
e-mail: laux@dfki.uni-sb.de

Abstract

In multi-agent systems a group of autonomous intelligent systems, called agents, co-operates in order to achieve certain goals. However, each of these agents can only perform actions that are based on his local knowledge and on his local beliefs. In the literature knowledge of agents is mostly represented under the view that knowledge is true belief. On the other hand, if agents are acting in a (real) world their knowledge often is obtained by perception and communication, and hence typically is not true. Thus, the use of belief—where agents may have false beliefs—seems more appropriate than knowledge for formalizing the reasoning and deduction of a knowledge base.

In this paper we present a language $\mathcal{ALC}_{\mathcal{B}}$ that extends the concept language $\mathcal{ALC}$ by a modal operator $\Box$, which is indexed by agents. Thereby, $\Box_i\varphi$ represents the fact "agent i believes φ". This belief operator will be interpreted in terms of possible worlds using the well-known modal logic KD45. In this approach an agent is said to believe a fact φ iff φ is true in all worlds he considers possible.

The language $\mathcal{ALC}_{\mathcal{B}}$ provides a uniform formalism to describe both, a world agents are acting in and the beliefs agents have about this world and about their own and other agents' beliefs. Thus, it can be seen as a two-dimensional extension of $\mathcal{ALC}$ which allows both, reasoning about objective facts that hold in the world and reasoning on the level of possible worlds. We will give sound and complete algorithms to check consistency of the represented beliefs and to decide whether an $\mathcal{ALC}_{\mathcal{B}}$-sentence is logically entailed by the beliefs of agents. Hence, when acting in a world agents can use beliefs which are explicitly represented as well as implicit beliefs that are entailed by their knowledge base.

1 Introduction

Research on the field of multi-agent systems deals with the question how a group of autonomous intelligent systems, called agents, can cooperate in order to achieve certain goals (see, e.g., [6, 14]). As an example, a forwarding agent a and a shipping agent b may cooperate in order to carry out overseas transportation orders.

Although the tasks that multi-agent systems are required to perform are normally stated in terms of the global behavior of the system, the actions that an agent performs can depend only on his local knowledge and on his local beliefs. Thus, there is a close relationship between knowledge, belief, and action in multi-agent systems (see, e.g., [27, 28, 15]). Suppose, in the above example agent a wants to offer a price for carrying out some transportation order o_1. If he believes that there is no other forwarding agent who also can carry out o_1, he will most likely offer another price as in the case where he believes that there is a competitor for this order. And if he even *knows* that there is no competitor for this order, he perhaps offers an exorbitant price. In a recent paper [20] we investigated how knowledge of agents can be represented on the basis of terminological logics, whereby we used the classical view of knowledge as true belief. That means, an agent knows φ if he believes that φ holds and φ actual does hold. On the other hand, as pointed out in, say [23], the knowledge represented in a knowledge base typically is not true. Thus, the use of beliefs—where agents may have false beliefs—seems more appropriate than knowledge for formalizing the reasoning and deduction of a knowledge base. In the current paper we concern with the question how agents can be equipped with beliefs about the world they are acting in, about beliefs of other agents, and also about their own beliefs. Thereby, it should be taken into consideration that different agents may have different beliefs about the same notions. For example, forwarding agent a may believe that *company XY* is a rich company and a good client, while forwarding agent B believes that *company XY* is rich but not a good client. Using the language $\mathcal{ALC}_{\mathcal{B}}$, which is presented in the next section, this can be formalized by

$$\Box_a(\textit{company XY}\text{: rich-company} \sqcap \text{good-client}) \quad \text{and}$$
$$\Box_b(\textit{company XY}\text{: rich-company} \sqcap \neg\text{good-client})$$

respectively, where $\Box_i\varphi$ is to be read as "agent i believes φ".

Since the work of Hintikka [17], modal logics have widely been accepted to be an adequate formalism for representing knowledge and belief of agents. The intuitive idea here is that besides the real world agents can imagine a number of other worlds (situations) to be possible. By imposing various conditions on this possibility relation, we can capture a number of interesting axioms. For example, if we require that the world that the agent finds himself in is always one of the worlds he considers possible (which amounts to saying that the possibility relation is reflexive), then it follows that the agent does not know false facts. When using a possibility relation which captures axioms of knowledge (belief) an agent is said to know (believe) a fact φ iff φ is true in all worlds he thinks to be possible. For example, an agent knows (believes) that there exists a monster of Loch Ness if there is such a monster in all worlds he considers possible. To express the beliefs of an agent a in this approach a binary operator *BELIEF* (a, φ) is used, where φ is a formula over some logical language $\mathcal{L}$. If we want to devise a formalism for representing the beliefs of agents we have to take two decisions. Firstly, we have to decide what the general properties of belief are we want this formalism to capture. Secondly, we have to choose a suitable logical representation language $\mathcal{L}$ which allows to describe the beliefs of agents.

There are many approaches to determine axioms characterizing belief (see, e.g., [22, 28, 24, 25, 26, 11, 16]). We will use the following axiomatization which has been most commonly used in the literature. The first of these properties states that an agent does not believe false facts. That means, an agent cannot believe both a fact and its negation, though he can believe facts which actually do not hold in the world. Secondly, if an agent believes a fact then he believes that he believes it (positive introspection), and if he does not believe in a fact then he believes that he does not believe in it (negative introspection). From this it follows, e.g., that agents believe that their beliefs are true (weak reflexivity). Finally, the probably most important property is that agents can reason on the basis of their beliefs. For example, suppose agent a believes that each truck which is owned by John can be used to transport gasoline and he believes that John owns the truck *truck-1*. In this case, agent a must be able to conclude that John's truck *truck-1* can (probably) be used to transport gasoline, and thus may negotiate with John for a transportation order.

As logical language to describe belief of agents we will use a terminological logic. Terminological logics provide a well-investigated and decidable fragment of first-order logics that is much more expressive than propositional logic. They are based on the work of Brachman and Schmolze [9] and have been developed as a structured formalism to describe the relevant concepts of a problem domain and the interactions between these concepts. Starting with atomic concepts (unary predicates) and roles (binary predicates), one therefore defines complex concepts with the help of operators provided by a concept language, and interactions between (complex) concepts are expressed by a set of so-called terminological axioms. On the other hand, by so-called assertional axioms, objects can be associated with concepts and relationships between objects can be defined via roles. For example, we can use these logics to represent facts like "each truck which is owned by John can be used to transport gasoline" or "John owns *truck-1* which is a truck".

In the literature, a lot of concept languages have been considered (see, e.g., [8, 29, 3]). But they all have in common that they are only suitable for representing objective facts about the world, and knowledge or beliefs of agents can only be represented in a very limited way. Thus, we need an extended concept language which allows the representation of belief according to the above given (informal) axiomatization. Since the work of Schild [31] it is known that the concept language $\mathcal{ALC}$ provides a terminological logic which is a notational variant of the propositional modal logic $\mathsf{K}_{(m)}$. However, it is not investigated there how to extend this logic to a two-dimensional logic which allows reasoning on both the objective level and the level of possible worlds. In order to combine both levels one has to define syntax and semantics of an extended language. Baader and Ohlbach [5] present a multi-dimensional extension of $\mathcal{ALC}$, where multi-modal operators can be used at all levels of the concept terms and they can be used to modify both concepts and roles. However, the underlying logic is simply the basic modal logic K, and it is not yet clear how to extend their approach in such a way that modal logics different from K can be handled. Moreover, they could not succeed in proving completeness of their satisfiability algorithm.

In this paper we will present a different extended language where (sequences of) modal operators are only allowed in front of terminological and assertional axioms. This language allows one to interpret the modal operators w.r.t. modal logics different from K, e.g., S4 (see [20]) or KD45 (in the present paper). This language, called $\mathcal{ALC}_{\mathcal{B}}$, can be seen as a two-dimensional representation language with terminological and assertional axioms as primitives where each primitive may describe a part of the world and each agent can believe a set of such primitives to hold in the world. The modal operators, which are indexed with agents, are interpreted in terms of possible worlds in such a way that they satisfy the above axiomatization of belief, what amounts in using the modal logic KD45. Thus, the resulting language provides a uniform formalism to describe both, a world agents are acting in as well as the beliefs agents have about this world and about their own and other agents' beliefs. We will give sound and complete algorithms for deciding satisfiability of $\mathcal{ALC}_\mathcal{B}$-formulas and for testing whether an $\mathcal{ALC}_\mathcal{B}$-formula is entailed by a given set of $\mathcal{ALC}_\mathcal{B}$-formulas. Hence, when acting in a world agents can use beliefs which are explicitly represented as well as implicit beliefs that are entailed by their knowledge base.

2 Syntax and Semantics of $\mathcal{ALC}_\mathcal{B}$

In the concept language $\mathcal{ALC}$ *concepts* are built up from atomic concepts, the *top concept* $\top$, the *bottom concept* $\bot$, and roles inductively by: Each atomic concept, $\top$, and $\bot$ are concepts. If C and D are concepts and R is a role, then $C \sqcap D$ (*concept conjunction*), $C \sqcup D$ (*concept disjunction*), $\neg C$ (*concept negation*), $\forall R.C$ (*value restriction*), and $\exists R.C$ (*exists restriction*) are concepts. An interpretation I is a function over some non-empty domain Δ^I which maps each atomic concept C to a subset C^I of Δ^I, each role R to a subset R^I of $\Delta^I \times \Delta^I$, $\top$ to Δ^I, and $\bot$ to $\emptyset$. Furthermore, $\sqcap$ is interpreted as set intersection, $\sqcup$ as set union, and $\neg$ as set complement w.r.t. Δ^I. The value and the exists restrictions are interpreted by

$$[\forall R.C]^I = \{d \in \Delta^I \mid \forall d' : (d,d') \in R^I \rightarrow d' \in C^I\}$$
$$[\exists R.C]^I = \{d \in \Delta^I \mid \exists d' : (d,d') \in R^I \wedge d' \in C^I\}$$

For example, if *man* and *truck* are atomic concepts and *owns* is a role we can define the concept of men who own trucks by $man \sqcap \exists\, owns.truck$.

The taxonomical knowledge of a problem domain can be defined by an $\mathcal{ALC}$-*TBox* (*terminology*), which consists of a finite set of terminological axioms. A *terminological axiom* is of the form $C = D$ (concept equivalence) or $C \neq D$ (negated concept equivalence) where C, D are concepts. An interpretation I satisfies $C = D$ iff $C^I = D^I$ and it satisfies $C \neq D$ iff $C^I \neq D^I$. An interpretation I satisfies an $\mathcal{ALC}$-TBox $\mathcal{T}$ iff I satisfies each axiom in $\mathcal{T}$. For example, if *carrier*, *person*, and *truck* are concepts and *owns* is a role, we can define exactly the persons who own a truck to be a carrier by $carrier = person \sqcap \exists\, owns.truck$.

The assertional formalism of $\mathcal{ALC}$ allows to introduce concrete objects by stating that they are instances of concepts and roles: If a is an object and C a concept, then $a : C$ is a *concept instance*. If a and b are objects and R is a role, then aRb is a *role instance*. Concept instances and role instances are called *assertional axioms*, and a finite set of assertional axioms is called an $\mathcal{ALC}$-*ABox*. An interpretation I maps objects to elements of its domain Δ^I and satisfies $a : C$ iff $a^I \in C^I$, and aRb iff $(a^I, b^I) \in R^I$. We assume that different objects in an ABox are mapped to different elements in Δ^I (*unique name assumption*). An interpretation I satisfies an $\mathcal{ALC}$-ABox $\mathcal{A}$ iff I satisfies each axiom in $\mathcal{A}$. As an example, if *John* and *truck-1* are objects, we can express that John owns truck-1 which is a truck by the assertional axioms *John owns truck-1* and *truck-1 : truck*. Thus, we can describe the relevant concepts of a problem domain by terminological axioms, i.e., by an $\mathcal{ALC}$-TBox, and properties of objects as well as relations between them by assertional axioms, i.e., by an $\mathcal{ALC}$-ABox. We say an interpretation I satisfies a set $Ax_1, \ldots, Ax_n$ of terminological and assertional axioms iff I satisfies each of these axioms. We then write $I \models Ax_1, \ldots, Ax_n$.

Now we will introduce the language $\mathcal{ALC}_{\mathcal{B}}$ which extends $\mathcal{ALC}$ by a new operator $\Box_i$ for each agent i.[1] We allow these operators in front of terminological and assertional axioms. Thereby, the operator $\Box_i$, read as "agent i believes", allows us to express the beliefs agent i has about the world, about beliefs of other agents, and about his own beliefs. Therefore, we extend the definition of terminological and assertional axioms as follows. If *TA* is a terminological axiom, then $\Box_i$ *TA* and $\neg\Box_i TA$ are terminological axioms as well. If *CI* is a concept instance, then $\Box_i$ *CI* and $\neg\Box_i$ *CI* are concept instances as well. Finally, if *RI* is a role instance, then $\Box_i$ *RI* is a role instance as well. Note, that we do not allow formulas of the form $\neg\Box_i(aRb)$. The reason for this restriction is that such axioms would be equivalent to stating that there exists a world in which the role instance *aRb* does *not* hold. And negation of roles is not allowed in $\mathcal{ALC}$. These extended assertional and terminological axioms are called *$\mathcal{ALC}_{\mathcal{B}}$-formulas* and can, e.g., be used to state that agent i believes that a carrier is a man who owns a truck by $\Box_i(carrier = man \sqcap \exists\, owns.truck)$. Analogously, the $\mathcal{ALC}_{\mathcal{B}}$-formulas $\Box_i \neg\Box_j$ (*vehicle-1 : truck*) and $\Box_i\neg\Box_i$ (*vehicle-1 : truck*) are to be read as "agent i believes that agent j doesn't believe that vehicle-1 is a truck" and "agent i believes that he doesn't believe truck-1 to be a truck", respectively. Allowing $\Box_i$ immediately in front of concepts (possibly $\Box_i C$ may be interpreted as "the set of individuals agent i believes to be a C") causes essential algorithmic problems and is out of the scope of this paper.

We will interpret the operators $\Box_i$ in terms of *possible worlds*, i.e., besides the real world there exist a number of worlds agents consider to be possible. If agent i considers world w' possible at world w, we say w' is *accessible from* w by agent i. The *accessibility relation* of agent i is given by all pairs (w, w') such that w' is accessible from w by agent i. Since different worlds are possible in our approach, the interpretation of concepts and roles in $\mathcal{ALC}_{\mathcal{B}}$-formulas depends on the world we are currently speaking of. That means, in different worlds concepts may contain different objects and roles may contain different pairs of objects. This will be expressed by taking an additional parameter, the *world parameter*, into consideration when interpreting concepts and roles. Formally, we use the notion of a *K-interpretation* K_I which consists of a non-empty domain Δ^{K_I} and maps objects to elements in Δ^{K_I} while satisfying the unique name assumption, atomic concepts to subsets of $\Delta^{K_I} \times \mathcal{W}$, $\top$ to $\Delta^{K_I} \times \mathcal{W}$, $\bot$ to $\emptyset \times \mathcal{W}$, and roles to subsets of $\Delta^{K_I} \times \Delta^{K_I} \times \mathcal{W}$. Furthermore, $\sqcap$ is interpreted as set intersection, $\sqcup$ as set union, and $\neg$ as set complement w.r.t. $\Delta^{K_I} \times \mathcal{W}$. Finally, the value and exists restrictions are interpreted by

$$[\forall R.C]^{K_I} = \{(d, w) \mid (d', w) \in C^{K_I} \text{ for each } d' \text{ with } (d, d', w) \in R^{K_I}\}$$
$$[\exists R.C]^{K_I} = \{(d, w) \mid (d', w) \in C^{K_I} \text{ for some } d' \text{ with } (d, d', w) \in R^{K_I}\}.$$

Definition 2.1 *A Kripke structure K is a triple $(\mathcal{W}, \Gamma, K_I)$. Thereby, $\mathcal{W}$ is a non-empty set of worlds, Γ is a finite set of accessibility relations, one accessibility relation γ_i for each agent i, and K_I is a K-interpretation.*

[1] In the following, we will abbreviate agents by numbers, and we suppose only a finite number of agents to be given.

The *satisfiability* of an $\mathcal{ALC}_\mathcal{B}$-formula F *in a Kripke structure* $K = (\mathcal{W}, \Gamma, K_I)$ *and a world* $w \in \mathcal{W}$, written as $K, w \models F$, is recursively defined by:

$$
\begin{array}{lll}
K,w \models C = D & \text{iff} & \{d \mid (d,w) \in C^{K_I}\} = \{d \mid (d,w) \in D^{K_I}\} \\
K,w \models C \neq D & \text{iff} & \{d \mid (d,w) \in C^{K_I}\} \neq \{d \mid (d,w) \in D^{K_I}\} \\
K,w \models a : C & \text{iff} & (a,w) \in C^{K_I} \\
K,w \models aRb & \text{iff} & (a,b,w) \in R^{K_I} \\
K,w \models \Box_i G & \text{iff} & K,w' \models G \text{ for each world } w' \text{ with } (w,w') \in \gamma_i \\
K,w \models \neg\Box_i G & \text{iff} & \text{there is a world } w' \text{ with } (w,w') \in \gamma_i \text{ and } K,w' \not\models G
\end{array}
$$

where G is an $\mathcal{ALC}_\mathcal{B}$-formula, C, D are concepts, a, b are objects, and R is a role. A set $F_1, \ldots, F_n$ of $\mathcal{ALC}_\mathcal{B}$-formulas is *satisfiable* iff there exists a Kripke structure $K = (\mathcal{W}, \Gamma, K_I)$ and a world $w_0 \in \mathcal{W}$ such that $K, w_0 \models F_i$ for $i = 1, \ldots, n$. We then write $K \models F_1, \ldots, F_n$.

In the following we will use the notion *modality* to denote (negated) indexed $\Box$ operators, and $\mathcal{ALC}_\mathcal{B}$-formulas without any modalities are called $\mathcal{ALC}$*-formulas*. For example, the $\mathcal{ALC}_\mathcal{B}$-formula $\Box_i\neg\Box_j$(*vehicle-1* : *truck*) contains the modalities $\Box_i$ and $\neg\Box_j$, and the $\mathcal{ALC}_\mathcal{B}$-formula *vehicle-1* : *truck* is an $\mathcal{ALC}$-formula. Furthermore, we use the standard notation $\Diamond_i F$ as an abbreviation for $\neg\Box_i\neg F$.

3 Testing Satisfiability of $\mathcal{ALC}_\mathcal{B}$-formulas

Using $\mathcal{ALC}_\mathcal{B}$-formulas, a "real world" and belief of agents can be defined as follows. The real world is given by a finite set of $\mathcal{ALC}$-formulas, and the belief of agent i is given by a finite set of $\mathcal{ALC}_\mathcal{B}$-formulas with the leading modality $\Box_i$. Of course, we do not only want to represent a world and beliefs of agents, but we are interested in algorithms to test (i) consistency of the represented facts, i.e., whether a given set of $\mathcal{ALC}_\mathcal{B}$-formulas is satisfiable, and (ii) whether an $\mathcal{ALC}_\mathcal{B}$-formula is a logical consequence of a given set of $\mathcal{ALC}_\mathcal{B}$-formulas. In this section we will give an algorithm for testing satisfiability of a set of $\mathcal{ALC}_\mathcal{B}$-formulas. Building upon this we will show how to decide whether or not an $\mathcal{ALC}_\mathcal{B}$-formula is a logical consequence from a given set of $\mathcal{ALC}_\mathcal{B}$-formulas in Section 4.

By definition, a set $F_1, \ldots, F_n$ of $\mathcal{ALC}_\mathcal{B}$-formulas is satisfiable iff there exists a Kripke structure K such that $K \models F_1, \ldots, F_n$. Of course, we are not interested in arbitrary Kripke structures to satisfy $F_1, \ldots, F_n$, but only in Kripke structures which interpret the belief operators $\Box$ in $F_1, \ldots, F_n$ in such a way that they satisfy the properties described in Section 1. We therefore introduce the notion of KD45 Kripke structures.

Definition 3.1 *A set* $F_1, \ldots, F_n$ *of* $\mathcal{ALC}_\mathcal{B}$*-formulas is* KD45-satisfiable *iff there exists a Kripke structure* $K = (\mathcal{W}, \Gamma, K_I)$ *which satisfies* $F_1, \ldots, F_n$ *and which has the properties*

$$
\begin{array}{ll}
(P1) & \text{if } K,w \models \Box_i F \text{ then } K,w \models \neg\Box_i\neg F \\
(P2) & \text{if } K,w \models \Box_i F \text{ then } K,w \models \Box_i\Box_i F \\
(P3) & \text{if } K,w \models \neg\Box_i F \text{ then } K,w \models \Box_i\neg\Box_i F
\end{array}
$$

for each $\mathcal{ALC}_\mathcal{B}$*-formula* F*, for each agent* i*, and for each world* $w \in \mathcal{W}$*. A Kripke structure which satisfies (P1), (P2), and (P3) is called* KD45 Kripke structure.

Property (P1) corresponds to "an agent cannot believe in both a fact and its negation", (P2) to "if an agent believes something, then he believes that he believes it", and (P3) to "if an agent does not believe in a fact then he believes that he does not believe in this fact". The property "agents must be able to reason on the basis of their beliefs", is guaranteed by choosing Kripke structures for the representation of belief (cf., e.g., [16]). We will assume in the following that all these properties are mutually believed, i.e., each agent's belief has these properties, each agent believes that each agent's belief has these properties and so on. It is a well-known fact that $K = (\mathcal{W}, \Gamma, K_I)$ is a KD45 Kripke structure if the accessibility relation γ_i of each agent i is serial, Euclidean, and transitive (see, e.g., [25]). A relation $\gamma \subseteq \mathcal{W} \times \mathcal{W}$ is *serial* iff for each u in $\mathcal{W}$ there is a v in $\mathcal{W}$ such that $(u,v) \in \gamma$, *Euclidean* iff for all u, v, w in $\mathcal{W}$ holds: if $(u,v) \in \gamma$ and $(u,w) \in \gamma$ then $(v,w) \in \gamma$, and *transitive* iff for all u, v, w in $\mathcal{W}$ holds: if $(u,v) \in \gamma$ and $(v,w) \in \gamma$ then $(u,w) \in \gamma$.

To keep notation simple we transform $\mathcal{ALC}_\mathcal{B}$-formulas into negation normal form. An $\mathcal{ALC}_\mathcal{B}$-formula (concept) is in *negation normal form* iff in the formula (concept) negation signs occur immediately in front of atomic concepts only. Concepts can be transformed into an equivalent negation normal form by rules like $\neg(\forall R.C) \rightarrow \exists R.\neg C$ where C is a concept and R is a role (see, e.g., [19]). Building upon this, $\mathcal{ALC}_\mathcal{B}$-formulas can be transformed into negation normal form by the rules $\neg\neg F \rightarrow F$, $\neg\Box_i F \rightarrow \Diamond_i \neg F$, $\neg\Diamond_i F \rightarrow \Box_i \neg F$, $\neg(C = D) \rightarrow a_n : (C \sqcap \neg D) \sqcup (\neg C \sqcap D)$, $\neg(C \neq D) \rightarrow C = D$, and $\neg(a : C) \rightarrow a : \neg C$ where F is an $\mathcal{ALC}_\mathcal{B}$-formula, C, D are concepts, a is an object, and a_n is a new object. It is easy to verify that an $\mathcal{ALC}_\mathcal{B}$-formula is KD45-satisfiable iff its negation normal form is KD45-satisfiable (for details, see [21]). If F is an $\mathcal{ALC}_\mathcal{B}$-formula in negation normal form it has a (possibly empty) leading sequence $\circ^* = \circ_{i_1} \ldots \circ_{i_m}$ of non-negated modalities where each $\circ_{i_j}$ is either $\Box$ or $\Diamond$ and each index i_j is an agent. We now replace each subsequence of modalities indexed with the same agent in $\circ^*$ by the last modality in this subsequence. The obtained $\mathcal{ALC}_\mathcal{B}$-formula is called the *KD45 normal form F' of F*. For example, the KD45 normal form of $\Box_1\Diamond_1\Diamond_2\Box_2\Box_1(a : C)$ is given by $\Diamond_1\Box_2\Box_1(a : C)$. As an immediate consequence of Proposition 4.27 in [10], for each KD45 Kripke structure $K = (\mathcal{W}, \Gamma, K_I)$ and for each world $w \in \mathcal{W}$ holds that $K, w \models F$ iff $K, w \models F'$. In the following we assume each $\mathcal{ALC}_\mathcal{B}$-formula to be in KD45 normal form (and thus especially in negation normal form).

To formulate a calculus for testing KD45 satisfiability of $\mathcal{ALC}_\mathcal{B}$ formulas we introduce the notions of labeled $\mathcal{ALC}_\mathcal{B}$-formulas and of a world constraint system. A *labeled $\mathcal{ALC}_\mathcal{B}$-formula* consists of an $\mathcal{ALC}_\mathcal{B}$-formula F together with a label w, written as $F \parallel w$. Thereby, w is a constant representing a world in which F holds. A *world constraint* is either a labeled $\mathcal{ALC}_\mathcal{B}$-formula or a term $w \bowtie_i w'$. The constants w and w' represent worlds and $\bowtie_i$ represents the accessibility relation of agent i. A *world constraint system* is a finite, non-empty set of world constraints. A Kripke structure $K = (\mathcal{W}, \Gamma, K_I)$ satisfies a world constraint system W iff for each label w in W there is a world $w^K \in \mathcal{W}$ such that (i) $K, w^K \models F$ for each world constraint $F \parallel w$ in W and (ii) $(w^K, v^K) \in \gamma_i$ for each world constraint $w \bowtie_i v$ in W. A world constraint system W is (KD45-)*satisfiable* iff there exists a (KD45) Kripke structure which satisfies W. For testing KD45-satisfiability of a set $F_1, \ldots, F_n$ of $\mathcal{ALC}_\mathcal{B}$-formulas we firstly translate them into a world constraint system. The world constraint system W is *induced by* $F_1, \ldots, F_n$ iff $W = \{F_1 \parallel w_0, \ldots, F_n \parallel w_0\}$, where w_0 is a new constant (which, intuitively, represents the real world). Obviously, $F_1, \ldots, F_n$ are KD45-satisfiable iff W is KD45-satisfiable. KD45-satisfiability of a world constraint system W is tested by the *frame algorithm* which has a world constraint system as input that is induced by a finite number of $\mathcal{ALC}_\mathcal{B}$-formulas and which successively adds new world constraints to W by applying the

$W \rightarrow_{\Diamond} \{w \bowtie_i v, F \,||\, v, \Diamond_i F_1 \,||\, v, \ldots, \Diamond_i F_n \,||\, v, G_1 \,||\, v, \Box_i G_1 \,||\, v, \ldots, G_m \,||\, v, \Box_i G_m \,||\, v\} \cup W$

if $\Diamond_i F||w, \Diamond_i F_1||w, \ldots, \Diamond_i F_n||w$ are the world constraints with leading modality $\Diamond_i$ in W, $\Box_i G_1 \,||\, w, \ldots, \Box_i G_m \,||\, w$ are the world constraints with leading modality $\Box_i$ in W, there is no label u in W such that the world constraints $F \,||\, u$, $\Diamond_i F_1 \,||\, u, \ldots, \Diamond_i F_n \,||\, u$, $G_1 \,||\, u, \ldots, G_m \,||\, u$, $\Box_i G_1 \,||\, u, \ldots, \Box_i G_m \,||\, u$ are exactly the labeled $\mathcal{ALC}_\mathcal{B}$-formulas with label u in W, and v is a new label.

$W \rightarrow_{\Diamond_\circ} \{w \bowtie_i u\} \cup W$

if $\Diamond_i F||w, \Diamond_i F_1||w, \ldots, \Diamond_i F_n||w$ are the world constraints with leading modality $\Diamond_i$ in W, $\Box_i G_1 \,||\, w, \ldots, \Box_i G_m \,||\, w$ are the world constraints with leading modality $\Box_i$ in W, there is a label u in W such that the world constraints $F \,||\, u$, $\Diamond_i F_1 \,||\, u, \ldots, \Diamond_i F_n \,||\, u$, $G_1 \,||\, u, \ldots, G_m \,||\, u$, $\Box_i G_1 \,||\, u, \ldots, \Box_i G_m \,||\, u$ are exactly the labeled $\mathcal{ALC}_\mathcal{B}$-formulas with label u in W, and $w \bowtie_i u$ is not in W

$W \rightarrow_{\Box} \{w \bowtie_i v, G_1 \,||\, v, \Box_i G_1 \,||\, v, \ldots, G_m \,||\, v, \Box_i G_m \,||\, v\} \cup W$

if no world constraint of the form $\Diamond_i F||w$ is in W, $\Box_i G_1||w, \ldots, \Box_i G_m||w$ are the world constraints with leading modality $\Box_i$ in W, there is no label u in W such that the world constraints $G_1||u, \ldots, G_m||u$, $\Box_i G_1||u, \ldots, \Box_i G_m||u$ are are exactly the labeled $\mathcal{ALC}_\mathcal{B}$-formulas with label $u$in W, and v is a new label.

$W \rightarrow_{\Box_\circ} \{w \bowtie_i u\} \cup W$

if no world constraint of the form $\Diamond_i F \,||\, w$ is in W, $\Box_i G_1 \,||\, w, \ldots, \Box_i G_m \,||\, w$ are the world constraints with leading modality $\Box_i$ in W, there is a label u in W such that $G_1 \,||\, u, \ldots, G_m \,||\, u$, $\Box_i G_1 \,||\, u, \ldots, \Box_i G_m \,||\, u$ are are exactly the labeled $\mathcal{ALC}_\mathcal{B}$-formulas with label u in W, and $w \bowtie_i u$ is not in W.

Figure 1: Propagation rules of the $\mathcal{ALC}_\mathcal{B}$ frame algorithm.

four propagation rules in Figure 1. Thus, the result of the frame algorithm with input W is a (modified) world constraint system W'.

The intuitive idea behind these propagation rules is as follows: Firstly, for $W \rightarrow_{\Diamond} W'$, if there is a world constraint $\Diamond_i F \,||\, w$ in W we add a world v such that (i) v is accessible from w by agent i and (ii) $F \,||\, v$ holds. Furthermore, whenever $\Diamond_i F_j \,||\, w$ is in W we add $F_j \,||\, v$ because of property (P_3), and whenever $\Box_i G_k \,||\, w$ is in W we add both $\Box_i G_k \,||\, v$ and $G_k \,||\, v$ because of property (P_2) and the definition of $\Box_i$. This rule is similar to the unsigned prefixed KD45 tableaux rules in [13]. Secondly, for $W \rightarrow_{\Box} W'$, if $\Box_i G_1 \,||\, w, \ldots, \Box_i G_m \,||\, w$ are in W but there is no world u accessible from w by agent i, we have to introduce a new world v—accessible from w by agent i—where $G_1 \,||\, v, \ldots, G_m \,||\, v$ and $\Box_i G_1 \,||\, v, \ldots, \Box_i G_m \,||\, v$ holds. This is due to the properties (P_1) and (P_2) of KD45 Kripke structures. Finally, the rules $\rightarrow_{\Diamond_\circ}$ and $\rightarrow_{\Box_\circ}$ are used to guarantee termination of applying propagation rules. Termination of the frame algorithm is stated by the next proposition which is proved in [21].

Proposition 3.2 *If W is a world constraint system which is induced by a finite set of $\mathcal{ALC}_{\mathcal{B}}$-formulas, there is no infinite chain of applications of propagation rules to W.*

Thus, the application of the frame algorithm to a world constraint system W induced by a finite set of $\mathcal{ALC}_{\mathcal{B}}$-formulas $F_1, \ldots, F_n$ terminates and results a world constraint system, say W'. In order to test KD45-satisfiability of W', for each label w in W' we compute the set of all those $\mathcal{ALC}_{\mathcal{B}}$-formulas in W' which are labeled by w and which do not contain any modality. That means, such a set contains only $\mathcal{ALC}$-formulas and is therefore called the *$\mathcal{ALC}$ test set of label w*. More formally, if W is a world constraint system, the $\mathcal{ALC}$ test set $A(w)$ of label w in W is given by the set $\{F \mid F \,||\, w \in W$ and F does not contain any modality$\}$. The following theorem states that a finite set $F_1, \ldots, F_n$ of $\mathcal{ALC}_{\mathcal{B}}$-formulas is KD45-satisfiable iff the $\mathcal{ALC}$ test set $A(w)$ of each label in W' is satisfiable. Thereby, W' is the result of the frame algorithm with input $\{F_1 \,||\, w_0, \ldots, F_n \,||\, w_0\}$. A proof is given in [21].

Theorem 3.3 *Let $F_1, \ldots, F_n$ be a finite set of $\mathcal{ALC}_{\mathcal{B}}$-formulas, and let W be the world constraint system which is induced by $F_1, \ldots, F_n$. If W' is the result of the frame algorithm with input W, then the set $F_1, \ldots, F_n$ is KD45-satisfiable iff the $\mathcal{ALC}$ test set $A(w)$ of each label w in W' is satisfiable.*

Summing up, we obtain an algorithm for testing KD45-satisfiability of $\mathcal{ALC}_{\mathcal{B}}$-formulas $F_1, \ldots, F_n$ which works as follows: Firstly, the frame algorithm is applied to the world constraint system W which is induced by $F_1, \ldots, F_n$. Then, for each label w in the resulting world constraint system the $\mathcal{ALC}$ test set $A(w)$ is tested on satisfiability. If one of these test sets is unsatisfiable, the algorithm returns "KD45-unsatisfiable", otherwise it returns "KD45-satisfiable". An algorithm for testing satisfiability of $\mathcal{ALC}$ test sets has been given in [20].

4 Restricted $\mathcal{ALC}$-TBoxes and Computing $\mathcal{ALC}_{\mathcal{B}}$-Inferences

In Section 2 we defined an $\mathcal{ALC}$-TBox as a set of terminological axioms of the form $C = D$ and $C \neq D$, where C and D are concepts. However, most of the existing terminological representation and inference systems (e.g., BACK [30], CLASSIC [7], $\mathcal{KRIS}$ [2]) only allow terminological axioms of the form $A = C$, where A is a primitive concept and C is a concept. Such a terminological axiom is called *(concept) definition* of A.[2] Building upon this, an $\mathcal{ALC}$-TBox is then defined as a finite set of terminological axioms which satisfies the following restrictions (*i*) each atomic concept appears at most once as the left hand side of a terminological axiom, and (*ii*) in this set cycles do not occur. Thereby, a set $\mathcal{S}$ of terminological axioms contains a *cycle* iff there exists a terminological axiom $A = C$ in $\mathcal{S}$ such that A occurs in the concept C' which arises from C by iterated substitutions of primitive concepts in C by the right hand sides of their definition in $\mathcal{S}$. For example, if A and B are primitive concepts the sets $\{A = A\}$ and $\{A = C \sqcap B, B = D \sqcup \exists R.A\}$ of terminological axioms contain cycles. In the following we will call $\mathcal{ALC}$-TBoxes satisfying the additional conditions described above *restricted $\mathcal{ALC}$-TBoxes* in order to distinguish them from the $\mathcal{ALC}$-TBoxes defined in Section 2.

[2] Often so-called concept specializations of the form $A \sqsubseteq C$ are allowed which abbreviate the terminological axiom $A = C \sqcap A^*$ where A^* is a new primitive concept.

It can be shown that each restricted $\mathcal{ALC}$-TBox $\mathcal{T}$ can be transformed into an equivalent restricted $\mathcal{ALC}$-TBox $\mathcal{T}'$ such that each right hand side of a concept definition in $\mathcal{T}'$ does only contain concepts which do not occur as a left hand side in $\mathcal{T}'$ (see, e.g., [19]). For example, if A_1, A_2, A_3 are primitive concepts, the restricted $\mathcal{ALC}$-TBox $\mathcal{T} = \{A_1 = A_2 \sqcap A_3, A_2 = C \sqcup D, A_3 = \exists R.C\}$ can be transformed into $\mathcal{T}' = \{A_1 = (C \sqcup D) \sqcap \exists R.C, A_2 = C \sqcup D, A_3 = \exists R.C\}$. Thus, each primitive concept A on the left hand side of a terminological axiom $A = C$ in $\mathcal{T}'$ can be seen as an abbreviation for the concept C. With this it is easy to verify that testing consistency of an $\mathcal{ALC}$-ABox $\mathcal{A}$ w.r.t. a restricted $\mathcal{ALC}$-TBox $\mathcal{T}$ is equivalent to testing consistency of the $\mathcal{ALC}$-ABox $\mathcal{A}'$, which arises from $\mathcal{A}$ by replacing each such abbreviation A in $\mathcal{A}$ by C, w.r.t. the empty $\mathcal{ALC}$-TBox. This test is known to be PSPACE-complete (see [18]). Using sophisticated optimization techniques, the runtime of this algorithm can even be compared with the runtime of polynomial (but incomplete) consistency test as, e.g., used in CLASSIC. This is a result in [1]. An algorithm for testing consistency of $\mathcal{ALC}$ test sets which may contain terminological axioms as defined in Section 2 has been given in [20]. As an easy consequence of a result by Fischer and Ladner [12] this test is exp-time complete. Moreover, when using more expressive terminological logics than $\mathcal{ALC}$ this test becomes undecidable. For the terminological logic $\mathcal{ALCF}$ this has been shown in [4].

Let now $\mathcal{S}$ be a set of $\mathcal{ALC}_{\mathcal{B}}$-formulas, W be the world constraint system which is induced by $\mathcal{S}$, and W' be the result of the frame algorithm with input W. Because of the above given effects on efficiency and decidability of testing satisfiability of an $\mathcal{ALC}$-TBox and an $\mathcal{ALC}$-ABox it is an interesting question whether or not the terminological axioms in each $\mathcal{ALC}$ test set $A(w)$ of a label in W' define a restricted $\mathcal{ALC}$-TBox. The answer to this question is not obvious: Consider the following example in which the terminological axioms in each $\mathcal{ALC}$ test set of a label in W' define a restricted $\mathcal{ALC}$-TBox though this is not the case for the set of terminological axioms occurring in $\mathcal{S}$. Let $\mathcal{S}$ be the set $\{\Diamond_1(A = C), \Diamond_1(A = D)\}$ where A is a primitive concept and C, D are concepts. Applying the frame algorithm to the world constraint system $W = \{\Diamond_1(A = C) \parallel w_0, \Diamond_1(A = D) \parallel w_0\}$ which is induced by $\mathcal{S}$ results a system W' which contains the $\mathcal{ALC}$ test sets $\{A = C\}$ and $\{A = D\}$. Thus, the terminological axioms in each $\mathcal{ALC}$ test set of W' define an restricted $\mathcal{ALC}$-TBox, respectively. On the other hand, starting with the set $\mathcal{S} = \{\Box_1(A = C), \Box_1(A = D)\}$ leads to one non-empty $\mathcal{ALC}$ test set, namely $\{A = C, A = D\}$ which does *not* define a restricted $\mathcal{ALC}$-TBox. The following theorem shows that it can be decided syntactically whether or not applying the frame algorithm to a world constraint system W—which is induced by a finite set of $\mathcal{ALC}_{\mathcal{B}}$-formulas—results only $\mathcal{ALC}$ test sets whose terminological axioms define a restricted $\mathcal{ALC}$-TBox. Its proof is given in [21]. Thereby, we use the following notions. If S is the sequence $o_1 \dots o_n$ of modalities, then $\mathsf{S}[j]$ denotes o_j and *indexes* (S) denotes $1, \dots, n$.

Theorem 4.1 *Let W be a world constraint system which is induced by a finite set of $\mathcal{ALC}_{\mathcal{B}}$-formulas and let W' be the result of the frame algorithm with input W. Then there is a label w in W' such that the $\mathcal{ALC}$ test set $A(w)$ contains the $\mathcal{ALC}$-formulas $F_1, \dots, F_n$ iff there are sequences $\mathsf{S}_1, \dots, \mathsf{S}_n$ in W such that (i) $\mathsf{S}_1 F_1 \parallel w_0, \dots, \mathsf{S}_n F_n \parallel w_0$ are in W, (ii) indexes$(\mathsf{S}_1) = \dots =$ indexes(S_n), and (iii) there is no position j such that for two sequences S' and S'' in $\{\mathsf{S}_1, \dots, \mathsf{S}_n\}$ holds $\mathsf{S}'[j] = \mathsf{S}''[j] = \Diamond_{i_j}$ for some agent i_j.*

Summing up, there are syntactical conditions which—if satisfied—guarantee that only restricted $\mathcal{ALC}$-TBoxes have to be tested in order to test KD45-satisfiability of a set of $\mathcal{ALC}_{\mathcal{B}}$-formulas. For example, these conditions could be given by (1) agents only have positive beliefs, i.e., negation signs do not occur in front of $\Box$-operators, and (2) for each sequence S of modalities holds that the set of $\mathcal{ALC}_\mathcal{B}$-formulas occurring in the scope of S define a restricted $\mathcal{ALC}$-TBox. For practical applications, however, such conditions seem not reasonable and, even worse, when computing logical consequences (see below) such syntactical conditions in general cannot be maintained. Hence, an algorithm for testing satisfiability of restricted $\mathcal{ALC}$-TBoxes only seems not to be appropriate. Nevertheless, Theorem 4.1 probably helps one to obtain an efficient implementation for testing KD45-satisfiability of a set of $\mathcal{ALC}_\mathcal{B}$-formulas.

We will now show how to use the KD45-satisfiability algorithm in order to test whether or not a given $\mathcal{ALC}_\mathcal{B}$-formula is a logical consequence from a set $F_1, \ldots, F_n$ of $\mathcal{ALC}_\mathcal{B}$-formulas. Again, we are only interested in KD45 Kripke structures and thus define: F is a *KD45 consequence* of $F_1, \ldots, F_n$ iff for each KD45 Kripke structure $K = (\mathcal{W}, \Gamma, K_I)$ and for each world w in $\mathcal{W}$ holds: if $K, w \models F_1, \ldots, F_n$, then $K, w \models F$. Firstly, let F be an $\mathcal{ALC}_\mathcal{B}$-formula of the form $\circ^*(C = D)$, $\circ^*(C \neq D)$, or $\circ^*(a : C)$, where $\circ^*$ is an abbreviation for a (possibly empty) sequence of modalities. Then, F is an KD45 consequence of $F_1, \ldots, F_n$ iff the set $F_1, \ldots, F_n, [\neg F]^*$ of $\mathcal{ALC}_\mathcal{B}$-formulas is not KD45-satisfiable, where $[\neg F]^*$ denotes the negation normal form of $\neg F$. Note, that $\neg F$ is an $\mathcal{ALC}_\mathcal{B}$-formula if F is of the above described form. If, on the other hand, F is of the form $\Box^*(aRb)$, where $\Box^*$ is an abbreviation for a (possibly empty) sequence of non-negated indexed $\Box$ operators, we cannot use this test method since negation signs are not allowed in $\mathcal{ALC}_\mathcal{B}$-formulas which contain a role instance. However, as a consequence of Theorem 4.1, it follows that $\Box_{i_1} \ldots \Box_{i_m}(aRb)$ is a KD45 consequence of a set $F_1, \ldots, F_n$ of $\mathcal{ALC}_\mathcal{B}$-formulas iff $\Box_{i_1} \ldots \Box_{i_m}(aRb)$ is one of the $\mathcal{ALC}_\mathcal{B}$-formulas in $F_1, \ldots, F_n$. For details, see [21].

Summing up, we have now given algorithms for deciding KD45-satisfiability of a given set of $\mathcal{ALC}_\mathcal{B}$-formulas, and, building upon this, for deciding whether or not a given $\mathcal{ALC}_\mathcal{B}$-formula F is a KD45 consequence of a given set $F_1, \ldots, F_n$ of $\mathcal{ALC}_\mathcal{B}$-formulas.

5 Conclusion

We have presented a two-dimensional extension of the concept language $\mathcal{ALC}$ which allows both reasoning on the objective level and reasoning on the level of epistemic alternatives. In the obtained language $\mathcal{ALC}_\mathcal{B}$, a world agents are acting in can be described by a set of terminological and assertional axioms. Furthermore, the beliefs agents have about this world, about the beliefs of other agents, and about their own beliefs can be described by terminological and assertional axioms with a leading indexed $\Box$ operator or a leading sequence of indexed $\Box$ operators. We presented sound and complete algorithms to check consistency of the represented beliefs and to decide whether an $\mathcal{ALC}_\mathcal{B}$-formula is logically entailed by a given set of $\mathcal{ALC}_\mathcal{B}$-formulas. Thus, it is possible to equip agents with a decidable component to represent beliefs that is much more expressive than representing beliefs via propositional logic.

The main restriction of the presented language $\mathcal{ALC}_\mathcal{B}$ lies in the fact that modalities are only allowed in front of terminological and assertional axioms. As an extension one might think of modalities in front of concepts as well. Such a language would allow to represent facts like "the things agent i believes to be expensive are exactly the things agent j believes to be cheap" by $\Box_i(\textit{expensive}) = \Box_j(\textit{cheap})$. Such an extended language, however, causes algorithmic problems that are beyond the scope of this paper and is currently investigated.

References

[1] F. Baader, E. Franconi, B. Hollunder, B. Nebel, and H.-J. Profitlich. An empirical analysis of optimization techniques for terminological representation systems or: Making $\mathcal{KRIS}$ get a move on. In *Proceedings of the 3rd International Conference on Knowledge Representation and Reasoning*, Cambridge, Mass., 1992.

[2] F. Baader and B. Hollunder. $\mathcal{KRIS}$: $\mathcal{K}$nowledge $\mathcal{R}$epresentation and $\mathcal{I}$nference $\mathcal{S}$ystem. *SIGART Bulletin*, 2(3):8–14, 1991.

[3] F. Baader and B. Hollunder. A terminological knowledge representation system with complete inference algorithms. In M. Richter and H. Boley, editors, *International Workshop on Processing Declarative Knowledge*, 1991.

[4] F. Baader and B. Hollunder. Embedding defaults into terminological knowledge representation formalisms. In *Proceedings of the 3rd International Conference on Knowledge Representation and Reasoning*, Cambridge, Mass., 1992.

[5] F. Baader and H.-J. Ohlbach. A multi-dimensional terminological knowledge representation language. In *Proceedings of IJCAI 93*, 1993. To appear.

[6] A. Bond and L. Gasser. *Readings in Distributed Artificial Intelligence.* Morgan Kaufmann, Los Angeles, CA, 1988.

[7] A. Borgida, R. J. Brachman, D. L. McGuinness, and L. A. Resnick. CLASSIC: a structural data model for objects. In *Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data*, pages 59–67, Portland, Oreg., 1989.

[8] R. Brachman and H. J. Levesque. Expressiveness and tractability in knowledge representation and reasoning. *Computational Intelligence*, 3:78–93, 1987.

[9] R. J. Brachman and J. G. Schmolze. An overwiev of the KL-ONE knowledge representation system. *Cognitive Science*, 9(2):171–216, 1985.

[10] B. F. Chellas. *Modal Logic: An Introduction.* Cambridge University Press, 1980.

[11] R. Fagin and J. Y. Halpern. Two views of belief. *Artificial Intelligence*, 54(3):275–317, April 1992.

[12] M. J. Fischer and R. E. Ladner. Propositional dynamic logic of regular programs. *Journal of Computer and System Science*, 18:194–211, 1979.

[13] M. Fitting. *Proof Methods for Modal and Intuitionistic Logics*, volume 169 of *Synthese Library*. D. Reidel Publishing Company, 1983.

[14] L. Gasser and M.N. Huhns. *Distributed Artificial Intelligence, Volume II.* Research Notes in Artificial Intelligence. Morgan Kaufmann, San Mateo, CA, 1989.

[15] J. Y. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, 37(3):549–587, 1990.

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

[17] J. Hintikka, editor. *Knowledge and Belief.* Cornell University Press, 1962.

[18] B. Hollunder. How to reduce reasoning to satisfiability checking of concepts in the terminological system $\mathcal{KRIS}$. To appear.

[19] B. Hollunder. Hybrid inferences in KL-ONE-based knowledge representation systems. In *14th German Workshop on Artificial Intelligence*, volume 251 of *Informatik-Fachberichte*, pages 38–47, Ebingerfeld, Germany, 1990. Springer.

[20] A. Laux. Integrating a modal logic of knowledge into terminological logics. Research Report RR-92-56, DFKI Saarbrücken, 1992.

[21] A. Laux. Representing belief in multi-agent worlds via terminological logics. Research Report RR-93-29, DFKI Saarbrücken, 1993.

[22] W. Lenzen. Recent work in epistemic logic. *Acta Philosophica Fennica*, 30:1–219, 1978.

[23] H. J. Levesque. Foundations of a functional approach to knowledge representation. *Artificial Intelligence*, 23:155–212, 1984.

[24] G. L. McArthur. Reasoning about knowledge and belief: A survey. *Computational Intelligence*, 4:223–243, 1988.

[25] J.-J. C. Meyer, W. van der Hoek, and G. A. W. Vreeswijk. Epistemic logic for computer science: A tutorial (part one). In *Bulletin of the EATCS*, volume 44, pages 242–270. European Association for Theoretical Computer Science, 1991.

[26] J.-J. C. Meyer, W. van der Hoek, and G. A. W. Vreeswijk. Epistemic logic for computer science: A tutorial (part two). In *Bulletin of the EATCS*, volume 45, pages 256–287. European Association for Theoretical Computer Science, 1991.

[27] R. C. Moore. Reasoning about knowledge and action. Technical Report 191, SRI International, 1980.

[28] R. C. Moore. A formal theory of knowledge and action. In J. R. Hobbs and R. C. Moore, editors, *Formal Theories of the Commonsense World*, pages 319–358. Ablex Publishing Corporation, 1985.

[29] B. Nebel. *Reasoning and Revision in Hybrid Knowledge Representation Systems.* Number 422 in LNAI. Springer, 1990.

[30] B. Nebel and K. von Luck. Hybrid reasoning in BACK. In Z. W. Ras and L. Saitta, editors, *Methodologies for Intelligent Systems*, volume 3, pages 260–269. North-Holland, 1988.

[31] K. Schild. A correspondence theory for terminological logics: Preliminary report. In *Proceedings of IJCAI 91*, pages 466–471, 1991.

On the Relationship between Actions and Beliefs

Claus-Rainer Rollinger
Universität Osnabrück
Institut für Semantische Informationsverarbeitung
D-19069 Osnabrück

Extended Abstract *

(1) We have two different classification schemata for what we know: **knowledge vs. belief** on the one hand and **plans, intentions, goals, explanations, actions, situations, and events** on the other hand. The first differs with respect to the certainty that something is true in the real world, and the second differs with respect to the function of what is known or believed to be true in the real world.

(2) In Philosophy the **standard definition of Knowledge** is the following:
A knows p if and only if
- p is true and
- A believes p and
- A has good reasons to believe p.

In AI we have great problems with *p is true*. Usually we cannot decide whether p is true in reality or not. One reason for this is the dynamic of reality not being under our control. If we do not claim that p has to be true, then good or very good reasons *to know* something and less good reasons *to believe* other things is all we have.

(3) In AI the knowledge of an agent about the world is seen as a **mental model of the world** which is incomplete and uncertain. That's why default logic, nonmontonic logic, fuzzy logic, and modal logic have to be very tide connected with epistemic logic.

(4) It makes a big difference with respect to what action we will perform if we know a proposition p or if we (only) believe it. If a proposition that we believe is very important for us, actions are performed in order to **change the epistemic status** from *believe p* to *know p* (or *know not p*). On the other hand on the basis of observed actions we can infer whether an actor knows p or just believes p.

(5) A says to B: I know p.
B answers: I don't believe that.

In this case the answer of B is ambiguous: 1. B may not believe that A knows p and says nothing about the truth of p and 2. B doubts p and says that B is wrong in what he believes to know. The point of interest are the justifications of A and B for what they believe or know. These **justifications relates the epistemic states to actions** A and/or B have performed by themselves, or to actions they have observed, or to actions someone has told them about.

(6) Whether we know p or believe p depends on the **knowledge source** p comes from. If the knowledge source for p is our own perceptional system we usually classify p as something we know. If someone who has heard it from a third party tells us p, then we are uncertain about p and will qualify p as something we only believe.

* This is an extended abstract of the internal working paper (in German) C.-R. Rollinger (1993): Zum Unterschied zwischen Glauben und Wissen, Universität Osnabrück, 49069 Osnabrück.

(7) Actions have to be seen in the context of plans and goals of the acting agents. Thus we are able to characterise actions by the following two principles:
1. The actor can **predict the effect of his action** on the basis of his mental model of the world and the action he will perform. The effect of an action is a change in the world that is observed and thus changes the mental model.
2. There has to be an **alternative to the action**. At least it must be possible for the agent to omit the action. Otherwise its not an action but an event.

(8) Can we *know* what will happen in the **future**? Of course we can. If we have a stone in our hand we know that it will fall down if we open the hand.

(9) The **quality of a mental model** (a model of some part of the real world, a theory about this part of the world) determines the quality of the explanations of epistemic states and the quality of the predicted effects of actions.

(10) The agents act in the real world (they change it) on the basis of a mental model and observe the real world (the changes) in order to compare the predictions with the observations. If a difference is found, the mental model has to be optimised. Thus acting is the same as **learning by doing experiments**. False predictions and/or explanations call for theory revision.

(11) A **plan is a sequence of actions**, leading to a goal state. Selecting a plan as the one that will be performed, presupposes, that all preconditions of all actions of this plan can be fulfilled. But it's not the case that each single possibility has to be checked before performing the first action of the plan.

(12) **Understanding events** means to assign meanings to observed events and to integrate them into some temporal event structure. There are no intentional states related to the meanings of events.

(13) **Understanding actions** on the other side means to assign meanings to observations, but to make explicit the goals of the actions and to identify the plans the actions are part of. At least the intentions of the agents have to be made explicit. This implies that the meaning of an observed action has an intentional part, a variable to be instantiated. By this actions can be differed from events.

(14) The intentional effect of a speech act is a specific change in the addressee's mental model that itself produces an action e.g. an answer. In this sense speech acts are **indirect actions**.

A Fugue on the Themes of Awareness Logic and Correspondence

(preliminary version)

Elias Thijsse[‡] and Heinrich Wansing[§]

1 Introduction

Kripke models for general awareness [FH88] in which no restrictions are imposed on the accessibility relation between worlds are called *sieve models* in [T92]. Sieve models form a very flexible and therefore powerful generalization of ordinary Kripke models for normal modal propositional logics. They are suitable for modelling epistemic logics which are free from the so-called paradoxes of logical omniscience ([FH88], [W90], [T92]) and nevertheless have 'empirical content' in the sense of not characterizing *every* system of normal modal logic ([W89], [PW89]). Notwithstanding their modelling capacities, Konolige has raised certain objections against sieve models. Konolige [K86, p. 246] states that in contrast to the possible worlds semantics for epistemically interpreted modal logic, in the case of models for general awareness "the connection between accessibility conditions and belief is ruptured". We consider this reservation against Fagin and Halpern's logic of general awareness as boiling down to doubts whether there exists a correspondence theory for awareness logic wrt sieve models. In this short note we present a non-compositional translation from epistemic formulas into first-order logic as the essential ingredient of such a correspondence theory and define a notion of bisimulation, which is appropriate for sieve models.

In combination, [W90] and [T92] show that the sieve model semantics is equivalent to Rantala's non-normal worlds semantics [R82a, 82b]. In order to further underline the power of sieve models, we shall present a canonical Rantala model for $S1$, a system which, as Cresswell [C92] pointed out, "is not so simple" and for which no relational semantics seems to be known.

2 Awareness logic and sieve models

The vocabulary of (one agent) awareness logic is that of classical propositional logic in $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$ together with the unary modal operators A 'awareness', L 'implicit

[‡]Institute for Language Technology & AI (ITK), Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands, e-mail: thysse@kub.nl

[§]Institute of Logic and Philosophy of Science, University of Leipzig, Augustusplatz 9, 04109 Leipzig, Germany, e-mail: wansing@rz.uni-leipzig.de

belief', and B 'explicit belief'. Well-formed formulas (wffs) are built up in the obvious way, starting from a set of propositional variables $\{p_0, p_1, \ldots\}$ (finite or infinite). The set of all wffs is denoted by WFF. We write $\chi[\varphi]$ to indicate that φ is a subformula of χ. The result of substituting ψ for a marked occurrence of φ in χ is denoted by $\chi[\varphi : \psi]$.

A sieve model is a structure $M = \langle W, \mathsf{R}, \mathcal{A}, V\rangle$ where W, the set of possible worlds, is non-empty, $\mathsf{R} \subseteq W \times W$, $V : \{p_i\}_i \times W \longrightarrow \{0, 1\}$, and $\mathcal{A} : W \longrightarrow 2^{WFF}$. Truth of a wff φ at $w \in W$ in M ($M, w \models \varphi$) is defined as follows:

$$
\begin{array}{lll}
M, w \models p_i & \text{iff} & V(p_i, w) = 1, \text{ for every } i \\
M, w \models \neg\varphi & \text{iff} & \text{not } M, w \models \varphi \\
M, w \models \varphi \wedge \psi & \text{iff} & M, w \models \varphi \;\&\; M, w \models \psi \\
M, w \models \varphi \vee \psi & \text{iff} & M, w \models \varphi \text{ or } M, w \models \psi \\
M, w \models \varphi \rightarrow \psi & \text{iff} & M, w \models \varphi \Rightarrow M, w \models \psi \\
M, w \models \varphi \leftrightarrow \psi & \text{iff} & M, w \models \varphi \Leftrightarrow M, w \models \psi \\
M, w \models A\varphi & \text{iff} & \varphi \in \mathcal{A}(w) \\
M, w \models L\varphi & \text{iff} & (\forall w' \in W)\, w\mathsf{R}w' \Rightarrow M, w' \models \varphi \\
M, w \models B\varphi & \text{iff} & \varphi \in \mathcal{A}(w) \;\&\; (\forall w' \in W)\, w\mathsf{R}w' \Rightarrow M, w' \models \varphi.
\end{array}
$$

A wff φ is said to be valid in a sieve model $M = \langle W, \mathsf{R}, \mathcal{A}, V\rangle$ ($M \models \varphi$) iff ($\forall w \in W$) $M, w \models \varphi$. If $M = \langle W, \mathsf{R}, \mathcal{A}, V\rangle$ is a sieve model, $\langle W, \mathsf{R}, \mathcal{A}\rangle$ is said to be the sieve frame on which M is based. Validity on a sieve frame F is defined as validity in every sieve model based on F.

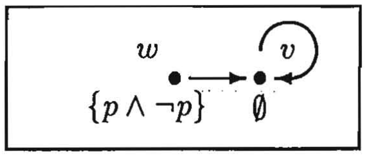
3 Correspondence for awareness logic

In his critical comments on Kripke models for general awareness, Konolige argues that the connection between accessibility conditions and belief is destroyed, since "[i]n the case of awareness, the formal correspondence between accessibility conditions and sets of awareness sentences breaks down" [K86, p. 246]. This explanation is not very precise. Should there be a formal correspondence between properties of the accessibility relation R and properties of the awareness function $\mathcal{A}$? This would then not be correspondence in the sense of modal correspondence theory. Or should axiom schemes for the awareness connective A correspond in the sense of modal correspondence theory with properties of R alone? To expect this would seem unwarranted, because the evaluation clause for wffs $A\varphi$ refers to $\mathcal{A}$ and not to R. And since there are no sets of awareness sentences in Kripke semantics for epistemically interpreted modal logic, it is not clear what *breaks down* where. Konolige continues to ask what, for instance, is required for BBp to be true if Bp is true. He claims that one has to impose the following two conditions:

(1) $A\varphi \rightarrow LA\varphi$,

(2) $B\varphi \rightarrow AB\varphi$.

This is in fact not entirely correct. The combination of (1) and (2) is not equivalent to the axiom of 'positive introspection' $B\varphi \rightarrow BB\varphi$. For even if we require R to be transitive, as [FH88] do, thus obtaining $L\varphi \rightarrow LL\varphi$, the equivalence does not hold for the logic



of general awareness. Indeed there are simple frames verifying $B\varphi \to BB\varphi$ while not verifying (1), e.g.
So $F = \langle\{w, v\}, \mathsf{R}, \mathcal{A}\rangle$, with $\mathsf{R} = \{\langle w, v\rangle, \langle v, v\rangle\}$, $\mathcal{A}(w) = \{p \wedge \neg p\}$ and $\mathcal{A}(v) = \emptyset$.[1] It is easily checked that $F \models \neg B\varphi$, thus $F \models B\varphi \to BB\varphi$, yet $F \not\models A\varphi \to LA\varphi$.

On the other hand $B\varphi \to BB\varphi$ *is* equivalent, modulo the *KD45* logic for L, to the combination of (2) with the weaker principle

(1′) $(L\varphi \wedge A\varphi) \to LA\varphi$.

For the sake of the argument we will focus on conditions (1) and (2), which, according to Konolige, "are not affected by the structure of accessibility". "Hence", he concludes, "the nice formal analysis of introspective properties obtainable in Kripke semantics is not present in the logic of general awareness". It is not clear what Konolige means by "not affected by the structure of accessibility". Schema (1), for example, corresponds with a second-order condition on sieve frames, saying that $\mathcal{A}$ is upwards monotone wrt R. To see this, consider any sieve model $M = \langle W, \mathsf{R}, \mathcal{A}, V\rangle$. M induces a first-order structure

$$\langle W, \mathsf{R}, \{[\![P_i]\!]\}_i, \{[\![P_\varphi]\!]\}_\varphi\rangle,$$

where $[\![P_i]\!] = \{w \in W \mid M, w \models p_i\}$, and $[\![P_\varphi]\!] = \{w \in W \mid \varphi \in \mathcal{A}(w)\}$. This construction gives rise to the following translation $(\cdot)^\sharp$ of wffs into the first-order language $\mathcal{L}_1$ with the binary predicate R such that $[\![R]\!] = \mathsf{R}$ and unary predicates P_i ($i \in \omega$) and P_φ ($\varphi \in WFF$):[2]

$$\begin{array}{lcl}
(p_i)^\sharp & = & P_i x \\
(\neg\varphi)^\sharp & = & \neg(\varphi)^\sharp \\
((\varphi \nabla \psi))^\sharp & = & ((\varphi)^\sharp \nabla (\psi)^\sharp), \quad \nabla \in \{\wedge, \vee, \to, \leftrightarrow\} \\
(L\varphi)^\sharp & = & \forall y(Rxy \to [y/x](\varphi)^\sharp) \\
(A\varphi)^\sharp & = & P_\varphi x \\
(B\varphi)^\sharp & = & (\forall y(Rxy \to [y/x](\varphi)^\sharp) \wedge P_\varphi x) \quad {}^3
\end{array}$$

Observation Let M^* be the first-order structure induced by the sieve model M. Then $M, w \models \varphi$ iff $M^*, w \models (\varphi)^\sharp$ for all $\varphi \in WFF$.

Applying $(\cdot)^\sharp$ to (1), we obtain

[1] F has been chosen such that it satisfies the conditions of *seriality*, *transitivity* and *Euclidicity* imposed by Fagin & Halpern.

[2] This translation was stimulated by a similar translation for Rantala semantics due to Johan van Benthem (private communication to the second author).

[3] Some technical details: Most of the parenthesis merely serves to make the scope of $\sharp$ unambiguous. Notice that the substitutions introduced in the clauses for L and B usually lead to renaming the quantified variable for embedded modals. (See the examples.)

$$\begin{aligned}&(A\varphi \to LA\varphi)^\sharp\\ =\;&(A\varphi)^\sharp \to (LA\varphi)^\sharp\\ =\;&P_\varphi x \to \forall y(Rxy \to [y/x](A\varphi)^\sharp)\\ =\;&P_\varphi x \to \forall y(Rxy \to P_\varphi y)\end{aligned}$$

Universal quantification over the world of evaluation x and the unary predicate P_φ gives $\forall x \forall P_\varphi(P_\varphi x \to \forall y(Rxy \to P_\varphi y))$, which is equivalent with $\forall x\forall y\forall P_\varphi(Rxy \to (P_\varphi x \to P_\varphi y))$. In other words, schema (1) is valid on a sieve frame $F = \langle W, \mathsf{R}, \mathcal{A}\rangle$ iff $\forall x\forall y(x\mathsf{R}y \Rightarrow \mathcal{A}(x) \subseteq \mathcal{A}(y))$. For Konolige's example $B\varphi \to BB\varphi$ notice that, since $B\varphi$ is interpreted as $L\varphi \wedge A\varphi$, we cannot expect the positive introspection scheme for explicit belief to correspond with a purely relational condition on sieve frames. Konolige's criticism simply is confused.

Yet we may wonder to which ordinary condition(s) the above translation amounts.

$$\begin{aligned}&(B\varphi \to BB\varphi)^\sharp\\ =\;&(B\varphi)^\sharp \to (BB\varphi)^\sharp\\ =\;&(\forall y(Rxy \to [y/x](\varphi)^\sharp) \wedge P_\varphi x) \to (\forall y(Rxy \to (\forall z(Ryz \to [z/x](\varphi)^\sharp) \wedge P_\varphi y)) \wedge P_{B\varphi}x)\end{aligned}$$

Here the last step is obtained by $[y/x](B\varphi)^\sharp = [y/x](\forall y(Rxy \to [y/x](\varphi)^\sharp) \wedge P_\varphi x) = [y/x](\forall z(Rxz \to [z/x](\varphi)^\sharp) \wedge P_\varphi x) = (\forall z(Ryz \to [z/x](\varphi)^\sharp) \wedge P_\varphi y)$. Universally quantifying over the free variables x, P_φ, $P_{B\varphi}$ and φ produces the higher-order formula

$$\forall x\forall\varphi\forall P_\varphi\forall P_{B\varphi}((\forall y(Rxy \to [y/x](\varphi)^\sharp)\wedge P_\varphi x) \to (\forall y(Rxy \to (\forall z(Ryz \to [z/x](\varphi)^\sharp)\wedge P_\varphi y))\wedge P_{B\varphi}x$$

which can be decomposed into:

(a) $\forall x\forall\varphi\forall P_\varphi\forall P_{B\varphi}(\forall y(Rxy \to [y/x](\varphi)^\sharp) \to (P_\varphi x \to P_{B\varphi}x))$.

(b) $\forall x\forall y\forall\varphi\forall P_\varphi((\forall u(Rxu \to [u/x](\varphi)^\sharp) \wedge Rxy) \to (P_\varphi x \to P_\varphi y))$.

(c) $\forall x\forall y\forall z\forall\varphi((\forall u(Rxu \to [u/x](\varphi)^\sharp) \wedge Rxy \wedge Ryz) \to [z/x](\varphi)^\sharp)$.

Since R is assumed to be transitive in the logic of general awareness, (c) automatically holds. The other two conditions amount to restricted forms of upwards monotonicity (of $\mathcal{A}$ wrt R) and B-closure (of $\mathcal{A}$), respectively:

- If $F, x \not\models \neg L\varphi$ (i.e. $L\varphi$ is satisfiable on F in x) and $\varphi \in \mathcal{A}(x)$ then $B\varphi \in \mathcal{A}(x)$.
- If $F, x \not\models \neg L\varphi$, $x\mathsf{R}y$ and $\varphi \in \mathcal{A}(x)$ then $\varphi \in \mathcal{A}(y)$.

These conditions are admittedly unusual and not even frame conditions in the strict sense, but adequately establish the correspondence.

Moreover, there are no obstacles to a correspondence theory based on $(\cdot)^\sharp$. The important notion of bisimulation, for example, can easily be modified so as to suit sieve models. A bisimulation C between sieve models $M = \langle W, \mathsf{R}, \mathcal{A}, V\rangle$ and $M' = \langle W', \mathsf{R}', \mathcal{A}', V'\rangle$ is defined by requiring that

$$wCw' \text{ implies } \mathcal{A}(w) = \mathcal{A}(w')$$

in addition to the standard conditions defining bisimulations between Kripke models (see [vB91,p.210]). By induction on φ one can show that epistemic formulas are invariant for bisimulations.

Observation For any bisimulation C between two sieve models M, M':

$$wCw' \text{ implies } M, w \models \varphi \text{ iff } M', w' \models \varphi.$$

One can then prove the following characterization:

Theorem A first order formula $\varphi(x)$ is equivalent to a $(\cdot)^\sharp$-translation of an epistemic formula iff $\varphi(x)$ is invariant for bisimulations.

PROOF. From the left to the right this is immediate by the above observation. The other direction is established by a simple reduction. Let $\varphi(x)$ be invariant for bisimulations between sieve models (or rather the induced first-order structures). Notice that a bisimulation between sieve models for the wffs of awareness logic amounts to an ordinary bisimulation for the standard uni-modal language $\mathcal{L}_0$ with modal operator L and propositional variables p_α (with $\alpha \in WFF$ or $\alpha \in \omega$). By bisimulation invariance for normal modal logic, the $\mathcal{L}_1$ formula φ is equivalent to the standard translation of the $\mathcal{L}_0$ formula ψ. Then the following backward translation $(\cdot)^\flat$ from $\mathcal{L}_0$ into WFF, essentially translating p_χ as $A\chi$ ($\chi \in WFF$), produces the desired formula $\psi^\flat$.

$$\begin{array}{lcl}
(p_i)^\flat & = & p_i \quad (i \in \omega) \\
(p_\chi)^\flat & = & A\chi \quad (\chi \in WFF) \\
(\neg\psi)^\flat & = & \neg(\psi)^\flat \\
((\psi\nabla\psi'))^\flat & = & ((\psi)^\flat\nabla(\psi')^\flat), \quad \nabla \in \{\wedge, \vee, \rightarrow, \leftrightarrow\} \\
(L\psi)^\flat & = & L(\psi)^\flat
\end{array}$$

It is easily checked that every sieve model $M = \langle W, \mathsf{R}, \mathcal{A}, V\rangle$ for WFF can be transformed into a standard Kripke model $M' = \langle W, \mathsf{R}, V'\rangle$ for $\mathcal{L}_0$, where V' extends V such that $V'(p_\chi, w) = 1 \Leftrightarrow \chi \in \mathcal{A}(w)$, and *vice versa*. If M^* is a first-order structure of the right similarity type, i.e. $M^* = \langle W, \mathsf{R}, \{[\![P_i]\!]\}_i, \{[\![P_\varphi]\!]\}_\varphi\rangle$, then it can be viewed as being induced by a sieve model M of the above form. So $M^*, w \models \varphi \Leftrightarrow M', w \models \psi \Leftrightarrow M, w \models \psi^\flat \Leftrightarrow M^*, w \models (\psi^\flat)^\sharp$. In all, φ is equivalent to $(\psi^\flat)^\sharp$. Q.E.D.

4 $S1$

The first Lewis system, $S1$, is a system of non-normal modal logic. According to Cresswell [C92] various attempted formulations of a relational semantics for $S1$ turned out to be defective, and no semantics of such a kind seems to be known. In this section we show that $S1$ is characterized by a suitable class of Rantala models and thus by a class of *relational models*. Although this is perhaps not the kind of intuitive semantics Cresswell is asking for, it may serve as a further indication of the power of Rantala models and hence of sieve models. $S1$ can be axiomatized as follows (cf. [HC68], [C92]):

(1) every wff obtained by uniform substitution from any theorem of classical propositional logic *CPL*
(2) $\Box\varphi \to \varphi$
(3) $(\Box(\varphi \to \psi) \wedge \Box(\psi \to \chi)) \to \Box(\varphi \to \chi)$
(4) the necessitated versions of (1) - (3)
(5) $\varphi, \varphi \to \psi / \psi$
(6) $\vdash \Box(\varphi \leftrightarrow \psi) / \vdash \Box(\chi[\varphi] \leftrightarrow \chi[\varphi : \psi])$.

A Rantala model is a structure $\langle W, W^*, \mathrm{R}, V\rangle$ where W and W^* are sets, $W \neq \emptyset$, $\mathrm{R} \subseteq (W \cup W^*) \times (W \cup W^*)$ and V is a function from $WFF \times (W \cup W^*)$ to $\{0,1\}$ such that for every $w \in W$ and every wff φ, ψ:

$$\begin{array}{lcll} V(\neg\varphi, w) & = 1 & \text{iff} & V(\varphi, w) = 0 \\ V(\varphi \wedge \psi, w) & = 1 & \text{iff} & V(\varphi, w) = V(\psi, w) = 1 \\ V(\varphi \vee \psi, w) & = 1 & \text{iff} & V(\varphi, w) = 1 \text{ or } V(\psi, w) = 1 \\ V(\varphi \to \psi, w) & = 1 & \text{iff} & V(\varphi, w) = 0 \text{ or } V(\psi, w) = 1 \\ V(\varphi \leftrightarrow \psi, w) & = 1 & \text{iff} & V(\varphi, w) = V(\psi, w) \\ V(\Box\varphi, w) & = 1 & \text{iff} & (\forall w' \in W \cup W^*)\, w\mathrm{R}w' \Rightarrow V(\varphi, w') = 1. \end{array}$$

The set W is called the set of *normal* worlds; elements from W^* are called *non-normal* worlds. There are no constraints on evaluating formulas at non-normal worlds. If $V(\varphi, w') = 1$, we also write $w' \models \varphi$, for any $w' \in W \cup W^*$. A wff φ is said to be valid in a Rantala model $\mathcal{M} = \langle W, W^*, \mathrm{R}, V\rangle$ ($\mathcal{M} \models \varphi$) iff $(\forall w \in W)\; w \models \varphi$. In order to characterize $S1$, we have to define a suitable class of valuations. A valuation V is said to be an $S1$ valuation iff for every wff φ, ψ, χ:

$\langle\Box 1\rangle$ $(\forall w^* \in W^*)\; w^* \models \varphi$ if φ is valid on the basis of *CPL*
$\langle\Box 2\rangle$ $(\forall w^* \in W^*)((\exists w \in W) w\mathrm{R}w^*) \Rightarrow w^* \models \Box\varphi \to \varphi$
$\langle\Box 3\rangle$ $(\forall w^* \in W^*)((\exists w \in W) w\mathrm{R}w^*) \Rightarrow w^* \models (\Box(\varphi \to \psi) \wedge \Box(\psi \to \chi)) \to \Box(\varphi \to \chi)$
$\langle 3\rangle$ $(\forall w^* \in W^*)\; (w^* \models \varphi \to \psi \;\&\; w^* \models \psi \to \chi) \Rightarrow w^* \models \varphi \to \chi$
$\langle 6\rangle$ $((\forall w \in W)\, (\forall w' \in W \cup W^*)\, w\mathrm{R}w' \Rightarrow w' \models \varphi \leftrightarrow \psi) \Rightarrow$
$(\forall w^* \in W^*)\; (w^* \models \chi[\varphi] \leftrightarrow \chi[\varphi : \psi]) \;\&\; (w^* \models \chi[\varphi] \Leftrightarrow w^* \models \chi[\varphi : \psi])$.

The class $\mathbf{A}$ of $S1$ models is the class of all Rantala models $\langle W, W^*, \mathrm{R}, V\rangle$ such that V is an $S1$ valuation and R is reflexive on W. A wff φ is said to be valid in $\mathbf{A}$ ($\mathbf{A} \models \varphi$) iff $\mathcal{M} \models \varphi$ for every $\mathcal{M} \in \mathbf{A}$.

Theorem $S1$ is characterized by $\mathbf{A}$: $\vdash_{S1} \varphi$ iff $\mathbf{A} \models \varphi$.

PROOF. Soundness is shown by induction on the length of proofs in $S1$. Among other things we have to show that instances of (6) preserve validity. Suppose for any $S1$ model that $(\forall w \in W)(\forall w' \in W \cup W^*)\; w\mathrm{R}w' \Rightarrow w' \models \varphi \leftrightarrow \psi$. We must show that, given an arbitrary $S1$-model, for every normal world w and every world w_1 such that $w\mathrm{R}w_1$: $(*)\; w_1 \models \chi[\varphi] \leftrightarrow \chi[\varphi : \psi]$. If w_1 is non-normal, then $(*)$ holds by $\langle 6\rangle$. If w_1 is normal, we proceed by induction on the complexity of χ. The only non-trivial case is $\chi = \Box\chi'$. It is sufficient to show that $(\forall w_2 \in W \cup W^*)\; w_1\mathrm{R}w_2 \Rightarrow (w_2 \models \chi'[\varphi] \Leftrightarrow w_2 \models \chi'[\varphi : \psi])$. If w_2 is normal, we use the induction hypothesis; if w_2 is non-normal, we use again $\langle 6\rangle$ again.

In order to prove completeness we construct the canonical model $\mathcal{M}_{S1} = \langle W, W^*, \mathrm{R}, V\rangle$ as follows:

$W = \{w \mid w \text{ is a maximal } S1\text{-consistent set of wffs}\}$

$W^* = \{w^* \mid w^* = \{\varphi \mid \Box\varphi \in w\} \text{ for some } w \in W\}$

$\mathsf{R} = \{\langle w, w^*\rangle \mid w \in W \;\&\; w^* = \{\varphi \mid \Box\varphi \in w\}\} \cup \{\langle w, w\rangle \mid w \in W\}$

$V : \mathit{WFF} \longrightarrow \{0,1\}$ is (partially) specified by:

(i) $(\forall w^* \in W^*)\ V(\varphi, w^*) = 1 \Leftrightarrow \varphi \in w^*$

(ii) $(\forall w \in W)\ V(p_i, w) = 1 \Leftrightarrow p_i \in w$, for all propositional variables p_i and

(iii) V satisfies the usual truth conditions on normal worlds.

One can easily verify that in $\mathcal{M}_{S1}$ $w \models \varphi$ iff $\varphi \in w$ for every wff φ and every (normal) world w (the *truth lemma*). Moreover, since $S1$ contains *CPL*, it is easily shown that if $\not\vdash_{S1} \varphi$, then there exists a maximal $S1$-consistent set which does not contain φ (the *Lindenbaum lemma*). Both lemmas can be used to show that $\mathcal{M}_{S1}$ in fact is in **A**. Reflexivity of R on W is immediate. It remains to be checked whether V is an $S1$ valuation. Consider $\langle 6\rangle$. Suppose $(\forall w \in W)(\forall w' \in W \cup W^*) w\mathsf{R}w' \Rightarrow w' \models \varphi \leftrightarrow \psi$. Then, in particular, for every $w^* \in W^*$, $w^* \models \varphi \leftrightarrow \psi$, and hence for every $w \in W$, $w \models \Box(\varphi \leftrightarrow \psi)$. It follows by the truth lemma that $\Box(\varphi \leftrightarrow \psi) \in w$ for every $w \in W$ and so (by the Lindenbaum lemma) that $\vdash_{S1} \Box(\varphi \leftrightarrow \psi)$. With (6) this implies $\vdash_{S1} \Box(\chi[\varphi] \leftrightarrow \chi[\varphi : \psi])$ (*) and so (reversing the above argument) $\Box(\chi[\varphi] \leftrightarrow \chi[\varphi : \psi]) \in w$ for every $w \in W$, thus for every $w \in W$: $w \models \Box(\chi[\varphi] \leftrightarrow \chi[\varphi : \psi])$. The latter implies $w^* \models \chi[\varphi] \leftrightarrow \chi[\varphi : \psi]$. But, since $w\mathsf{R}w$, we also have $w \models \chi[\varphi] \Leftrightarrow w \models \chi[\varphi : \psi]$. Now applying (6) to (*) yields $\vdash \Box(\Box\chi[\varphi] \leftrightarrow \Box\chi[\varphi : \psi])$, i.e. $w \models \Box(\Box\chi[\varphi] \leftrightarrow \Box\chi[\varphi : \psi])$ for all $w \in W$. Since again $w\mathsf{R}w$ this implies $w \models \Box\chi[\varphi] \leftrightarrow \Box\chi[\varphi : \psi]$ for any $w \in W$, and therefore $w \models \Box\chi[\varphi] \Leftrightarrow w \models \Box\chi[\varphi : \psi]$, and thus $w^* \models \chi[\varphi] \Leftrightarrow w^* \models \chi[\varphi : \psi]$. The usual contrapositive argument establishes completeness. Q.E.D.

References

[vB84] van Benthem, J., 'Correspondence theory', *Handbook of Philosophical Logic*, Gabbay & Günthner (eds.), Vol. 2, 1984, 167–247.

[vB91] van Benthem, J., *Language in Action. Categories, Lambdas and Dynamic Logic*, Studies in Logic Vol. 130, North-Holland, Amsterdam, 1991.

[C92] Cresswell, M., '$S1$ is not so simple', ms., Victoria University of Wellington/University of Massachusetts at Amherst, 1992.

[FH88] Fagin, R. & Halpern, J., 'Belief, awareness and limited reasoning', *Artificial Intelligence*, 34 (1988), 39–76.

[HC68] Hughes, G. & Cresswell, M., *An Introduction to Modal Logic*, Methuen, London, 1968.

[K86] Konolige, K., 'What awareness isn't: a sentential view of implicit and explicit belief', *Proceedings of* TARK I (Monterey CA), Morgan Kaufmann, 1986, 241–250.

[PW89] Pearce, D. & Wansing, H., 'On the methodology of possible worlds semantics, II: nonnormal worlds and propositional attitudes', technical report, Gruppe Logik, Wissenstheorie und Information, FU Berlin, 1989.

[R82a] Rantala, V., 'Impossible worlds semantics and logical omniscience', *Acta Philosophica Fennica* 35 (1982), 106–115.

[R82b] Rantala, V., 'Quantified modal logic: non-normal worlds and propositional attitudes', *Studia Logica* 41 (1982), 41–65.

[T92] Thijsse, E., *Partial Logic and Knowledge Representation*, Eburon Publishers, Delft, 1992.

[W89] Wansing, H., 'Bemerkungen zur Semantik nicht-normaler möglicher Welten', *Zeitschrift für mathematische Logik und Grundlagen der Mathematik* 35 (1989), 551–557.

[W90] Wansing, H., 'A general possible worlds framework for reasoning about knowledge and belief', *Studia Logica* 49 (1990), 523–539, 50 (1991) 359.

Epistemic Modalities and the Closed-World Assumption in Disjunctive Knowledge Representation Systems

Gerd Wagner
Gruppe Logik, Wissenstheorie und Information
Institut für Philosphie, Freie Universität Berlin
Habelschwerdter Allee 30, 14195 Berlin, Germany
gw@inf.fu-berlin.de

Abstract

After defining the general concept of a ***knowledge representation and reasoning system (KRS)*** we show how with each KRS $\boldsymbol{K}$ a ***disjunctive*** KRS $\boldsymbol{DK}$ can be associated in a natural way. The fundamental case of this construction is the extension of ***fact bases*** to ***disjunctive fact bases*** which correspond to Belnap's [1977] system of ***epistemic states***. In disjunctive fact bases one can distinguish between exclusive and inclusive disjunctive information, and express various kinds of incomplete information which can only be queried by means of appropriate epistemic modal operators. Introducing a modal operator for definite knowledge in $\boldsymbol{DK}$ yields an extended KRS $\boldsymbol{D^m K}$ by a construction based on an operational proof theory and not on the usual Kripke style model theory.

By adding rules to fact bases, resp. disjunctive fact bases, one obtains extended, resp. disjunctive, logic programs. We point out that there is a certain ambiguity in disjunctive rule-based systems concerning the local versus global application of rules. We also present a formalization of the CWA in the proposed framework of KRSs, and compare it with that of Gelfond [1992] where epistemic modal operators are used to express the CWA.

1 Introduction

The concept of a knowledge representation and reasoning system (KRS) consists essentially of two main components: an inference and an update operation manipulating knowledge bases as abstract objects,[1] together with a set of formal properties these operations may have. In general, there are no specific restrictions on the internal structure of a knowledge base. We will see, however, that a computational design can be achieved by 'compiling' incoming information into some normal form rather than leaving it in the form of arbitrarily complex formulas. This is the case, for instance, in Belnap's KRS[2] which can be considered as the paradigm for KRSs.

The concept of a KRS constitutes a useful framework for the classification and comparison of various AI systems and formalisms. It is more general than that of a logic (i.e. a consequence relation). A standard logic can be viewed as a special kind of KRS. On the other hand, by defining the inference and update operations procedurally, KRSs can serve as the basis for the operational definition of logics. There is even a strong analogy between the concept of a KRS and that of a Gentzen sequent system.

[1]This with respect to KR fundamental distinction was first proposed in [Levesque 1984] where the resp. operations are called *ASK* and *TELL*.

[2]Cf. [Belnap 1977].

Both relational and deductive databases can be considered as computational paradigms of real world KRSs. They implement a form of nonmonotonic reasoning caused by the use of negation-as-failure expressing default-implicit negative information. On the other hand, relational and deductive databases, as well as normal logic programs, are not capable of representing and processing explicit negative information. This shortcoming has led to the extension of logic programming by adding a second negation (in addition to negation-as-failure) as proposed independently in [Gelfond & Lifschitz 1990+1991] and in [Pearce & Wagner 1990] and [Wagner 1991]. We will call the general concept of an operator expressing default-implicit negative information in the style of negation-as-failure *weak negation*, while the concept of an operator expressing explicit negative information will be called *strong negation*.

The concept of a *vivid* KRS *(VKRS)* is a two-fold generalization:

1. it extends already known logics, such as Belnap's 4-valued or Nelson's constructive logic,[3] by adding *weak* negation, and
2. it extends already known knowledge representation systems, such as relational or deductive database systems, by adding *strong* negation.

In the framework of a VKRS, a specific meaning is assigned to the *Closed-World Assumption*:[4] it connects the use of weak and strong negation in combination with partially and totally represented predicates. If the Closed-World Assumption holds for a predicate, its weak negation implies its strong negation, in other words, an atomic sentence formed with such a predicate is already false if it is false by default. In the presence of disjunctive information the CWA can be related to epistemic introspection expressed by appropriate modal operators. We shall compare our approach to the CWA in disjunctive fact bases with the modal one proposed in [Gelfond 1992].

2 Knowledge Representation and Reasoning

The language of KRSs consists of the logical operator symbols $\wedge, \vee, \sim, -$ and 1 standing for conjunction, disjunction, strong negation, weak negation and the verum, respectively; the predicate symbols $p, q, r, \ldots$; the constant symbols $c, d, \ldots$ and variables $x, y, \ldots$. Notice that there are no functional terms but only variables and constants. Further connectives, such as epistemic modalities, and the (possibly restricted) quantifiers $\exists$ and $\forall$ can be optionally added to the language.

An *atom* is an atomic formula, it is called *proper*, if it is not 1. *Literals* are either atoms or strongly negated atoms. *Extended literals* are either literals or weakly negated literals. We use $a, b, \ldots$, $l, k, \ldots$, $e, f, \ldots$, and $F, G, H, \ldots$ as metavariables for atoms, literals, extended literals and formulas, respectively.[5] A variable-free expression is called *ground*. The set of all proper ground atoms (resp. literals, resp. extended literals) of a given language is denoted by At (resp. Lit, resp. XLit). If not otherwise stated, a formula is assumed to be ground. If *op-list* is a set of logical operators, say *op-list* $\subseteq \{1, -, \sim, \wedge, \vee, \rightarrow, \exists, \forall, \ldots\}$, then $L(\textit{op-list})$ denotes the respective set of well-formed formulas.

With each negation a complement operation for the resp. type of literal is associated: $\tilde{a} = \sim a$ and $\widetilde{\sim a} = a$, $\bar{l} = -l$ and $\overline{-l} = l$. These complements are also defined for sets of resp. literals

[3] See [Belnap 1977], resp. [Almukdad & Nelson 1984].

[4] Cf. [Reiter 1978].

[5] I will frequently just say "literal" when I, precisely speaking, mean a proper ground literal.

$L \subseteq$ Lit, and $E \subseteq$ XLit: $\tilde{L} = \{\tilde{l} : l \in L\}$, resp. $\overline{E} = \{\overline{e} : e \in E\}$. We distinguish between the positive and negative elements of $E \subseteq$ XLit by writing $E^+ := E \cap \text{Lit}$ and $E^- := \{l : -l \in E\}$.

The inference relation of a KRS is not uniform, in general. Only certain formulas may make sense for representing vivid knowledge, that is, there will be a specific representation language L_{Repr}, and a KB will be a finite collection of elements of L_{Repr}, possibly constrained in some way determined by the set L_{KB} of all admissible KBs: KB $\in L_{\text{KB}} \subseteq 2^{L_{\text{Repr}}}$.

Likewise, since not every formula may be appropriate as a sensible query, the set of admissible queries is specified by L_{Query}. The inference relation of a KRS, thus, is in general not based on a single universal language applying to premises as well as to queries (resp. consequences), but on two, usually different, languages: $\vdash \subseteq L_{\text{KB}} \times L_{\text{Query}}$.

Moreover, the axiomatization of standard consequence seems to be overidealised and not adequate for commonsense reasoning. This is no longer debated today in the case of the monotonicity postulate which requires that all previously obtainable inferences remain valid whenever new information comes in. In many forms of commonsense reasoning, such as default reasoning, and in many computational systems, such as Prolog, monotonicity is violated, and one has to look for other principles replacing it, or one can just drop it and allow for unrestricted nonmonotonicity.[6]

The basic scenario of a KRS consists of two operations: an inference operation processing queries posed to the KB, and an update operation processing input formulas entered by users or by other (e.g. sensoric) information suppliers. While in standard logics an update is a simple addition of a formula $F \in L$ to the premise set $X \subseteq L$, i.e. $X \cup \{F\}$, a KRS restricts the admissible inputs to elements of a specific input language L_{Input}, and an update is performed by processing the input formula in an appropriate way in order to add its information content to the KB.

In general, a KB can consist of any kind of data structures capable of representing knowledge, e.g. a set, or multiset, or sequence, of (logical) expressions, or a directed graph, etc. For the sake of simplicity, I will assume that a KB $\in L_{\text{KB}}$ is a finite set of expressions from a representation language, i.e. $L_{\text{KB}} \subseteq 2^{L_{\text{Repr}}}$. There will always be an informationally 'empty' KB, denoted by 0, which is not necessarily equal to the empty set.

2.1 Knowledge Representation Systems

A KRS is a quintuple:

$$\langle L_{\text{KB}}, \vdash, L_{\text{Query}}, \mathbf{Upd}, L_{\text{Input}} \rangle$$

where the inference relation $\vdash \subseteq L_{\text{KB}} \times L_{\text{Query}}$ is asscociated with a resp. inference operation C in the usual way:

$$C(\text{KB}) = \{F \in L_{\text{Query}} : \text{KB} \vdash F\}$$

and the update operation **Upd** takes a KB and an input formula F, and provides an appropriately updated KB:

$$\mathbf{Upd} : L_{\text{KB}} \times L_{\text{Input}} \rightarrow L_{\text{KB}}$$

An update may be simply an addition of a formula $F \in L_{\text{Input}} \cap L_{\text{Repr}}$ to the KB:

$$\mathbf{Upd}(\text{KB}, F) = \text{KB} \cup \{F\}$$

[6] It is an open debate if, and in what form, nonmonotonicity should be restricted in a KRS. The currently most popular proposal is *Cautious Monotonicity* which will be called *Lemma Compatibility* below.

But it may also be necessary to process the input formula in some way, especially if it is not an L_{Repr} expression. For instance, an RDB can be updated by a conjunction of atoms by simply adding all atoms to it.

Updates may be constrained by ***integrity constraints***. Following [Reiter 1990], we define a set of integrity constraints specified for a certain KB as any set of closed query formulas, $IC \subseteq L_{\text{Query}}$, such that an input $F \in L_{\text{Input}}$ is considered *illegal* whenever $\mathbf{Upd}(\text{KB}, F) \not\vdash G$ for some constraint $G \in IC$. Update attempts violating a constraint have to be rolled back by the system.

Notice that L_{Input} and L_{Repr} may be totally distinct (one may expect that an input formula is a kind of logical expression whereas a KB can be any kind of data structure).

The formulation of a KRS in terms of query and input processing was already implicitly present in Belnap's [1977] view of a KRS. In [Levesque 1984] it was proposed as a 'functional approach to knowledge representation'.[7]

Standard logics, such as classical or intuitionistic logic, can be viewed as special (very idealized) cases of KRSs. Here, the KB consists of a (possibly infinite) set of arbitrary first order sentences, and is usually called a 'theory'. The inference operation is monotonic. Updates are simple additions of formulas, and the representation language is equal to the query and input language, $L_{\text{Repr}} = L_{\text{Query}} = L_{\text{Input}}$.

2.2 Some Formal Properties of KRSs

If the language of a KRS contains the constant 1 for truth, it holds that $\text{KB} \vdash 1$, and $\mathbf{Upd}(\text{KB}, 1) = \text{KB}$.

Although it seems desirable, lemmas are not always redundant and compatible in KRSs. In other words, there are reasonable cases of KRSs, such as disjunctive fact bases (see below), which are not cumulative.

Lemma Redundancy (alias: Cut, Transitivity)

$$\text{KB} \vdash F \;\; \& \;\; \mathbf{Upd}(\text{KB}, F) \vdash G \;\Rightarrow\; \text{KB} \vdash G$$

Lemma Compatibility (alias Cautious Monotonicity, due to [Gabbay 1985])

$$\text{KB} \vdash F \;\; \& \;\; \text{KB} \vdash G \;\Rightarrow\; \mathbf{Upd}(\text{KB}, F) \vdash G$$

Lemma Redundancy and Compatibility can be combined in the following condition of

Cumulativity $\quad \text{KB} \vdash F \;\Rightarrow\; C(\mathbf{Upd}(\text{KB}, F)) = C(\text{KB})$

The following condition of Monotonicity seems to be too strong for commonsense reasoning. In a specifically restricted KRS, however, it may hold.

Monotonicity $\quad C(\text{KB}) \subseteq C(\mathbf{Upd}(\text{KB}, F))$

Clearly, Monotonicity implies Lemma Compatibilty.

It will be useful to compare knowledge bases in terms of their information content, that is, to have an informational ordering between KBs such that

$$\text{KB}_1 \leq \text{KB}_2 \text{ if } \text{KB}_2 \text{ contains more information than } \text{KB}_1.$$

[7] However, Levesque is not so much concerned with the investigation of the formal properties of KRSs in general, but rather with the modelling of the concept of meta-knowledge by means of an epistemic modality on the basis of classical logic.

The informational ordering '$\leq$' should be defined in terms of the structural components of knowledge bases and not in terms of higher-level notions (like derivability). The informationally empty KB will be denoted by 0. By definition, $0 \leq \text{KB}$ for all KBs.

Definition 1 (Definite Information) *A KB is called* definite *with respect to a constructive inference relation* $\vdash$, *if a disjunction cannot be inferred unless one of its disjuncts can,*

$$\text{KB} \vdash F \vee G \;\Rightarrow\; \text{KB} \vdash F \text{ or } \text{KB} \vdash G\,.$$

Notice that if the DeMorgan laws hold (which will be the case in all systems under consideration), this implies that a negated conjunction can only be inferred from a definite KB if the negation of one of its conjuncts can,

$$\text{KB} \vdash {\sim}(F \wedge G) \;\Rightarrow\; \text{KB} \vdash {\sim}F \text{ or } \text{KB} \vdash {\sim}G\,.$$

Definition 2 (Unique Representation) *A KRS enjoys the property of* Unique Representation *if the information contained in a KB is uniquely represented, i.e. if*

$$C(\text{KB}_1) = C(\text{KB}_2) \;\Rightarrow\; \text{KB}_1 = \text{KB}_2$$

A standard logical system, where a KB is a set of well-formed formulas, does not enjoy the Unique Representation property. On the other hand, a relational database, for instance, is a unique representation.

In general, it will not suffice in order to infer $-F$ that F fails (this will only be the case for definite KBs). However, the following restriction characterizes weak negation:

$$\text{(Inherent Consistency)} \qquad \text{KB} \vdash -F \;\Rightarrow\; \text{KB} \not\vdash F$$

Notice that this does not hold for classical negation. In fact, Inherent Consistency is violated by any negation satisfying the classical principle *ex contradictione sequitur quodlibet*, $\{F, {\sim}F\} \vdash G$. Also strong negation, in general, does not satisfy Inherent Consistency. Yet, it seems desirable that the following *coherence*[8] property holds:

$$\text{(Coherence)} \qquad \text{KB} \vdash {\sim}F \;\Rightarrow\; \text{KB} \vdash -F \;\Rightarrow\; \text{KB} \not\vdash F$$

In general, more information does not mean more consequences. In other words: answers are not necessarily preserved under growth of information. Queries, for which this is the case, are called *persistent.*

Definition 3 (Persistence) *A closed query formula F is called* persistent *if* $\text{KB}_1 \vdash F \Rightarrow \text{KB}_2 \vdash F$, *whenever* $\text{KB}_1 \leq \text{KB}_2$. *If all* $F \in L_{\text{Query}}$ *are persistent, the KRS and its inference relation* $\vdash$ *are called persistent. An operator of the query language is called persistent, if every query formed with it and with persistent subformulas is again persistent.*

Definition 4 (Ampliative Input) *An input formula F is called (i)* ampliative[9] *if* $\text{KB} \leq \mathbf{Upd}(\text{KB}, F)$, *or (ii)* reductive *if* $\text{KB} \geq \mathbf{Upd}(\text{KB}, F)$. *A KRS and its update operation* **Upd** *are called* ampliative, *if all inputs* $F \in L_{\text{Input}}$ *are ampliative.*

If a KRS is monotonic, its update operation is ampliative. Ampliative updating is the formal counterpart of persistent answering.

[8]The name is adopted from [Pereira & Alferes 1992].

[9]The name is adopted from [Belnap 1977].

2.3 Vivid Knowledge Representation Systems

Depending on the specific requirements of an application, and on the computational resources available, different inference and update operations constituting different KRSs may be appropriate. However, there are some minimal requirements any vivid KRS has to satisfy, namely Restricted Reflexivity, Constructivity and Non-Explosiveness.[10] A KRS that satisfies these conditions is called a *basic VKRS*.

Additionally, besides a principal negation (called *strong*), expressing explicit falsity, there should be a second negation (called *weak*) which handles default-implicit negative information in the style of negation-as-failure. Thus, the query language, and possibly also the input language, should contain $\sim$ and $-$.

Some form of CWA, restricted to specific predicates (namely those which are totally represented in the knowledge base) then relates explicit with default-implicit falsity, i.e. strong with weak negation: an atomic sentence formed with a totally represented predicate is (explicitly) false if it is false by default, i.e. its strong negation holds if its weak negation does. The inference relation of a basic VKRS, thus, is extended in the following way:

$$\begin{array}{lrl}(\mathrm{CWA}+) & \mathrm{KB} \vdash_{\mathsf{cwa}} {\sim} p(t) & \text{if} \quad \mathrm{KB} \vdash {\sim} p(t), \text{ or } p \in \mathrm{CWA}^{+} \ \& \ \mathrm{KB} \vdash -p(t) \\ (\mathrm{CWA}-) & \mathrm{KB} \vdash_{\mathsf{cwa}} p(t) & \text{if} \quad \mathrm{KB} \vdash p(t), \text{ or } p \in \mathrm{CWA}^{-} \ \& \ \mathrm{KB} \vdash -{\sim} p(t)\end{array}$$

In a VKRS, three kinds of predicates can be distinguished. The first distinction, introduced by Körner [1966], reflects the fact that many predicates (especially in empirical domains) have truth value gaps: neither $p(c)$ nor ${\sim}p(c)$ has to be the case for specific instances of such *inexact* predicates, like color attributes which can in some cases not be determined because of vagueness.

Other predicates, e.g. from legal or theoretical domains, are *exact*, and we then have, for instance, $m(S) \vee {\sim} m(S)$ and $r(S) \vee {\sim} r(S)$, stating that Susan is either married or unmarried and that she is either a resident of some city or she is not, depending on the information represented in the resp. database of that city. Only exact predicates can be totally represented. Therefore, the predicates subject to the CWA in a KB form a subset of EXACT, the set of all predicates declared as exact in KB, CWA $\subseteq$ EXACT. For an exact predicate p and any term t from its domain, the resp. *tertium non datur* has to be derivable:

$$\mathrm{KB} \vdash_{\mathsf{cwa}} p(t) \vee {\sim} p(t)$$

no matter whether p is subject to the CWA or not. For a non-CWA predicate p, this can only be achieved by adding all instances of the resp. *tertium non datur*, $\{p(t) \vee {\sim} p(t) : t \in U_{\mathrm{KB}}\}$ to the KB which is not possible in definite KRSs where the input language does not contain $\vee$.

Definition 5 (VKB) *A vivid KB, in its general form, is a quadruple*

$$\langle \mathit{EXACT}, \mathit{CWA}, \mathrm{KB}, \mathit{IC} \rangle$$

where EXACT is the set of exact predicates, CWA $= \langle \mathit{CWA}^{+}, \mathit{CWA}^{-} \rangle$ *specifies the predicates subject to the Closed-World Assumption, and IC* $\subseteq L_{\mathsf{Query}}$ *is a set of integrity constraints. A query F is derivable from a VKB if*

$$\mathbf{Upd}(\mathrm{KB}, \bigwedge \{p(t) \vee {\sim} p(t) : p \in \mathit{EXACT} - \mathit{CWA}, t \in U_{\mathrm{KB}}\}) \vdash_{\mathsf{cwa}} F$$

[10] See [Wagner 1993].

Notice that in KBs of definite KRSs, like fact bases or extended logic programs, it is not possible to declare exact predicates not subject to the CWA. Therefore, in definite KRSs, EXACT = CWA.

We shall in most cases neglect integrity constraints in this paper. In cases where CWA^- is empty, we identify CWA with CWA^+. If neither the CWA nor integrity constraints matter, we only mention KB.

Definition 6 (VKRS) *A basic VKRS extended by adding weak negation and the above CWA rules is called a (full)* VKRS.

If a VKRS enjoys the properties of Lemma Redundancy and Compatibility, it will be called *cumulative.* In certain respects, a cumulative VKRS is preferable to a non-cumulative one since it is more regular. However, there seems to be a trade-off between regularity and the capability to capture all aspects of commonsense reasoning. In particular, the price for cumulativity seems to be a more cautious inference operation not allowing for certain inferences suggested by respective commonsense reasoning schemes.

In a VKRS we have the following DeMorgan-like rewrite rules in order to simplify complex formulas:

$$
\begin{array}{rcl@{\qquad}rcl}
-(F \wedge G) & \longrightarrow & -F \vee -G & \sim\sim F & \longrightarrow & F \\
\sim(F \wedge G) & \longrightarrow & \sim F \vee \sim G & -\sim\sim F & \longrightarrow & -F \\
-\sim(F \wedge G) & \longrightarrow & -\sim F \wedge -\sim G & \sim -F & \longrightarrow & F \\
-(F \vee G) & \longrightarrow & -F \wedge -G & --F & \longrightarrow & F \\
\sim(F \vee G) & \longrightarrow & \sim F \wedge \sim G & -\sim -F & \longrightarrow & -F \\
-\sim(F \vee G) & \longrightarrow & -\sim F \vee -\sim G & & &
\end{array}
$$

For inductive definitions, then, it is sufficient to treat the cases of the verum (1), of extended literals (e), of conjunctions ($\wedge$), and of disjunctions ($\vee$). All other cases are covered by the above rewrite rules.[11]

In the sequel we frequently use the fact that formulas $F \in L(1, -, \sim, \wedge, \vee)$ can be normalized according to the following definition:

$$
\begin{array}{lcl}
\mathrm{DNS}(1) & = & \{\emptyset\} \\
\mathrm{DNS}(e) & = & \{\{e\}\} \\
\mathrm{DNS}(F \wedge G) & = & \{K \cup L \,:\, K \in \mathrm{DNS}(F),\, L \in \mathrm{DNS}(G)\} \\
\mathrm{DNS}(F \vee G) & = & \mathrm{DNS}(F) \cup \mathrm{DNS}(G)
\end{array}
$$

This formulation is inspired by a similar one (without weak negation) in [Miller 1989]. The disjunctive normal form of a formula $G \in L(1, -, \sim, \wedge, \vee)$ is obtained as

$$\mathrm{DNF}(G) = \bigvee_{K \in \mathrm{DNS}(G)} \bigwedge K$$

A formula F is called *definite* if its disjunctive normal set is a singleton: $\mathrm{DNS}(F) = \{E\}$.

We will use the following notation for a KRS: let K denote a KRS where only the query but not the input language contains weak negation, then K^+ denotes its weak-negation-free fragment not allowing for weak negation in queries, and K^- denotes its extension by adding weak negation to the input language.

[11] For their justification see [Wagner 1993].

3 Fact Bases

A KB consisting of ground literals (viewed as positive and negative facts) is actually a slight generalization of a *vivid knowledge base* in the restricted sense of Levesque [1988], where only positive facts, i.e. ground atoms, are allowed. For example, $X_1 = \{\sim b(S), m(P, L)\}$ represents the information that Susan is not blonde, and that Peter is married to Linda.

Definition 7 (Informational Ordering) *Let X and X' be fact bases, i.e. $X, X' \subseteq$ Lit. Then X' is an informational extension of X, symbolically $X' \geq X$, if $X' \supseteq X$.*

As a kind of natural deduction from positive and negative facts an inference relation $\vdash$ between a set of proper ground literals X and a ground formula $F \in L(1, -, \sim, \wedge, \vee)$ is defined:

$(\vdash l)$	$X \vdash l$	if	$l \in X$
$(\vdash -l)$	$X \vdash -l$	if	$l \notin X$
$(\vdash \wedge)$	$X \vdash F \wedge G$	if	$X \vdash F$ & $X \vdash G$
$(\vdash \vee)$	$X \vdash F \vee G$	if	$X \vdash F$ or $X \vdash G$

For example, $X_1 \vdash \sim b(S) \wedge -\sim b(L)$. Notice that according to this definition both a literal l and its complement $\bar{l}$ are acceptable at the same time, and independently from each other. This kind of inference is called *liberal*, as opposed to certain neutralization-based inference procedures (see e.g. [Wagner 1993a]) where contradictory information is discarded.

Observation 1 *A liberal inference relation is not coherent, i.e. $X \vdash \sim p$ does not imply that $X \vdash -p$. In other words, it is possible to infer contradictory queries, such as $p \wedge \sim p$.*

Observation 2 *In addition to a standard disjunction $\vee$, an* exclusive *disjunction, $|$, can be defined:*

$$F \mid G \quad := \quad (F \wedge -G) \vee (G \wedge -F)$$

For instance, $\{p, q\} \vdash p \vee q$, but $\{p, q\} \not\vdash p \mid q$. Notice that $|$ is not persistent.

Definition 8 *Updating by extended literals is defined as insertion, resp. deletion:* $\mathbf{Upd}(X, l) := X \cup \{l\}$, *resp.* $\mathbf{Upd}(X, -l) := X - \{l\}$. *More generally,* **Upd** *can be defined for sets (corresponding to conjunctions) of extended literals, $E \subseteq$ XLit, as* $\mathbf{Upd}(X, E) := X \cup E^+ - E^-$.[12] *Notice that new information which is inconsistent with the already represented one does not lead to an explosion of X, or in other words, destroy the information content of X.*

For example, if we learn that Peter gets divorced from Linda and marries Susan, we perform the following update: $\mathbf{Upd}(X_1, -m(P, L) \wedge m(P, S)) = \{\sim b(S), m(P, S)\}$. Such a sequence of basic inputs (insertions and deletions of atoms or literals) is also called a *knowledge base transaction.*

Definition 9 *The KRS* $\langle 2^{\mathrm{Lit}}, \vdash, L(1, -, \sim, \wedge, \vee), \mathbf{Upd}, \mathrm{XLit} \rangle$ *is denoted by* V_o^-.

Claim 1 V_o^- *is a cumulative VKRS.*

Proof: See [Wagner 1993].

[12] Thus, an input formula can be any definite formula F.

4 The Disjunctive Extension of a KRS

Let K be a definite KRS, that is, KBs in K are definite, and hence, any input formula is also definite, implying that the input laguage does not allow for disjunctions. The *disjunctive extension* of K, symbolically DK, is defined as the KRS $\langle L^d_{\text{KB}}, \vdash_d, L^d_{\text{Query}}, \mathbf{Upd}_d, L^d_{\text{Input}}\rangle$ where $L^d_{\text{KB}} := 2^{L_{\text{KB}}}$ and for $Y \in L^d_{\text{KB}}$,

$$Y \vdash_d F \stackrel{def}{\Longleftrightarrow} \text{for all } X \in Y : X \vdash F$$

In the sequel, we will not write d as superscript, resp. subscript if it is clear from context. The elements $X \in Y \subseteq L_{\text{KB}}$ of a disjunctive KB are also called *(possible) situation descriptions.*

Notice that in DK, $Y = \emptyset$ is not a meaningful KB: it trivially confirms every sentence. The 'empty' KB of DK contains as its only element the empty set: $0 = \{\emptyset\}$.

The informational ordering of KBs is extended from K to DK:

Definition 10 (Informational Ordering) *Let $\geq$ be an informational ordering of KBs in K, then an ordering relation between the disjunctive KBs of DK can be defined according to*

$$Y' \geq_d Y \stackrel{def}{\Longleftrightarrow} \forall X' \in Y' \exists X \in Y : X' \geq X$$

Finally, a disjunction operator is added to the input language of K, $L^d_{\text{Input}} := L_{\text{Input}}(\vee)$, and the update operation is inductively extended in the following way:

$$\begin{array}{lll}
(Ue) & \mathbf{Upd}_d(Y, e) & = \{\mathbf{Upd}(X, e) : X \in Y\} \\
(U\wedge) & \mathbf{Upd}_d(Y, F \wedge G) & = \mathbf{Upd}_d(\mathbf{Upd}_d(Y, F), G) \\
(U\vee) & \mathbf{Upd}_d(Y, F \vee G) & = \mathbf{Upd}_d(Y, F) \cup \mathbf{Upd}_d(Y, G) \cup \mathbf{Upd}_d(Y, F \wedge G) \\
(U\mid) & \mathbf{Upd}_d(Y, F \mid G) & = \mathbf{Upd}_d(Y, F) \cup \mathbf{Upd}_d(Y, G)
\end{array}$$

All other inductive cases are handled by the above rewrite rules for complex formulas.[13] Notice that the definition of disjunctive update, $(U\vee)$, is *inclusive* (as is standard disjunction).

Again, new inputs contradicting the KB do not lead to an explosion or total loss of information as in classical logic. Classical behavior can be achieved by modifying (Ue), now discarding those situation descriptions which become inconsistent after being updated:

$$\mathbf{Upd}_{cl}(Y, e) = \{\mathbf{Upd}(X, e) : X \in Y \ \& \ \mathbf{Upd}(X, e) \text{ is consistent}\}$$

If $Y \vdash l$ then $\mathbf{Upd}_{cl}(Y, \tilde{l}) = \{\}$ is an 'exploded' KB in the sense that everything follows from $\{\}$. A vivid alternative to this hypersensitive inconsistency handling mechanism of classical logic is the principle of *minimal inconsistency* proposed in [Priest 1989]:

$$\mathbf{Upd}_{mi}(Y, e) = \{\mathbf{Upd}(X, e) : X \in Y \ \& \ \mathbf{Upd}(X, e) \text{ is minimally inconsistent}\}$$

which is nonexplosive and leads to a restricted form of disjunctive syllogism. Inconsistency handling is further discussed in [Wagner 1993a+b].

Since in 'real world' KBs the amount of disjunctive information can be expected to be small, and for reasons of computational efficiency has to be small, compared to the amount of definite information represented in a typical KB, one could choose to represent a disjunctive KB as a pair $\langle X, Y\rangle$ where $X \in L_{\text{KB}}$ contains the definite and $Y \subseteq L_{\text{KB}}$ the disjunctive information, and

$$\langle X, Y\rangle \vdash F \stackrel{def}{\Longleftrightarrow} X \vdash F, \text{ or } X \cup X' \vdash F \text{ for all } X' \in Y$$

Thus, if $Y \subseteq L_{\text{KB}}$ is a disjunctive KB, then its 'folding' is

$$\langle \bigcap Y, \{X - \bigcap Y : X \in Y\}\rangle$$

[13] $(\wedge U)$ requires that $F \wedge G$ are weakly consistent. Alternatively, the update operation can be defined by means of the disjunctive normal set which also works for weakly inconsistent inputs.

4.1 Disjunctive Fact Bases

Following Belnap [1977] we call a set Y of partial interpretations (represented as sets of literals) an *epistemic state*. Since an epistemic state is able to represent disjunctive information, it can also be viewed as a *disjunctive fact base* where – in addition to Belnap's considerations – we also allow for weak negation in the query and in the input language.

Definition 11 $V_B := DV_0$. *The weak-negation-free fragment of the disjunctive extension of fact bases,* V_B^+, *is called* Belnap's KRS.

Example 1 $\{\{p\},\{\sim q\}\} \vdash (p \vee \sim q) \wedge -(p \wedge \sim q)$. *However, neither p nor $\sim q$, and neither $-p$ nor $-\sim q$ are derivable.*

Example 2 *Let* $Y_1 = \{X_1\}$ *where* $X_1 = \{\sim b(S), m(P,L)\}$. *If we learn that not both Peter and Linda are blonde, we perform the following update:*

$$Y_2 := \mathbf{Upd}(Y_1, \sim(b(P) \wedge b(L))) = \{\, X_1 \cup \{\sim b(P)\},\ X_1 \cup \{\sim b(L)\}\ X_1 \cup \{\sim b(P), \sim b(L)\}\,\}$$

We obtain, for instance, $Y_2 \not\vdash -\sim b(L)$, *and also* $Y_2 \geq Y_1$.

4.1.1 Belnap's KRS

Observation 3 *In* V_B^+, *KBs are not unique representations:* $C(Y) = C(Y')$ *does not imply that* $Y = Y'$. *For instance,* $C(\{\{p\}\}) = C(\{\{p\}, \{p,q\}\})$.

Claim 2 V_B^+ *is a monotonic basic VKRS.*

Proof: See [Wagner 1993].

Observation 4 *In Belnap's KRS, one cannot express exclusive disjunctions. In the epistemic states of* V_B^+ *only the minimal elements count,*

$$C(Y) = \bigcap\{C(X) : X \in Y\} = \bigcap\{C(X) : X \in Min(Y)\}$$

Therefore, exclusive and inclusive disjunctive information cannot be distinguished:

$$\{\{p\},\{q\}\} \cong \{\{p\},\{q\},\{p,q\}\}$$

4.1.2 Belnap's KRS with Weak Negation

Claim 3 *In* V_B, *KBs are unique representations.*

Proof: See [Wagner 1993].

Claim 4 V_B *is a non-cumulative VKRS. It does not satisfy Lemma Compatibility.*

Proof: See [Wagner 1993]. We only show the failure of Lemma Compatibility by means of a simple example:

$$\begin{array}{ll} & \{\{p\}\} \vdash p \vee q \\ \text{but} & \mathbf{Upd}(\{\{p\}\}, p \vee q) = \{\{p\},\{p,q\}\} \not\vdash -q \\ \text{while} & \{\{p\}\} \vdash -q \qquad \square \end{array}$$

Not only are lemmas no longer compatible by the addition of weak negation to Belnap's KRS, but also is the inference operation no longer persistent and the update operation no longer ampliative.

Observation 5

1. *The inference operation of V_B is not persistent, in general. However, query formulas without weak negation and exclusive disjunction are persistent.*

2. *In V_B^-,* **Upd** *is no longer ampliative, that is, $Y \not\leq$* **Upd**(Y, F)*, in general. However, input formulas $F \in L(1, \sim, \wedge, \vee)$ are ampliative.*

4.1.3 Representing Three Kinds of Predicates

Among the exact predicates one can distinguish between those which are totally represented in a KB and those which are not. Exact and totally represented predicates are subject to the CWA. For example, the local KB of some city knows all residents of the city, i.e. the CWA holds for r, but it does not know of every resident whether she is married or not because she might have married in another city and this information would only be retrievable if all city KBs were interconnected as a distributed KB. Consequently, the CWA does not apply to m.

The CWA helps to reduce disjunctive complexity which is exponential in the number of exact non-CWA predicates: if n is the number of ground atoms which can be formed by means of predicates declared as exact but not subject to the CWA by KB, then KB contains at least 3^n possible situation descriptions (2^n if incoherent situation descriptions are discarded).

We illustrate these distinctions with the following example:

$$\mathrm{KB} \quad = \quad \begin{cases} \mathrm{EXACT(married, resident)} \\ \mathrm{CWA(resident)} \\ \mathrm{married(\,Mary),\ resident(\,Mary),\ blonde(\,Mary)} \\ \sim\mathrm{married(\,Susan),\ \sim blonde(\,Susan)} \\ \mathrm{LookingAt(\,Mary,\ Peter)} \\ \mathrm{LookingAt(\,Peter,\ Susan)} \end{cases}$$

The interesting queries we can ask this KB and the resp. answers are:

1. Does a married person look at an unmarried one ? Yes, but the KB does not know who.

2. Does a resident look at a non-resident ? Yes, Mary at Peter.

3. Does a blonde person look at a non-blonde one ? No. The KB is completely ignorant about the blondeness of Peter: neither is he blonde, nor is he non-blonde, nor is he blonde or non-blonde. (He might be in-between.)

Notice that the explicit form of KB obtained by updating by all tertium non datur disjunctions specified implicitly through the EXACT declaration of *resident* and *married* would yield $3^6 = 729$ update steps, resp. possible situation descriptions, which is reduced to $3^3 = 27$ by the CWA declaration of *resident*.

If all predicates are assumed to be exact, the resp. 'completion' of a disjunctive fact base by adding all instances of the tertium non datur leads half way to classical logic: if all $X \in Y$ are consistent, then

$$\mathbf{Upd}(Y, \bigwedge\{p(t) \vee \sim p(t) : t \in U_Y\}) \vdash F \quad \text{iff} \quad Y \vdash_{cl} F$$

where $F \in L(\sim, \wedge, \vee)$ and $\vdash_{cl}$ denotes classical derivability. The second half consists of the *ex contradictione sequitur quodlibet* principle according to which all inconsistent $X \in Y$ have to be discarded (see the definition of $\mathbf{Upd}_{cl}$ above).

4.2 Adding an S5-like Modal Operator for Definite Belief

The query language of DK can be extended by adding the modal operator B for definite belief, yielding the *modal query extension* $D^m K$. For $X \in Y \in L^d_{\mathrm{KB}}$, we define[14]

(B)	$X \vdash \mathrm{B}F$	if	for all $X' \in Y : X' \vdash F$
($\sim$B)	$X \vdash \sim\mathrm{B}F$	if	for some $X' \in Y : X' \vdash \sim F$
($-$B)	$X \vdash -\mathrm{B}F$	if	for some $X' \in Y : X' \vdash -F$
($-\sim$B)	$X \vdash -\sim\mathrm{B}F$	if	for all $X' \in Y : X' \vdash -\sim F$

In this setting, two different possibility operators are definable:

$$\mathrm{E}F := \sim\mathrm{B}\sim F$$
$$\mathrm{C}F := -\mathrm{B}\sim F$$

EF captures the situation where there is some evidence in favour of F, whereas CF captures the situation where $\sim F$ cannot be inferred, i.e. where F is – in this sense – consistent.

Observation 6 *The modal operators* B *and* E *are persistent, whereas* C *is not.*

Observation 7 (Definite vs. Indefinite Existence) *By means of* B*, dishonest disjunctive information can be distinguished from honest one: if $F \vee G$ is dishonest on the basis of Y, then $Y \vdash \mathrm{B}F \vee \mathrm{B}G$. Likewise, indefinite existential queries can be distinguished from definite ones, as for instance in $D^m V_0$ where, for $X = \{\{p(c), q(d)\}, \{p(d), q(d)\}\}$, we obtain $X \vdash \exists x p(x)$ as well as $X \vdash \exists x q(x)$, but only $X \vdash \exists x\, \mathrm{B}q(x)$, whereas $X \not\vdash \exists x\, \mathrm{B}p(x)$. Gelfond [1992] proposes to use such existential queries as integrity constraints requiring that a KB knows definitely, or indefinitely, certain entities satisfying certain predicates.*

Observation 8 (Representing and Querying the Unknown) *While the problem whether some fact p is unknown in a fact base can be simply decided by answering the query $-p \wedge -\sim p$ this does not work in a disjunctive fact base. Instead, the indeterminacy query can be reformulated in three different versions which are also discussed in [Thijsse 1992] where they are named 'simple', 'strong', and 'semi-strong ignorance', respectively. In the following three examples we express complete knowledge, $\mathrm{B}p$, about p, and three different forms of incomplete knowledge, $-\mathrm{B}q \wedge -\mathrm{B}\sim q$, about q:*

1. *Complete ignorance wrt q represented by*

$$\{\{p\}\} \vdash p \wedge -q \wedge -\sim q$$

2. *Strong ignorance wrt q:*

$$\{\{p\}, \{p,q\}, \{p,\sim q\}\} \vdash \mathrm{B}p \wedge \mathrm{E}q \wedge \mathrm{E}\sim q$$

3. *Semi-strong ignorance wrt q:*

$$\{\{p\}, \{p,\sim q\}\} \vdash \mathrm{B}p \wedge \mathrm{E}\sim q$$

or symmetrically,

$$\{\{p\}, \{p,q\}\} \vdash \mathrm{B}p \wedge \mathrm{E}q$$

[14] *Strictly speaking, we should rather write $Y, X \vdash F$, specifying the set of 'possible worlds' Y along with the 'actual world' X (as usual in possible worlds semantics), instead of $X \vdash F$.* We prefer, however, the more economic notation, keeping the context Y in the background whenever it is clear.

Observation 9 (Honest and Dishonest Disjunctive Information) *Knowing $p \vee q$ subsumes four (!) cases, two of them 'honest' and two 'dishonest'. In the case of a dishonest disjunction, more information is, in fact, known than expressed by the resp. disjunction:*

- *Honest disjunctive information:*
  - *Only knowing $p \vee q$, and nothing more about p or q.*

    $$\{\{p\}, \{q\}, \{p,q\}\} \vdash (p \vee q) \wedge -\mathrm{B}p \wedge -\mathrm{B}q$$

  - *Knowing that either p or q but not both, and not knowing which of them:*

    $$\{\{p\}, \{q\}\} \vdash (p \mid q) \wedge -\mathrm{B}p \wedge -\mathrm{B}q$$

- *Dishonest disjunctive information:*
  - *Knowing exactly one of both items: either p, or q. This corresponds to the KB $\{\{p\}\}$, resp. $\{\{q\}\}$, and to the query $p \wedge -q$, resp. $q \wedge -p$.*
  - *Knowing both p and q which corresponds to the KB $\{\{p,q\}\}$, and to the query $p \wedge q$.*

Observation 10 (Modal Inputs) *Epistemic states cannot represent arbitrary subjective (i.e. modal) knowledge. For instance, updating by dishonest disjunctive information, $\mathbf{Upd}(0, \mathrm{B}p \vee \mathrm{B}q)$, is not possible. For this purpose, one had to extend the framework and take sets of epistemic states as KBs (yielding the* **S5** *extension S5K). Updating by subjective literals, however, is straightforward:*

$$\begin{aligned}
\mathbf{Upd}(Y, \mathrm{B}e) &:= \mathbf{Upd}(Y, e) \\
\mathbf{Upd}(Y, \mathrm{E}e) &:= Y \cup \mathbf{Upd}(Y, e) \\
\mathbf{Upd}(Y, \mathrm{C}e) &:= Y \cup \mathbf{Upd}(Y, -\!\sim\! e)
\end{aligned}$$

Example 3 *Let rhb, resp. lhb, denote the facts 'right hand broken', resp. 'left hand broken'. In the beginning we only have the information that the right hand is broken. We then learn that there is a serious possibility that the left hand is broken as well:*

$$\mathbf{Upd}(\{\{rhb\}\}, \mathrm{E}\, lhb) = \{\{rhb\}, \{rhb, lhb\}\}$$

Finally however, as the result of an examination, we are happy to be told that there is no conclusive evidence that the right hand is broken, i.e. it is consistent that it is not broken:

$$\mathbf{Upd}(\{\{rhb\}, \{rhb, lhb\}\}), \mathrm{C} \sim rhb) = \{\{rhb\}, \{rhb, lhb\}, \{\}, \{lhb\}\}$$

5 Logic Programs as Rule Knowledge Bases

With each KRS K a *rule-based* extension, RK, can be associated. In RK a knowledge base $X \in L_{\mathrm{KB}}$ is supplemented by a set $R \subseteq L_{\mathrm{Input}} \times L_{\mathrm{Query}}$ containing rules $r = \langle Conclusion, Premise \rangle$ with $Conclusion \in L_{\mathrm{Input}}$ and $Premise \in L_{\mathrm{Query}}$, also written as '$Conclusion \leftarrow Premise$'. These rules are mappings between KBs,

$$r : L_{\mathrm{KB}} \rightarrow L_{\mathrm{KB}}$$

since – in the standard case – their application is defined as

$$r(X) = \begin{cases} \mathbf{Upd}(X, Conclusion) & \text{if } X \vdash Premise \\ X & \text{otherwise} \end{cases}$$

Definition 12 *A mapping $f : A \to A$ from a preorder $\langle A, \leq \rangle$ into itself is called* monotonic *if $f(x) \leq f(y)$ whenever $x \leq y$. It is called* ampliative *if $x \leq f(x)$. A rule is called* monotonic *(resp.* ampliative*) if it is a monotonic (resp. ampliative) mapping.*

Observation 11 *The rule $F \leftarrow G$ is monotonic if its conclusion F is ampliative, and its premise G is persistent.*

The semantics of an ampliative rule knowledge base $\langle X, R \rangle$ is determined by the definition of a preferred closure of X under R, being a knowledge base $Z \in L_{\mathrm{KB}}$ extending X and closed under all rules of R:

1. $Z \geq X$
2. $r(Z) = Z$ for all $r \in R$

In general, however, there may be several preferred closures, or none. We denote their collection by $R(X)$. If there are several preferred closures, a valid consequence must be inferrable from all of them:

$$C(\langle X, R \rangle) \quad := \quad \bigcap \{C(Z) : Z \in R(X)\}$$

In the simplest case, where all rules are monotnonic, the preferred closures are the minimal ones (like, e.g., in logic programs without negation-as-failure).

Observation 12 (Local vs. Global Rules) *In the case of a disjunctive KB $Y \subseteq L_{\mathrm{KB}}$, a* local closure *of $\langle Y, R \rangle$, corresponding to the local application of rules, can be defined as the set of all informationally minimal $X \in L_{\mathrm{KB}}$ satisfying the rules of R as constraints:*

$$R_{loc}(Y) \quad := \quad \mathrm{Min}\{X' : X' \geq X \text{ for some } X \in Y, \text{ such that}$$
$$\text{for all rules } F \leftarrow G \in R,\ X' \vdash F \text{ whenever } X' \vdash G\}$$

Syntactically, a disjunctive logic program can be interpreted either as a KB of RDV_0, or as a KB of DRV_0. In the latter case, rules are local, while in the former case they are global. For instance, the program $\Pi = \{p \vee q, r \leftarrow p, r \leftarrow q\}$ has two readings:

1. *As a KB of DRV_0, i.e. as*

   $$\{\ \{p, r \leftarrow p, r \leftarrow q\},\ \{q, r \leftarrow p, r \leftarrow q\},\ \{p, q, r \leftarrow p, r \leftarrow q\},\ \}$$

   from which r is inferrable.

2. *As a KB of RDV_0, i.e. as*

   $$\langle \{\{p\}, \{q\}, \{p, q\}\}, \{r \leftarrow p, r \leftarrow q\} \rangle$$

   from which r cannot be inferred, since $R(Y) = Y$. However, using the local closure one gets

   $$R_{loc}(Y) = \{\{p, r\}, \{q, r\}, \{p, q, r\}\}$$

   and, as in DRV_0, r can be inferred.

It is conceivable that a disjunctive KB uses both local and global rules, like KBs of $RDRV_0$.

Observation 13 (Disjunction in the Premise) *The following property,*

$$(Or) \quad \mathbf{Upd}(\mathrm{KB}, F) \vdash H \;\&\; \mathbf{Upd}(\mathrm{KB}, G) \vdash H \;\Rightarrow\; \mathbf{Upd}(\mathrm{KB}, F \vee G) \vdash H$$

does not hold in RDK where K is any KRS.

In most proposals on the semantics of disjunctive logic programs,[15] rules are interpreted locally in a model-theoretic fashion, like implications. While this could be called a *constraint semantics* of rules, since they are read as a kind of constraints on models, the interpretation of rules as mappings proposed here amounts to an *operational semantics*.

Determining the 'right' preferred closures $R(X)$ is especially difficult if the dependency graph of R contains loops involving weak negation, or in other words, if R is not weakly wellfounded.[16] In this case the concept of a *stable closure* generalising the stable model semantics of [Gelfond & Lifschitz 1988] offers help.

By means of stable closures, a general form of negation-as-failure, called *stable negation*, can be added to the premises of rules of any ampliative RKRS provided that it has unique closures $R(X)$ for any rule knowledge base $\langle X, R\rangle$. Instead of rules $F \leftarrow G$ we allow for rules $F \leftarrow G_1, \ldots, G_i, \mathtt{not}\, G_{i+1}, \ldots, \mathtt{not}\, G_m$, where not is a newly introduced operator (not occuring in the query and input language, that is, F, G and all G_k do not contain not).

Definition 13 (Stable Closure) *For any RKB $\langle X, R\rangle$ and any $Z \in L_{\mathrm{KB}}$ define R^Z as the set of* not *-free rules obtained from R by*

1. *removing all rules containing a premise* $\mathtt{not}\, G$ *such that* $Z \vdash G$, *and*

2. *deleting in the remaining rules all premises of the form* $\mathtt{not}\, G$.

$Z \in L_{\mathrm{KB}}$ *is called a* stable closure *of* $\langle X, R\rangle$ *if* $Z = R^Z(X)$.

5.1 Extended Logic Programs

Extended logic programs correspond to rule knowledge bases of RV_0. For instance, the following program

$$\Pi_1 = \{\, m(P, L),\ \sim b(S),\ \sim m(x, y) \leftarrow -m(x, y) \,\}$$

corresponds to the rule knowledge base $\langle X_1, R_1\rangle$ in RV_0 such that X_1 is as above, and $R_1 = \{\sim m(x, y) \leftarrow -m(x, y)\}$. We obtain as the unique preferred closure

$$R_1(X_1) = \{\sim b(S), m(P, L), \sim m(P, S)\},$$

and consequently, $\langle X_1, R_1\rangle \vdash \sim m(P, S)$. In the case of a weakly wellfounded (i.e. locally stratified) program $\Pi = \langle X, R\rangle$, there is a unique preferred closure $R(X)$, obtained by applying the rules of R in the order given by the natural stratification of R. This closure is also a supported, perfect, wellfounded and stable model. In the general case, the stable closures of Π can be regarded as its preferred closures.

[15] See, e.g., [Gelfond & Lifschitz 1991], or [Lobo, Minker & Rajasekar 1992].

[16] The notion of *weak wellfoundedness*, corresponding to local stratifiability, is introduced in [Wagner 1993]. It excludes 'negative loops' while simple *wellfoundedness*, also defined in [Wagner 1993], excludes in addition positive loops, and *strong wellfoundedness*, defined in [Wagner 1993a], excludes in addition 'complementary loops'.

5.2 Extended Disjunctive Logic Programs

Extended disjunctive logic programs correspond to rule knowledge bases of RDV_o, resp. RV_B. For instance, the following program about broken left hands (lh) and right hands (rh),

$$\Pi_2 = \begin{cases} \text{lh\_broken} \lor \text{rh\_broken} \\ \text{lh\_usable} \leftarrow -\text{lh\_broken} \\ \text{rh\_usable} \leftarrow -\text{rh\_broken} \\ \text{get\_50\$\_compensation} \leftarrow \text{rh\_broken} \\ \text{get\_50\$\_compensation} \leftarrow \text{lh\_broken} \\ \text{go\_to\_work} \leftarrow -(\text{rh\_broken} \land \text{lh\_broken}) \end{cases}$$

corresponds to the rule knowledge base $\langle Y_2, R_2 \rangle$ where

$$Y_2 = \{\{\text{lh\_broken}\}, \{\text{rh\_broken}\}, \{\text{lh\_broken}, \text{rh\_broken}\}\}$$

In the case where the rules of Π_2 are global, none of them is applicable, so the closure of Y_2 is Y_2 itself and we can neither conclude that any hand is usable, nor that compensation can be obtained (we had to tell the insurance company which hand is broken), nor that we are expected to go to work. We would not get compensation because (Or) fails if the resp. rules are read globally.

The local closure of $\langle Y_2, R_2 \rangle$, in abbreviated form, is:

$$\{\{lhb, rhu, 50, gtw\}, \{rhb, lhu, 50, gtw\}, \{lhb, rhb, 50\}\}$$

and thus, 50 \$ are obtained as compensation.

5.3 Epistemic Specifications

Gelfond [1992] has proposed to extend the framework of logic programming by the addition of epistemic modal operators in the premise of rules. An *epistemic specification*, in this sense, corresponds to a rule knowledge base of RD^mV_o. It specifies an epistemic state, called *world view* in [Gelfond 1992], as a collection of *belief sets*: $Y \subseteq 2^{\text{Lit}}$ is an epistemic state specified by Π if each $X \in Y$ is a belief set of Π with respect to Y, i.e.

$$Y = \bigcup_{X \in Y} \alpha(\Pi^{Y,X}, Y)$$

where $\Pi^{Y,X}$ is the following **not**-eliminating transformation of Π:

1. remove from Π all rules containing premise formulas $\mathbf{not}\, G$ such that $X \vdash G$,
2. delete in the remaining rules all premise formulas of the form $\mathbf{not}\, G$,

and $\alpha(\Pi, Y)$ collects all minimal literal closures of Π with respect to Y in the following sense: let Π be a specification not containing negation-as-failure, then $X \subseteq$ Lit is a minimal literal closure of Π with respect to Y if X is a minimal set satisfying $X \vdash F$ whenever $Y, X \vdash G$ for all rules $F \leftarrow G \in \Pi$.

This definition simplifies that of [Gelfond 1992] where a two-step reduction of Π is defined: first modal operators are eliminated from rules with respect to the epistemic state Y, yielding Π^Y, and then, with respect to single belief sets $X \in Y$, negation-as-failure is eliminated, yielding the resp. $(\Pi^Y)^X$. In this way one gets the fixpoint equation

$$Y = \bigcup_{X \in Y} \alpha((\Pi^Y)^X)$$

If Π does not contain modal operators, or does only contain them in the scope of negation-as-failure, then for every $X \in Y \subseteq 2^{\text{Lit}}$, $\Pi^{Y,X} = (\Pi^Y)^X$.

Notice that meanwhile Gelfond has changed his original definition, now defining the falsity of modal formulas as non-satisfiability (instead of strong falsifiability) and no longer allowing subjective formulas prefixed by negation-as-failure. This change, however, is not essential for our discussion.

5.4 The Closed-World Assumption in Disjunctive Fact Bases

The above definition of the CWA rule reducing the derivability of a strongly negated atom formed with a CWA predicate to the derivability of its weak negation is applicable in every KRS with strong and weak negation. It is, therefore, more general than the definiton of the CWA given in [Gelfond & Lifschitz 1990] for extended logic programs (corresponding to the rule-based extension of fact bases) which was modified in [Gelfond 1992] in order to account properly for disjunctive information. We shall discuss this now by means of an example.

In [Gelfond & Lifschitz 1990+1991] it was proposed to express the CWA for a predicate p by the rule

$$\sim p(x) \leftarrow \text{not}\, p(x)$$

where not denotes stable negation as defined by the *stable model semantics*. Consider the following example. A member of the scientific council must be both a lecturer and an active researcher. Suppose we know that Peter is a lecturer, $l(P)$, and we also know that Tom is a lecturer or a researcher, $l(T) \vee r(T)$, and for both predicates the CWA holds, then, according to [Gelfond & Lifschitz 1991] we get the following 'disjunctive database':

$$\Pi_1 = \begin{cases} l(P) \\ l(T) \vee r(T) \\ \sim l(x) \leftarrow \text{not}\, l(x) \\ \sim r(x) \leftarrow \text{not}\, r(x) \end{cases}$$

which has two *answer sets*,[17] namely $\{l(P), \sim r(P), l(T), \sim r(T)\}$ and $\{l(P), \sim r(P), r(T), \sim l(T)\}$, and consequently, $\sim r(P)$ as well as $\sim l(T) \vee \sim r(T)$ can be inferred, that is, we get the answer no not only to the query 'Does Peter belong to the scientific council ?' but also to 'Does Tom belong to the scientific council ?'.

However, as Gelfond argues in [1992], the fact that $r(T)$ does not hold in some answer set, does not guarantee that a rational reasoner does not have a reason to belief $r(T)$. Such a reason may be given by the existence of another answer set containing $r(T)$. Gelfond, therefore, proposes to express the CWA for a predicate p in the presence of disjunctive information by means of an epistemic modality defined within his framework of *epistemic specifications* in the following way:

$$\sim p(x) \leftarrow \text{not}\, \mathrm{E}p(x)$$

where the premise requires that there is no possible evidence for $p(x)$ in none of the answer sets. We thus obtain the following *epistemic specification* as the correct formalization of our problem:

$$\Pi_2 = \begin{cases} l(P) \\ l(T) \vee r(T) \\ \sim l(x) \leftarrow \text{not}\, \mathrm{E}l(x) \\ \sim r(x) \leftarrow \text{not}\, \mathrm{E}r(x) \end{cases}$$

[17] Recall that $S \subseteq \text{Lit}$ is an ***answer set*** of a 'disjunctive database' Π if $S \in \alpha(\Pi^S)$ where Π^S is the Gelfond-Lifschitz transformation eliminating not from Π, and $\alpha(\Pi)$ collects all minimal literal closures of Π.

Π_2 has the unique world view

$$\{\{l(P), \sim r(P), l(T)\}, \{l(P), \sim r(P), r(T)\}\}$$

yielding the conclusion that $\sim r(P)$, but not that $\sim l(T) \vee \sim r(T)$, and hence as desired only the query 'Does Peter belong to the scientific council ?' gets the answer **no**, while the same query with respect to Tom gets the answer **unknown** now.

This is exactly the same result as in the corresponding disjunctive fact base obtained by stipulating the CWA for the predicates l and r, and compiling the facts that Peter is a lecturer and that Tom is a lecturer or a researcher:

$$\begin{aligned} Y_1 &= \langle\{l,r\}, \mathbf{Upd}(0, l(P) \wedge (l(T) \vee r(T)))\rangle \\ &= \langle\{l,r\}, \{\{l(P), l(T)\}, \{l(P), r(T)\}, \{l(P), l(T), r(T)\}\}\rangle \end{aligned}$$

By means of (CWA+), we get the following closure:

$$\{\{l(P), l(T), \sim r(P)\}, \{l(P), r(T), \sim r(P)\}, \{l(P), l(T), r(T), \sim r(P)\}$$

yielding the same answers as above to the queries whether Peter and Tom belong to the scientific council.

6 Conclusion

We have shown how the framework of databases and logic programs can be constructively extended with respect to disjunctive information and modal queries, resp. subjective knowledge. In these extensions there are different design options how to model the interaction of negations and the other operators. And it is not obvious at all what is the 'right' generalization of negation-as-failure and the CWA in these settings. We hope, however, that the proposed framework of vivid knowledge representation systems helps to realize and evaluate the different design options and modelings.

References

[Almukdad & Nelson 1984] A. Almukdad and D. Nelson: Constructible Falsity and Inexact Predicates, *JSL* 49:1 (1984), 231–233.

[Belnap 1976] N.D. Belnap: How a Computer Should Think, in *Contemporary Aspects of Philosophy*, Proc. Oxford International Symposium 1975, Oriel Press 1976, 30–56.

[Belnap 1977] N.D. Belnap: A Useful Four-valued Logic, in G. Epstein and J.M. Dunn (Eds.), *Modern Uses of Many-valued Logic*, Reidel 1977, 8–37.

[Clark 1978] K.L. Clark: Negation as Failure, in H. Gallaire and J. Minker (Eds.), *Logic and Databases*, Plenum Press, New York, 1978, 293–322.

[Gabbay 1985] D. Gabbay: Theoretical Foundations for Nonmonotonic Reasoning in Expert Systems, in K.R. Apt (Ed.), *Proc. NATO Advanced Study Institute on Logics and Models of Concurrent Systems*, Springer Verlag, 1985, 439–457.

[Gelfond 1992] M. Gelfond: Logic Programming and Reasoning with Incomplete Information, *Proc. Workshop on Extensions of Logic Programming 1992*

[Gelfond & Lifschitz 1988] M. Gelfond and V. Lifschitz: The Stable Model Semantics for Logic Programming, *Proc. ICLP 1988*, MIT Press, 1988.

[Gelfond & Lifschitz 1990] M. Gelfond and V. Lifschitz: Logic Programs with Classical Negation, *Proc. ICLP 1990*, MIT Press, 1990.

[Gelfond & Lifschitz 1991] M. Gelfond and V. Lifschitz: Classical Negation in Logic Programs and Disjunctive Databases, *J. New Generation Computing* 9 (1991), 365–385.

[Körner 1966] S. Körner: *Experience and Theory*, Kegan Paul, London, 1966.

[Levesque 1984] H.J. Levesque: Foundations of a Functional Approach to Knowledge Representation, *AI* 23:2, 1984, 155-212.

[Levesque 1986] H.J. Levesque: Making Believers out of Computers, *AI* 30 (1986), 81-107.

[Levesque 1988] H.J. Levesque: Logic and the Complexitiy of Reasoning, *J. Philosophical Logic* 17 (1988), 355–389.

[Lobo, Minker & Rajasekar 1992] J. Lobo, J. Minker and A. Rajasekar: *Foundations of Disjunctive Logic Programming*, MIT Press, 1992.

[Miller 1989] D. Miller: A Logical Analysis of Modules in Logic Programming, *J. Logic Programming* 1989, 79–108.

[Nelson 1949] D. Nelson: Constructible falsity, *JSL* 14 (1949), 16–26.

[Pearce & Wagner 1990] D. Pearce and G. Wagner: Reasoning with Negative Information I – Strong Negation in Logic Programs, in L. Haaparanta, M. Kusch and I. Niiniluoto (Eds.), *Language, Knowledge and Intentionality*, Acta Philosophica Fennica 49, 1990.

[Pereira & Alferes 1992] L.M. Pereira and J.J. Alferes: Wellfounded Semantics for Logic Programs with Explicit Negation, *Proc. ECAI'92*, Wiley, 1992.

[Priest 1989] G. Priest: Reasoning about Truth, *AI*, 39 (1989), 231–244.

[Rautenberg 1979] W. Rautenberg: *Klassische und nichtklassische Aussagenlogik*, Vieweg, 1979.

[Reiter 1978] R. Reiter: On Closed-World Databases, in J. Minker and H. Gallaire (Eds.): *Logic and Databases*, Plenum Press, 1978.

[Reiter 1990] R. Reiter: On asking what a database knows, in J. Lloyd (Ed.), *Computational Logic*, Proceedings, Springer, 1990, 96–113.

[Thijsse 1992] E. Thijsse: *Partial Logic and Knowledge Representation*, dissertation, Tilburg university, 1992.

[Wagner 1991] G. Wagner: Logic Programming with Strong Negation and Inexact Predicates, *J. Logic and Computation* 1:6 (1991), 835–859.

[Wagner 1993] G. Wagner: *Vivid Logic – Knowledged-Based Reasoning with Two Kinds of Negation*, Dissertation, Freie Universität Berlin, 1993, to appear as Springer LNAI.

[Wagner 1993a] G. Wagner: Reasoning with Inconsistency in Extended Deductive Databases, L.M. Pereira and A. Nerode (Eds.), *Proc. 2nd Int. Workshop on Logic Programming and Nonmonotonic Reasoning*, MIT Press, 1993.

[Wagner 1993b] G. Wagner: Update, Contraction and Revision in Knowledge Representation Systems, in H. Boley, F. Bry and U. Geske (eds.), *Neuere Entwicklungen der deklarativen KI-Programmierung*, DFKI-Report, 1993.

Eine epistemische Erweiterung von Classic als Interaktionssprache für Classic-Wissensbasen

Andreas Becker Gerhard Lakemeyer

Universität Bonn
Institut für Informatik III
Römerstraße 164, 53117 Bonn

e-mail: becker@gmdzi.gmd.de,
gerhard@cs.uni-bonn.de

16. November 1993

Zusammenfassung

Wie Levesque gezeigt hat, läßt sich die Interaktion mit einer Wissensbasis funktional mit Hilfe von Operationen ASK und TELL beschreiben, die es erlauben, Anfragen an das System zu stellen und neue Informationen hinzuzufügen. Insbesondere stellt Levesque heraus, daß eine epistemische Interaktionssprache die Expressivität deutlich erhöht, selbst wenn die Wissensbasis nur aus Formeln der Logik 1. Ordnung (FOL) besteht und keine epistemischen Operatoren enthält. Interessanterweise lassen sich jedoch dabei epistemische ASK- und TELL-Operationen auf herkömmliche nicht-epistemische Operationen reduzieren.

Die Interaktionssprache für Wissensbasen der Konzeptsprache Classic, die mit Abstrichen als FOL-Fragment angesehen werden kann, besteht bisher nur aus Classic-Ausdrücken zusammen mit metalogischen Konzepten, um die Ausdruckskraft zu erhöhen. Das im Workshop-Beitrag dargestellte Projekt hat eine epistemische Erweiterung von Classic (K-Classic) zum Inhalt, mit dem Ziel, eine mächtigere Interaktionssprache für Operationen wie ASK und TELL zu erhalten. Die Erweiterung in der Mächtigkeit der Interaktionssprache besteht darin, daß nicht ausschließlich die Domäne Interaktionsgegenstand ist, sondern auch das Wissen der Wissensbasis über die Domäne zum Interaktionsgegenstand werden kann. Neben der Sprachdefinition ist unter anderem zu überprüfen, ob ähnliche Reduktionsverfahren von epistemischen auf nicht-epistemische Operationen wie im Levesqueschen Ansatz möglich sind. Außerdem wird auf Integritätsbedingungen eingegangen, die als epistemische Anfragen formuliert stets die Antwort 'ja' erhalten müssen, und insofern als spezielle ASK-Operationen anzusehen sind. Schließlich kann mittels epistemischer Ausdrücke als Argumente der TELL-Operation u.a. die Closed World Assumption lokal für Konzepte und Rollen durchgesetzt werden.

1 Eine epistemische Erweiterung von Classic als Interaktionssprache für Classic-Wissensbasen

Wie Levesque in [Levesque 84] gezeigt hat, läßt sich die Interaktion mit einer Wissensbasis funktional mit Hilfe von Operationen ASK und TELL beschreiben, die es erlauben, Anfragen an das System zu stellen und neue Informationen hinzuzufügen. Insbesondere stellt Levesque heraus, daß eine epistemische Interaktionssprache die Expressivität deutlich erhöht, selbst wenn die Wissensbasis nur aus Formeln der Logik 1. Ordnung (FOL) besteht und keine epistemischen Operatoren enthält. Interessanterweise lassen sich jedoch dabei epistemische ASK- und TELL-Operationen auf herkömmliche nicht-epistemische Operationen reduzieren.

Die Interaktionssprache für Wissensbasen der Konzeptsprache Classic, die mit Abstrichen als FOL-Fragment angesehen werden kann, besteht bisher nur aus Classic-Ausdrücken zusammen mit metalogischen Konzepten, um die Ausdruckskraft zu erhöhen. Uns geht es in diesem Projekt um eine epistemische Erweiterung von Classic (K-Classic) mit dem Ziel, eine mächtigere Interaktionssprache für Operationen wie ASK und TELL zu erhalten. Neben der Sprachdefinition geht es uns unter anderem darum zu überprüfen, ob ähnliche Reduktionsverfahren von epistemischen auf nicht-epistemische Operationen wie im Levesqueschen Ansatz möglich sind. Außerdem wird auf den Einsatz epistemischer Ausdrücke als Integritätsbedingungen und als Argument der TELL-Operation eingegangen.

2 Classic

Classic ist eine bei AT&T entwickelte und in Common LISP, neuerdings auch in C implementierte Konzeptsprache. Bezüglich Syntax und Semantik der in Classic enthaltenen Sprachkonstrukte sei auf [Resnick et al. 91] und [Brachman et al. 90] verwiesen. Oberstes Designziel bei der Konzeption von Classic ist Effizienz im Subsumptionsverhalten gewesen, die durch eine vergleichende empirische Studie ([Heinsohn et al. 92]) im wesentlichen auch bestätigt worden ist. Das vergleichsweise günstige Laufzeitverhalten geht jedoch zu Lasten der Expressivität der Sprache, indem beispielsweise eine wesentliche Quelle von Ineffizienz, nämlich Disjunktion als expliziter konzeptbildender Operator (vgl. [Donini et al. 91]), nicht Bestandteil von Classic ist. Disjunktion ist in Classic nur auf der Ebene der Individuen mit Hilfe des ONE-OF-Operators möglich. Ebensfalls nur sehr eingeschränkt sind die Möglichkeiten, Negation auszudrücken, entweder wiederum über das ONE-OF-Konstrukt (Individuen schließen sich implizit gegenseitig aus) oder mit Hilfe von DISJOINT-PRIMITIVE.

Neben der vergleichsweise stark eingeschränkten Expressivität von Classic ist eine weitere Folge der Priorisierung der Effizienz, daß der verwendete Subsumptionsalgorithmus zwar korrekt, aber nicht vollständig ist, so daß nicht alle Subsumptionen erkannt werden (vgl. zum Beispiel [Heinsohn et al. 92]). Als eine Ursache für die Unvollständigkeit wird unter anderem die für Termsubsumptionssprachen untypische minimale Trennung von konzeptuellem (T-Box) und assertionalem (A-Box) Wissen identifiziert. Beispielsweise können Konzepte (T-Box) unter Rückgriff auf Individuen (A-Box) definiert werden, und zwar durch Aufnahme der Konstrukte (FILLS R $\text{individuum}_1 \ldots \text{individuum}_n$) und (ONE-OF $\text{individuum}_1 \ldots \text{individuum}_n$) in die Konzeptdefinition.

3 Motivation für eine epistemische Erweiterung von Classic

Die Motivation für eine epistemische Erweiterung von Classic besteht in den folgenden Zielen:

- Die Möglichkeiten für ASK-Operationen sollen erweitert werden, indem auch Anfragen an eine Wissensbasis darüber möglich werden, was dieser Wissensbasis bekannt beziehungsweise unbekannt ist. Nicht der in der Wissensbasis modellierte Weltausschnitt ist Gegenstand der Anfragen, sondern das, was der Wissensbasis über diesen Weltausschnitt bekannt ist. Dieser Unterschied ist relevant, solange das Wissen der Wissensbasis über den Weltausschnitt unvollständig ist.
- Als Sonderfall von ASK-Operationen soll die Durchsetzung von Integritätsbedingungen in der Wissensbasis ermöglicht werden. Integritätsbedingungen sind epistemische Anfragen über den Zustand einer Wissensbasis, die zu jedem Zeitpunkt mit 'ja' beantwortet werden müssen, damit die Integrität der Wissensbasis garantiert ist.
- Es sollen TELL-Operationen möglich werden, die den Zustand einer Wissensbasis insbesondere dahingehend verändern, daß einzelne Konzepte oder Rollen als geschlossen gelten in dem Sinne, daß ihre Extension vollständig bekannt ist. Mittels solcher TELL-Operationen wird somit eine lokale Closed World Assumption durchgesetzt, die neue (vor allem allquantifizierte) Inferenzen ermöglicht.
- Classic enthält Triggerregeln, Regeln des Inhalts „wenn ein Individuum unter ein Konzept C subsumiert wird, so auch unter ein Konzept D". Wie [Donini et al. 92] aufzeigen, ist die Semantik derartiger Regeln mit Hilfe eines epistemischen Operators formalisierbar.

4 Syntax und Semantik von K-Classic

4.1 Syntax von K-Classic

Zur Darstellung der Syntax und Semantik von K-Classic übernehmen wir weitgehend die Terminologie aus [Donini et al. 92]. Im folgenden stellen C Konzepte (einstellige Prädikate), A primitive Konzepte und P Rollen (zweistellige Prädikate) dar. $\mathcal{O}$ sei ein Alphabet von Symbolen a_i. n stehe für eine natürliche Zahl, g und j für natürliche Zahlen oder Zeichenketten.

Unsere epistemische Erweiterung von Classic sieht syntaktisch folgendermaßen aus:

$C, C_i \longrightarrow$ A
CLASSIC-THING
(AND $C_1 \ldots C_n$)
(ALL R C)
(AT-LEAST n R)
(AT-MOST n R)
(SAME-AS (R_1 $R_2 \ldots R_n$) (R'_1 $R'_2 \ldots R_m$))
(PRIMITIVE C j)
(DISJOINT-PRIMITIVE C g j)
(ONE-OF a_1 $a_2 \ldots a_n$)
(FILLS R a_1 $a_2 \ldots a_n$)
(K C)

$R \longrightarrow$ P | (K P)

Classic enthält außer den genannten konzeptbildenden Operatoren auch die Test-Prädikate TEST-C und TEST-H, die Schnittstellen zur LISP-Umgebung darstellen, sowie die primitiven Konzepte THING und HOST-THING, die die Welt der LISP-Objekte subsumieren. Diese Konstrukte werden der Einfachheit halber nicht berücksichtigt.

4.2 Semantik von K-Classic

p_1 und p_2 seien Parameter, Elemente einer Domäne Δ, γ eine injektive Abbildung von $\mathcal{O}$ auf Δ. $\mathcal{I}$ sei eine Interpretation, die jedem Konzept eine Teilmenge aus Δ und jeder Rolle eine Teilmenge aus $\Delta \times \Delta$ zuordnet, wobei Zuordnungen auch für durch konzeptbildende Operatoren formierte Konzepte entsprechend der Semantik dieser Operatoren vorgenommen werden. Da es sich bei K-Classic um eine epistemische Sprache handelt, bestehen Modelle aus zwei Teilen, einer Interpretation $\mathcal{I}$, die die reale Welt repräsentiert, und einer Menge von Interpretationen $\mathcal{W}$, die das Wissen des Systems beschreiben. Kurz gesagt weiß das System genau die Sätze, die in allen Interpretationen in $\mathcal{W}$ wahr sind (dies entspricht der Mögliche-Welten-Semantik der Logik K45, siehe [Halpern 92]). Eine epistemische Interpretation ist ein Paar $(\mathcal{I}, \mathcal{W})$, wobei $\mathcal{I}$ eine Interpretation und $\mathcal{W}$ eine Menge von Interpretationen ist, so daß die folgenden Gleichungen erfüllt werden:

$$
\begin{array}{lcl}
\text{CLASSIC-THING}^{\mathcal{I},\mathcal{W}} & = & \Delta \\
\text{A}^{\mathcal{I},\mathcal{W}} & = & \text{A}^{\mathcal{I}} \\
\text{p}^{\mathcal{I},\mathcal{W}} & = & \text{p}^{\mathcal{I}} \\
(\text{AND } \text{C}_1\ \text{C}_2 \ldots \text{C}_n)^{\mathcal{I},\mathcal{W}} & = & \text{C}_1^{\mathcal{I},\mathcal{W}} \cap \text{C}_2^{\mathcal{I},\mathcal{W}} \cap \ldots \text{C}_n^{\mathcal{I},\mathcal{W}} \\
(\text{ALL R C})^{\mathcal{I},\mathcal{W}} & = & \{p_1 \in \Delta \mid \forall p_2 (p_1,p_2) \in \text{R}^{\mathcal{I},\mathcal{W}} \rightarrow p_2 \in \text{C}^{\mathcal{I},\mathcal{W}}\} \\
(\text{AT-LEAST n R})^{\mathcal{I},\mathcal{W}} & = & \{p_1 \in \Delta \mid |\{p_2 \in \Delta \mid (p_1,p_2) \in \text{R}^{\mathcal{I},\mathcal{W}}\}| \geq n\} \\
(\text{AT-MOST n R})^{\mathcal{I},\mathcal{W}} & = & \{p_1 \in \Delta \mid |\{p_2 \in \Delta \mid (p_1,p_2) \in \text{R}^{\mathcal{I},\mathcal{W}}\}| \leq n\} \\
(\text{SAME-AS } (\text{R}_1\ \text{R}_2 \ldots \text{R}_n) & & \\
\quad (\text{R}'_1\ \text{R}'_2 \ldots \text{R}_m))^{\mathcal{I},\mathcal{W}1} & = & \{p_1 \in \Delta \mid \ (p_1,p_2) \in \text{R}_1^{\mathcal{I},\mathcal{W}} \wedge (p_2,p_3) \in \text{R}_2^{\mathcal{I},\mathcal{W}} \\
& & \quad \wedge \ldots \wedge (p_{n-1},p_n) \in \text{R}_n^{\mathcal{I},\mathcal{W}} \wedge \\
& & \quad (p_1,p'_2) \in \text{R}_1'^{\mathcal{I},\mathcal{W}} \wedge (p'_2,p'_3) \in \text{R}_2'^{\mathcal{I},\mathcal{W}} \\
& & \quad \wedge \ldots \wedge (p_{m-1},p_m) \in \text{R}_m^{\mathcal{I},\mathcal{W}} \\
& & \quad \rightarrow p_n = p_m\} \\
(\text{PRIMITIVE C j})^{\mathcal{I},\mathcal{W}} & \subseteq & \text{C}^{\mathcal{I},\mathcal{W}} \\
(\text{DISJOINT-PRIMITIVE C g j})^{\mathcal{I},\mathcal{W}} & \subseteq & \text{C}^{\mathcal{I},\mathcal{W}} \wedge \\
& & ((\text{DISJOINT-PRIMITIVE C g j})^{\mathcal{I},\mathcal{W}} \\
& & \cap\ (\text{DISJOINT-PRIMITIVE C g j'})^{\mathcal{I},\mathcal{W}} = \emptyset), \text{j} \neq \text{j'} \\
(\text{ONE-OF } \text{a}_1\ \text{a}_2 \ldots \text{a}_n)^{\mathcal{I},\mathcal{W}} & = & \{\gamma(\text{a}_1), \gamma(\text{a}_2), \ldots, \gamma(\text{a}_n)\} \subseteq \Delta \\
(\text{FILLS R } \text{a}_1\ \text{a}_2 \ldots \text{a}_n)^{\mathcal{I},\mathcal{W}} & = & \{p \in \Delta \mid \{(p,\gamma(\text{a}_1)), (p,\gamma(\text{a}_2)), \ldots, (p,\gamma(\text{a}_n))\} \subseteq \text{R}^{\mathcal{I},\mathcal{W}}\} \\
(\text{K C})^{\mathcal{I},\mathcal{W}} & = & \bigcap_{\mathcal{J} \in \mathcal{W}} (\text{C}^{\mathcal{J},\mathcal{W}}) \\
(\text{K P})^{\mathcal{I},\mathcal{W}} & = & \bigcap_{\mathcal{J} \in \mathcal{W}} (\text{P}^{\mathcal{J},\mathcal{W}})
\end{array}
$$

(K C) wird also in $\mathcal{W}$ als die Menge derjenigen Parameter interpretiert, die in allen Modellen von $\mathcal{W}$ zur Extension des Konzepts C gehören. Im Zusammenhang mit einer K-Classic Wissensbasis interessiert als $\mathcal{W}$ insbesondere die Menge aller Modelle der Wissensbasis relativ zu einem gegebenen Δ.[2] Informell besteht (K C) dann aus den Objekten, von denen in der Wissensbasis bekannt ist, daß sie Instanzen von C sind ((K P) analog).

4.3 Ein Beispiel für K-Classic

Wir zeigen an einem kleinen Beispiel, daß K-Classic gegenüber Classic als Interaktionssprache einen Gewinn an Expressivität bringt. Angenommen, in der Wissensbasis ist notiert, daß Mary zwei Kinder hat, John und Peter, und daß John und Peter männlichen Geschlechts sind. Die Individuen *john* und *peter* sind demnach Rollenfüller der Rolle *child* in der Definition des Individuums *mary*, beide Individuen werden vom Konzept *male* subsumiert. Die Definition von *mary* enthält dann den Bestandteil

```
(AND  (FILLS child peter)
      (FILLS child john))
```

[1]In Classic sind beim SAME-AS-Operator aus Effizienzgründen nur Attribute, d.h. funktionale Relationen zugelassen.

[2]Der Einfachheit halber kann als Domäne das Herbrand-Universum angenommen werden, was der Menge aller Parameter entspricht.

Unter der Voraussetzung, daß die Rolle *child* für das Individuum *mary* nicht geschlossen ist, würde die Anfrage

(ALL child male)(mary)

die Antwort 'unknown' bekommen, da nicht ausgeschlossen werden kann, daß es noch andere, der Wissensbasis nicht bekannte Rollenfüller für die Rolle *child* gibt, die nicht vom Konzept *male* subsumiert werden. Die Anfrage

(ALL (K child) male)(mary)

wird mit 'yes' beantwortet, da alle bekannten Rollenfüller der Rolle *child* in der Definition von *mary* unter das Konzept *male* fallen. Eine Anfrage

(AT-MOST 2 child)(mary)

würde die Antwort 'unknown' erhalten, da unbekannt ist, ob Mary nicht noch mehr Kinder hat. Dagegen würde auf die Anfrage

(AT-MOST 2 (K child))(mary)

die Antwort 'yes' gegeben werden, da nicht mehr als 2 ihrer Kinder bekannt sind. Es lassen sich also durchaus Beispiele dafür aufzeigen, daß eine epistemische Erweiterung von Classic zu einer größeren Expressivität der Interaktionssprache führt.

5 ASK

5.1 Die Semantik des ASK-Operators

Ist Σ eine Classic-Wissensbasis und $\mathcal{M}(\Sigma)$ die Menge aller Modelle von Σ und α eine K-Classic-Anfrage, so ist die Semantik des ASK-Operators definiert als:
$\text{ASK}[\Sigma, \alpha] =$ yes genau dann, wenn für jedes Modell $\mathcal{I} \in \mathcal{M}(\Sigma)$ α wahr in $(\mathcal{I}, \mathcal{M}(\Sigma))$ ist.
Für K-Classic gibt es zwei Verfahren zur Realisierung epistemischer ASK-Operationen.

5.2 Ein constraintbasiertes Verfahren zur Beantwortung epistemischer Anfragen

Eine Aussage α ist für jedes Modell $\mathcal{I} \in \mathcal{M}(\Sigma)$ wahr in $(\mathcal{I}, \mathcal{M}(\Sigma))$ genau dann, wenn es für die Wissensbasis $\Sigma \cup \{\neg\alpha\}$ kein epistemisches Modell gibt. In [Donini et al. 92] wird der aus [Schmidt-SchaußSmolka 91] bekannte tableaux-basierte Algorithmus verwendet, $\Sigma \cup \{\neg\alpha\}$ auf ein Constraintsystem abzubilden, das genau dann unerfüllbar ist, wenn es für $\Sigma \cup \{\neg\alpha\}$ kein epistemisches Modell gibt. Zur Feststellung der Erfüllbarkeit eines Constraintsystems werden sukzessive die unmittelbar konstituierenden Bestandteile komplexer Constraints als eigenständige Constraints hinzugenommen (Vervollständigung), und zwar in einer Weise, daß die Erfüllbarkeitseigenschaft durch diese Erweiterung nicht berührt ist.

Bei Unerfüllbarkeit eines Constraintsystems manifestiert sich schließlich ein syntaktisch erkennbarer Widerspruch in der Constraintmenge, der mit Hilfe sogenannter Clash-Regeln festgestellt werden kann. Vervollständigungsregeln und Clash-Regeln haben sprachspezifischen Charakter.

Im Zusammenhang mit K-Classic entsteht das Problem, daß die Sprache nicht Negation als expliziten konzeptbildenden Operator enthält. Um eine K-Classic-Anfrage negieren zu können, ist K-Classic um die Negation auszuweiten. Für die so erweiterte Sprache haben wir Vervollständigungsregeln und Clash-Regeln ermittelt.

Der wesentliche Vorteil des Verfahrens ist seine Vollständigkeit, der wesentliche Nachteil jedoch seine Ineffizienz.

5.3 Reduktion epistemischer Ausdrücke auf ONE-OF-Konzepte

Eine andere, effizientere Möglichkeit, epistemische Anfragen an eine Classic-Wissensbasis auszuwerten, besteht, wie in [Donini et al. 92] angedeutet[3], in sukzessiven Bottom-Up Ersetzungen von im Skopus eines epistemischen Operators liegenden Konzepten durch jeweils ein ONE-OF-Konzept mit den in der Wissensbasis bekannten Instanzen des Konzepts. Die Ergebnisse solcher Bottom-Up-Ersetzungen hängen somit vom Zustand der Wissensbasis ab. Sind beispielsweise in der Wissensbasis die Individuen $a_1, a_2, \ldots, a_n$ als Instanzen des Konzepts C bekannt, und sind die Individuen $b_1, b_2, \ldots, b_m$ als Instanzen des Konzepts (ALL P (ONE-OF a_1 a_2 ... a_n)) bekannt, so würden sich für eine Anfrage (K (ALL P (K C))) (c) folgende Transformationen ergeben:

(K (ALL P (K C))) (c) ->

(K (ALL P (ONE-OF a_1 ... a_n))) (c) ->

(ONE-OF b_1 ... b_m) (c)

Soweit im Skopus des epistemischen Operators ein Konzept liegt, erhält man durch die einzelnen Transformationen in ONE-OF-Konzepte K-Classic-Ausdrücke und letzten Endes einen Classic-Ausdruck. Im Prinzip wird ähnlich dem Reduktionsverfahren von Levesque vergegangen. Die Korrektheit des Verfahrens kann entsprechend durch Rückbezug auf die Levesquesche Reduktionsabbildung gezeigt werden.

Ein Vorteil des Verfahrens ist, daß es zum großen Teil in Classic selbst realisiert werden kann, indem ein Classic-Konzept im Skopus des K-Operators in die Classic-Wissensbasis eingeführt wird, anschließend der Subsumptionsgraph neu aufgebaut wird, und die Menge der bekannten Instanzen des Classic-Konzepts zu einem ONE-OF-Konzept zusammengefaßt wird. Allerdings können in Classic einmal in die Wissensbasis eingetragene Konzepte in der gegenwärtigen Implementierung nicht wieder entfernt werden, so daß die Wissensbasis allmählich aufgebläht wird. Die Implementierung ist insoweit zu modifizieren, daß Konzepte nach Bearbeitung einer epistemischen Anfrage wieder gelöscht werden können (der Subsumptionsgraph muß dann erneut angestoßen werden).

Liegt im Skopus des epistemischen Operators eine Rolle, so kann die Verarbeitung nicht in Classic erfolgen. Bei einer Anfrage der Form (AT-LEAST n (K P)) beispielsweise sind alle

[3] außerdem persönliche Kommunikation mit Werner Nutt

Individuenpaare zu ermitteln, die in der Wissensbasis in der Relation P stehen. Man erhält damit Individuen*paare* (a_i, b_i), die nicht als Argumente des ONE-OF-Operators auftreten können. Es gibt in Classic kein Pendant zum ONE-OF-Operator für Individuenpaare, mit dessen Hilfe sich eine Menge der Form $((a_1, b_1), \ldots, (a_n, b_n))$ ausdrücken ließe, d.h. eine Rolle extensional definiert werden könnte. Daher kann ein Konstrukt der Form (K P) nicht unmittelbar auf einen nicht-epistemischen Ausdruck reduziert werden. Selbst wenn ein Sprachmittel zur extensionalen Definition von Rollen zur Verfügung stünde, bräuchte man zusätzlich noch Projektionen und die Möglichkeit, die Projektionen Tests zu unterziehen, d.h. man bräuchte zusätzliche rollenbildende Operatoren. Eine solch umfassende Erweiterung von Classic würde andere Schwierigkeiten nachsichziehen, etwa daß neue Regeln zum Erkennen von Inkonsistenz oder zur Subsumption formuliert werden müssen.

Die Lösung des Problems besteht darin, einen Teil der Verarbeitung mittels (relativ einfacher) LISP-Programmierung vorzunehmen. Es wird hierbei ausgenützt, daß Konstrukte der Form (K P) nur innerhalb bestimmter, unmittelbar umschließender Konstrukte auftreten können. Ein solches Konstrukt ist beispielsweise (AT-LEAST n (K P)). Nach Feststellung der (K P)$^{\mathcal{I},\mathcal{M}(\Sigma)}$ entsprechenden Menge von Individuenpaaren wird die Teilmenge der ersten Projektion berechnet, deren Elemente jeweils mindestens n bekannte Rollenfüller (in der zweiten Projektion) haben. Diese Teilmenge der ersten Projektion ist wiederum eine als ONE-OF-Konzept darstellbare *Individuen*menge. Der gesamte Ausdruck (AT-LEAST n (K P)) kann somit schließlich durch ein ONE-OF-Konzept ersetzt, und die weitere Verarbeitung in Classic fortgesetzt werden. Die möglichen, ein Konstrukt (K P) unmittelbar umschließenden Konstrukte stellen wegen ihres konzeptuellen Charakters Individuenmengen dar, die mit einem ONE-OF-Konzept dargestellt werden können. In der gegenwärtigen Implementierung wird der epistemische Operator allerdings nicht auf den Argumentstellen des SAME-AS-Operators zugelassen, da es nur für sehr spezielle Fälle sinnvolle Beispiele gibt.

Ein Nachteil des Verfahrens ist, daß der in Classic implementierte Subsumptionsalgorithmus, wie eingangs festgestellt, unvollständig ist. Um korrekte und vollständige Antworten zu erhalten, wäre eine Reimplementierung des Subsumptionsalgorithmus erforderlich.

6 Integritätsbedingungen

Integritätsbedingungen stellen Forderungen bezüglich der internen Beschaffenheit einer Wissensbasis dar, die zu jedem Zeitpunkt, in jedem Zustand der Wissensbasis erfüllt sein müssen. Aussagen über den Zustand einer Wissensbasis sind Aussagen über die Wissensbasis, d.h. die Wissensbasis selbst wird zum Interaktionsgegenstand. Derartige Aussagen können daher in natürlicher Weise mit einem epistemischen Operator formuliert werden. Integritätsbedingungen können entsprechend als epistemische Anfragen formuliert werden, eine Sichtweise, die auf [Reiter 88] zurückgeht, und zwar Anfragen, die zu jedem Zeitpunkt, in jedem Zustand der Wissensbasis die Antwort 'ja' erhalten müssen, damit die jeweilige Integritätsbedingung erfüllt ist. Integritätsbedingungen können daher als Spezialfall epistemischer ASK-Operationen aufgefaßt werden.

Integritätsbedingungen können zum Teil durch entsprechende Kunstgriffe quasi in die Modellierung eingebaut werden. Dies erfordert allerdings viel Erfahrung, ist meistens umständlich und steht dem Prinzip entgegen, möglichst direkt eine Domäne deklarativ zu modellieren, ohne dabei technische Constraints im Auge behalten zu müssen. Von daher ist es vorzuziehen, Integritätsbedingungen als epistemische Anfragen zu formulieren, die von der Wissensbasis separiert werden, aber nach jeder Veränderung im Zustand der Wissensbasis ausgewertet werden. Bei den Beispielen für Integritätsbedingungen in [Reiter 88] zeigt sich, daß K-Classic allein zur Formulierung vieler Integritätsbedingungen nicht ausreicht. Dies wird deutlich in folgender Integritätsbedingung:

> Für ein Individuenpaar, das in der Relation *mother* steht, muß das erste Individuum von dem Konzept *person*, das zweite Individuum von den Konzepten *person* und *female* subsumiert werden.

$$\forall x, y\ \mathrm{K}mother(x, y) \supset \mathrm{K}(person(x) \wedge person(y) \wedge female(y))$$

In Classic kann eine Konzeptdefinition (etwa durch Triggerregeln in Kombination mit einem Test-Prädikat und einer geeignet definierten LISP-Funktion) so ausgerichtet werden, daß die Integritätsbedingung für alle Instanzen des Konzepts erfüllt ist. Es gibt in Classic jedoch keine allgemeine Möglichkeit, das Vorkommen einer Rolle oberhalb eines bestimmten Konzepts in der Subsumptionshierarchie — in diesem Fall oberhalb des Konzepts *person* — zu untersagen. Die genannte Integritätsbedingung (oder allgemeiner Typspezifizierungen für Rollenargumente) kann für eine Classic-Wissensbasis nicht *global*, d.h. unabhängig von einer einzelnen Konzeptdefinition, durchgesetzt werden. Es gibt also zahlreiche Integritätsbedingungen, die mit Modellierungskniffen nur sehr schwer oder gar nicht für eine Classic-Wissensbasis umgesetzt werden können, aber als epistemische Anfragen problemlos ausgedrückt werden können.

Die als Beispiel angeführte Integritätsbedingung illustriert auch, daß man mit bloßem K-Classic bei den Integritätsbedingungen nicht auskommt. Viele Integritätsbedingungen stellen Implikationen dar, d.h. sie erfordern eine ASK-Sprache, die auch Negation und Disjunktion, sowie Variablenquantifzierung als Sprachmittel zur Verfügung stellt. Es ist somit eine neuerliche Ausweitung der ASK-Sprache um derartige Junktoren erforderlich. Offenbar können nicht alle Bildungen, die eine so mächtige ASK-Sprache hervorbringt, als Argumente von ASK-Operationen für eine Classic-Wissensbasis gehandhabt werden. Hierbei steht eine möglichst allgemeine Definition der im Zusammenhang mit Classic handhabbaren Klassen von Integritätsbedingungen noch aus. Die zugrundeliegende Intuition einer solchen Definition ist, daß eine epistemische ASK-Operation mit einer Integritätsbedingung als Argument in letzter Konsequenz auf Tests reduzierbar sein muß, ob bekannte Individuen in einem Konzept (oder einer Relation) sind oder nicht.

7 TELL

7.1 Die Semantik des TELL-Operators

Mit dem TELL-Operator können neue Informationen zur Wissensbasis hinzugefügt werden. Wir betrachten hierbei nur den monotonen Fall, d.h. durch das Hinzufügen von Informationen zur Wissensbasis verlieren die in der Wissensbasis enthaltenen Aussagen nicht ihre Gültigkeit. Ist $\mathcal{M}(\Sigma)$ wieder die Modellmenge einer Classic-Wissensbasis Σ und *assertion* eine zur Wissensbasis hinzuzufügende Aussage, so ist die Semantik des TELL-Operators:

$$\text{TELL: } \mathcal{M}(\Sigma) \times \mathcal{M}(\textit{assertion}) \mapsto \mathcal{M}(\Sigma) \cap \mathcal{M}(\textit{assertion})$$

Die Modellmenge der um *assertion* erweiterten Wissensbasis entspricht der Schnittmenge der Modellmengen von Σ und *assertion*.

Bei einer so restriktiven Repräsentationssprache wie Classic erhebt sich stets die Frage, ob das Ergebnis einer TELL-Operation mit einem epistemischen Argument in Classic adäquat repräsentierbar ist. Im folgenden geben wir beispielhaft Typen epistemischer Aussagen an, die als Argument einer TELL-Operation zu einem in Classic darstellbaren Ergebnis führen.

7.2 Open World Semantik in Classic und die Durchsetzung einer lokalen Closed World Assumption

Classic liegt, wie allen Konzeptsprachen, eine Open World Semantik zugrunde. Es wird nicht davon ausgegangen, daß alle in einer Wissensbasis bekannten Instanzen eines Prädikats die einzig möglichen sind (vgl. [Resnick et al. 91]). Beispielsweise ist aus dem Umstand, daß alle bekannten Personen männlich sind, nicht folgerbar, daß alle Personen männlich sind. Ist umgekehrt ein Individuum nicht als Instanz eines Konzepts bekannt, wird nicht ausgeschlossen, daß das Individuum dennoch Instanz des Konzepts sein könnte.

Es gibt in Classic jedoch eine sehr spezielle Möglichkeit, lokal eine Closed World Assumption einzuführen. Für ein bekanntes Individuum kann eine Rolle geschlossen werden in dem Sinne, daß nach Schließung der Rolle alle Rollenfüller der Rolle für das Individuum als bekannt gelten. Individuen, die nicht als Rollenfüller der Rolle für das Individuum bekannt sind, können demnach keine Rollenfüller sein. Eine derartige Schließung einer Rolle für ein Individuum kann vom Anwender über eine LISP-Funktion veranlaßt werden, oder vom System aus der Attributeigenschaft der Rolle oder über vorhandene Zahlenrestriktionen geschlossen werden. Formalisiert werden kann diese in Classic nur informell beschriebene Aktion als eine TELL-Operation mit dem epistemischen Argument

$$\forall y \; P(a,y) \supset \mathbf{K}P(a,y)$$

Eine Gruppe von in Classic repräsentierbaren TELL-Operationen stellen Operationen zur Einführung einer lokalen Closed World Assumption für ein Konzept oder eine Menge von Rollenfüllern dar. Ein Konzept beispielsweise kann durch eine TELL-Operation mit dem Argument

$$\forall x \; C(x) \supset \mathbf{K}C(x)$$

geschlossen werden. Technisch wird diese Operation realisiert, indem in die Definition des Konzepts C ein ONE-OF-Konzept mit der Menge der bekannten Instanzen von C als Extension eingefügt wird. Eine andere Variante stellt die Aussage

$$\forall x\ C(x) \supset \mathrm{K}D(x)$$

dar, wobei die Menge der bekannten Instanzen des Konzepts D die *möglichen* Instanzen des Konzepts C festlegt. In diesem Fall wird für das Konzept C eine abgeschwächte Form der Closed World Assumption eingeführt. Unterscheiden sich die auf beiden Seiten des Implikationszeichens vorkommenden Prädikate, kann es bei der Ausführung der entsprechenden TELL-Operation zu Konsistenzproblemen kommen. Ist nämlich in der Wissensbasis eine Instanz des Konzepts C bekannt, ohne auch als Instanz des Konzepts D zu gelten, so führt die Ausführung einer TELL-Operation mit obigem Argument zur Inkonsistenz der Wissensbasis. Die TELL-Operation führt nur dann nicht zur Inkonsistenz, wenn für die Wissensbasis Σ die Voraussetzung

$$\mathrm{K}C^{\mathcal{I},\mathcal{M}(\Sigma)} \subseteq \mathrm{K}D^{\mathcal{I},\mathcal{M}(\Sigma)}$$

erfüllt ist. Es handelt sich hierbei um eine Integritätsbedingung, die in der Wissensbasis erfüllt sein muß, damit die TELL-Operation mit obigem Argument nicht die Konsistenz der Wissensbasis gefährdet. Als Integritätsbedingung kann die Voraussetzung als epistemische ASK-Operation auf ihre Gültigkeit überprüft werden. Unter dem Aspekt der Konsistenzerhaltung können epistemische ASK- und TELL-Operationen in Relation zueinander gesetzt werden. Es bleibt zu sehen, inwieweit diese Relation formal spezifizierbar ist.

7.3 Komplexe Updates und Triggerregeln

Andere TELL-Operationen mit epistemischem Argument erlauben komplexe Updates ähnlich dem Konzept der Triggerregeln in Classic. Die Bedeutung des Regelformats der Triggerregeln in Classic ist in der Classic-Literatur nur informell beschrieben als „ist ein Individuum eine Instanz eines Konzepts C, so gilt das Individuum auch als Instanz eines anderen Konzepts D". Die Semantik läßt sich als TELL-Operation mit dem Argument

$$\forall x\ \mathrm{K}C(x) \supset D(x)$$

formalisieren, wobei allerdings diese TELL-Operation nach jedem Update der Wissensbasis ausgeführt werden muß.

Offenbar gibt es zwei Möglichkeiten der Formalisierbarkeit von Triggerregeln. Neben der genannten Formalisierung als epistemische TELL-Operation läßt sich die Semantik der Regeln auch als epistemische ASK-Operation, genauer als Integritätsbedingung, erfassen. Demnach wird eine Triggerregel spezifiziert durch eine Integritätsbedingung, die in der Wissensbasis nach Ausführung der Triggerregel erfüllt ist. Eine Wissensbasis nach Ausführung einer Triggerregel des in Classic realisierten Formats erfüllt die Integritätsbedingung

$$\forall x\ \mathrm{K}C(x) \supset \mathrm{K}D(x)$$

(vgl. die Formalisierung in [Donini et al. 92]). Während die Formalisierung als epistemische TELL-Operation die ***Aktion*** bei der Ausführung der Triggerregel akzentuiert, spezifiziert die Formalisierung als epistemische ASK-Operation (beziehungsweise Integritätsbedingung) den ***Zustand*** der bei Ausführung der Triggerregel resultierenden Wissensbasis.

Wird obige TELL-Operation nicht nach jedem Update der Wissensbasis ausgeführt, so erhält man eine Variante der Triggerregeln in Classic. Ähnlich gibt es eine Reihe von Varianten der obigen epistemischen Aussage, die jeweils als Argument einer TELL-Operation eine Variierung des in Classic realisierten Konzepts der Triggerregeln darstellt. Beispiele hierfür sind

$$\forall y\ \mathrm{K}C(y) \supset P(a, y)$$

und

$$\forall x \forall y\ \mathrm{K}P(x, y) \supset Q(x, y)$$

Durch Variierung epistemischer Argumente von TELL-Operationen und Variierung in Hinsicht auf einmalige, mehrmalige oder ständige Ausführung der TELL-Operation ergibt sich eine im Vergleich zu dem in Classic realisierten Regelformat allgemeinere Konzeption von Triggerregeln. Die zweifache Formalisierbarkeit ist auch für diese allgemeinere Konzeption gegeben.

7.4 Bei Einschränkung auf Mengen bekannter Individuen realisierbare TELL-Operationen

Einige Typen epistemischer TELL-Operationen führen an sich zu nicht in Classic darstellbaren Ergebnissen, weil sie Konsequenzen für (noch) nicht der Wissensbasis bekannte Individuen involvieren. Ein solches Beispiel ist die globale Schließung einer Rolle P:

$$\forall x, y\ P(x, y) \supset \mathrm{K}P(x, y)$$

Für später der Wissensbasis bekanntwerdende Individuen wäre aus einer solchen TELL-Operation zu folgern, daß diese Individuen keinen Rollenfüller der Rolle P haben. Es gibt jedoch keine Möglichkeit, in einer Classic-Wissensbasis etwas zu notieren, das die Wissensbasis bei Bekanntwerden eines neuen Individuums genau diese Schlußfolgerung ausführen läßt (in anderen Konzeptsprachen größerer Expressivität wird dies kaum anders sein). Eine zunächst nicht in Classic repräsentierbare TELL-Operation wie die genannte wird jedoch ausführbar, wenn sie durch Aufnahme einer epistemischen Prämisse in das Argument auf Mengen *bekannter* Individuen eingeschränkt wird:

$$\forall x, y\ \mathrm{K}C(x) \wedge P(x, y) \supset \mathrm{K}P(x, y)$$

Die Rolle P ist demnach nur für die bekannten Instanzen des Konzepts C zu schließen, was in Classic darstellbar ist. C kann dabei auch das eingebaute Prädikat CLASSIC-THING sein, d.h. $\mathrm{K}C$ entspräche der Menge aller in der Wissensbasis bekannten Individuen.

7.5 Im Ergebnis in Classic nicht darstellbare TELL-Operationen

Daß epistemische TELL-Operationen Konsequenzen für der Wissensbasis (noch) nicht bekannte Individuen haben können, ist ein Beispiel dafür, daß das Ergebnis von TELL-Operationen nicht in Classic repräsentierbar ist. Ein anderes Beispiel ist die Asymmetrie der Notation von Rollen in Classic. Assertionen bezüglich Rollen werden ausschließlich beim Rollenträger notiert. Beim Rollenfüller kann nicht notiert werden, für welche Individuen und welche Rollen er Rollenfüller ist. Aus diesem Grunde sind beispielsweise Existenzaussagen über Rollenfüller in Classic möglich (mit Hilfe des AT-LEAST-Konstrukts), nicht aber über Rollenträger. So erklärt sich, daß eine TELL-Operation mit dem Argument

$$\exists y\ P(a,y) \land \neg \mathrm{K}P(a,y)$$

in Classic im Ergebnis darstellbar ist, mit dem Argument

$$\exists x\ P(x,b) \land \neg \mathrm{K}P(x,b)$$

jedoch nicht. In der Asymmetrie in der Darstellung von Rollen liegt auch begründet, daß eine TELL-Operation mit der Aussage

$$\forall y\ P(a,y) \supset \mathrm{K}P(a,y)$$

im Ergebnis repräsentiert werden kann, nicht aber mit dem Argument

$$\forall x\ P(x,b) \supset \mathrm{K}P(x,b)$$

7.6 Redundante oder zu Inkonsistenz führende TELL-Operationen

Bereits Levesque (vgl. [Levesque 84]) hatte festgestellt, daß bestimmte Klassen epistemischer Aussagen als Argumente von TELL-Operationen entweder redundant sind oder zu Inkonsistenz führen. Es handelt sich hierbei insbesondere um subjektive Aussagen etwa der Form

$$\mathrm{K}C(a)$$

Der Wissensbasis wird mitgeteilt; daß ihr bekannt ist, daß a eine Instanz des Konzepts C ist. Ist in der Wissensbasis das Individuum a als Instanz des Konzepts C bekannt, so ist diese Mitteilung redundant. Ist a dagegen nicht als Instanz von C bekannt, so kommt es durch die TELL-Operation zum Widerspruch, d.h. die Modellmenge der aus der TELL-Operation resultierenden Wissensbasis ist leer. TE LL-Operationen mit derartigen Argumenten machen ausschließlich Aussagen über das Wissen der Wissensbasis, deren Wahrheitsgehalt unabhängig von Domänenzusammenhängen ist. Syntaktisch sind die subjektiven Aussagen daran erkennbar, daß alle in ihnen vorkommenden Prädikate im Skopus eines K-Operators liegen. Aber auch gemischte Aussagen, wie beispielsweise

$$\forall x\ C(x) \supset \neg \mathrm{K}C(x)$$

können als Argumente von TELL-Operationen ein ähnliches Verhalten zeigen.

7.7 Probleme bei der Realisierung epistemischer TELL-Operationen für Classic

Zwei Probleme stellen sich im Zusammenhang mit der Realisierung epistemischer TELL-Operationen:

1. Ähnlich wie bei den Integritätsbedingungen ist eine *Sprache* zu definieren, die alle die epistemischen Aussagen umfaßt, die als Argumente von TELL-Operationen zu in Classic darstellbaren Ergebnissen führen und nicht redundant oder Inkonsistenz bewirkend sind. Hierbei ist nicht klar, ob es vor dem Hintergrund der geringen Expressivität von Classic wirklich gelingt, allgemeine Constraints für die Beschaffenheit epistemischer Argumente für im Ergebnis darstellbare und 'sinnvolle' TELL-Operationen zu formulieren. Schlimmstenfalls läßt sich nur eine Liste von Aussagetypen oder Templates zusammenstellen, die instantiiert als Argumente zu handhabbaren TELL-Operationen verwendet werden können.

2. Obwohl die meisten der als handhabbar befundenen TELL-Operationen relativ einfach mit den in Classic zur Verfügung stehenden Update-Möglichkeiten realisiert werden können, ist noch unklar, inwieweit eine *allgemeine* Abbildung spezifiziert werden kann, die ausgehend von der syntaktischen Form eines Arguments adäquat die Classic Update-Funktionen instantiiert und zur Ausführung bringt, also die TELL-Operation eigentlich erst realisiert.

8 Zusammenfassung

Wir haben an einem Beispiel verdeutlicht, daß eine epistemische Erweiterung von Classic zu größerer Expressivität der Interaktionssprache führt. Es können so viele Anfragen an eine Classic-Wissensbasis über das, was der Wissensbasis bekannt beziehungsweise unbekannt ist, gestellt werden, die als reine Classic-Anfragen nicht formulierbar wären. Ein Verfahren zur Beantwortung epistemischer Anfragen wurde in prinzipieller Anlehnung an das Reduktionsverfahren von Levesque realisiert.

Integritätsbedingungen können als spezielle epistemische ASK-Operationen aufgefaßt werden. Zur Formulierung nichttrivialer Integritätsbedingungen reicht K-Classic allein offenbar nicht aus. Hier ist noch das Format einer Erweiterung der Anfragesprache um solche Konstrukte wie Disjunktion und Negation festzulegen.

TELL-Operationen ermöglichen u. a. die Durchsetzung einer lokalen Closed World Assumption für Konzepte und Rollen und führen zu einer allgemeineren Konzeption der in Classic realisierten Triggerregeln. Wegen der Eingeschränktheit von Classic als Repräsentationssprache sind nicht alle TELL-Operationen mit beliebigem Argument in ihrem Ergebnis in Classic darstellbar. Auch für die TELL-Operationen ist eine Sprache der handhabbaren Argumente zu spezifizieren.

Literatur

[Brachman et al. 90] R. J. Brachman, D. L McGuinness, P. F. Patel-Schneider, L. A. Resnick, *Living with Classic: When and How to Use a KL-ONE-Like Lanuage*, in: J. Sowa (ed.), Principles of Semantic Networks, Morgan Kaufmann, San Mateo 1990.

[Donini et al. 91] F. M. Donini, M. Lenzerini, D. Nardi, W. Nutt, *The complexity of concept languages*, in: J. Allen, R. Fikes, E. Sandewall (eds.), Proc. of the 2nd International Conference on Principles of Knowledge Representation and Reasoning KR-91, Morgan Kaufmann, San Mateo,1991,151-162.

[Donini et al. 92] F. M. Donini, M. Lenzerini, D. Nardi, W. Nutt, A. Schaerf, *Adding Epistemic Operators to Concept Languages*, in: Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning, Morgan Kaufmann, San Mateo 1992, 342-53.

[Halpern 92] J. Y. Halpern, Y. Moses, *A guide to completeness and complexity for modal logics of knowledge and belief*, in: Artificial Intelligence 54 (1992), 319–379.

[Heinsohn et al. 92] J. Heinsohn, D. Kudenko, B. Nebel, H.-J. Profitlich, *An Empirical Analysis of Terminological Representation Systems*, DFKI-Research-Report RR-92-16, Saarbrücken, Mai 1992.

[Levesque 84] H. J. Levesque, *Foundations of a functional approach to knowledge representation*, in: Artificial Intelligence, 23:155-212, 1984.

[Reiter 88] R. Reiter, *On Integrity Constraints*, Proceedings of the 2nd Conference on Theoretical Aspects of Reasoning about Knowledge, Morgan Kaufmann, San Mateo 1988, 97-111.

[Resnick et al. 91] L. A. Resnick et al., *Classic Description and Reference Manual For the Common Lisp Implementation Version 1.2*, 1991.

[Schmidt-SchaußSmolka 91] M. Schmidt-Schauß, G. Smolka, *Attributive concept descriptions with complements*, in: Artificial Intelligence, 48(1):1-26, 1991.

Zum Unterschied zwischen Glauben und Wissen

Claus-Rainer Rollinger
Universität Osnabrück
Arbeitsbereich CL&KI

Abstract

Der Unterschied zwischen Glauben und Wissen, den die Philosophie im Rahmen der epistemischen Logik behandelt, spielt in der KI in verschiedenster Hinsicht eine wichtige Rolle. Die mit dieser Unterscheidung verbundene Differenzierung macht es uns möglich, die verschiedensten Aspekte der Unexaktheit (Unvollständigkeit, Unsicherheit, Vagheit, Defaults) zusammenzufassen und natürlichsprachlich auszudrücken. Wir betrachten insbesondere Wissen und Glauben in Relation zu Handlungen und zeigen eine Beziehung zwischen den Erklärungen von Handlungen und den Gründen auf, die dafür ausschlaggebend sind, daß ein Sachverhalt geglaubt bzw. gewußt wird. Dabei sind gute Gründe von weniger guten zu unterscheiden, was durch die in der Begründung verwendeten Schlußweisen sowie durch den Status der beteiligten Wissenselemente möglich wird.

Es macht nicht nur für Sally einen großen Unterschied, ob sie *glaubt*, daß sie schwanger ist, oder ob sie *weiß*, daß sie schwanger ist. Sally's Glauben oder Wissen wird ihre nächsten Handlungen entscheidend beeinflussen. So wäre es nicht verwunderlich, falls sie nur glaubt schwanger zu sein, wenn sie Handlungen vollzieht, die ihr Gewißheit verschaffen, die sie also in einen Zustand versetzen, in dem sie entweder weiß, daß sie schwanger ist, oder in dem sie weiß, daß sie es nicht ist. Diese Handlungen sind unsinnig, wenn sie bereits weiß, daß sie schwanger ist. Wenn Sally nun in einer Apotheke einen Schwangerschaftstest kauft, dann hat der Apotheker gute Gründe anzunehmen, daß sie glaubt, schwanger zu sein, dieses aber nicht weiß. An diesem Beispiel können wir bereits zwei wichtige Beobachtungen machen:

(1) Wenn A glaubt, daß p, dann kann dieses ganz andere Handlungen von A zur Folge haben, als wenn A weiß, daß p.

(2) Aus den Handlungen von A läßt sich von einem Beobachter B u.U. erschließen, ob A weiß, daß p, oder ob er glaubt, daß p.

Dabei stehen die jeweiligen Handlungen in engem inhaltlichen Bezug zu p. Zwei Arten von Handlungen können voneinander getrennt werden: Einerseits die Handlungen, die "dazu beitragen", aus dem Glauben von p Wissen über p zu machen, und andererseits diejenigen Handlungen, die unter dem Aspekt einer globaleren Zielverfolgung auf der Grundlage von p möglich sind. Je nachdem wie wichtig p für die Zielerreichung ist, wird darüber befunden werden, ob es ausreicht, p zu glauben, oder ob es als notwendig erachtet wird, p zu verifizieren. Da ein großer Alternativenraum von potentiellen Handlungen die Regel sein dürfte, ist es von Bedeutung für ein handlungsfähiges Subjekt, entscheiden zu können, ob p gewußt oder geglaubt wird. Offensichtlich sind wir in der Lage Handlungen auszuführen, die uns im Hinblick auf p in die Lage versetzen, nach Ausführung der Handlungen p zu wissen bzw. $\neg$ p zu wissen. Diese Handlungen stellen dann die Begründungsbasis des Gewußten dar. Wird Sally nach Vollzug eines Schwangerschaftstests gefragt, wie sie denn wissen könne, daß sie schwanger sei, kann sie auf diese Handlung und deren positives Ergebnis verweisen.

(3) Wenn A weiß, daß p, dann kann er auch Gründe für p angeben.

Begründungen werden also eine wichtige Rolle spielen, wobei wir bislang noch nicht in der Lage sind, Glauben von Wissen unterscheiden zu können. Denn auch der Glaube von Sally wird nicht gänzlich unbegründet gewesen sein. Beim Wissen müssen die Gründe allerdings bessere sein als beim Glauben, weshalb man auch von guten Gründen spricht. Was gute Gründe von weniger guten unterscheidet wollen wir später diskutieren.

(3') Wenn A weiß, daß p, dann muß er gute Gründe für p angeben können.

Der Zusammenhang zwischen Plänen, Handlungen, Begründungen, Glauben und Wissen wird im folgenden weiter ausgeführt werden, wobei sprachliche Handlungen im Vordergrund stehen werden. Zunächst aber werden wir uns mit dem Wissensbegriff auseinandersetzen.

1 Der Wissensbegriff in der Philosophie, der KI und der Kognitionswissenschaft

In der Philosophie ist es durchaus gängig, Wissen derart zu definieren, daß der Wahrheit des Gewußten eine wichtige Bedeutung zukommt.

(4) A weiß p gdw. gilt: **p ist wahr**, A glaubt p und A hat gute Gründe, p zu glauben.

Das heißt, daß in (4) der Unterschied zwischen Glauben und Wissen über den Wahrheitsbegriff festgelegt wird. Das Problem hiermit ist, daß irgendwie festgestellt werden muß, was wahr ist. Da dieses nicht von einer neutralen Institution festgelegt werden kann, sondern dadurch, daß Wahrheit empirisch über den Erfolg bzw. Mißerfolg von Handlungen bestimmt wird, hilft uns diese Definition nur dann weiter, wenn wir bereit sind, den Wahrheitsbegriff durch einen empirischen Bestätigungsbegriff zu ersetzen. Die Frage ist, ob dieser von den *guten Gründen* abgegrenzt werden kann, oder aber, ob wir *gute Gründe* von *weniger guten Gründen* unterscheiden werden können.

In der Künstlichen Intelligenz werden dem Begriff Wissen mindestens zwei Bedeutungen zugeordnet:

(5a) Wissen = Repräsentation + Zugriff + Modifikation

(5b) Aussagen über die Welt, die gewußt werden, im Gegensatz zu den Aussagen, die (nur) geglaubt werden.

Wissen umfaßt in (5a) alles, das repräsentiert, auf das zugegriffen und das modifiziert werden muß. Also auch das, was über die Welt geglaubt wird. Geglaubtes Wissen (im Sinne von (5a)) unterscheidet sich von gewußtem Wissen durch Unterschiede in der Repräsentation, im Zugriff und in der Modifikation. Der Unterschied bezieht sich in erster Linie auf die Dimension der Sicherheit, mit der wir davon ausgehen, daß die Welt bestimmte Eigenschaften etc. hat.

Im Rahmen der wahrscheinlichkeitstheoretischen Ansätze zur Behandlung von Unsicherheit bzw. Vagheit (z.B. Zadeh 1971 oder Shafer 1976) kann sehr hohe Sicherheit für p gleichgesetzt werden mit **p wird gewußt**, während eine geringere Sicherheit für p als **p wird geglaubt** interpretiert werden kann. Problematisch ist neben der Frage, wo die numerischen Werte herkommen, daß hierbei der Übergang von Glauben zu Wissen über Schwellwerte geregelt wird.

Die symbolischen Ansätze hingegen (Default-Logiken, Nichtmonotonie) ermöglichen eine Unterscheidung zwischen Glauben und Wissen auf die folgende Art: Ein per Default abgeleitetes p wird geglaubt. Gibt es eine Begründung von p ohne Default, wird p gewußt. Fraglich ist der Status der Axiome (der nicht begründbaren Aussagen)[1].

Beide Familien von Lösungsvorschlägen sind allerdings nur für die Behandlung einzelner Aspekte der Inexaktheit geeignet. Unsicherheit, Vagheit, Unvollständigkeit und Standardannahmen müssen als unterschiedliche Dimensionen der Inexaktheit behandelt werden (Rollinger 1984). Bislang sind wir nicht in der Lage, die Bezüge zwischen diesen Dimensionen zu verstehen[2].

In der Kognitionswissenschaft wird Wissen verstanden als mentales Modell der Welt[3], auf dessen Grundlage Probleme gelöst, Handlungen geplant und das aufgrund von Erfahrungen und Erkenntnisgewinn modifiziert wird. Diese Sichtweise ist eng verwandt mit der Sichtweise der Künstlichen Intelligenz, wobei in der Kognitionswissenschaft die kognitive Adäquatheit von Repräsentation, Zugriff und Modifikation eine vorrangige Rolle spielt.

Vor dem Hintergrund der Frage nach einem formalisierbaren Unterschied zwischen Glauben und Wissen wollen wir im weiteren betrachten, was Handlungen sind und in welchem Zusammenhang Handlungen und Erklärungen stehen.

2 Handlungen

Zunächst wollen wir die notwendigen Bedingungen formulieren, die eine Handlung von einem Reflex, einem Ereignis unterscheiden:

(6) Handeln setzt die Existenz eines Modells der Welt voraus, mit dem Wirkungen (zukünftige Ereignisse) vorausgesagt werden können. Diese Wirkungen müssen intendiert sein. (Prinzip der Wirkungsvorhersage)

(7) Es muß darüber hinaus die Möglichkeit der Unterlassung einer Handlung gegeben sein. (Prinzip möglicher Alternativen), so ist z.B. die unterlassene Hilfeleistung ein strafbarer da willentlicher Akt[4].

---

[1] Wenn ich **weiß**, daß Vögel üblicherweise fliegen können, und daß Tweety ein Vogel ist, dann **glaube** ich, daß Tweety fliegen kann.

[2] Wie formalisiere ich z.B. den (sicherlich bestehenden) Zusammenhang (Äquivalenz ?) zwischen folgenden Aussagen: "Ich bin mir sicher, daß Harry eine Portion Spagetti gegessen hat." und "Ich bin mir unsicher, ob Harry eine große Portion Spagetti gegessen hat".

[3] Man ist sich dabei nicht einig darüber, ob dieses nur für das Kurzzeitgedächtnis gilt, oder auch für das Langzeitgedächtnis bzw. das Hintergrundwissen, wobei davon auszugehen ist, daß die Repräsentationen zumindest sehr gut kompatibel sein müssen. Auch ist unklar, wieviele Arten von mentalen Modellen es gibt, bzw. wie das Wissen über mentale Modelle strukturiert ist.

[4] Zur Veranschaulichung ein Beispiel von S. Kanngießer, der mit dem Alternativenbegriff wesentlich zu der Definition beigetragen hat: Der Stein, der fallen gelassen wird, wird fallen (Wirkungsvorhersage), und hat keine

Wir finden hier noch keinen Ansatzpunkt für eine Unterscheidung zwischen Glauben und Wissen, und der Standpunkt, daß zukünftige Ereignisse nicht gewußt werden können, da die Zukunft nicht deterministisch vorhergesagt werden kann, wird offensichtlich nicht bezogen. Wenn wir einen Stein anheben und dann loslassen, dann wissen wir, daß er fallen wird und nicht in der Luft stehen bleibt. Derartig empirische Tatsachen haben wir nicht nur in Bezug zu den Naturgesetzen entwickelt.

Einen solchen Ansatzpunkt finden wir erst, wenn wir einerseits die Intentionalität als wesentlichen Bestandteil von Handlungen und andererseits Handlungen als Bestandteile von Plänen betrachten. Laut (Searle 1986) hat ein intentionaler Zustand zwei Bestandteile: 1. einen (propositionalen) Gehalt (wie z.B. "ins Zimmer gehen") und 2. einen psychischen Modus (Typ) (wie z.B. "beabsichtigen, ...", "hoffen, ..."). Glauben und Wissen können demnach als Typ eines intentionalen Zustandes verstanden werden, was allerdings nichts weiter austrägt, solange wir nicht die Beziehungen zwischen den verschiedenen Typen sowie deren Eigenschaften kennen.

(8) Handlungen haben ausgezeichnete Beschreibungen (Searle 1986)

Handlungen als solche zu erkennen gelingt uns aufgrund der Kenntnis der relevanten Eigenschaften einer Handlung. Hierzu sind insbesondere die Vor- und Nachbedingungen von Handlungen zu zählen. Eine nur geglaubte aber relevante Vorbedingung führt zu einer ebenfalls nur geglaubten Wirkungsvorhersage.

3 Erklärungen und Handlungen

Das Verstehen von Handlungen[5] bezeichnet den Prozeß der Erklärungskonstruktion auf der Grundlage (naiver) Theorien. Eine Handlung läßt üblicherweise aufgrund der Ambiguität des Verstehens mehrere Erklärungen zu. Mittels Erklärungen können z.B. Handlungen von Ereignissen abgegrenzt werden:

(9) Das Verstehen von Ereignissen erfordert in erster Linie die Zuordnung einer Bedeutung zu einem beobachteten Ereignis (lauter Knall -> Donner), und in zweiter Linie die Einordnung in einen (ev. beobachteten) Ereignisablauf im Hinblick auf Ursache - Wirkung.

(10) Das Verstehen von Handlungen erfordert die Explikation der Handlungsziele und der Pläne, in denen die Handlungen eine Rolle für die Zielerreichung spielen.

Grundlage der Erklärungskonstruktion sind (naive) Theorien (Problemlösungswissen) sowie das Wissen über die faktische Beschaffenheit der Welt. Die gefundenen Erklärungen sind wiederum die Grundlage für eigenes Handeln. Handeln stellt in Verbindung mit der Beobachtung der erzielten Wirkung(en) eine Beziehung zwischen dem Weltmodell und der Realität her. Handeln ist demnach auch immer (gezieltes) Experimentieren. Da der Mensch in der Realität agieren muß, um

---

Möglichkeit, sich zu entscheiden, ob er fallen will oder nicht. Die Hand, die ihn fallen gelassen hat, hätte auch anderes mit dem Stein tun können, zum Beispiel ihn nicht fallen lassen (mögliche Alternativen).

[5] Hier ist nicht gemeint das Erkennen einer Handlung über deren ausgezeichnete Eigenschaften. Verstehen meint hier die Gründe zu bestimmen, aufgrund derer eine Handlung vollzogen wurde.

seine (Grund-) Bedürfnisse befriedigen zu können, und Erkenntnisgewinn dazu beiträgt, die Bedürfnisbefriedigung zu optimieren, kann jedes Handeln als Experiment zur Bestätigung der (naiven) Theorien interpretiert werden. Der aktive Eingriff in die Realität stellt sicher, daß die Wirkungen von Handlungen zu einem (mehr oder minder) selbstgewählten Zeitpunkt beobachtet werden können, der Lernprozess kann somit selbst gesteuert werden. Die Selbststeuerung ist dann nicht gegeben, wenn das eigene Handeln ausgeschlossen wird und man sich ausschließlich auf die Beobachtung von Handlungen und ihren Wirkungen zurückzieht.

Die Qualität einer Theorie (eines Weltmodells) mißt sich damit an der Qualität der Erklärung einerseits und der Qualität der Wirkungsvorhersage andererseits. Sollte sich eine Erklärung bzw. eine Wirkungsvorhersage als mangelhaft erweisen, dann bilden diese Einschätzungen die Grundlage zur Revision der Theorie.

Da die Welt nicht still steht und der Mensch abhängiger Bestandteil der Welt ist, können zukünftige Weltzustände nur angestrebt werden. Dies drückt sich einerseits durch unterbestimmte Beschreibungen der angestrebten Zustände aus, und andererseits dadurch, daß nicht alle Abhängigkeiten kontrolliert werden können.

Die Entscheidung für eine bestimmte Handlungssequenz als Abwägung (aller) verschiedener Intentionen und damit verbundenen Zielzuständen bedingt die Einschätzung, daß für alle Einzelhandlungen die Handlungsvoraussetzungen erreicht werden können (Voraussetzungsprinzip)[6].

Handlungsmuster (Scripts) spielen auch auf der Sprachebene eine wichtige Rolle, nicht nur auf der Weltebene, so wie bei (Schank / Abelson 1977). Nicht jede Sprechäußerung wird vollständig neu geplant, sondern orientiert sich an Mustern, die aus sehr vielen vorherigen Erfahrungen aufgebaut worden sind.

Die intendierte Wirkung einer Sprechhandlung ist die Modifikation eines Weltmodells, eines mentalen Modells / Zustands, eines Informationszustandes[7]. D.h. sprachliches Handeln ist indirektes Handeln: Zuerst wird ein mentales Modell modifiziert, und erst dann kann diese Modifikation eine Handlung auslösen (z.B. wieder eine Sprechhandlung).

4 Begründbarkeit

Der Unterschied zwischen Glauben und Wissen wird also nur deutlich werden, wenn wir wissen, was eine Begründung bzw. eine Erklärung ist. Betrachten wir folgende verallgemeinerte Situation gegensätzlicher Auffassungen:

---

[6] Das Voraussetzungsprinzip gilt auch für die Planung einzelner Teiläußerungen. Es muß geprüft werden, ob der Hörer alle Voraussetzungen erfüllt, um eine Äußerung verstehen zu können. Ist dem nicht so, dann müssen zuvor die Äußerungen getätigt werden, die die Voraussetzungen beim Hörer schaffen, die wiederum das Verstehen der ursprünglich intendierten Äußerung ermöglichen. Beispiel: A kann "Frida hatte einen Unfall." gegenüber B nur dann äußern, wenn er der Meinung ist, daß B Frida kennt und weiß, wer mit Frida gemeint ist. Erfüllt B diese Voraussetzung nicht, dann muß A dieser Äußerung die Mitteilung vorausschicken, daß Frida z.B. die Freundin seines, B's Sohnes ist.

[7] Eine nicht intendierte Wirkung (unsystematische Nebenwirkung) ist z.B. das Aufschrecken eines Hasen, der am Wegesrand sitzt und die Aufmerksamkeit des Sprechers bislang nicht erregt hat.

A zu B:	(1) Ich weiß, daß p.
B:	(2) Das glaube ich nicht.

(2) ist mehrdeutig: Es kann einerseits bedeuten, daß B nicht glaubt, daß A p weiß, p aber damit *nicht in Frage stellt. Es kann aber auch bedeuten, daß p angezweifelt wird, und damit natürlich* auch, daß A p weiß. Wir kommen zu dem Begriff der Begründbarkeit, wenn wir uns überlegen, welche Reaktionsmöglichkeiten A nun offenstehen (hier nur drei wichtige Schemata):

A: (3a) Ich war an (der Herstellung von) p beteiligt.
(3b) Ich habe p mit eigenen Augen gesehen.
(3c) Eine an der Herstellung von p beteiligte Person (die wir beide für zuverlässig halten) hat es mir berichtet.

Jede dieser Reaktionen ist geeignet, B sowohl davon zu überzeugen, daß A berechtigtermaßen behaupten kann, p zu wissen, als auch B von p selbst zu überzeugen. In allen drei Fällen werden gute Gründe genannt, die A berechtigen, p als gesichertes Wissen einzustufen, wobei bei (3c) bereits die Glaubwürdigkeit einer dritten Person bemüht wird, was einen Sicherheitsverlust bedeuten müßte. Wenn keine derartige Begründung abgegeben werden kann, dann wird es auch nicht gelingen, den Zweifel zu beseitigen[8]. Die Äußerung der Gründe wiederum verpflichtet B allerdings nicht dazu, seine Meinung zu ändern. Wenn er A als in dieser Sache redlich einstuft (daß A ihn also nicht anlügt), dann kann er (in Abhängigkeit von seinem eigenen Wissen über die Sache und seinen eigenen Handlungszielen) z.B. wie folgt reagieren:

B:	auf (3a+b)	(4a) Das hast Du nur geträumt.
	auf (3c)	(4b) Mir hat diese Person aber $\neg$ p erzählt.
		usw.

Die Begründbarkeit spielt bei allen Handlungen, und somit auch in der Kommunikation eine wichtige Rolle. Sie ist die Basis für einen Abgleich unserer mentalen Modelle mit der Realität.

5 Was sind gute Gründe ?

Eberle hat neben den epistemischen Operatoren Glauben und Wissen die Relation $I_A(p,q)$ "aus p erschließt die Person A rational q" (Eberle 1974) eingeführt, u.a. um diese Frage zu beantworten. Diese Relation macht es möglich, einen Bezug zwischen Glauben und Wissen folgender Art herzustellen: Wenn p von A geglaubt wird, und A q aus p rational erschlossen hat, dann wird auch q von A geglaubt, aber nicht gewußt. Wird p hingegen gewußt, dann wird auch q von A gewußt werden. Diese Vorgehensweise ist sehr ähnlich derjenigen, die wir oben im Hinblick auf die Default-Logik angedeutet haben.

---

8 Einen besonderen Status haben die Elemente unseres Wissens, die nicht begründet werden können, die aber dennoch als gewußt eingestuft werden. "Bitte bleibe Zuhause, ich weiß, daß du heute einen Unfall haben wirst." Hier wird die Stärke der Ahnung als große Sicherheit begriffen und deshalb als Wissen artikuliert, das nicht rational begründet werden kann.

(11) Die Gründe werden mit den Einstellungen zu den in der Begründung verwendeten Propositionen entweder gut, wenn sie alle gewußt werden, oder weniger gut, wenn mindestens eine verwendete Proposition geglaubt wird.

Ein anderer Ansatzpunkt, eingeschränkt auf die Festlegung dessen, was gute Gründe für das Wissen über Handlungsziele, Pläne und Intentionen anderer Aktanten sind, scheint möglich, wenn wir die oben skizzierte Handlungstheorie ausnützen. Betrachten wir folgendes Beispiel:

(12) a. Harry weiß, daß Sally morgen zu ihm kommt.
b. Harry glaubt, daß Sally morgen zu ihm kommt.

Es ist in (12) die Rede von einer zukünftigen Handlung von Sally. Diese Handlung muß einerseits zu einem intentionalen Zustand von Sally in Beziehung stehen, und andererseits müssen die Vorbedingungen für die Handlung erfüllbar sein. Harry ist dann berechtigt, (12a) zu behaupten, wenn er gute Gründe dafür angeben kann, daß Sally morgen zu ihm kommt. Entscheidend scheint die Kenntnis der Intentionen von Sally zu sein. Wenn er von Sally selbst mitgeteilt bekommen hat, daß sie kommen will, bzw. wenn er bei einer solchen Mitteilung von Sally an einen Dritten zugegen war, dann ist Harry tatsächlich im Besitz der Kenntnis der Intention von Sally. Wenn die Intention jedoch erschlossen werden muß, dann ist es entscheidend, ob es ein rationaler[9] Schluß war, der die Intention von Sally aufgedeckt hat, oder ein abduktiver[10]. Wenn nämlich aus der Beobachtung von Einzelhandlungen auf den Plan, das Handlungsziel und den dazugehörigen intentionalen Zustand eines Aktanten geschlossen wird, dann sind dieses in der Regel abduktive Schlüsse.

> Wenn Harry Sally vor zwei Tagen gebeten hat, morgen zu kommen und Sally sich ihm gegenüber nicht festgelegt hatte, weil sie auch Lust hatte, am nächsten Tage nach Berlin zu fliegen und dort Fred zu besuchen, und wenn Harry von Fred erfahren hat, daß Sally in Berlin nicht aufgetaucht ist, dann kann er durchaus annehmen, daß Sally beabsichtigt, ihn zu besuchen.

Intention	→	Handlungsziel
Handlungsziel	→	Plan
Plan	→	Prüfen der Handlungsvoraussetzungen

Eine besonders wichtige Rolle spielen hierbei die Pläne, da diese die Umsetzung von Intention in (eine Sequenz von) Handlung(en) signalisieren. Da weder die Existenz einer Intention (die sich in dem gesamten Netz aller Intentionen behaupten muß), noch die Existenz eines Plans alleine die tatsächliche Ausführung einer Handlung determiniert und die Realisierungsabsicht nur abduktiv erschlossen werden kann, wird ein solcher abduktiver Schluß keinen guten Grund darstellen können, Wissen zu rechtfertigen.

(13) *Damit kann festgestellt werden, daß ein guter Grund für das Wissen über die Handlungen* eines anderen Aktanten dann vorliegt, wenn die Intention, der dazugehörige Plan und die

[9] Es ist die Frage, ob ein rationaler Schluß notwendigerweise ein deduktiver Schluß sein muß. Wir müssen uns auch fragen, ob eine Kausalbeziehung nicht per Induktion seinen Weg in ein Wissenssystem gefunden hat. Es könnte ja sein, daß morgen Montag ist und Sally immer montags zu Harry kommt.
[10] Ein abduktiver Schlu0ß liegt dann vor, wenn im Rahmen partieller Definitionen von der Konklusion auf die Prämissen geschlossen wird.

Realisierungsabsicht des Aktanten rational erschlossen oder auf anderem Wege festgestellt werden konnte, nicht aber durch Anwendung der Abduktion.

6 Das mutuelle Wissen

Ein ganz anderer Aspekt kommt im Hinblick auf die Unterscheidung zwischen Glauben und Wissen ins Spiel, wenn wir die Wissenszustände zweier Kommunikationspartner betrachten. Die Rolle der Partnermodellierung ist hinlänglich bekannt. Wichtigster Untersuchungsgegenstand ist das Wissen, das der eine Gesprächspartner über den anderen akkumulieren kann und bei der Planung seiner weiteren Handlungen (auch Sprechhandlungen) ausnutzt. Der etwas weitere Begriff des mutuellen Wissens umfaßt nun nicht nur das Wissen über den Gesprächspartner, sondern auch das, was gemeinsam gewußt wird, also nicht eingeschränkt auf einen Kommunikationspartner. Dieses mutuelle (von Sprecher und Hörer geteilte) Wissen (im Sinne von (5a)) ist von einer objektiven Warte aus leicht zu bestimmen: Es sind diejenigen Aussagen über die Welt, die in der Schnittmenge zweier Wissenssysteme liegen.

Da in einer Kommunikation weder der Sprecher noch der Hörer diese objektive Position einnehmen kann, kann hier nur die Rede sein von dem, was der Sprecher über das Wissen des Hörers vermutet (glaubt), bzw. welche seiner Wissenselemente er dem mutuellen Wissen und damit auch dem Wissen des Hörers zurechnet und welche nicht. Der Hörer vermutet Entsprechendes über das Wissen des Sprechers. Von diesem geglaubten Wissen ist das zu unterscheiden, was von beiden deshalb gewußt wird, weil es z.B. im aktuellen Dialog geäußert worden ist und auf der conversational record (Thomason 1990) vermerkt wurde. Da aber diese conversational records von den an einer Kommunikation Beteiligten geführt werden, haben die Einträge auf der record den Status des Gewußten, ebenso wie gewußt wird, daß ebendieses auch von dem Gesprächspartner gewußt wird. Die Unterscheidung in Wissen und Glauben hilft hier, Erlebtes von Vermutungen zu trennen.[11]

Eine mögliche Charakterisierung des Wissens eines Sprechers über einen Hörer wird mit (14) versucht[12]:

(14)

$$
\begin{aligned}
&\mathbf{MK(des\text{-}Sprechers,\ über\text{-}den\text{-}Hörer,\ ti)} = \\
&\{\, w \mid \quad \mathbf{K(Sprecher,\ w,\ ti)} \wedge \\
&\qquad \mathbf{B\ (Sprecher,\ K\ (Hörer,\ w,\ ti))} \wedge \\
&\qquad \mathbf{B\ (Sprecher,\ B\ (Hörer,\ K\ (Sprecher,\ w,\ ti))} \wedge \\
&\qquad \mathbf{B\ (Sprecher,\ B\ (Hörer,\ B\ (Sprecher,\ K\ (Hörer,\ w,\ ti)))}\,\}
\end{aligned}
$$

MK (mutuelles Wissen), ti (zu dem Zeitpunkt ti), K (Wissen), B (Glauben),

---

[11] Es ist von Bedeutung, welcher der epistemischen Operatoren **Glauben** (**B**) und **Wissen** (**K**) tatsächlich im Hinblick auf das Wissen über das Wissen des Gegenübers gerechtfertigt ist, wenn es z.B. um mögliche Reaktionen auf Präsuppositionsverletzungen geht, da die Frage zu entscheiden sein wird, welches Wissen dem Sprecher trotz der Präsuppositionsverletzung zugebilligt wird (da der Hörer weiß, daß der Sprecher es weiß), und über welches Wissen der Sprecher möglicherweise nicht verfügt. Die Reaktion des Hörers hängt durchaus davon ab, ob der Sprecher glaubt, daß der Hörer über ein bestimmtes Wissen verfügt, ob er es weiß, oder ob er nicht davon ausgeht.

[12] Vergleichbare Ansätze über mutual knowledge sind z.B. die Arbeiten von Clark und Carlson (1982), Sperber und Wilson (1982), Joshi (1982) in (Smith 1982).

w (Wissenselement).

Die Ereignisse / Zustände, die ich weiß und von denen ich glaube, daß mein Gegenüber sie weiß und von denen ich glaube, daß mein Gegenüber von ihnen glaubt, daß ich sie weiß, und von denen ich glaube, daß mein Gegenüber von ihnen glaubt, daß ich von ihnen glaube, daß er sie weiß.

Es ist (in der KI) üblich, spätestens hier abzubrechen und Axiome wie (15) zum Einsatz zu bringen, die in gewisser Weise aus dem unendlichen Regreß herausführen:

(15) "A weiß, daß A weiß, daß p" ist äquivalent zu "A weiß, daß p" (Hintikka 1962)

Von (14) zu trennen ist der Teil meines Wissens über die Ereignisse / Zustände, die ich weiß und von denen ich glaube, daß mein Gegenüber sie weiß und von denen ich glaube, daß mein Gegenüber von ihnen glaubt, daß ich sie **nicht** weiß (usw.):

(16) **MK**(des-**Sprechers**, über-den-**Hörer**, **ti**) =
{ w | **K (Sprecher , w , ti)** $\wedge$
B (Sprecher, K (Hörer , w , ti) $\wedge$
B (Sprecher, B (Hörer, $\neg$ **K (Sprecher , w , ti)))}**

7 Ausblick

Wenn es darum geht, eine umfassende Theorie des Verstehens und der Interpretation sprachlicher Äußerungen zu entwerfen, dann wird die hier behandelte Unterscheidung zwischen Glauben und Wissen im Rahmen einer Handlungstheorie ein zentrale Rolle spielen, weil (um mit Searle zu sprechen) sich der psychische Modus eines intentionalen Zustandes in der pragmatischen Realisierung einer sprachlichen Äußerung immer wiederfinden wird. Egal ob es darum geht, einen Sprechakt festzulegen, Präsuppositionen zu überprüfen, die Intention eines Sprechers zu erschließen, oder die eigene Reaktion zu planen, es wird immer entscheidend sein, was gewußt und was geglaubt wird. So muß z.B. die Entscheidung darüber, ob eine Aussage als Präsupposition mitgeteilt kohärenzstiftend sein soll, oder ob mit ihr (als rhetorischem Mittel) jemandem etwas auf eine Art mitgeteilt werden soll, die es schwer macht zu widersprechen, berücksichtigen, was jeweils gewußt bzw. geglaubt wird. Ohne eine fundierte Kenntnis dieses Unterschieds wird es nicht gelingen, zufriedenstellende (also einerseits adäquate und andererseits implementierbare) Theorien über die pragmatischen Aspekte der Sprachbeherrschung zu entwickeln.

Literatur

Clark, H. H.; Carlson, T. B. (1982): Speech Acts and Hearers' Beliefs. In (Smith 1982).

Eberle, R. A. (1974): A Logic of Believing, Knowing, and Inferring, Synthese 26, 356 - 382.

Hintikka, J. (1962): Knowledge and Belief, Ithaka.

Johnson-Laird, P. N. (1988): The Computer and the Mind. Fontana Press, London.

Joshi, A. K. (1982): Mutual Beliefs in Question Answer Systems. In: (Smith 1982)

Rollinger, C.-R. (1984): Die Repräsentation natürlichsprachlichen Wissens - Behandlung der Aspekte Unsicherheit und Satzverknüpfung. Dissertation an der TU-Berlin.

Schank, R., Abelson, R. (1977): Scripts, Plans, Goals, and Understanding. Hillsdale, New Jersey.

Searle, J. R. (1986): Geist, Hirn und Wissenschaft, suhrkamp taschenbuch wissenschaft stw 591, S. 56 - 70

Shafer, G. (1976); A Mathematical Theory of Evidence. Princeton, NJ, Princeton University Press.

Smith, N. V. (ed.) (1982): Mutual Knowledge. Academic Press.

Sperber, D.; Wilson, D. (1982): Mutual Knowledge and Relevance in Theories of Comprehension. In: (Smith 1982).

Thomason, R. H. (1990): Accomodation, Meaning, and Implicatur: Interdisciplinary Foundations for Pragmatics. In: Cohen, P. R.; Morgan, J.; Pollack, M. E. (eds.) Intentions in Communication. The MIT Press, Cambridge, London.

Zadeh, L. (1971): Quantitative Fuzzy Semantics". Information Sciences 3, 159-176.

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
67608 Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse oder per anonymem ftp von ftp.dfki.uni-kl.de (131.246.241.100) unter pub/Publications bezogen werden.

Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all published papers so far are obtainable from the above address or per anonymous ftp from ftp.dfki.uni-kl.de (131.246.241.100) under pub/Publications.

The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-92-48
Bernhard Nebel, Jana Koehler:
Plan Modifications versus Plan Generation: A Complexity-Theoretic Perspective
15 pages

RR-92-49
Christoph Klauck, Ralf Legleitner, Ansgar Bernardi:
Heuristic Classification for Automated CAPP
15 pages

RR-92-50
Stephan Busemann:
Generierung natürlicher Sprache
61 Seiten

RR-92-51
Hans-Jürgen Bürckert, Werner Nutt:
On Abduction and Answer Generation through Constrained Resolution
20 pages

RR-92-52
Mathias Bauer, Susanne Biundo, Dietmar Dengler, Jana Koehler, Gabriele Paul: PHI - A Logic-Based Tool for Intelligent Help Systems
14 pages

RR-92-53
Werner Stephan, Susanne Biundo:
A New Logical Framework for Deductive Planning
15 pages

RR-92-54
Harold Boley: A Direkt Semantic Characterization of RELFUN
30 pages

RR-92-55
John Nerbonne, Joachim Laubsch, Abdel Kader Diagne, Stephan Oepen: Natural Language Semantics and Compiler Technology
17 pages

RR-92-56
Armin Laux: Integrating a Modal Logic of Knowledge into Terminological Logics
34 pages

RR-92-58
Franz Baader, Bernhard Hollunder:
How to Prefer More Specific Defaults in Terminological Default Logic
31 pages

RR-92-59
Karl Schlechta and David Makinson: On Principles and Problems of Defeasible Inheritance
13 pages

RR-92-60
Karl Schlechta: Defaults, Preorder Semantics and Circumscription
19 pages

RR-93-02
Wolfgang Wahlster, Elisabeth André, Wolfgang Finkler, Hans-Jürgen Profitlich, Thomas Rist:
Plan-based Integration of Natural Language and Graphics Generation
50 pages

RR-93-03
Franz Baader, Berhard Hollunder, Bernhard Nebel, Hans-Jürgen Profitlich, Enrico Franconi:
An Empirical Analysis of Optimization Techniques for Terminological Representation Systems
28 pages

RR-93-04
Christoph Klauck, Johannes Schwagereit:
GGD: Graph Grammar Developer for features in CAD/CAM
13 pages

RR-93-05
Franz Baader, Klaus Schulz: Combination Techniques and Decision Problems for Disunification
29 pages

RR-93-06
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux: On Skolemization in Constrained Logics
40 pages

RR-93-07
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux: Concept Logics with Function Symbols
36 pages

RR-93-08
Harold Boley, Philipp Hanschke, Knut Hinkelmann, Manfred Meyer: COLAB: A Hybrid Knowledge Representation and Compilation Laboratory
64 pages

RR-93-09
Philipp Hanschke, Jörg Würtz:
Satisfiability of the Smallest Binary Program
8 Seiten

RR-93-10
Martin Buchheit, Francesco M. Donini, Andrea Schaerf: Decidable Reasoning in Terminological Knowledge Representation Systems
35 pages

RR-93-11
Bernhard Nebel, Hans-Juergen Buerckert:
Reasoning about Temporal Relations:
A Maximal Tractable Subclass of Allen's Interval Algebra
28 pages

RR-93-12
Pierre Sablayrolles: A Two-Level Semantics for French Expressions of Motion
51 pages

RR-93-13
Franz Baader, Karl Schlechta:
A Semantics for Open Normal Defaults via a Modified Preferential Approach
25 pages

RR-93-14
Joachim Niehren, Andreas Podelski, Ralf Treinen:
Equational and Membership Constraints for Infinite Trees
33 pages

RR-93-15
Frank Berger, Thomas Fehrle, Kristof Klöckner, Volker Schölles, Markus A. Thies, Wolfgang Wahlster: PLUS - Plan-based User Support Final Project Report
33 pages

RR-93-16
Gert Smolka, Martin Henz, Jörg Würtz: Object-Oriented Concurrent Constraint Programming in Oz
17 pages

RR-93-17
Rolf Backofen:
Regular Path Expressions in Feature Logic
37 pages

RR-93-18
Klaus Schild: Terminological Cycles and the Propositional μ-Calculus
32 pages

RR-93-20
Franz Baader, Bernhard Hollunder:
Embedding Defaults into Terminological Knowledge Representation Formalisms
34 pages

RR-93-22
Manfred Meyer, Jörg Müller:
Weak Looking-Ahead and its Application in Computer-Aided Process Planning
17 pages

RR-93-23
Andreas Dengel, Ottmar Lutzy:
Comparative Study of Connectionist Simulators
20 pages

RR-93-24
Rainer Hoch, Andreas Dengel:
Document Highlighting —
Message Classification in Printed Business Letters
17 pages

RR-93-25
Klaus Fischer, Norbert Kuhn: A DAI Approach to Modeling the Transportation Domain
93 pages

RR-93-26
Jörg P. Müller, Markus Pischel: The Agent Architecture InteRRaP: Concept and Application
99 pages

RR-93-27
Hans-Ulrich Krieger:
Derivation Without Lexical Rules
33 pages

RR-93-28
Hans-Ulrich Krieger, John Nerbonne, Hannes Pirker: Feature-Based Allomorphy
8 pages

RR-93-29
Armin Laux: Representing Belief in Multi-Agent Worlds viaTerminological Logics
35 pages

RR-93-30
Stephen P. Spackman, Elizabeth A. Hinkelman:
Corporate Agents
14 pages

RR-93-31
Elizabeth A. Hinkelman, Stephen P. Spackman:
Abductive Speech Act Recognition, Corporate Agents and the COSMA System
34 pages

RR-93-32
David R. Traum, Elizabeth A. Hinkelman:
Conversation Acts in Task-Oriented Spoken Dialogue
28 pages

RR-93-33
Bernhard Nebel, Jana Koehler:
Plan Reuse versus Plan Generation: A Theoretical and Empirical Analysis
33 pages

RR-93-34
Wolfgang Wahlster:
Verbmobil Translation of Face-To-Face Dialogs
10 pages

RR-93-35
Harold Boley, François Bry, Ulrich Geske (Eds.):
Neuere Entwicklungen der deklarativen KI-Programmierung — *Proceedings*
150 Seiten
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

RR-93-36
Michael M. Richter, Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Gabriele Schmidt: Von IDA bis IMCOD: Expertensysteme im CIM-Umfeld
13 Seiten

RR-93-38
Stephan Baumann: Document Recognition of Printed Scores and Transformation into MIDI
24 pages

RR-93-40
Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt, Andrea Schaerf:
Queries, Rules and Definitions as Epistemic Statements in Concept Languages
23 pages

RR-93-41
Winfried H. Graf: LAYLAB: A Constraint-Based Layout Manager for Multimedia Presentations
9 pages

RR-93-42
Hubert Comon, Ralf Treinen:
The First-Order Theory of Lexicographic Path Orderings is Undecidable
9 pages

RR-93-44
Martin Buchheit, Manfred A. Jeusfeld, Werner Nutt, Martin Staudt: Subsumption between Queries to Object-Oriented Databases
36 pages

RR-93-45
Rainer Hoch: On Virtual Partitioning of Large Dictionaries for Contextual Post-Processing to Improve Character Recognition
21 pages

RR-93-46
Philipp Hanschke: A Declarative Integration of Terminological, Constraint-based, Data-driven, and Goal-directed Reasoning
81 pages

DFKI Technical Memos

TM-92-01
Lijuan Zhang: Entwurf und Implementierung eines Compilers zur Transformation von Werkstückrepräsentationen
34 Seiten

TM-92-02
Achim Schupeta: Organizing Communication and Introspection in a Multi-Agent Blocksworld
32 pages

TM-92-03
Mona Singh:
A Cognitiv Analysis of Event Structure
21 pages

TM-92-04
Jürgen Müller, Jörg Müller, Markus Pischel, Ralf Scheidhauer:
On the Representation of Temporal Knowledge
61 pages

TM-92-05
Franz Schmalhofer, Christoph Globig, Jörg Thoben:
The refitting of plans by a human expert
10 pages

TM-92-06
Otto Kühn, Franz Schmalhofer: Hierarchical skeletal plan refinement: Task- and inference structures
14 pages

TM-92-08
Anne Kilger: Realization of Tree Adjoining Grammars with Unification
27 pages

TM-93-01
Otto Kühn, Andreas Birk: Reconstructive Integrated Explanation of Lathe Production Plans
20 pages

TM-93-02
Pierre Sablayrolles, Achim Schupeta:
Conlfict Resolving Negotiation for COoperative Schedule Management
21 pages

TM-93-03
Harold Boley, Ulrich Buhrmann, Christof Kremer:
Konzeption einer deklarativen Wissensbasis über recyclingrelevante Materialien
11 pages

TM-93-04
Hans-Günther Hein: Propagation Techniques in WAM-based Architectures — The FIDO-III Approach
105 pages

TM-93-05
Michael Sintek: Indexing PROLOG Procedures into DAGs by Heuristic Classification
64 pages

D-92-28
Klaus-Peter Gores, Rainer Bleisinger: Ein Modell zur Repräsentation von Nachrichtentypen
56 Seiten

D-93-01
Philipp Hanschke, Thom Frühwirth: Terminological Reasoning with Constraint Handling Rules
12 pages

D-93-02
Gabriele Schmidt, Frank Peters, Gernod Laufkötter: User Manual of COKAM+
23 pages

D-93-03
Stephan Busemann, Karin Harbusch(Eds.): DFKI Workshop on Natural Language Systems: Reusability and Modularity - Proceedings
74 pages

D-93-04
DFKI Wissenschaftlich-Technischer Jahresbericht 1992
194 Seiten

D-93-05
Elisabeth André, Winfried Graf, Jochen Heinsohn, Bernhard Nebel, Hans-Jürgen Profitlich, Thomas Rist, Wolfgang Wahlster:
PPP: Personalized Plan-Based Presenter
70 pages

D-93-06
Jürgen Müller (Hrsg.):
Beiträge zum Gründungsworkshop der Fachgruppe Verteilte Künstliche Intelligenz, Saarbrücken, 29. - 30. April 1993
235 Seiten
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-07
Klaus-Peter Gores, Rainer Bleisinger:
Ein erwartungsgesteuerter Koordinator zur partiellen Textanalyse
53 Seiten

D-93-08
Thomas Kieninger, Rainer Hoch:
Ein Generator mit Anfragesystem für strukturierte Wörterbücher zur Unterstützung von Texterkennung und Textanalyse
125 Seiten

D-93-09
Hans-Ulrich Krieger, Ulrich Schäfer:
TDL ExtraLight User's Guide
35 pages

D-93-10
Elizabeth Hinkelman, Markus Vonerden,Christoph Jung: Natural Language Software Registry
(Second Edition)
174 pages

D-93-11
Knut Hinkelmann, Armin Laux (Eds.):
DFKI Workshop on Knowledge Representation Techniques — Proceedings
88 pages

D-93-12
Harold Boley, Klaus Elsbernd, Michael Herfert, Michael Sintek, Werner Stein:
RELFUN Guide: Programming with Relations and Functions Made Easy
86 pages

D-93-14
Manfred Meyer (Ed.): Constraint Processing – Proceedings of the International Workshop at CSAM'93, July 20-21, 1993
264 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-15
Robert Laux: Untersuchung maschineller Lernverfahren und heuristischer Methoden im Hinblick auf deren Kombination zur Unterstützung eines Chart-Parsers
86 Seiten

D-93-16
Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Gabriele Schmidt: Design & KI
74 Seiten

D-93-20
Bernhard Herbig:
Eine homogene Implementierungsebene für einen hybriden Wissensrepräsentationsformalismus
97 Seiten

D-93-21
Dennis Drollinger:
Intelligentes Backtracking in Inferenzsystemen am Beispiel Terminologischer Logiken
53 Seiten

D-93-22
Andreas Abecker: Implementierung graphischer Benutzungsoberflächen mit Tcl/Tk und Common Lisp
44 Seiten

D-93-24
Brigitte Krenn, Martin Volk:
DiTo-Datenbank: Datendokumentation zu Funktionsverbgefügen und Relativsätzen
66 Seiten

D-93-25
Hans-Jürgen Bürckert, Werner Nutt (Eds.):
Modeling Epistemic Propositions
118 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-26
Frank Peters: Unterstützung des Experten bei der Formalisierung von Textwissen
INFOCOM - Eine interaktive Formalisierungskomponente
58 Seiten

Modeling Epistemic Propositions

Hans-Jürgen Bürckert, Werner Nutt (Eds.)

D-93-25
Document