CIRCUMSCRIPTION IN A MODAL LOGIC

Fangzhen Lin
Computer Science Department
Hua Chiao University

Computer Science Department
Stanford University

ABSTRACT

In this paper, we extend circumscription [McCarthy, 80] to a propositional modal logic of knowledge of one agent. Instead of circumscribing a predicate, we circumscribe the knowledge operator "K" in a formula. In order to have a nontrivial circumscription schema, we extend S5 modal logic of knowledge by adding another modality "Val" and a universal quantifier over base sentences (sentences which do not contain modality). Intuitively, "$Val(P)$" means that P is a valid formula. It turns out that by circumscribing the knowledge operator in a formula , we completely characterize the maximally ignorant models of the formula (models of the formula where agents have minimal knowledge).

1. Introduction

Recently, it has been made clear that reasoning about knowledge is not only an issue of concern to philosophy, but also an issue that has great importance to computer science and AI. More recently, it has been argued that logic of knowledge is also a suitable framework for formalizing common sense reasoning : much common sense reasoning can be formalized by minimizing agent's knowledge (cf. [Halpern and Moses, 84], [Levesque, 87], [Moore, 83], [Shoham, 86]).

In this paper we propose a way of capturing the notion of minimal knowledge, using a technique similar to that of circumscription [McCarthy, 80]. Briefly speaking, a circumscriptive axiom is a first-order schema or a second order sentence which minimizes the set of objects having property P (for simplicity, we assume P is a unary predicate). It is plausible, therefore, that there might be a similar schema in the logic of knowledge which will minimize the set of properties known by the agent.

The transition to a modal version of circumscription is not completely straightforward. We argue that in order to have a nontrivial circumscription schema in logic of knowledge, we must extend the logic of knowledge to a logic of validity with a universal quantifier over base sentences (sentences which do not contain the knowledge operator).

In this paper, we only consider the S5 propositional modal logic of knowledge of single agent. In section 2, we briefly review S5 modal logic of knowledge and introduce maximally ignorant models ([Halpern and Moses, 84], [Shoham, 86]) as a semantical formalization of minimized knowledge. We argue that in order to use circumscription to formalize reasoning in maximally ignorant models, we need to expand the logic of knowledge to a logic of validity with a universal quantifier over base sentences. This will be achieved in section 3 and section 4. In section 3, we introduce the validity modality "Val", intuitively, $Val(P)$ means that the proposition P is valid in the resulted modal logic itself. In particular, if P is a formula in the S5 logic of knowledge, $Val(P)$ is true iff P is a S5-valid formula. A simple semantics and a complete axiom system is proposed for the logic of validity. And in section 4, the logic of validity is further extended by adding a universal quantifier over base sentences.

Finally, in section 5, we are ready to have a circumscription schema in logic of knowledge. First we propose a simple circumscription schema which successfully minimizes an agent's knowledge in many cases. But there are some cases that the simple circumscription schema is not enough and so we extend it to a circumscription schema which allows primitive propositions to vary. It turns out that this circumscription schema completely characterizes reasoning in maximally ignorant models in every case.

2. Maximally Ignorant Models

In this secion we briefly review the one agent S5 modal logic of knowledge, and present the notion of maximally ignorant models in the logic, which has appeared, explicitly or implicitly, in the work of Halpern and Moses (84), Shoham (86), among others.

The primitive symbols of the S5 modal logic of knowledge include a sequence of propositional symbols: $p, q, p_1, q_1, \ldots$; two connectives: $\neg$, $\wedge$; and one modal operator: the knowledge operator K. Formulas are defined conventionally. Intuitively, if P is a formula, KP means that the agent knows that P is true (we assume that there is only one agent in the logic). In this paper, formulas which do not contain the knowledge operator K will play an important role, we will call them base formulas ([Shoham, 86]).

The semantics of the logic are conventional possible worlds semantics. A *valuation* v is a mapping from the set of primitive propositions to the set $\{0, 1\}$ of truth values. An *intepretation* M is a set of valuations. A *model* is a pair (M, v) such that M is an interpretation and v is a valuation in M. For any model (M, v) any formula P, the notion P being satisfied in (M, v), or (M, v) being a model of P, written $(M, v) \models P$, is defined inductively as follows:

1. $(M, v) \models p$ iff $v(p) = 1$, for any primitive proposition p.
2. $(M, v) \models P \wedge Q$ iff $(M, v) \models P$ and $(M, v) \models Q$
3. $(M, v) \models \neg P$ iff $(M, v) \models P$ is not true
4. $(M, v) \models KP$ iff $(M, v') \models P$ for any v' in M

Some familiar semantic notions are readily definable. For example, a formula P is *satisfiable* if there is a model (M, v) such that $(M, v) \models P$; a formula P is *valid* (written $\models P$), if P is satisfied in all models; and a set of formulas S *entails* a formula P (written $S \models P$), if the formula P is true in all models (M, v) such that (M, v) is a model of all members of S.

We now define the notion of maximally ignorant models. Intuitively, a maximally ignorant model of a formula P is one which satisfies P and where the agent knows as few facts as possible. Thus the key is how we compare two models M_1 and M_2 so that we can say , for example, that in M_1 the agent has more knowledge than he has in M_2.

Definition 1. *Let M and M' be two interpretations. $M \prec M'$, that is, M' is more ignorant than M iff for any base formula P, if $M' \models KP$ then $M \models KP$, and there is a*

base formula P such that $M \models KP$ but $M' \models KP$ is not true.

A model (M, v) is a *maximally ignorant model* of a formula P if $(M, v) \models P$ and there are no models (M', v') such that $(M', v') \models P$ and $M \prec M'$.

The notion 'maximally ignorant models' is due to Shoham. In fact, in [Shoham, 86], Shoham defined the concept of 'chronologically ignorant models' in a modal temporal logic of knowledge. Essentially the same definition of maximally ignorant models appeared also in [Halpern and Moses, 84].

Example 1. Consider the formulas Kp, $Kp \vee Kq$ and $p \vee Kq$. Readers familiar with the results in [Halpern& Moses,84] can easily see that the maximally ignorant models of Kp are the knowledge states where the agent knows only that p is true. The maximally ignorant models of $Kp \vee Kq$ are either the knowledge states where the agent knows only that p is true or the knowledge states where the agent knows only that q is true. Finally, the maximally ignorant models of $p \vee Kq$ are the knowledge states where p is true and the agent knows nothing.

In general, the maximally ignorant models of P correspond to the knowledge states where the agent's knowledge is determined solely by P. This is a more general notion than 'knowing only'. For example, according to [Halpern and Moses, 84], it is meaningless to say that the agent only knows that $Kp \vee Kq$ is true, since in the logic, our agent has perfect knowledge about himself, he knows whether he knows p is true, similarly, he knows whether he knows q is true. But it does make sense to say that the agent's knowledge is determined solely by $Kp \vee Kq$: what this amounts to is that either the agent knows only that p is true or he konws only that q is true. This means that *we* do not have a complete knowledge about our agent . This kind of situation often arise in AI applications. For example, when we formalize in our logic (appropriately expanded to the many-agents case) the so called 'wise men' or 'cheating husbands' puzzle ([Moses, Dolev and Halpern, 86]), we need such 'indeterminate knowing only'.

Finally, we come to the main aim of this paper: how to capture reasoning in maximally ignorant models by an axiom schema like circumscription. First we briefly recall the definition of McCarthy's circumscription in first-order logic with *equality*.

Suppose A is a first-order sentence and $P(x)$ is a unary predicate in A. According to [McCarthy, 80], the circumscription of P in A is the following schema:

$$A \wedge (A(B) \rightarrow \neg P < B)$$

where $\neg P < B$ is

$$\forall x(B(x) \rightarrow P(x)) \rightarrow \forall x(P(x) \rightarrow B(x))$$

$B(x)$ is any first-order formula with one free variable, and $A(B)$ is the result of replacing every $P(t)$ in A by $B(t)$. Semantically, circumscribing P in A means we minimize the set of objects having the property P without violating the condition A. For example, suppose A is $block(a)$ and $P(x)$ is $block(x)$, denote the circumscription of P in A by $C(A;P)$, it is easy to see that

$$block(a) \wedge C(A;P) \vdash \forall x(block(x) \leftrightarrow x = a)$$

so we have

$$a \neq b \wedge block(a) \wedge C(A;P) \vdash \neg block(b)$$

that is, after circumscribing *block* in $block(a)$, we minimize the predicate *block* so that for any object x, x is a block iff x is a. Note that this effect can only be achieved and represented in the presence of the special "=" predicate.

Thus it seems reasonable that we can have a similar schema to minimize the set of base formulas having the property K, that is, minimize the agent's knowledge. For example, suppose the agent knows that p is true, that is Kp is true, we would like to expect that the following schema will minimize the agent's knowledge without violating the condition that Kp is true:

$$Kp \wedge (Tp \rightarrow \neg T < K)$$

where $\neg T < K$ is

$$\forall X(TX \rightarrow KX) \rightarrow \forall X(KX \rightarrow TX)$$

and T is something behaves like K and $\forall X$ means 'for all base sentences X'. Unfortunately, we can not go very far with this schema in S5 logic, for example, even the desired result: KX is true iff $Kp \vdash KX$ (compared with the desired result: $block(x)$ is true iff $x = a$ in the above ordinary circumscription) can not be represented in S5 modal logic of knowledge. What we need is a way of representing the meta-logical concept 'P is a logical consequence of Q' in the object language.

To sum things up, we still need two things in order to minimize knowledge using circumscription :

(1) A way of representing the statement: 'P is a logical consequence of Q. This will be done in Section 3.

(2) A universal quantifier over base sentences. This will be done in Section 4.

3. Formalizing Validity

In this section, we extend S5 modal logic of knowledge to a logic of knowledge and validity by introducing a new modal operator "Val." Let us call the new logic $\mathcal{L}$.

$\mathcal{L}$-Formulas are the standard S5 formulas augmented by another unary modality Val. Intuitively, if P is a formula $Val(P)$* means that P is valid. Interpretations and models in $\mathcal{L}$ are the same as those in S5. Semantics for formulas of the form $Val(P)$ are defined as follows:

5. $(M, v) \models Val(P)$ iff $(M', v') \models P$ for any model (M', v')

The rest of this section is devoted to study some properties of Val. In the remainder of this paper, we write a vector of primitive propositions as $\vec{p}, \vec{q}, \ldots$. If $\vec{p} = (p_1, p_2, \ldots, p_n)$, we call n the *dimension* of the vector $\vec{p}$. Similarly, we write vectors of formulas as $\vec{P}, \vec{Q}, \ldots$. Let P be a formula, $\vec{p} = (p_1, p_2, \ldots, p_n)$ be a vector of primitive propositions, $\vec{Q} = (Q_1, Q_2, \ldots, Q_n)$ be a vector of formulas having the same dimension as that of $\vec{p}$, and $S = (\vec{Q}, \vec{Q}_1, \ldots, \vec{Q}_m)$ be a set of vectors containing $\vec{Q}$. The formula $P(\vec{p}/(\vec{Q}, S))$ is defined inductively as follows:

1. $q(\vec{p}/(\vec{Q}, S)) = q$, if q is a primitive proposition different from p_i for any $1 \leq i \leq n$.

2. $p_i(\vec{p}/(\vec{Q}, S)) = Q_i$, $1 \leq i \leq n$.

3. $(\neg P)(\vec{p}/(\vec{Q}, S)) = \neg(P(\vec{p}/(\vec{Q}, S)))$; $(P \wedge P')(\vec{p}/(\vec{Q}, S)) = P(\vec{p}/(\vec{Q}, S)) \wedge P'(\vec{p}/(\vec{Q}, S))$.

4. $KP(\vec{p}/(\vec{Q}, S)) = P(\vec{p}/(\vec{Q}, S)) \wedge P(\vec{p}/(\vec{Q}_1, S)) \wedge \ldots \wedge P(\vec{p}/(\vec{Q}_m, S))$

5. $Val(P)(\vec{p}/(\vec{Q}, S)) = Val(P)$*.

Intuitively, we can consider $\vec{p}/\vec{Q}$ as the partial valuation, or the situation ([Barwise and Perry, 84]), such that for any p_i of $\vec{p}$, we assign the corresponding Q_i of $\vec{Q}$ to p_i. Thus $\vec{p}/(\vec{Q}, S)$ behaves like the partial model: $(\{\vec{p}/\vec{Q}, \vec{p}/\vec{Q}_1, \ldots, \vec{p}/\vec{Q}_m\}, \vec{p}/\vec{Q})$ and $P(\vec{p}/(\vec{Q}, S))$ is the 'truth value' of P in the partial model $\vec{p}/(\vec{Q}, S)$.

---

* We used "$Prov$" instead of "Val" in the former manuscript and we understood $Prov(P)$ as 'P is provable'. Yoav Shoham persuaded us to use the new terminology which fits the following semantical definition better.

* the requirment of the substitution outside the scope of "Val" is due to Joe Halpern. The restriction is necessary for Theorem 1 to be true.

Example 2. Denote the valid formula $p \vee \neg p$ as t, and denote the false formula $p \wedge \neg p$ as f, let $p = \vec{p} = (p)$, $t = \vec{t} = (t)$, $f = \vec{f} = (f)$ and $S = \{f, t\}$, then

$$p(p/(t,S)) = t, \quad Kp(p/(t,S)) = t \wedge f, \quad Val(p)(p/(t,S)) = Val(p)$$

Recall that $Val(P)$ means that P is true in all models and that $P(\vec{p}/(\vec{Q},S))$ is the truth value of P in the partial model $\vec{p}/(\vec{Q},S)$, it is easy to understand the following central theorem about Val:

Theorem 1. *For any formula P, any vector of primitive propositions $\vec{p} = (p_1, \ldots, p_n)$, any vector of formulas $\vec{Q} = (Q_1, \ldots, Q_n)$ and any set of vectors of the same dimension $S = \{\vec{Q}, \vec{Q}_1, \ldots \vec{Q}_m\}$, we have*

$$\models Val(P) \rightarrow Val(P(\vec{p}/(\vec{Q},S)))$$

The theorem is proved by that given any model (M,v), we can always construct a model (M',v') such that $(M,v) \models P(\vec{p}/(\vec{Q},S))$ iff $(M',v') \models P$.

Theorem 1 expresses an important property of "Val". It enables us to prove the invalidity of some propositions. In fact, we can think of Theorem 1 as formalizing the process of constructing counter-examples. For example, it is easy to deduce $\models \neg Val(p)$ from $\models \neg Val(f)$ using the theorem by constructing the counter-model $p/(f,\{f\})$ for p (for detail see example 3 and example 4). Furthermore we can have a complete axiom system based on Theorem 1.

The Axiom System $\mathcal{KP}$

Axioms

(1) Propositional tautologies in the language of $\mathcal{L}$.

(2) $(KP \rightarrow P) \wedge (Val(P) \rightarrow P)$

(3) $(K(P \rightarrow Q) \rightarrow (KP \rightarrow KQ)) \wedge (Val(P \rightarrow Q) \rightarrow (Val(P) \rightarrow Val(Q)))$

(4) $(KP \rightarrow KKP) \wedge (Val(P) \rightarrow Val(Val(P)))$

(5) $(\neg KP \rightarrow K(\neg KP)) \wedge (\neg Val(P) \rightarrow Val(\neg Val(P)))$

(6) $Val(P) \rightarrow Val(P(\vec{p}/(\vec{Q},S)))$, where $\vec{p}$ is any vector of primitive propositions, $\vec{Q}$ is

any vector of formulas, S is any finite set of vectors of formulas containing $\vec{Q}$ and all the vectors $\vec{p}, \vec{Q}$ and vectors in S have the same dimension.

(7) $Val(P) \rightarrow KVal(P)$

(8) $\neg Val(P) \rightarrow K\neg Val(P)$

Rules of Inferences

[MP] if P and $P \rightarrow Q$, then Q;

[GK] if P, then KP;

[GV] if P, then $Val(P)$.

As usual, if P is a theorem of the axiom system $\mathcal{KP}$, then we write it as $\vdash P$, and we write $S \vdash P$ if there is a proof of the formula P from the set S of formulas in $\mathcal{KP}$.

Theorem 2. *(soundness and completeness) For any formula P and any finite set S of formulas*

$$S \vdash P \iff S \models P$$

The proof of the theorem 2 is a little tedious. The proof is given in the full paper.

As you can see in the axiom system $\mathcal{KP}$, the validity orerator "Val" behaves like the knowledge operator "K" except it satisfies the axiom (6). In fact, it is the axiom (6) makes our axiom system $\mathcal{KP}$ uniquer and distinguishes our logic of validity from Boolos' logic of provability [Boolos, 79]. In order to fully appreciate the axiom (6), it is best to see some examples. In the following, we identify a one-dimensional vector with its element, for example, P with (P).

Example 3 $\vdash \neg Val(p)$. It is proved by constructing the counter-model $p/(f, \{f\})$ for p:

1. $\vdash Val(p) \rightarrow Val(p(p/(f, \{f\})))$ axiom (6)

2. $\vdash Val(p) \rightarrow Val(f)$ from 1.

3. $\vdash Val(f) \rightarrow f$ axiom (2)

Example 4 $\vdash \neg Val(Kp \vee \neg p)$. The proof is as similar to that of Example 3 by constructing the counter-model $p/(t, \{t, f\})$ for $Kp \vee \neg p$.

4. Propositional Quantifiers

In this section we briefly describe how to extend the language of $\mathcal{L}$ to include a propositional quantifier "$\forall X$". Intuitively, "$(\forall X)P$" means that "for all base formulas X, the formula P is true."

Let us call the new logic $\mathcal{L}'$. In addition to the primitive symbols of $\mathcal{L}$, $\mathcal{L}'$ has the following primitive symbols: a sequence of (base formula) variables: $X, Y, X_1, Y_1, \ldots$; and a logical quantifier: $\forall$. The well-formed formulas of $\mathcal{L}'$ are defined as follows. In addition to the formation rules of $\mathcal{L}$, we have: every variable is a formula; and if P is a formula, X is any variable, then $(\forall X)P$ is also a formula. So p, KX, $\forall X(Val(X) \rightarrow X)$ are formulas of $\mathcal{L}'$.

The semantics of $\mathcal{L}'$ are defined as follows. A valuation v of $\mathcal{L}'$ is a mapping which not only assigns a truth value to every primitive proposition but also assigns a base formula to every variable. Again, an interpretation is a set of valuations and a model is a pair (M, v) where M is an interpretation and v is a valuation in M. For the definition of $(M, v) \models P$, in addition to those in $\mathcal{L}$, we have:

6. $(M, v) \models X$ iff $(M, v) \models v(X)$, where X is any variable in $\mathcal{L}'$

7. $(M, v) \models \forall XP$ iff $(M, v) \models P(X/Q)$ for any base formula Q, where $P(X/Q)$ is the result of substituting every free occurence of X in P by Q.

The axiom system $\mathcal{KP}$ is extended to include the following axioms and the rules of inference

Additional Axioms For $\mathcal{L}'$

(9) $\forall XP \rightarrow P(X/Q)$, where Q is any base formulas

(10) $\forall XKP \leftrightarrow K(\forall XP)$

(11) $\forall XVal(P) \leftrightarrow Val(\forall XP)$

(12) $\forall X(P \rightarrow Q) \rightarrow (\forall XP \rightarrow \forall XQ)$

Additional Rule of Inference For $\mathcal{L}'$

[G] if P, then $\forall X P$

5. Circumscription In The Modal Logic $\mathcal{L}$

Now we have enough machinery to express circumscription in the modal logic $\mathcal{L}$. As we have said, the key idea is that instead of circumscribing a predicate in first-order logic, we circumscribing the knowledge operator in $\mathcal{L}$. Thus the first question will be what expressions can we use to replace the knowledge operator K in a formula.

In the following, we call a formula T in $\mathcal{L}'$ an *X-formula*, if T contains neither occurences of the propositional quantifier "$\forall$" nor the variables other than X. For example, $Val(X) \rightarrow X$ is a formula with the variable X but neither $Val(X) \rightarrow Y$ nor $\forall X(Val(X) \rightarrow X)$ are formulas with the variable X.

Definition 2. *Suppose $T(X)$ is an X-formula, T is called a knowledge expression if*

1. $\models \forall X \forall Y(T(X) \wedge Val(X \rightarrow Y) \rightarrow T(X/Y))$

2. $\models \forall X \forall Y(T(X) \wedge T(X/Y) \rightarrow T(X/X \wedge Y))$

Intuitively, $T(X)$ is a knowledge expression if $T(X)$ behaves like KX. For example, the formula $Val(X)$ is a knowledge expression for it is easy to see that $\models Val(X) \wedge Val(X \rightarrow Y) \rightarrow Val(Y)$, and $\models Val(X) \wedge Val(Y) \rightarrow Val(X \wedge Y)$. In fact, we can prove the following proposition:

Proposition 2. *If P is a base formula, then the formula $Val(P \rightarrow X)$ is a knowledge expression.*

As we shall see in the following, knowlege expressions of the form $Val(P \rightarrow X)$ are enough for our purpose.

We now minimize knowledge using circumscription. We begin with a definition which comes close to achieve our goal. In the following, if P is a formula, T is a knowledge expression, then we use $P(K/T)$ to denote the result of substituting any KQ which is not in the scope of Val in P by $T(Q)$.

Definition 3. *If P is a formula in $\mathcal{L}$ such that there is no nesting of the knowledge operator "K" in P, then the circumscription of the knowledge operator K in the formula*

P is the following schema:

$$P \wedge (P(K/T) \rightarrow \neg T < K)$$

where $\neg T < K$ *is*

$$\forall X(T(X) \rightarrow KX) \rightarrow \forall X(KX \rightarrow T(X)))$$

and $T(X)$ *is any knowledge expession.*

Let us denote the above schema by $C(P)$. As we can see in the following examples, in many cases, $C(P)$ characterizes the maximally ignorant models of P.

Example 5. Consider the formula Kp, and the knowledge expression $T(X) = Val(p \rightarrow X)$, from $\models Kp \rightarrow (Val(p \rightarrow p) \wedge \forall X(Val(p \rightarrow X) \rightarrow KX)$ and $Kp(K/T) = Val(p \rightarrow p)$, by the above definition of $C(Kp)$, we have

$$C(Kp) \models \forall X(KX \leftrightarrow Val(p \rightarrow X))$$

Therefore,

$$C(Kp) \models K(p \vee q) \wedge \neg Kq \wedge K(\neg Kq) \wedge \ldots$$

As we can see in Example 2, this is exactly what we need: in maximally ignorant models, for any base formula B, the agent knows that B is true iff B is a logical consequence of p. Intuitively, choosing the knowedge expression T in $C(Kp)$ as $Val(p \rightarrow X)$ means that we are trying to construct a model of Kp where for any base formula Q, KQ is true iff $Val(p \rightarrow Q)$ is true, that is, he knows only that p is true.

Example 6. Consider the formula $Kp \vee Kq$. Let $T(X)$ in $C(Kp \vee Kq)$ be $Val(p \rightarrow X)$, we have

$$C(Kp \vee Kq) \models Kp \rightarrow \forall X(KX \rightarrow Val(p \rightarrow X))$$

Similarly

$$C(Kp \vee Kq) \models Kq \rightarrow \forall X(KX \rightarrow Val(q \rightarrow X))$$

So

$$C(Kp \vee Kq) \models \forall X(KX \rightarrow Val(p \rightarrow X)) \vee \forall X(KX \rightarrow Val(q \rightarrow X))$$

Again, this is exactly what we need: a maximally ignorant model of $Kp \vee Kq$ is one where either the agent knows only that p is true, or the agent knows only that q is true (see

Example 2) and the facts known by the agent in both models are exactly characterized by $C(Kp \vee Kq)$. Unfortunately, there are some cases that the circumscription defined in Definition 3 is not strong enough.

Example 7. Consider the formula $p \vee Kq$. The maximally ignorant models of $p \vee Kq$ are those where p is true and the agent knows nothing, so it seems reasonable to choose the knowledge expression in $C(p \vee Kq)$ to be $T(X) = p \wedge Val(X)$. Unfortunately, we only get:

$$C(p \vee Kq) \models p \rightarrow \forall X(KX \rightarrow Val(X))$$

The reason is that although by choosing $T(X)$ to be $p \wedge Val(X)$ we expect to have a model where in the actual world p is true and the interpretation is derermined by $\forall X(KX \leftrightarrow Val(X))$, the axiom schema in Definition 3 only enables us to pin down the interpretation, we have no way to say anything about actual worlds. For example, if we let $T(X)$ be $Val(X)$ instead of being $p \wedge Val(X)$, then we will get the same result. Choosing an actual world means that we assign true or false to primitive propositions. Generally, this means we substitute primitive propositions by formulas (and this yields relative partial valuations).

Briefly speaking, a model is a pair (KI, v), where KI is an interpretation and v is a valuation in KI. So far Definition 3 only enables us to choose an interpretation, we also need to choose a valuation as the actual world, this leads to the following definitons.

Definition 4. *Suppose $T(X)$ is a knowledge expreesion, $\vec{p}$ is a vector of primitive propositions and $\vec{Q}$ is a vector of base formulas which has the same dimension as that of $\vec{p}$, we say that T is consistent with the partial valuation $\vec{p}/\vec{Q}$ if for any base formula Q,*

$$T(Q) \models T(Q(\vec{p}/\vec{Q}))$$

where $Q(\vec{p}/\vec{Q})$ is the result of replacing every p_i of $\vec{p}$ in Q by the corresponding Q_i of $\vec{Q}$.

By Theorem 1, we have

Proposition 3. *For any $\vec{p}$ and $\vec{Q}$, $Val(X)$ is a knowledge expression consistent with $\vec{p}/\vec{Q}$.*

Definition 5. *If P is a formula in $\mathcal{L}$ such that there is no nesting of the knowledge operator "K" in P, then the circumscription of the knowledge operator K in the formula P with the truth values of the primitive propositions allowd to vary is the following schema:*

$$P \wedge (P(\vec{p}/\vec{Q})(K/T) \rightarrow \neg T < K)$$

where $T(X)$ is any knowledge expression which is consistent with $\vec{p}/\vec{Q}$, $\vec{p}$ is any vector of primitive propositions , $\vec{Q}$ is any vector of base formulas which has the same dimension as that of $\vec{p}$, and $P(\vec{p}/\vec{Q})(K/T)$ is $(P(\vec{p}/\vec{Q}))(K/T)$.

Let us still write the above schema as $C(P)$. Intuitively, the above schema means that if a model M satisfies the schema, then the "model" $(T(X), \vec{p}/\vec{Q})$ can not be more ignorant than M. Note that Definition 5 contains Definition 3 as a special case because any knowledge expression is consistent with $\vec{p}/\vec{p}$ and $P(\vec{p}/\vec{p}) = P$. Also Note that the requirement that P ccntains no nesting of K does not restrict the usefulness of Definition 3, because in $\mathcal{L}$, every formula is logically equivalent to a formula which contains no nesting of "K".

Now let us continue Example 7 with the new definition of $C(P)$.

Example 7 (Continued). Consider the formula $p \vee Kq$. Let $T(X)$ in $C(p \vee Kq)$ be $Val(X)$. Let $\vec{p}$ and $\vec{Q}$ be p and t, respectively (by Proposotion 3, $Val(X)$ is a knowledge expression which is consistent with p/t). We have

$$C(p \vee Kq) \models \forall X(KX \rightarrow Val(X)) \wedge (p \vee Kq)$$

So

$$C(p \vee Kq) \models \forall X(KX \leftrightarrow Val(X)) \wedge p$$

exactly what we desired.

Our main result of this paper is that for any formula P, the maximally ignorant models of P are completely characterized by $C(P)$ as defined in Definition 5.

Theorem 3. *Let P be a formula in $\mathcal{L}$ such that P contains no nesting of the orperator "K". If (M, v) is a maximally ignorant model of P, then*

$$(M, v) \models C(P)$$

Theorem 4. *Let P be a formula in $\mathcal{L}$ which contains no nesting of the knowledge operator "K". For any formula Q, if Q is true in all maximally ignorant models of P, then $C(P) \models Q$.*

In order to prove this theorem, we first prove that for any formula P and P', if P and P' are logically equivalent, then $C(P)$ and $C(P')$ are also logically equivalent. Then we

transform P into the following logically equivalent formula

$$(P_1' \wedge KP_1 \wedge \neg KP_{11} \wedge \ldots \wedge \neg KP_{1m}) \vee \ldots \vee (P_n' \wedge KP_n \wedge \neg KP_{n1} \wedge \ldots \wedge \neg KP_{nk})$$

where $P_1', P_1, P_{11}, \ldots, P_{1m}, \ldots, P_n', P_n, P_{n1}, \ldots, P_{mk}$ are basic formulas and every disjunct is not trivial, that is, not equivalent to the false "f". For formulas of the above form, we use the same techniques as those in Example 5 and Example 6 to get the result of circumscription.

6. Concluding Remarks

We think that the most important thing we would like to convey to you in this paper is that we not only can circumscribe a predicate in the first-order logic as that in [McCarthy, 80], we can also circumscribe a modal operator in a modal logic. In particular, in this paper, we have shown that the circumscription of the knowledge operator in a formula is nontrivial and interesting, it includes the well-known notion of "knowing only" as a special case.

Acknowledgements

I would like to thank Yoav Shoham for encouragement, helpful discussions, and comments on both the technical issues and English. I also thank Joe Halpern for helpful comments, John McCarthy for kindly letting me use the Sail system at Stanford. Most importantly, I am grateful for the friendships of Lois Dewart and her family, without their help, many things would turn out to be really difficult to me. Thanks.

References

[1] Barwise, J. and J. Perry (83), *Situations and Attitudes,* Cambridge, MA:MIT Press; 1983

[2] Boolos, G. (79), *The Unprovability of Consistency: An Essay In Modal Logic,* Cambridge University Press,1979.

[3] Halpern, J. and Y. Moses (84), *Towards a Theory of Knowledge and Ignorance: Preliminary Report,* IBM technical Report RJ 4448 (48136), 1984.

[4] Konolige, K. (82), Circumscriptive Ignorance, *Conference Proceedings of AAAI-82,* 202–204.

[5] Levesque, H. (87), All I Know: An Abridged report, *Conference Proceedings of AAAI-87, 426–431.*

[6] McCarthy, J. (80), Circumscription — A Form of Non-Monotonic Reasoning, *Artificial Intelligence* 13(1980) 27–39.

[7] Moore, R. (83), Semantical considerations on Nonmonotonic Logic, *Conference Proceedings of IJCAI-83.*

[8] Moses, Y, D. Dolev and J. Halpern (86), Cheating husbands and other stories: a case study of knowledge, action, and communication, *Distributed Computing,* 1:3, 1986, pp. 167–176.

[9] Shoham, Y. (86), *Reasoning About Change: Time and Causation from the Standpoint of Artificial Intelligence*, Ph.D. Thesis, Computer Science Department, Yale University, 1986.