Intensional Contexts, Common Knowledge and Meta-Logic Programming*

Hiroto OHNISHI

We propose a significant extension of the indexing procedure, which Jiang [12] used in his epistemic logic programming, to treat recalcitrant data of what has been called common knowledge. Combining the extended indexing procedure with meta-level approach to modal logic, we gain a system that covers philosophically and linguistically much debated data with flexibility and computational efficiency. The analysis proposed here can fully be implemented within the framework of meta-logic programming.

1. INTRODUCTION

Knowledge representation is one of much debated fields that need computational methods of logic programming and logically oriented natural language semantics. This is solely because we need to step into the domain of human minds. Taking into consideration that each person (agent) has his own cognitive world, the complexity of a system radically increases since we ordinarily commit ourselves to only one world: reality. We should be able to describe each person's view about the world, but that is not enough. We all have our specific view on the world, but this does not mean we are totally isolated. We sometimes share with others pieces of knowledge through communication, without which we could not live along even a single day. Therefore a desirable logical system of knowledge should be able to treat each person's specific view on the one hand, and points where two or more persons' view coincide on the other. The latter has been called the problem of common knowledge (shared or mutual knowledge), which is the main topic here.

Recently we see a number of extensions of logic programming in the Horn clause theory. One of such extensions is modal logic programming, which seems to fill our purpose here since modal logic is the logic that can formalize various intensional concepts such as knowledge and belief. We find several approach on this line in the literature, e.g., Akama [1], Fariñas del Cerro [9], Jiang [12]. Especially Akama & Ohnishi [3] proposed a system called epistemic knowledge base to handle epistemic reasoning in logic data base, which, in the meta-level, simulates the logic of knowledge, i.e., modal logic S5 (with its negation interpreted as negation as failure, theoretical implication of which will be made clear in the sequel), and the meta-level is combined with the object level through the amalgamation technique due to Bowen &

Kowalski [7] [1] The potential advantage in adopting such meta-programming approach is that it can easily be extended to many other types of epistemic logic programming if we change the logic governing the meta-level system; to cite just a few examples, if we are to develop a system of belief, all we need to do is to change the logic governing the meta-level into S4 (roughly, see Hintikka [11]), and for generics S 4.3 will be a possible candidate. See Akama [1] and Fariñas del Cerro [9] for more on this topic

There we also developed a similar indexing procedure originally due to Jiang [12] to inject agents' view about the world into our system: to explicitly show on whose knowledge the interpretation of a given expression depends. The procedure is rooted in the idea that when several agents are involved it must be necessary to anchor the referent of a given expression onto an agent's world, since its referent may vary agent by agent. Here we propose an extended version of the indexing procedure which can deal with pieces of knowledge that agents involved have in common We also show how our extended indexing procedure can be embedded into the framework of logic programming

In section 2, we review Jiang's [12] indexing procedure and show, if interpreted properly, this procedure per se is of great help to solve various philosophical/linguistic problems. In section 3, we proceed to more complicated cases where two or more agents are involved (common knowledge problems) Our new version of indexing procedure will enable us to treat the famous Hob-Nob problems and other issues. The formal semantics and meta-programming technique essential to this proposal, will be presented later in section 4 and 5. Section 6 concludes this paper

2. JIANG'S EPISTEMIC LOGIC PROGRAMMING AND SOME PROBLEMS IN KNOWLEDGE REPRESENTATION

In this section we first survey Jiang's indexing procedure and its contribution to the field of logic programming. Second we discuss how this procedure can be applied to solve much debated problems in the field of philosophy as well as linguistics.

Jiang [12] proposed epistemic logic programming based on the epistemic SLD-resolution, which guarantees its efficiency, as an extension of ordinary SLD-resolution In addition, epistemic logic programming can represent non-monotonic reasoning in the framework of logic programming although we do not go into details here (see Moore [18] on this point) One of the difficulties with epistemic reasoning lies in the unification mechanism since the classical unification method cannot be applied in a modal context His analysis of modal contexts can be best surveyed by the following example:

(1) $\mathrm{Black}\ (\mathrm{mayor}\ (\mathrm{Tokyo}))$

$\neg \mathbb{B}\ (\mathrm{Simon},\ \mathrm{Black}\ (\mathrm{mayor}\ (\mathrm{Tokyo})))$

$\forall x\ (\mathrm{Black}(x) \rightarrow \mathbb{B}\ (\mathrm{Simon},\ \mathrm{Black}(x)))$,

,where **B** is the belief operator. As is clear, though this set of epistemic formulas is satisfiable, the ground substitution into the variable x is not. This example shows that Herbrand's Theorem does not generally hold for modal logic. Konolige's [13] solution to this problem was to change the meaning of instance. He introduced the so-called bullet operator, which is attached whenever there is a substitution of variables in the scope of a modal operator. Terms with the bullet operator, for example • *t*, always (i.e., regardless of the context) take the denotation in the actual world. See the following set, where the variable x is substituted for Cmayor(Tokyo):

(2) Black (mayor (Tokyo))
$\neg$**B** (Simon, Black (mayor (Tokyo)))
Black (mayor (Tokyo)) → **B** (Simon, Black (•mayor (Tokyo)))

Since the substituted term •*maynr* (*Tokyo*) is different from *mayor* (*Tokyo*) in its denotation, the set remains satisfiable. We should note, however, that his method is quite limited in its expressive power. Consider the following sentence, where two modal operators are nested:

(3) $\mathbf{B}_1$ (Tom, $\forall$x (Black(x) → $\mathbf{B}_2$ (Simon, Black(x)))

Though the variable is outside the modal $\mathbf{B}_2$, still it does not refer to an actual world object. This situation can not be expressed in his system, since his system is restricted to the cases where variables denote objects either in the actual world or in the world denoted by one modal operator. This implies that his analysis has to be extended to multi-intensional contexts.

A promising alternative to Konolige's can be found in Jiang [12]. His overall strategy is summarized as the following principles:

a. The varying domain across possible worlds accessible for an agent
b. no assumption of rigid designators (two same terms need not have the same denotation across possible worlds.)
c. multi-leveled intensionality
d. predicates/ function terms are flexible depending on the agent involved

He introduced level of intension scheme. Each term is indexed accordingly as the nestness of modal operators. The default level is zero. See the following formula taken from Jiang [12]:

(4) $\mathbb{B}$ (Simon, $\mathbb{B}$ (Tom, L (Venus$_2$, Mars$_1$)))

This formula for Simon believes that Tom believes that Venus likes Mars, where the concept Mars is in Simon's mind and that of Venus in Tom's mind from Simon's view. Note that in his analysis predicates (e.g., *like*), and functional terms (e.g., *Father* (*John*)) are also indexed It seems plausible for the interpretations of them to vary across possible worlds, since like in Simon's mind naturally differs from that in Tom's. Now the indexed formulas of (1) come to be as follows:

(5) Black (mayor (Tokyo))
$\neg\mathbb{B}$ (Simon, Black (mayor (Tokyo)$_1$))
$\forall x_0$ (Black(x_0) $\rightarrow$ $\mathbb{B}$ (Simon, Black (x_0))

Since mayor (Tokyo) denotes an actual world individual and mayor (Tokyo)$_1$ that in Simon's accessible world, the second and the third sentences do not contradict. Hence this set is satisfiable.

Now we apply his indexing procedure to treat more interesting examples for formal semantics for natural language related to the problem of substitutivity in intensional contexts.

(6) a John believes that $2+2=4$.
b. John believes that square-root $(144)=12$

Obviously John has different beliefs in (6a b). But they have been formally indistinguishable: they share the same intension Bauerle and Cresswell [6] argues that meaning is not equal to intension but rather to intension and a syntactical make up (this is called structured meaning) following Lewis [15]. But still their analysis, as they admit, left unsolved such sentences as below, where two sentences differ oy in their coreferring terms

(7) a. John believes that Kupe landed in New Zealand.
b John believes that Kupe landed in Aoteraora

New Zealand and *Aoteraora* are not distinguishable for they have the same intension, that is, "if we consider an essential function of a name is denoting". Furthermore, in this case the two sentences share the syntactic structure. We can use here the indexing technique of Jiang quite effectively; the following will be the indexed representation of (7a-b):

(8) a. $\mathbb{B}$ (John, landed-in (Kupe, New Zealand$_1$)
b $\mathbb{B}$ (John, landed-in (Kupe, Aoteraora$_1$)

Note that here New Zealand and Aoteraora are clearly distinguished (they may be different in John's mind), hence generally they are not interchangeable.

Precisely in the same way, several puzzles called Kripke's Puzzle in Kripke [14] falls within this analysis. First of these is a story of Pierre; a Parisian Pierre had a belief that Londres is not pretty. Afterwards he mmoved to London and came to have another belief that London is pretty, not knowing that Londres denotes the same city as London. How can we describe this situation? The two beliefs about the same city is completely coherent in Pierre's mind (since in his mind, Londres and London refer to two distinct cities, his two beliefs do not contradict each other), and at the same time they are, of course, externally incoherent. If names are assumed to be rigid designators, this problem is insurmountable: since London and Londres denote the same entity, if Pierre has two externally incoherent beliefs, his beliefs should be incoherent also internally (in his mind). But once we give up the assumption, the paradox naturally resolves. The following set of sentences represents a: external incoherence of Pierre's beliefs, and b: internal coherence: 2

(9) a. $\mathbf{B}$ (Pierre, $\neg$Pretty (Londres$_0$))
$\mathbf{B}$ (Pierre, Pretty (London$_0$)
b. $\mathbf{B}$ (Pierre, $\neg$Pretty (Londres$_1$))
$\mathbf{B}$ (Pierre, Pretty (London$_1$))

His second puzzle, though a little more complicated, is similar to the first in essence; Peter, not knowing that a famous pianist Paderewski and Polish statesman Paderewski are one and the same person, came to have the following two beliefs:

(10) a. Peter believes that Paderewski had musical talent.
b. Peter believes that Paderewski had no musical talent.

This situation seems to arise when: on one occasion Peter, who is skeptical of the musical talent of politicians, learned the name Paderewski as a name of a famous pianist, and later on another occasion got to know Polish politician Paderewski. In such cases, he will naturally be led to have the above externally contradictory beliefs (though, of course, internally these two beliefs are coherent as above). The point that differs from the first puzzle and hence makes it more difficult, is only that two homonyms are receiving different interpretations. Hence if we distinguish two Paderewskis (e.g., Paderewski* and Paderewski**), this puzzle collapses into the first: in Peter's beliefs these two names have their own reference, Paderewski^{*1} and Paderewski^{**1}, and in reality Paderewski^{*0} and Paderewski^{**0} refers to the same person. As above, examples beyond the reach of Montague semantics [17] can be resolved by using the indexing procedure.

There is a further advantage in this method: we can formalize intensional concepts in first-order logic without relying on higher-order mechanism such as intensional logic As Akama [2] argued, Montague's intensional logic and its variants are computationally expensive in general and, hence, not practical.

3. INDEXING PROCEDURE AND COMMON KNOWLEDGE

As discussed in the previous section, Jiang's indexing procedure is appealing for knowledge representation But we can point out that more complicated problems cannot be described so easily. We here consider the well-known example of Geach [10] and discuss that Jiang's analysis cannot cover this example. His inability to solve this particular problem leads us to the general problem of the so-called common knowledge

(11) Hob believes that a witch has killed Cob's cow and Nob believes that she has blighted Bob's mare

Now situation is that we have to guarantee that the speaker does not believe that there exists a witch, and, at the same time, Hob and Nob have beliefs about the very same witch. As is clear from this, if we are to give this sentence a logical representation, we have to fulfill two contradictory requirements. one is that the existential quantifier has to take a wide scope so that it can bind both a witch and she, since both words refer to the same instance of witch. The other is that we have to keep the scope of the quantifier narrow, since the speaker does not believe in its existence in the actual world. Clearly Jiang's indexing procedure, as it is, does not provide a suitable basis for the analysis of this example.

(12) [Hob believes that witch(x_1) has killed Cob's cow and Nob believes that x_1 has blighted Bob's mare.]

The would-be representation in (12) does not guarantee that Hob and Nob has their beliefs about the same individual since the first x1 is in Hob's mind, and the second in Nob's (i.e., they may be quite different)

The next simple sentence reveals that his inability to treat this sentence stems from general problem of representing common knowledge.

(13) John and Mary know that Nancy is kind

In the reading where John and Mary both have the same object in their minds Jiang's analysis simply fails, since if we decompose the sentence into two sentences below, we have no way to guarantee that they share the same knowledge about the same person

(14) John knows that Nancy_1 is kind.
and
Mary knows that Nancy_1 is kind.

In Akama & Ohnishi [3] we develop the concatenation technique, which allows us to concatenate agents when they share the common knowledge. The representation for the above sentence will be, roughly, as follows:

(15) John and Mary know that $\text{Nancy}_{\langle j,m \rangle}$ is kind.

where the series of letters j,m within the angled bracket depicts the concatenated agents. Intuitively, Nancy with concatenated $\text{agents}_{\langle j,m \rangle}$ denotes Nancy from the common viewpoint of John and Mary: John and Mary share the knowledge about the same object denoted by Nancy. The formal definition of this common view will be found in the next section.

Now it should be clear how we can treat the above Hob-Nob problem. Since the only remaining problem is how we can guarantee that Hob and Nob have beliefs about the same entity (i.e. the denotation of witch), the following representation satisfies our requirements:

(16) $\exists x_{\langle h,n \rangle}$ [Hob believes that witch($x_{\langle h,n \rangle}$) has killed Cob's cow
and Nob believes that $x_{\langle h,n \rangle}$ has blighted Bob's mare]

where the variable $x_{\langle h,n \rangle}$ ranges over objects in the shared view of Hob and Nob. We omit indices on the other expressions irrelevant to this topic. As is clear, this representation guarantees, as required, (1) Hob and Nob has the same witch in their beliefs since the variable $x_{\langle h,n \rangle}$ ranges only over the shared view of Hob and Nob (2) the speaker does not have to be responsible for the existence of the witch since the variable $x_{\langle h,n \rangle}$ is assigned its value not in the real but in the shared belief world.

Of course, the problem of common knowledge is not restricted to this particular problem. We can easily find various situations where we utilize this type of knowledge, since we always need a basis of communication with others. To get a feel of how we utilize such knowledge, imagine a perceptional situation (which provides a most typical example of common knowledge), in which Ken and John are seeing a teenager smoking. If Ken says to Tom, "That should not happen again around here," we understand that Ken is making an assertion based on the fact (i.e., the fact referred to by the pragmatic anaphor *that*) shared with Tom. More complicated example of this kind can be found in Parikh [22]; when we dance, we are (at least pretend to be) sure that we share knowledge with our partners, say, which direction to go, or which foot to step first.

We next consider another example, which, without reference to common knowledge, could not be fully analyzed: definite descriptions. In the course of the discussion, we explicate our system in more explicit terms in relation to existing analyses.

According to Clark & Marshall [8], for a definite description to be used felicitously, not only must the speaker know the referent of a definite description, but also he must be sure that the hearer knows it, too. Consider the following dialogue:

(17) John: I met the man yesterday.
Mary: What happened then?

Suppose John is referring to Ken with the definite description *the man.* In this case the felicitous use of the man requires:

(18) John knows that the referent of the man is Ken

Since here John is expecting the hearer Mary to grasp the referent of the man, the following is also necessary.

(19) John knows that Mary knows that the referent of the man is Ken.

But this is not the whole story Clark & Marshall [8], using a story of Marx Brothers films as an example, argue that we need to check the validity of infinitely many propositions to attain actual common knowledge: K_jA, K_iA, K_jK_iA, $K_jK_iK_jA$, $K_jK_iK_jK_iA$ ....., where A is a proposition and i, j are the agents involved and Kip means an agent i knows p. This approach to common knowledge is what Barwise [4] calls iterate approach, which is in essence equivalent to *fixpoint approach* below (with some provisos, see Barwise [4]):

(19) $B = A \wedge K_i(B) \wedge K_j(B)$

where B is a fixpoint Notice that these approaches to common knowledge is assuming an omniscient observer since we, as human beings, cannot check such infinitely many propositions This observation led Clark & Marshall to the third approach along the same line with Lewis [15]: *shared environment approach* On their account we attain common knowledge without checking infinitely many propositions, when we are copresent at a situation s such that simplies A and we are attending to s. See Clark & Marshall [8] for details.

Now let us get back to the problem of definite descriptions Our analysis on them is simply to introduce a new term E, which denotes every agent in a contextually defined universe of discourse For example, if a discourse is made up of only two persons, E denote these two persons But when talking about a rule or convention observed by all community members, it is construed as denoting all of the members in the community. The first sentence of the dialogue in (16), which is between two persons John and Mary, is represented by attaching ⟨E⟩

to the definite description:

(17) I met the man$_{\langle E\rangle}$ yesterday.

where $\langle E\rangle$ is equivalent to $\langle j, m\rangle$. This representation means that the referent of the man is in the knowledge shared by John and Mary. Here recall that we adopted S5 as a system of knowledge, where its negation is taken as negation as failure. If the system works ideally, it is simply S5, hence, includes the following modal axiom 4 as its theorem:

(18) $Lp \rightarrow LLp$ (4)

where L is the necessity operator. In the present context,

(19) $K_{\langle E\rangle}p \rightarrow K_{\langle E\rangle} K_{\langle E\rangle}p$

where p is the proposition that the referent of the man is Ken. Notice that 4 makes it possible to deduce all the proposition required in the first two approaches, and moreover attain common knowledge without checking infinitely many propositions. The cases of an omniscient observer and visual situation cases, which are typical examples of the third approach, seem to proceed in this way. But ordinary human beings in usual circumstances cannot entertain this nature of powerful S5, because their information is bounded and negation must be interpreted as negation as failure rather than classical negation. Therefore we need to check all the list of infinitely many propositions one by one to attain actual common knowledge, but since it is impossible to do so, we must be satisfied, in practice, with only approximate common knowledge attained by checking only several of them. In this sense the system proposed here rightly reflects (simulates) the situation we are in.

4. FORMAL SEMANTICS FOR INTENSIONAL CONTEXTS

We semantically formalize the above discussion of common knowledge as an extension of the model provided by Jiang [12]. For this reason, we only discuss the point of modifications here. Jiang's model is adopted intact for the rest of the formalization. Before going into details some introduction to Jiang's model might be in order.

Jiang's model has the following Kripke-like model structure:

(20) $M = \langle W, D, p, F\rangle$

where W is a non-empty set of possible worlds, Di a domain for each possible world, pj the accessibility of an agent j, F the interpretation function. The key feature of his model is that the nesting of modal operator constitutes a chain of possible worlds each of which is linked by an agent's accessibility. Remember the example (4) reproduced in the following:

(21) B (Simon, B(Tom, L($Venus_2$, $Mars_1$)))

The Venus is interpreted as Venus in Tom's mind from Simon's view: in a world accessible from Simon's belief world by Tom's accessibility which eventually comes from the actual world by Simon's accessibility The indices simply indicates which possible world in the chain is responsible for the interpretation of a term/predicate See Jiang(1990) for details

Our model for common knowledge, while essentially adopting Jiang's strategy, brings partiality of agents into each world (i e, domains). It is plausible that an agent's knowledge does not cover the entire domain of a possible world, but only a portion of it. The definition of the domain is as follows

1. D_i: a domain associated with each possible $world_i$, where the universal domain $\mathscr{D} = \bigcup_i D_i$, in particular D_0 is a domain for the actual world Obviously, $D_0 \subseteq \mathscr{D}$
2. $D_{i\langle a\rangle}$: a domain for an agent a of a possible world i.
3. $D_{i\langle a\rangle} \subseteq D_i$, and $\bigcup_a \{D_{i\langle a\rangle}\} = D_i$
4. Domain for concatenated agents
   $D_{i\langle a,b\rangle} = D_{i\langle a\rangle} \cap D_{i\langle b\rangle}$, especially $D_{i\langle E\rangle} = \bigcap a \{D_{i\langle a\rangle}\}$
   for every $\in$ Agent (a set of agents)

The clause 3 defines that an agent's domain constitutes a portion of an entire domain of a possible world The clause 4 is the specification of a domain for concatenated agents Since each of concatenated agents are considered to share entities in its domain, the domain of the new agent ⟨*concatenated agents*⟩ is safely considered to be the intersection of the domains of each member Especially, the domain of ⟨E⟩ is the intersection of all the agents involved. A ⟨concatenated agents⟩ acts like an agent and, therefore, has its own accessibility It provides a shared view point of its members

5. META-LEVEL CONTROL FOR INDEXED KNOWLEDGE

So far, we presented a formal account of the proposed indexing procedure. In this section, we show how our theory can be achieved in epistemic logic programming by suitable meta-level control In particular, classical unification is revised to handle substitutions in a modal context. As is well known, classical unification can be written in the meta-language of logic programming in the manner of Bowen and Kowalski [7]

Now we show the simulation of our proposal in meta-programming in logic programming as shown in Akama & Ohnishi [3]. Since we assume that each agent has a different domain to describe the same object in the world, the following integrity constraint is necessary:

(22) $\leftarrow$ not(a=b), same(B(a,P(t_a)), B(b, P(t_b)))

Here the meta-predicate same(A,B) means that terms A and B are equal under unification. This integrity constraint plays a key role in our unification procedure. In other words, terms with the same index are unifiable in a modal context.

(23) demo(kb, B (kb, P θ)) $\leftarrow$ demo (kb, B(kb, P)),
demo (kb, B (kb, Q)), unify (P, Q, θ)

Here demo(kb,P) is the Bowen and Kowalski meta-predicate for provability in logic programming showing that P is provable from the knowledge base kb. The meta-predicate unify(P,Q, θ) intends to say that P and Q are unifiable with most general unifier(mgu) θ to be defined by the next meta-program.

(24) unify (P, Q, θ) $\leftarrow$ a=b, sub (θ , v_a, P, P'), sub (θ , v_b, Q, Q'),
same (P', Q')

where sub(a,b,c,d) is a meta-predicate that means the result d is obtainable by replacing all free occurrences of the variable b in the formula c by mgu a.

Seen from the above, the control strategy above enables us to give modal unification in a first-order meta-language.

Finally we turn to the simulation of our treatment of common knowledge. As elaborated above, common knowledge can be represented by concatenated agents. This idea is easily incorporated by meta-programming technique:

(25) unify (P, Q, θ) $\leftarrow$ $\langle$a,b$\rangle$, sub (θ , v_a,P,P'), sub (θ , v_b, Q, Q'),
same (P', Q')

The concatenation relation can be extended for a general case because it is regarded as an n-place predicate. For example, we say that P is common knowledge if every agent in the discourse knows P:

(26) B (a,P_a), B (b,P_b),..., E = $\langle$a,b,...$\rangle$

(26) can be rewritten as below:

(27) B (E,P), B2 (E,P),..., E = $\langle$a,b,...$\rangle$

where B_2 (E,P) is an abbreviation of B(E, B(E,P)), and so on.

If the interpreter of epistemic logic programming behaves ideally, the above representation agrees with the fixpoint definition of common knowledge And the agent is proved to be logically omniscient. But we cannot generally obtain such consequence even if we simulate S5. This is because *negation as failure* is weaker than classical negation unless we use a computationally sound fragment, e g stratified logic programs. Akama & Ohnishi [3] focused on this aspects of non-monotonicity of knowledge in connection with metaprogramming.

6. CONCLUSION

We have presented an improved version of Jiang's indexing procedure in epistemic logic programming We also related this method to the problem of common knowledge. The meta-level simulation of our approach can be given in first-order meta-language. Though this paper provides only a rough sketch of our system, it is fully implementable.

We are currently working on the formalism of indexing for nigher-order knowledge within first-order logic. The simulation of higher-order modal logic itself is a difficult task since this, of course, involves self-reference. The treatments of indexing and common knowledge will thus be more complicated to control. For instance, they require a nontrivial handling of consistency check, in particular, for liar paradox. We hope to report the extended account of indexed knowledge in the near future.

NOTES

*This paper is an extended version of Ohnishi & Akama [19] [20] and it provides a theoretical foundation to the rather informal discussion in Ohnishi [21] As is clear from this, I owe much to the discussions with Seiki Akama All errors are entirely my own, though.

[1] In Akama and Ohnishi(1990) we claimed that S5, which includes negative introspection, is too strong for a system of knowledge, and proposed a detuned version of S5 (i e, S5 with negation as failure) as a viable alternative, which serves as the basis of the analysis here

[2] For another solution in the framework of Situation Semantics, see Barwise & Perry (1983)

REFERENCES

[1] Akama, S (1986a): A proposal of modal logic programming, *Proc of the 6th Canadian Artificial Intelligence Conference*, 99-10 , Montreal, Canada

[2] Akama,S (1986b): Methodology and verifiability in Montague grammar *Proc. of COLING* '86 88-90, Bonn, West Germany.

[3] Akama,S. and Ohnishi, H (1990) Outline of epistemic knowledge base," in Z W Ras, M Zemankowa and M Emrich (eds) *Methodologies for Intelligent Systems 5 Proc. of the Fifth International Symposium on Methodologies for Intelligent Systems*, 110-117, October, Knoxville, Tennessee, U S A , North-Holland, Amsterdam

[4] Barwise,J (1988)· Three views of common knowledge, *Proc of the 1st Conference on Theoretical Aspects of Reasoning abnut Knowledge*, 365-379, Morgan Kaufmann Publishers, Los Altos

[5] Barwise,J and Perry, J (1983)· *Situations and Attitudes* MIT Press, New York

[6] Bauerle, R and Cresswell, M J (1987)· Propositional attitudes, in D Gabbay and F Guenthner (eds.) *Handbook of Philosophical Logic*, Volume IV, 491-512 Reidel, Dordrecht

[7] Bowen, K A and Kowalski, R A , (1982). Amalgamating language and metalanguage in logic programming, in Clark, K L and Tarnlund, S A (eds), *Logic Programming*, 153-172, Academic Press, New York

[8] Clark, H H and Marshall, C R (1981)· Definite reference and mutual knowledge, in Joshi, Webber and Sag (eds.), *Elbments of Discourse Understanding*, Cambridge University Press

[9] Fariñas del Cerro, L. (1986): MOLOG, a system that extends Prolog with modal logic, New Generation Computing 4, 35-51.

[10] Geach, P. T. (1962): *Reference and Generality,* Cornell University Press, Ithaca.

[11] Hintikka, J. (1962): *Knwoledge and Belief,* Cornell University Press, Ithaca.

[12] Jiang, Y. J. (1990): An Epistemic Model of Logic Programming, *New Generation Computing 8*, 33-59.

[13] Konolige, K. (1986): *A Deduction Model of Belief,* Morgan Kaufmann Publishers, Los Altos

[14] Kripke, S. A. (1979): A puzzle about belief, L. Margalit (ed.), *Meaning and Use,* 239-283, Reidel, Dordrecht,

[15] Lewis, D. K. (1969): *Convention,* Harvard University Press, Cambridge.

[16] Lewis, D. K. (1970): General semantics, *Synthese 22,* 18-67.

[17] Montague, R. (1974): *Formal Philosophy,* R. H. Thomason (ed.), Yale University Press, New Haven.

[18] Moore, R. C. (1985): Semantical considerations on nonmonotonic logic, *Artificial Intelligence 25,* 75-94.

[19] Ohnishi, H. and S. Akama (1992): Indexed Knowledge in Epistemic Logic Programming, Pettorossi, A. (ed), *Proceedings of Meta-92 Third International Workshop on Meta-programming in Logic,* Uppsala University, Sweden.

[20] Ohnishi, H. and S. Akama (1993): Intensional Contexts and Common Knowledge, in *Proceedings of the First Pacific Asia Conference on Formal & Computational Linguistics (PACFoCoL I),* Huang, C., Cang, C., Chen, K., & Liu, C. (eds), The Computational Linguistics Society of R.O.C, Taiwan.

[21] Ohnishi, H. (1996): Tough Adjectives, Intensional Contexts and Speakers' View, 東洋女子短期大学紀要 28.

[22] Parikh, R. (1990): Recent issues in reasoning about knowledge, in R. Parikh (ed.) *Proc. of the 3rd Conference on Theoretical Aspects of Reasoning about knowledge,* 3-10, Morgan Kaufmann Publishers, Palo Alto.