FRANCIS J. PELLETIER

Identity in Modal Logic Theorem Proving

Abstract. THINKER is an automated natural deduction first–order theorem proving program. This paper reports on how it was adapted so as to prove theorems in modal logic. The method employed is an "indirect semantic method", obtained by considering the semantic conditions involved in being a valid argument in these modal logics. The method is extended from propositional modal logic to predicate modal logic, and issues concerning the domain of quantification and "existence in a world's domain" are discussed. Finally, we look at the very interesting issues involved with adding identity to the theorem prover in the realm of modal predicate logic. Various alternatives are discussed.

1. Introduction

This paper is a report on some issues concerning the addition of identity to my automated theorem proving system, THINKER, in the realm of modal logic. Although there is much background material which is of relevance to the overall enterprise (for some of it, see [11], [12]), fortunately not much of it is crucial for understanding the philosophico–logical issues involved with the addition of identity to modal logics. In this introductory section, I mention some of this background without going into details; in the following sections we look at some deeper issues.

THINKER is an automatic theorem proving system, employing a natural deduction format, for the full first–order logic with identity. While it is not important for the logic of what is to be discussed below that the system embodies a natural deduction format, this perhaps explains why the emphasis below is on rules of inference rather than axioms and rather than on resolution–style strategies. The particular system which is implemented mirrors the Kalish & Montague system [3], [4]. The system implements the full first–order predicate logic with identity (but without arbitrary function symbols).

A basic distinction can be made between *direct* and *indirect* methods of (automated) theorem proving in general, and not just in modal logic. For example, were one interested in proofs in the simple propositional logic, there are numerous proof theories available — differing axiomatic developments, different tableaux methods, different natural deduction formulations, and also propositional resolution. A *direct* method of theorem proving is to

Studia Logica **52**: 291–308, 1993.

construct proofs within one of these proof theories — by which I mean both that the result generated would be recognized as a proof in [say] Whitehead & Russell's axiom system and also that the "machine internal" strategies and methods are applications of what it is legal to do within the proof theory. (In Whitehead & Russell, this amounts to finding substitution instances of formulas for propositional variables in the axioms, and applying Modus Ponens). Were one directly constructing proofs in Smullyan [14] tableaux system, the output should be a list of subformulas of the original formula, each with a "sign" (indicating whether they are "true" or "false"), and arranged in such a way that the "dependencies amongst the decompositions" reflect a genuine proofs as defined by the tableaux system. Furthermore, the internal representation of the problem should involve this decomposition and dependency formulation, and actually make use of it in determining whether the original argument was valid or not. A propositional resolution system ought to have an internal representation of clauses each as a representation of a disjunction of literals and ought to use this representation in some way so as to generate resolvants. The output should be a listing of the clauses with an ordering of which formulas gave rise to the null clause by resolution. In this sense, THINKER is a direct theorem proving system for first order logic with identity of Kalish & Montague. Its internal representation and method of constructing a proof is just like the way a student would construct a Kalish & Montague proof — everything THINKER does internally is a legitimate Kalish & Montague proof-step; and its output is straightforwardly a proof in Kalish & Montague's system.

But direct proofs are not the only possibility. An *indirect* method is to use a procedure of one system in order to determine whether *there is a proof* in some other system. In contrast to the direct methods, it does not construct the proof within this other system (except, possibly, by means of a post-processor) nor is the internal representation and strategy directly isomorphic to what is "legal in the system." For example, rather than attempt to generate proofs directly in Whitehead & Russell's propositional system, one could instead construct a truth table for the formula and evaluate it. Should the truth table have all "true" in its final column then there is a proof within Whitehead & Russell's system.

It is never completely obvious whether a method is direct or indirect, as for example whether semantic tableaux are "direct for the proof theory of semantic tableaux" or are indirect in the same way that truth tables are. No doubt the answer has something to do with the *intentions* of the person writing the system. Nevertheless, there are clear cases of indirect methods, and two of them stand out: the "syntactic method" and the "semantic

method".[1] Intuitively, the former method involves an attempt to *represent* the syntactic proof theory within some "more universal logic" — such as first order predicate logic. All formulas of the original logic are taken to be objects in the more universal logic; the structure of complex formulas of the original logic is represented by deploying functions within the more universal logic; and there is a special monadic predicate of the more universal logic which is interpreted as "x is a theorem of the original logic" [let us symbolize it as $ThmO(x)$]. Rules of inference of the original logic are represented as saying that there is a certain relation between two objects of which $ThmO$ is true. For example, if the axioms of the original logic included

$$\neg\neg p \rightarrow p$$

and the rules of inference included Modus Ponens, the more universal logic might represent '$\neg$' as the function 'n' and '$\rightarrow$' as the function 'i', and thereby represent this axiom and the rule as

$$(\forall x)ThmO[i(n(n(x)),x)]$$

$$(\forall x)(\forall y)[ThmO(x)\,\&\,ThmO(i(x,y)) \rightarrow ThmO(y)].$$

All axioms and rules of the original logic would be represented in this manner, and whenever a formula Φ is to be checked for theoremhood in the original logic, we construct a proof of $ThmO(\Phi')$ in the universal logic (where Φ' is the result of representing Φ by functions, constants, and variables) using the above representations of the axioms and rules as premises to the argument. If we can construct such a proof within more universal logic, then we know that there is a proof in the original logic. And depending on our methods in the universal logic, we can sometimes even postprocess this generated proof of the more universal logic to find the proof of the original logic. (As for example by the method mentioned in [9]).

The second indirect method again involves using a "more universal logic", but this time rather than using first order logic to mirror the syntactic metatheory, we use it to formulate the validity–conditions of the semantic metalanguage. To apply this semantic method, one starts with some basic semantic notion in terms of which the notion of validity is defined. One *translates* the object language sentence into one which characterizes it in terms of this basic semantic notion, and then one uses the definition of validity to try to prove that the translation obeys this definition. For example,

[1] The terms are taken from Morgan [8], who was the first (I believe) to describe the indirect methods as being useful in non–classical theorem proving to the automated theorem proving audience. (See also [18]).

consider the propositional logic formulated with $\neg$ and &. The fundamental semantic notion for this logic is "true in a particular interpretation" (or: "true in a row of a truth table"). Validity is "true in all interpretations" (or: "true in every row of a truth table"). So our basic semantic concept is "true in [row, or interpretation] i". Now we wish to translate every sentence of the object language into a statement concerning its truth. Here is one method: we define the translation function $TRANS(A, i)$ as follows:

$TRANS(A, i)$:

if A is a propositional letter p, then $TRANS(A, i) = P(i)$;
if A has the form $\neg B$, then $TRANS(A, i) = \neg TRANS(B, i)$;
if A has the form $(B \,\&\, C)$, then $TRANS(A, i) = (TRANS(B, i)$ $\&\; TRANS(C, i))$.

Each sentence gets a unique translation. Note that "q is true at row i of the truth table" gets translated as $Q(i)$ — meaning, roughly, that i is one of the Q rows. The sentence connectives get "translated" into themselves, but that is just because we were assuming the usual truth tables for the object language $\neg$ and &. Had they had different truth conditions from these, we would have translated them accordingly. Now, to show a sentence Φ to be a theorem, we wish to show that it is true at every row of the truth table; that is, we want to prove

$$(\forall x)(TRANS(\Phi, x)).$$

Thus, rather than providing a proof of Φ within some particular proof theory, we have "ascended to the (semantic) metalanguage" and shown that there must be a proof of the formula within the system. For example, if the object language sentence were $\neg(p \,\&\, \neg p)$, we would try to prove

$$(\forall x)\neg(P(x) \,\&\, \neg P(x))$$

in our more universal theory, first order logic.

I think this basic division between direct and indirect proof methods will help organize our discussion of proof methods in modal logic[2], even though the distinction between direct and indirect methods can easily blur. As remarked above, it is never clear whether a semantic tableaux method

[2]Another indirect method would be to construct models of set of sentences. Such a method has been explored in the realm of Relevant Logics by Thistlewaite and colleagues. See especially [7], but also [15], [16]. However, we shall not consider this method here.

is direct or is an indirect semantic method. There are other philosophical issues that are occasioned by reflection upon the difference between direct and indirect proof methods; some of these are discussed in [13].

In the realm of modal logics, almost all presentations of the logic of these systems are given in terms of axioms. But no one who is interested in providing automated proofs within modal logic uses an axiomatic system, and so it would therefore seem that all these methods of implementing them must be indirect (on the grounds that they import some other methodology for proofs over and above what is allowed in the axiom system). However, I would prefer to count such developments as 'direct' if they employ rules of inference which directly apply to the formulas of modal logic, and only call the method 'indirect' when it eliminates the distinctive modal operators in favour of first–order predicates, relations, or models. (On the other hand, one could look at these developments as providing some other, new proof theory for the modal logics — e.g., a resolution proof theory for modal logics or a tableaux proof theory for modal logics.)

Various authors have developed indirect semantic methods, for example Jackson & Reichgelt [2] and Ohlbach [10] who treat modal formulas by separating the modal portion from the rest of the formula. The modal information (e.g., which worlds are alleged by the formula to be accessible from which other worlds) is given a separate representation from the propositional letters themselves. They then develop unification and resolution methods to show how the negated–conclusion clause form of a modal sentence can be treated. The method employed by THINKER is also an indirect semantic method, but in the context of a natural deduction system rather than within resolution system.

2. Indirect Modal Propositional Logic in THINKER

Modal logics are formed from classical logic by adding 'L' ("necessarily") and 'M' ("possibly") as sentence operators. As is well known, the various systems of normal modal logics are characterized by validity in a frame, wherein the differences amongst the different systems are manifested by different restrictions on a binary accessibility relation between possible worlds, R. This means that if we could convert each formula of a modal logic into a statement that describes the formula as being true in the correct set of possible worlds, then we could use this new statement as a conclusion of an argument whose premises are the particular conditions upon R which are relevant to the specific modal system in which we are interested. This is the method pursued in THINKER — an indirect semantic method.

Of course there are direct methods available for natural deduction systems, but from a practical point of view they have two shortcomings. The suggestion would be to expand the inference rules (and the rules of proof completion) of the natural deduction system so as to be correct for the modal system under investigation. However, this would require *distinct* programs for different modal systems, since the rules of inference, the proof completion rules, and indeed perhaps even the proof strategy itself would all be different in these different systems. The indirect semantic method only requires one program — a good first–order logic with identity theorem prover. Differences amongst the modal systems is a matter of different premises concerning R. Secondly, not every modal system, not even those systems that are popular in the literature, has been given a well–defined natural deduction formulation; indeed, it is very difficult to see how to accommodate certain logics. Better, then, to go with what is known!

In all the modal logics under consideration, the deduction theorem holds:

$$\Gamma, A \vdash B \text{ iff } \Gamma \vdash (A \rightarrow B).$$

Therefore, in any such modal system, the claim that there is a proof of B from a finite set of premises can be equivalently represented as claiming that a single formula is a theorem:

$$A_1, A_2, \ldots, A_n \vdash B \text{ iff } ((A_1 \,\&\, A_2 \,\&\, \cdots \,\&\, A_n) \rightarrow B).$$

The first step in THINKER's method of proof of $\Gamma \vdash B$ in an arbitrary modal system is to apply the deduction theorem, so that the task becomes one of proving some formula to be a theorem (with no premises). Secondly, we "translate" this alleged theorem into a statement about possible worlds. This translation comes in two steps. The first is simply the observation that a formula A is a theorem of a normal modal logic just in case it is true at every possible world. Letting W be a one–place predicate meaning "x is a possible world", this becomes requirement:

$$(\forall x)(Wx \rightarrow TRANS(A, x))$$

where $TRANS(A, x)$ is a function that translates the sentence A into one that says "A is true at (world) x". This function is recursively defined thus:

$TRANS(A, y)$:

1. if A is a propositional letter p, then $TRANS(A, y) = P(y)$
2. if A has the form $\neg B$, then $TRANS(A, y) = \neg TRANS(A, y)$

3. if A has the form $(B \bullet C)$, where $\bullet$ is one of the binary connectives $\rightarrow, \leftrightarrow, \&, \vee$, then

$$TRANS(A,y) = (TRANS(B,y) \bullet TRANS(C,y))$$

4. if A has the form LB, then

$$TRANS(A,y) = (\forall z)(Wz \,\&\, Ryz \rightarrow TRANS(B,z))$$

5. if A has the form MB, then

$$TRANS(A,y) = (\exists z)(Wz \,\&\, Ryz \,\&\, TRANS(B,z)).$$

(In steps 4 and 5, z is distinct from any variable occurring in A and from y). As can easily be seen from the definition of $TRANS(A,y)$, the way we represent that a propositional letter p is true at a possible world y is just to invent a new one–place predicate (which we represent as the upper case of the propositional letter) meaning "p is true at y", $P(y)$. The truth functional connectives (clauses 2 and 3) contribute nothing new to the translation, but the modal connectives L and M (clauses 4 and 5) do. LB is true at a world y just in case B is true at every world related to y; and MB is true at a world y just in case B is true at some world related to y. The function $TRANS(A,y)$ is well–defined and yields a unique translation (up to choice of variables) of A given initial y, in a finite number of steps (depending linearly only on the length of A). For example:

$$TRANS(L(Lp \rightarrow Mp), x) =$$
$$(\forall y)(Wy \,\&\, Rxy \rightarrow ((\forall z)(Wz \,\&\, Ryz \rightarrow Pz) \rightarrow (\exists z)(Wz \,\&\, Ryz \,\&\, Pz));$$

$$TRANS(LMp \rightarrow p, x) =$$
$$((\forall y)(Wy \,\&\, Rxy \rightarrow (\exists z)(Wz \,\&\, Ryz \,\&\, Pz)) \rightarrow Px);$$

$$TRANS(L(Lp \leftrightarrow p) \rightarrow Lp, x) =$$
$$[(\forall y)(Wy \,\&\, Rxy \rightarrow [(\forall z)(Wz \,\&\, Ryz \rightarrow Pz) \leftrightarrow Py]) \rightarrow (\forall y)(Wy \,\&\, Rxy \rightarrow Py)].$$

Such example translations should give the flavour of the operation of $TRANS(A,x)$.

In the current implementation of THINKER, a user specifies which modal system he wishes a proof to be attempted in. THINKER goes to a special file in which the semantic conditions on R corresponding to each axiom are

stored and adds them as premises to the argument to be proved.[3] As is well–known, certain of these premises imply others, certain combinations of the premises are equivalent to other combinations, and indeed certain combinations are equivalent to other (simpler) formulas. For example, it is well–known that T implies D; so whenever a user happens to specify a system by using both T and D, only T is added.[4] It is also true that G is provable in any system with either B or 5; thus any user attempt to specify G in combination with B or 5 will just result in the B or 5 premise being added, respectively. System KB4 is the same as system KB5 (which of course is the same as system KB45). In this case it was "empirically" determined that the most efficient formulation had all of B, 4, and 5 added as premises for any of these combinations entered by the user. The combinations $T5$, $TB4$, $T45$, $DB4$, and $DB5$ are equivalent ways of specifying system S_5. Furthermore, it is also well–known that S_5 is determined by the class of worlds in which

[3]The nomenclature used here follows Chellas [1]. K is the smallest normal system. It is formed from classical logic by adding the interdefinability of L and M plus

$\vdash_k L(\Phi \rightarrow \Psi) \rightarrow (L\Phi \rightarrow L\Psi)$

if $\vdash_k \Phi$ then $\vdash_k L\Phi$.

The other systems under consideration here are formed from K by adding combinations of these axioms:

D. $L\Phi \rightarrow M\Phi$;

T. $L\Phi \rightarrow \Phi$;

G. $ML\Phi \rightarrow LM\Phi$;

B. $\Phi \rightarrow LM\Phi$;

4. $L\Phi \rightarrow LL\Phi$;

5. $M\Phi \rightarrow LM\Phi$.

There are certain dependencies amongst these axioms, so that of the 64 different possible combinations of the axioms, only 21 distinct systems are generated. The semantic conditions on the accessibility relation R corresponding to the axioms are:

(d) [seriality] $(\forall x)(x \in W \rightarrow (\exists y)(y \in W \,\&\, Rxy))$;

(t) [reflexivity] $(\forall x)(x \in W \rightarrow Rxx)$;

(b) [symmetry] $(\forall x)(\forall y)(x, y \in W \rightarrow (Rxy \rightarrow Ryx))$;

(g) [incestuality] $(\forall x)(\forall y)(\forall z)(x, y, z \in W \rightarrow (Rxy \,\&\, Rxz \rightarrow (\exists w)(w \in W \,\&\, Ryw \,\&\, Rzw)))$;

(4) [transitivity] $(\forall x)(\forall y)(\forall z)(x, y, z \in W \rightarrow (Rxy \,\&\, Ryz \rightarrow Rxz))$;

(5) [euclidean] $(\forall x)(\forall y)(\forall z)(x, y, z \in W \rightarrow (Rxy \,\&\, Rxz \rightarrow Ryz))$.

[4]More accurately, we should always say "the semantic condition corresponding to T is added" (cf. the previous footnote for these conditions), but we will just indifferently refer to the axioms and allow context to determine whether we mean the axioms or the semantic conditions on the accessibility relation.

the relation R is a "universal relation":

$$S5: \qquad (\forall x)(\forall y)(Wx \,\&\, Wy \rightarrow Rxy).$$

This premise turns out to be much easier for THINKER to use then any of the other, equivalent, combinations of restrictions on R. Thus when any of $S5, KT5, KTB4, KT45, KDB4, KDB5$ are entered by the user, the formula described by $S5$ is used instead. Also well known is that, in the presence of T, G can be reduced to

$$G': \qquad (\forall x)(\forall y)(Wx \,\&\, Wy \rightarrow (\exists z)(Wz \,\&\, Rxz \,\&\, Ryz)).$$

And so when the user enters some combination of T and G, G' is entered rather then G (T is still entered separately, of course).

Lastly it should be noted that the user can specify that an argument should be attempted in a system other than one of the above, if he knows the relevant accessibility relation that characterizes the system. For example, one might be interested in a proof within the system $S_{4.3}$. This system is S_4 plus any one of a large number of axioms, such as $[(MLA \,\&\, MLB) \rightarrow M(LA \,\&\, LB)]$. If one knows that the addition of

$$(4.3) \qquad (\forall x)(\forall y)(Wx \,\&\, Wy \rightarrow (Rxy \vee Ryx))$$

to T and 4 will describe this system $S_{4.3}$, then one can merely add this as an extra premise within KT4 to any argument that one wishes to investigate the validity of. Overall, then, when a person wishes to prove formula A in the modal systems X, what is done is to show that it is provable that the translation of A is true at every world, given the semantic accessibility conditions corresponding to system X as premises.

Thus THINKER has the capacity, with no further user input, to investigate the validity of arguments of any of these 21 propositional modal systems: K, KD, KT, KB, K4, KG, K5, KDB, KD4, KDG, KG4, KGT, KD5, K45, KB4, KTB (Browerische), KT4 (S_4), KT4G ($S_{4.2}$), KT5 (S_5), KD45, KDG4. In addition, if the user knows the relevant accessibility relation for an extension of any of these systems, he can add it as an extra premise and investigate proofs in such a new system.

3. Modal Predicate Logic

The scheme described above for investigating the 21 propositional modal systems involves finding a formula of the semantic metalanguage that has the truth conditions relevant to the original sentence. The main idea was

that atomic propositions, p or q etc., were represented as $P(x)$ or $Q(y)$ etc. — which said that "p is true at world x" or "q is true at world y", etc. This same idea could be extended to arbitrary predicates: the first–order (monadic) sentence $F(a)$ could be represented as $F(a,x)$ — which would say that "$F(a)$ is true at world x". More generally, any n–place predicate would be represented as an $(n+1)$–place predicate whose last argument records the world relevant to evaluation. (The treatment of an object language '=' will have to be different. For now we restrict our attention to first order predicate logic without identity.) To say that $LF(a)$ is true at world x, one would represent this as $(\forall y)(W(y)\,\&\,R(x,y) \rightarrow F(a,y))$ in precisely the same manner as before. Indeed, this just *is* the method reported earlier, if one thinks of propositions as being 0–place predicates. So the only new machinery needed concerns n–place predicates and quantifiers:

$TRANS(A,w)$:

if A is an n–place predicate F^n followed by n terms $a_1,\ldots,a_n$,
then $TRANS(A,w) = F^{n+1}(a_1 \ldots a_n, w)$;

if A has the form $(\forall x)B$,
then $TRANS(A,w) = (\forall x)TRANS(B,w)$;

if A has the form $(\exists x)B$,
then $TRANS(A,w) = (\exists x)TRANS(B,w)$.

The semantic postulates or axioms on the accessibility relation remain exactly the same as before, giving us our 21 modal predicate logic systems. As a matter of form, one would probably wish to restrict the object language not to have the monadic predicate $R(x)$, since that would get translated into the binary $R(x,w)$ — which could then be confounded with the (metalinguistic) accessibility relation R. [We tacitly made a similar restriction in the propositional language in not allowing w to be a propositional variable — since its translation, $W(y)$, might have been confounded with the metalinguistic predicate "is a possible world". We maintain that restriction here.]

It would be nicest if we had a two–sorted logic so that there were different sets of variables for possible worlds and for "ordinary" individuals in those worlds. (THINKER does not have this feature). If not, we will have to state as a special axiom, which is to be used for every argument in every system, that no possible world is an ordinary individual. There are a variety of ways this might be done, but one that fits in well with other features to be considered shortly is this. We have another metalinguistic predicate $D(x,y)$

which says that "x is in the domain of world y". (And we correspondingly restrict the object language not to have a monadic predicate D.) And in place of the translations just given for quantifiers, we use

$TRANS(A, w)$:

if A has the form $(\forall x)B$,
then $TRANS(A, w) = (\forall x)(D(x, w) \rightarrow TRANS(B, w))$;
if A has the form $(\exists x)B$,
then $TRANS(A, w) = (\exists x)(D(x, w) \,\&\, TRANS(B, w))$.

We would then have some special "world–domain axioms", to be used as premises in all arguments:

$[w - d1]$ $\qquad (\forall w)(\forall x)(W(x) \,\&\, W(y) \rightarrow \neg D(x, y))$;

$[w - d2]$ $\qquad (\forall x)(\neg W(x) \rightarrow (\exists w)(W(w) \,\&\, D(x, w)))$

telling us that no worlds is in the domain of another world and that everything which is not a world is in the domain of some world. We also wish to ensure that no world's domain is empty:

$[w - d3]$ $\qquad (\forall w)(W(w) \rightarrow (\exists y)(D(y, w))$.

Thus the formula $D(a, w)$ should be read as saying "a exists in world w".

As stated so far, neither the Barcan Formula (BF) nor its converse (CBF) holds. Indeed, there have been no restrictions of any sort on what the relationship is between the entities in the domains of two worlds where one is accessible from the other.

(BF) $\qquad (\forall x)LF(x) \rightarrow L(\forall x)F(x)$;

(CBF) $\qquad L(\forall x)F(x) \rightarrow (\forall x)LF(x)$.

(CBF) in effect says that the worlds accessible to w cannot "shrink in domain" — i.e., that everything in w's domain is in their domain also. Should one wish such an "expanding domain" modal logic, the relevant semantic axiom is

(cbf) $\qquad (\forall w_1)(\forall w_2)(W(w_1) \,\&\, W(w_2) \,\&\, R(w_1, w_2)$
$\rightarrow (\forall x)(D(x, w_1) \rightarrow D(x, w_2)))$.

The (BF) axiom is a "non–expansion" axiom, saying that if w_2 is accessible from w_1 then there are no new items in the domain of w_2 than there were in w_1. The relevant semantics axiom is

$$(bf) \qquad (\forall w_1)(\forall w_2)(W(w_1) \,\&\, W(w_2) \,\&\, R(w_1, w_2) \rightarrow (\forall x)(D(x, w_2) \rightarrow D(x, w_1))).$$

Should one wish a "constant domain" logic, one could add both (cbf) and (bf), but it would probably be easier to work with

$$(cd) \qquad (\forall w_1)(\forall w_2)(W(w_1) \,\&\, W(w_2) \rightarrow (\forall x)(D(x, w_1) \rightarrow D(x, w_2))).$$

It might also be noted that adding one of (cbf) or (bf) to any logic which includes symmetry of the accessibility relation (any logic containing the B axiom, for example S_5) will generate a constant domain logic.

The extension to modal predicate logic has not been done for THINKER, but there seems to be no obstacle here — all arguments continue to be in first order predicate logic, only with some further axioms (premises to each argument). THINKER has already demonstrated that predicate logic arguments of this complexity pose no particular problem. We thus can have any of the 21 modal systems, with or without constant, shrinking, or expanding domains (so long as it is logically possible — S_5 cannot have a strictly expanding domain, for example).

4. Identity.[5]

Modal predicate logic raises a host of philosophically interesting questions. (For a classic statement of them see Kripke [5], [6]).We stay here with the formal issues raised in adding identity to the framework just outlined, but nonetheless there seem to be places where even this impinges upon the philosophical issues.

There seem to be two different approaches to identity available to us using the method of translation into the semantic metalanguage. The one is to treat an identity statement just like any predicate, true at a world — that is, to treat identity as a "world relativized relation". The order is to treat identity as a "trans world relation" — that is, an identity statement is simply true, not merely true in some world.

[5] THINKER adds classical identity in first order logic by means of three rules of inference: Reflexivity of Identity (which requires no premises), Leibniz's Law, and Negation of Identity:

[REFL] $\Rightarrow a = a$;
[LL] $\Phi(a), a = b \Rightarrow \Phi(b)$;
[NEGID] $\Phi(a), \neg\Phi(b) \Rightarrow a \neq b$.

The world–relativized approach would extend the $TRANS(A,w)$ functions as follows:

$TRANS(A,w)$:

if A is of the form $(\alpha = \beta)$, then $TRANS(A,w) = I(\alpha,\beta,w)$.

That is, we replace the object language binary identity relation with the world–relativized, ternary relation $I(\alpha,\beta,w)$ — which says that α and β are identical in world w. In such an approach, the various Rules of Inference concerning identity also need to be relativized. The following three seem unproblematic: For all a, b, and x

(Id1) $D(a,x) \Rightarrow I(a,a,x)$;

(Id2) $D(a,x), D(b,x), \Phi(a,x), I(a,b,x) \Rightarrow \Phi(b,x)$;

(Id3) $D(a,x), D(b,x), \Phi(a,x), \neg\Phi(b,x) \Rightarrow \neg I(a,b,x)$.

That is, within the domain of any world x, $I(a,b,x)$ works just like identity. But, we might ask, what shall be said about $I(a,b,x)$ when one or both of a and b are not in domain of x? One possibility is to say nothing. This has the effect of allowing such cases to be true in some worlds and false in other worlds. (Since the semantic metalanguage is extensional, at any particular world w, either $I(a,a,w)$ or $\neg I(a,a,w)$ is true [for example] even if $\neg D(a,w)$ — they aren't undefined, nor do they take on some other truth value. But for different w's it will happen that $(\neg D(a,w_1)\,\&\, I(a,a,w_1))$ and $(\neg D(a,w_2)\,\&\, \neg I(a,a,w_2))$.) This is perhaps the "least philosophically loaded" decision to make, since it presumes no special truths in a world about "objects that don't exist in that world." But it is not the only choice that could be made. My own intuitions would have the following

(Id1′) $\neg D(a,x) \Rightarrow I(a,a,x)$;

(Id2′) $\neg D(a,x), \neg D(b,x), I(a,b,x), \Phi(a,x) \Rightarrow \Phi(b,x)$;

(Id3′) $\neg D(a,x), \neg D(b,x), \Phi(a,x), \neg\Phi(b,x) \Rightarrow \neg I(a,b,x)$;

(Id4′) $\neg D(a,x), D(b,x) \Rightarrow \neg I(a,b,x)$;

(Id5′) $D(a,x), I(a,b,x) \Rightarrow D(b,x)$;

(Id6′) $\neg D(a,x), I(a,b,x) \Rightarrow \neg D(b,x)$.

In this view, self identities are true in a world even when the object doesn't exist in that world. (Id1′) can be put together with (Id1) to form world–relative reflexivity, a rule with no premises.

[w–r REFL] $\Rightarrow I(a,a,x)$.

(Id4′), (Id5′), and (Id6′) are rather like instances LL and NEGID, applied to existence–in–a–world. My intuitions tell me that if α does not exist in w but β does, then α and β cannot be the same in w; or alternatively put, if α exists in w and is identical–to–β–in–w, then β must exist in w. This group of rules tell us that if α and β are identical–in–a–world, then either they both exist in that world or both don't exist in that world. That is, existence–in–a–world is a property of that world in the sense that LL applies to existence–in–a–world, that is, it applies to the property of being in the domain of a world — we might call it "domain LL":

$$[\text{dom LL}] \qquad I(a,b,w) \Rightarrow D(a,w) \leftrightarrow D(b,w)).$$

Now, given that there can be some true identities in a world even if the objects involved do not exist in that world, we need to be able to reason about them; (Id2′) and (Id3′) just mirror our identity rules LL and NEGID, but in a world relative way, and apply them to things that do not exist in the world. Of course once we have [dom LL], then the (Id2) and (Id2′) rules and the (Id3) and (Id3′) rules can be more simply stated as world–relative rules:

$$[\text{w–r LL}] \qquad I(a,b,x), \Phi(a,x) \Rightarrow \Phi(b,x);$$

$$[\text{w–r NEGID}] \qquad \Phi(a,x), \neg\Phi(b,x) \Rightarrow \neg I(a,b,x).$$

Notice that in this version of world–relative identity, there is no requirement to the effect that there be some terms which "denote the same thing in each possible world", or, to weaken it somewhat, "denote the same thing in each possible world in which they exist". It is not quite clear how this requirement might be stated in the language anyway, but we can at least note that there are no a and b such that $I(a,b,w_1)$ and $\neg I(a,b,w_2)$ are mutually inconsistent (unless $w_1 = w_2$ or unless $a = b$). For any two distinct terms α and β, an expression of identity can be true in one world without being true in another — whether or not α and β exist in the relevant worlds. Of course one could add the postulate

$$(\forall x)(\forall y)[(\exists w)I(x,y,w) \rightarrow (\forall w)I(x,y,w)]^6.$$

Another opinion would be to restrict the principle just to hold of certain constants — "rigid designators" a and b:

$$(\exists w)I(a,b,w) \rightarrow (\forall w)I(a,b,w)$$

[6] Or, perhaps, the weaker

$$(\forall x)(\forall y)[(\exists w)I(x,y,w) \rightarrow (\forall w)(D(x,w) \rightarrow I(x,y,w))]$$

if one wanted to add the quantification "in worlds in which they exists". (And perhaps this last principle and these formulas should be restricted to *accessible* possible worlds.)

or the weaker

$$(\exists w)I(a,b,w) \rightarrow (\forall w)(D(a,w) \rightarrow I(a,b,w))$$

(and again, maybe this should be restricted to accessible possible worlds). But I see no reason to try to force the notion of "world–bound identity" into service for "trans world identity". A better strategy would be to take the other avenue for identity.

The second of the two paths one might take for identity in a modal predicate logic is to treat identity as essentially a "trans world concept" in its own right. This means that

$TRANS(A,x)$:

if A is of the form $(\alpha = \beta)$, then $TRANS(A,x) = (\alpha = \beta)$

Identity statements are not relativized to a world; they are true or false, simpliciter. Thus, from the original LL and NEGID rules of classical logic, whenever we are given a true identity $a = b$, we can infer that *every* property of a is one of b and conversely — and this includes "modal properties", since they are merely expressed as a formula which quantifies over possible worlds. Consider then a sentence like "a is necessarily an F", which might be represented as $LF(a)$, or in our framework as

(1) $\qquad (\forall w_1)(W(w_1) \rightarrow (\forall w_2)(W(w_2)\,\&\,R(w_1,w_2) \rightarrow F(a,w_2))).$

If $a = b$, it follows by LL that

(2) $\qquad (\forall w_1)(W(w_1) \rightarrow (\forall w_2)(W(w_2)\,\&\,R(w_1,w_2) \rightarrow F(b,w_2)))$

that is to say, given $a = b$ it follow that b is necessarily an F. Contrapositively, if we had (1) and

(3) $\qquad \neg(\forall w_1)(W(w_1) \rightarrow (\forall w_2)(W(w_2)\,\&\,R(w_1,w_2) \rightarrow F(b,w_2)))$

then NEGID would conclude $a \neq b$. (The same would hold true if the translation (1) had a clause saying that a and b had to exist in the relevant worlds, such as $D(a,w_2)$ or $D(a,w_1)$. By LL and NEGID, if $a = b$ then they exist in exactly the same possible worlds.)

Perhaps this trans–world identity is most suitable to cases in which both terms are "right designators", for it entails that such an identity implies the mutual possession of all qualities including modal ones. Thus if t ('Tully') and c ('Cicero') are such terms, and if '$t = c$' is true, then they share all modal properties.

However, the classic test for rigid designation is whether a true identity is necessarily true. In the present framework the translation of such a claim would seem to be

$$(4) \qquad t = c \rightarrow (\forall w_1)[W(w_1)\ \&\ R(w, w_1) \rightarrow t = c].$$

But far from being an interesting and controversial claim, as "rigid identity" is supposed to be, this is a mere truth functional tautology (almost), and seems unlikely to be what is meant by the classic test. Perhaps the classic test actually mixes the two notions of identity. If $t = c$ and they are both rigid terms, then the world–bound identity holds in each related world.

$$(5) \qquad t = c \rightarrow (\forall w_1)[W(w_1)\ \&\ R(w, w_1) \rightarrow I(t, c, w_1)]$$

or if one prefers a formulation mentioning "in worlds in which they exist",

$$(6) \qquad t = c \rightarrow (\forall w_1)[W(w_1)\ \&\ R(w, w_1)\ \&\ D(t, w_1) \rightarrow I(t, c, w_1)].$$

This last is still a logical truth (given the two views of $I(a, b, w)$ expressed above), but no longer merely an (almost) truth–functional tautology like (4). Recall that $D(a, w) \Rightarrow I(a, a, w)$ [for all w and a] is a rule of inference even in the weaker account of I (i.e., the account with (Id1), (Id2), and (Id3)). It follows then that

$$(7) \qquad (\forall w_1)[W(w_1)\ \&\ R(w, w_1)\ \&\ D(t, w_1) \rightarrow I(t, t, w_1)]$$

is logically true. From the logically true (7), and with the assumption of $t = c$, we can infer the consequent of (6); thus (6) itself is logically true. Note that in this weaker account of I, the formula (5) is not logically true; however, in my preferred, stronger, account of I, in which $\Rightarrow I(a, a, w)$ [for any a and w], formula (5) would be. For, we have

$$(8) \qquad (\forall w_1)[W(w_1)\ \&\ R(w, w_1) \rightarrow I(t, t, w_1)]$$

being logically true, and if $a = b$ we would derive the consequent of (5) by LL, and hence (5) itself must be logically true.

My own feeling about the two avenues for treating identity is that both '$=$' and 'I' should be used, at least in certain circumstances and for certain purposes. It seem to me that we want 'I' in order to be able to express "non rigid identities" such as *the inventor of the bifocals = the first postmaster general of the US*. (I do believe that this really is an identity, and not some other form of predication.) Here we would want to say it is true in some possible worlds but not in others, and this calls for using the world–bound

predicate 'I' in its translation. On the other hand though, for some purposes we wish to make "trans world identifications" and this seems to call for some relation other than 'I'. For example, in the use of modal logic to analyze distributed systems, we sometimes would like to say that the object (or process or memory location) that one processor is working on is the same as the one that another processor is working on (now, or perhaps at a different time). This cannot be adequately captured by 'I', which only "talks about identities from the point of view of one processor." So perhaps we would wish to use both. Some terms would be marked as "rigid designators" in the underlying logic, and an identity statement between two of them would be translated into the semantic metalanguage as $\alpha = \beta$. Terms not so marked would be translated as $I(\alpha, \beta, x)$. Although there might be choices, I would suppose that an identity between a rigid term and a non–rigid term would be translated with 'I' (at least that is how I would to translate *Ben Franklin = the inventor of the bifocals*) It seems to me that there is considerable promise in developing logical systems in which both 'I' and '=' play a role.

5. Acknowledgments

My research into automated theorem proving, and the development of THINKER, has been supported over the years by the (Canadian) NSERC grant OGP5525. I am extremely grateful for this assistance, as are the students who worked on THINKER during parts of this period: Dan Wilson, Gilles Chartrand, Randy Kopach, Xinying Gu, Jeff Kelly, Hong–qi Lu. Thanks also to Charles Truscott for bibliographical help. This particular paper was first drafted when I visited the Seminar für natürlich–sprachliche Systeme at Universität Tübingen in Summer 1989. I sincerely thank Franz Guenthner for his hospitality during this period, and for providing me with a pleasant environment in which to think about automated theorem proving for modal logics. I am also a member of the Institute for Robotics and Intelligent Systems, and I wish to acknowledge the support of the Networks of Centers of excellence Program of the Government of Canada, NSERC, and the participation of PRECARN Associates Inc.

References

[1] B. CHELLAS, **Modal Logic**, Cambridge University Press , 1980.

[2] P. JACKSON end H. REICHGELT, *A general proof method for first order modal logic*, **IJCAI–10**, 1987, pp. 942 - 944.

[3] D. KALISH and R. MONTAGUE, **Logic**, Hartcourt, Brace, Janovich, 1964.

[4] D. KALISH, R. MONTAGUE and G. MAR, **Logic,** Hartcourt, Brace, Janovich, 1980.

[5] S. KRIPKE, *Identity and necessity,* in M. Munitz, ed., **Identity and Individuation,** New York University Press, 1971, pp. 135 – 164.

[6] S. KRIPKE, *Naming and necessity,* in D. Davidson and G. Harman, eds, **Semantics of Natural Language,** D. Reidel Publ. Co., Dordrecht 1972, pp. 253 – 355 and 763 – 769.

[7] M. MCROBBIE, R. MEYER and P. THISTLEWAITE, *Towards efficient 'knowledge-based' automated theorem proving for non-standard logics,* **CADE–9,** Springer-Verlag 1988, pp. 197 – 217.

[8] C. MORGAN, *Methods for automated theorem proving in non-classical logics,* **IEEE Trans. on Computers,** 1976, pp. 852 – 862.

[9] C. MORGAN, *Autologic,* **Logique et Analyse,** 1985, pp. 257 – 282.

[10] H. OHLBACH, *A resolution calculus for modal logics,* **CADE–9,** Springer-Verlag 1988, pp. 500 – 516.

[11] F. J. PELLETIER, **Completely Non–Causal, Completely Heuristic–Driven, Automatic theorem Proving,** Tech. Report TR82–7, Dept. Computing Science, Univ. Alberta, 1982.

[12] F. J. PELLETIER, **Further Developments in THINKER , an Automated Theorem Prover,** Tech. Report TR–ARP–16/87 Automated Reasoning Project, Australia National University, 1987.

[13] F. J. PELLETIER, *The philosophy of automated theorem proving,* **Proceedings of IJCAI–91,** 1991.

[14] R. SMULLYAN, **First Order Logic,** Springer-Verlag 1968.

[15] P. THISTLEWAITE, M. MCROBBIE and R. MEYER, **Automated Theorem Proving in Non–Classical Logics,** Pitman, 1987.

[16] P. THISTLEWAITE, R. MEYER and M. MCROBBIE, *Advanced theorem proving techniques for relevant logics,* **Logique et Analyse,** 1985, pp. 233 – 258.

[17] A. WHITEHEAD and B. RUSSELL, **Principia Mathematica,** Cambridge University Press 1910.

[18] G. WRIGHTSON, *Non-classical logic theorem proving,* **Journal of Automated Reasoning,** 1985, pp. 35 – 37.

DEPARTMENT OF COMPUTING SCIENCE
UNIVERSITY OF ALBERTA
615 GENERAL SERVICES BUILDING
EDMONTON, ALBERTA T6G 2H1, CANADA

Received June 17, 1992

Studia Logica 52, 2 (1993)