ELSEVIER

Artificial Intelligence 74 (1995) 311-350

Artificial Intelligence

Naming and identity in epistemic logic Part II: a first-order logic for naming

Adam J. Grove *

Stanford University, Stanford, CA 94305, USA

Received January 1994; revised March 1994

Abstract

Modal epistemic logics for many agents sometimes ignore or simplify the distinction between the agents themselves, and the *names* these agents use when reasoning about each other. We consider problems motivated by practical computer science applications, and show that the simplest theories of naming are often inadequate. The issues we raise are related to some well-known philosophical concerns, such as indexical descriptions, *de re* knowledge, and the problem of referring to nonexistent objects. However, our emphasis is on epistemic logic as a descriptive tool for distributed systems and artificial intelligence applications, which leads to some nonstandard solutions.

The main technical result of this paper is a first-order modal logic, specified both axiomatically and semantically (by a variant of possible-worlds semantics), that is expressive enough to cope with all the difficulties we discuss.

1. Introduction

There are several well-known modal logics that can represent an agent's knowledge or beliefs about the world. These formalisms help clarify what it means to ascribe "knowledge" to individual agents, and they can suggest or verify correct patterns of reasoning. Another promising application for epistemic logic is to model systems of many interacting agents. Then not only does an agent reason about the state of the world, it also reasons about what other agents know about the world, what these agents know about other agents' knowledge, and so on. In particular, such multi-agent reasoning can involve subtle issues of *naming*. That is, how does one agent refer to others?

* E-mail: grove@research.nj.nec.com. Current address: NEC Research Institute, 4 Independence Way, Princeton NJ 08540, USA.

SSDI 0004-3702(94)00015

Many treatments of multi-agent epistemic logic have all but ignored this question; we explain why below. This is the second of two papers in which we develop epistemic logics that include a useful and general concept of naming. Our approach is to find simple examples from multiple-agent AI and distributed systems research, and then investigate the problems concerning names that arise.

The first paper, [11], investigated naming in propositional epistemic logics. Some of the issues raised there are important for this work, and we review these in Section 2. In this paper we look at first-order logics. The logic we develop in Section 4.2 is far more expressive than any of our propositional logics (although it is correspondingly more complex). In part this is for the usual reasons, to do with predicates, functions, equality, quantification, and the much more realistic first-order semantics. However, there are also more interesting reasons. In this paper we discuss two quite subtle problems relating to naming, *scope* and *multiple ways of referring*. In Section 3.1 we introduce these issues in detail. It is difficult to deal these issues in a general way within propositional logic; in contrast, the first-order logic we develop is well suited to handling both problems.

In the remainder of this introduction, we discuss why "naming" is important in epistemic logic, and the two problems of scope and multiple ways of referring. Some of this discussion repeats part I ([11]), and we refer the reader to that paper for more details.

In a traditional single-agent modal logic there is just one modal operator (say K for "It is known that ..."). The general question we ask in this paper and in [11] is: what happens with more agents? One straightforward answer is to include one operator K_a for every agent a in the system, and make minimal changes to the rest of the logic. For instance, this is the approach taken in [13]. However, observe that:

- The language is fixed, and so the set of agents is fixed also.
- The only way we can refer to a is by means of the operator K_a. If by "name" we simply mean any way of referring to an agent, we see that each agent has just one name.
- Conversely, each name denotes just one agent.
- The composition of the system and the names of the agents in it are *common knowledge* so that every agent knows them, and knows that every agent knows them, etc.

These assumptions are often quite reasonable, particularly for applications involving interactions among a small, fixed set of agents; for instance, see [3, 6, 12, 13, 15, 30, 33, 34, 39]. But there are equally many situations where they are much too restrictive. In part I, we noted that:

- Sometimes there is no fixed set of agents. This happens when the composition of the system changes (e.g., we remove robots from a factory when they break down, and reintroduce them when repaired). This also occurs when an agent does not know what the composition of the system is, and so the agent must reason about various possibilities.
- We often need a way to refer to groups of agents; for instance, "every node in the network knows that the power will be shut off soon."
- We need *non-rigid* names, that is, names that denote different agents in different possible worlds. An agents can reason using these names, such as "the manager",

"the nodes in the network that haven't failed", "the most reliable robot", even when it doesn't know who the name "really" refers to. Note that it is not necessarily common knowledge who these names refer to.

- Sometimes agents must reason about themselves, as in "I know that I must be at work before eight". This becomes an especially important, and nontrivial, issue in *anonymous* systems.
- Often we need ways of referring to agents that are *relative* to ourselves (or to whoever is making an assertion). Typical examples are "the node to my left", "the person behind me", "everyone within 10 meters of me". Also known as *indexical* reference, this has becomes an increasingly prominent issue in AI planning research [1,26].

In [11], we present simple propositional logics that have all of these features. However, these logics are deficient in other significant, and quite subtle, ways. Let us introduce the problems with two examples. First, consider an unreliable network in which some processes may have failed. Let the formula φ stand for "p_1 knows that all correct processes received p_1's last message". Suppose that, in fact, the correct processes are p_1, p_2, and p_3, and p_1 knows that all three of these processes actually did receive the message. Then, under one reading, we could argue that φ is true: p_1 does know that all the processes which are in fact correct did receive its message. On the other hand, p_1 may not know that these processes are the only correct ones (and, perhaps, may spend more time—fruitlessly—waiting for additional acknowledgments). This observation leads to a second reading which makes φ false at w. All the propositional logics we considered assumed that the second of these interpretations is the correct one. Both readings are meaningful, however, and we may need to reason about either.

This example shows that there are different *scopes* that can be used in evaluating names. This can also be seen by thinking about the possible-worlds semantics that are often used for epistemic logic. The non-rigid name "the correct processes" can be evaluated—its referent determined—just once in the *actual* world, or else be re-evaluated in each situation that p_1 considers possible. We call these possibilities *outermost* scope and *innermost* scope, respectively.[1] Somewhat informally, we may look at the general issue in the following way. Although we ask about the truth of some sentence like φ at a particular world (w, say), every knowledge operator we encounter in φ will force us to consider possible worlds other than w. If $\mathbf{n}$ is a name occurring in φ, which world(s) should we evaluate $\mathbf{n}$ at? Because $\mathbf{n}$ is potentially non-rigid, the meaning of the formula will depend on this decision. The problem is that there seems to be several reasonable answers: for example, we can evaluate $\mathbf{n}$ at w (we call this outermost scope); or at the most "recent" worlds we have considered (innermost scope); or at any other world that has been introduced by a modal operator in whose scope the occurrence of $\mathbf{n}$ lies. If $\mathbf{n}$ is within m such operators, there will be $m+1$ choices. In the following, we shall refer to all but innermost scope as being varieties of *outer-scope* reference. In the example above, the only modality was "p_1 knows" and so two scopes were possible for the interpretation of "the correct processes".

[1] These two possibilities for the example are sometimes described as being *de re* and *de dicto* reference, respectively. We will discuss this terminology in much more detail later.

This observation about scope does not explain all the difficulties that can arise in interpreting names. It is not enough to know *who* a name refers to—which the question of scope addresses—because we must also decide *how* the reference is made. We illustrate this with another example.

Suppose there are two robotic agents, A and B, and A has just broken down. He sends a cry for help over a public broadcast system. B, who is the agent responsible for dealing with such matters, may or may not have heard. So A's subsequent action depends on whether he can deduce that "I (A) know that B knows that I need help" (if this is true, he can just wait, but otherwise he should try something else).

So what is a good formalization of "I know that B knows that I need help"? That is, what sentence should A try to deduce from what he knows? The logic we review in Section 2 permits the formula $K_I K_B(I\_need\_help)$, which certainly seems appropriate. But the situation is really much more complex, because a formula like this can be read many ways.

One possibility is that it could mean that that A knows that B knows that he (B) needs help (the arrangement of quotation symbols in "I know that 'B knows that "I need help" ' " captures what we mean here). This is another example of innermost scope, because the second occurrence of I is bound by the innermost modal operator which is K_B. This interpretation is useful in many contexts, and is all our propositional logic allows. But it is clearly not appropriate in the story above.

The alternative is that the second use of the word or name I refers to agent A; i.e., we can use outer scope. The sentence might be read as saying that A knows that B knows that the agent named A needs help. However, this is not the only possible outer-scope reading. The problem, which we refer to as *multiple ways of referring*, is that there are usually several reasonable ways to read a name using outer scope. Note that A might not have sent a message explicitly saying "A needs help", and instead he might have said simply "I need help". So perhaps B didn't know that it was A who sent the message. All B knows then is that "the agent who just sent a message needs help". Clearly, this is not so good for A. Nevertheless, in some sense A is still justified in saying "I know that B knows that I need help". After all, "I need help" was the content of the message, so if B heard it there should be some sense in which B knows it, no matter what B can infer about the identity of the sender. Here, B knows that "the agent who just sent a message" needs help, and it is true that this agent is A. The point we are making is that B might refer to A using some description other than the name "A".

For yet another outer-scope interpretation, suppose that B can always tell from which room the message was sent, and that there is never more than one agent in any room. Then, again, A can say "I know that B knows that I need help". If the assertion is true in the sense that B knows that "the agent in room X needs help", where A is the agent in room X, then A need do nothing more. He does not even need to tell B where he is. We still call this outer scope (the "I" refers to A somehow), but we must be careful about *how* B refers to A. Let us suppose that the robots in this factory are more or less interchangeable (they have similar abilities). Then the agents might identify themselves to each other using the room they are in or the task they are performing. In that case, names like A or B could be similar to serial numbers: useful to us as external observers for describing what is going on, but totally irrelevant to the interaction of the robots

among themselves.

There is sometimes an assumption that the "correct" reading of an assertion like "I know that you know that I know..." has you referring to me by my "proper name". While referring to an agent by his/her proper name is often the appropriate choice, it is not always the appropriate choice. In the situation described at the end of the previous paragraph, the correct way for robots to refer to each other is using position and not using names like A and B. The reading of "I know that B knows that I need help" given there is not just plausible, it is in fact the most natural interpretation there is.

In summary, we have seen two issues that must be addressed whenever we interpret a name or description. First, we must decide on the *scope* (in the formula $K_I K_B(I\_need\_help)$, is it A or B who needs assistance?). Second, we must decide on the appropriate *manner of reference* (which description or name is B using for A in formula $K_I K_B(I\_need\_help)$, when the second "I" is read with outer scope?). There are many possibilities for each, and in different situations we may need to express and reason with any of them. An expressive formalism should allow us to deal with both the issue of scope and the issue of multiple ways of referring in a general way, and the construction of a logic that does this is a major goal of this paper. In particular, our logic allows us to express various outer-scope readings of a sentence such as "I know that you know that I know" yet does not allow outer-scope sentences that contain ambiguity.

In one sense, it is true that the two issues just raised are already quite well known. For instance, the observation about potential scope ambiguity with non-rigid names was already implicit in Russell's Theory of Descriptions ([41,42]; see [43] for a good discussion). In essence, the scope distinction can also be viewed as an aspect of the *de dicto/de re* distinction made in philosophical logic. And our concern with how agents are referred to echoes earlier controversy about the meaning of *de re* reference. We discuss these ideas, and explain their relevance to this paper, in Section 5.

On the other hand, our perspective on these issues is not traditional, and this leads to new results. First, we note that most philosophical work assumes that there is exactly one correct way of interpreting outer-scope (*de re*) knowledge. In our terminology, this amounts to assuming that exactly one "way of referring" is really important. (The two most relevant exceptions we know of are due to Hintikka [17] and Thomason [45], who both consider relaxing this assumption. We discuss their work later.) Even in work in artificial intelligence, those workers who have thought carefully and explicitly about these issues, in practice end up making the assumption that there is a special set of "standard names" underlying *de re* semantics (for example, see [27,31]). All this may be plausible if the goal of logic is to understand human thought and natural language; we will not enter this controversy. And the standard name assumption is without doubt sensible and appropriate in applications where agents' knowledge about each other's identities is comprehensive and straightforward. But our interest is quite different. We want a general concept of "knowledge" that we can ascribe to agents, as a model of their mental states. If we cannot be precise about how agents are referred to, then outer scope is quite useless. And when we look at simple systems, there are many instances where there really are *many* (or perhaps no) important ways of referring. So the first contribution of this work is that we discuss the importance of being able to reason using

all scopes and ways of referring.

The second contribution is our integration of these questions with ideas in our earlier propositional logic, that allows agents to refer to themselves and others relatively (indexically). We show that this integration can be essential to giving a complete account of naming and identity, because sometimes the appropriate "way of referring" is relative. We note that, in discussing this, we are continuing ideas advanced by Lewis [29].

Finally, we emphasize that we discuss a complete logic, with precise syntax, simple yet powerful possible-worlds semantics, and give a sound and complete axiomatization. So, even aside from our differences in motivation and philosophy, our work is novel in that we give a technical as well as philosophical account of naming, identity, relative reference, and scope.

The remainder of this paper is organized as follows. Section 2 introduces possible-worlds semantics in a propositional context, and summarizes the relevant features of one of the logics in [11]. Section 3 discusses more examples that illustrate the subtleties of outer scope and manner of reference. Section 4 is the heart of the paper. In it, we briefly continue our analysis of the problems, and then present the logic we propose as our solution. Also included in this section is an axiomatization for the logic, examples of the logic's application, and some discussion of related work. In Section 5 we discuss the issues we have seen once again, but from a much more philosophical perspective.

2. Propositional logics and possible-worlds semantics

In this section, we review two propositional logics. The first subsection contains an overview of a logic called $S5_n$, and discusses standard possible-worlds semantics for knowledge. Our review of this is extremely brief because the issues involved are quite well known. For more details see, for instance, [4, 14, 16, 20], as well as [11].

The logic $S5_n$ is subject to all the restrictions mentioned in the introduction. In Section 2.2 we present a simple logic, based on material in part I, that is more general in several ways. Some of the new features in this logic, *varying domains of agents, relative names* and *knowledge about self-identity*, become important features of the first-order logic we develop later.

2.1. Possible-worlds semantics and $S5_n$

In propositional epistemic logic agents reason about the world in terms of some fixed collection Φ of primitive propositions. A formula in the logic can be any Boolean combination of the primitive propositions in Φ (formed using $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$). In addition, for the logic $S5_n$ we allow n modal operators $K_1, K_2, \ldots, K_n$, so that if φ is a formula, then so is $K_i\varphi$. We read $K_i\varphi$ as *agent i knows* φ.

An $S5_n$ possible-worlds structure (or $S5_n$ *Kripke* structure) over the vocabulary Φ is a tuple $M = (W, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$. Here W is a set of states, or possible worlds, and π associates with each possible world a truth assignment to the propositions Φ. More formally, $\pi : W \rightarrow (\Phi \rightarrow \{\mathtt{true}, \mathtt{false}\})$. Each $\mathcal{K}_i$ is an equivalence relation on the set of worlds W.

The intuition is that a binary relation $\mathcal{K}_i$ on worlds can represent agent i's knowledge, because it can capture his ignorance about what the world is really like. Suppose that the "real world" is represented by $w \in W$. The agent probably doesn't know this; otherwise, he would be uncertain about nothing. On the other hand, some of the other w' will be inconsistent with what the agent knows. So there is some subset of W which contains just the worlds that are consistent with the agent's knowledge. A relation $\mathcal{K}_i$ can represent this set: we say that $\{w' \in W : (w, w') \in \mathcal{K}_i\}$ is *the set of worlds agent i considers possible from w*. $\mathcal{K}_i$ is sometimes called an epistemic *accessibility* relation. Note that this intuition suggests that K_i can be any binary relation over W. The principal semantic feature which distinguishes $S5_n$ from other modal logics based on the same language are the further conditions that $\mathcal{K}_i$ be reflexive, transitive, and symmetric (i.e., an equivalence relation).

Although some of the properties of $S5_n$ entailed by these conditions have been criticized in the philosophical literature (see [24] for an overview), $S5_n$ has been shown to be useful and well motivated in the context of analyzing multi-agent systems [13, 39]. We briefly review the motivation here. In any possible world, we can suppose that agent i is in a particular *local state*. We imagine the local state as a structure containing all information the agent can reason with and base decisions on. This picture is particularly appropriate for robotic agents because we can often say precisely what memory and sensory capabilities the agent has. Now consider a possible world w, and suppose the agent is in local state s. We can argue that the agent must consider possible *all* worlds, including w, where he is in state s because he has no information to rule any of them out. On the other hand, he should not consider possible any worlds where his state is not s, because the agent can tell that such worlds are inconsistent with what he knows about himself. These arguments show that the accessibility relation partitions the possible worlds according to the state of the agent; thus, the accessibility relation is an equivalence relation. We note, however, that none of our arguments are critically dependent on the nature of the accessibility relation, or on the particular logic $S5_n$. It would be quite straightforward to reformulate our work in the context of other logics, such as the logic $KD45_n$ that is often used to model agent's *beliefs*.

A formula φ is either true or false at a pair (M, w) consisting of a structure M and a world w in M. We define what it means for φ to be true at world w in structure M, written $(M, w) \models \varphi$, by induction on the structure of φ:[2]

- $(M, w) \models p$ iff $p \in \Phi$ and $\pi(w)(p) = \texttt{true}$,
- $(M, w) \models \neg\varphi$ iff not $(M, w) \models \varphi$,
- $(M, w) \models \varphi \wedge \psi$ iff both $(M, w) \models \varphi$ and $(M, w) \models \psi$,
- $(M, w) \models K_i\varphi$ iff $(M, w') \models \varphi$ for all w' such that $(w, w') \in \mathcal{K}_i$.

A formula φ is said to be valid in a structure M, written $M \models \varphi$, iff $(M, w) \models \varphi$ for all worlds w in M. Formula φ is said to be ($S5_n$-) valid iff $M \models \varphi$ for all $S5_n$ structures M. There are several well-known axiomatizations for the class of all $S5_n$-valid formulas (see [4, 14]).

Note that the familiar logic S5 is simply the special case of $S5_n$ where $n = 1$.

[2] The cases for the boolean connectives $\vee, \Rightarrow, \Leftrightarrow$ are omitted because they are definable in terms of $\neg$ and $\wedge$.

2.2. A propositional logic with relative names

We now review material from [11]. As we noted in the introduction, $S5_n$ is a restrictive logic because it makes almost no distinction between the n agents and their names. Three of the more significant weaknesses are the following:

- Each possible world is a model of how the system might be. In general, the number and identity of the agents comprising the system might not be fixed. So $S5_n$-semantics, which assumes that there are exactly n agents, is frequently unrealistic.
- It is not always possible to model an agent's knowledge as a set of worlds that the agent considers possible. We want a concept of knowledge that models the internal "state" of an agent. Now consider the following example. We have a system S, and at time 0 it contains two agents in the same state s. The system is set up so that, at time 1, one of the two agents is chosen at random and enters a new state, t. Note that after time 1 both agents agree on which worlds are possible; all they know is that there is one agent in state s and one in state t. So any possible-worlds structure where the worlds are some "objective" model of reality gives the two agents identical knowledge. But we don't want this, because such a concept of knowledge clearly cannot model the two agents' (different!) states completely. The problem is that possible worlds alone cannot capture an agents' knowledge or ignorance about *who he is*.
- Many applications need relative names. For instance, the denotation of "the agent to my left" is not determined just by what the world is like, it also depends on who is using the name. One special relative name is I, which refers directly to the agent speaking or reasoning.

Much more discussion of these issues is contained in part I. Even more detailed arguments for the second point, that knowledge should include some component of what we call *knowledge about self-identity*, can be found in [29,35]. A formal logic that allows some forms of relative reference, as well as convincing arguments for the usefulness of this in robotic applications, can be found in [25,26]. We also note that *indexical* is a term frequently used to describe objects, like names, that are relative to one agent's perspective. We have used the word "relative" in preference in "indexical" in this paper. In part this is to avoid confusion; in [6,33] the word indexical is instead used as a synonym for non-rigid.

The problem of varying domains has a simple solution. For each world w in the model, we explicitly specify a set A_w which consists of the agents that exist in w. The second and third issues are more subtle, but turn out to be closely related and in fact share a common solution. Note that standard possible-worlds semantics regards the truth of any formula as being determined by the world alone and ignores the identity of the agent uttering it. Thus, in $S5_n$ the truth of a formula is defined with respect to a *world*. But in order to give the semantics of relative names this is not good enough; we must also specify the speaker of the assertion. We can do this formally by defining the truth of a formula with respect to a *pair* (w, a) consisting of world w and an agent a (where $a \in A_w$). All relative names are interpreted with respect to a. More formally, there is a name-interpretation function μ that maps a world, a name and an agent to

some other agent in that world. If $\mu(w, \mathbf{n}, a) = b$ then we say that, in w, a calls b by name $\mathbf{n}$. Note that at world w every name refers to some agent in A_w; the logic will not allow meaningless assertions about an agent's knowledge at a world where the agent does not exist. Our logic also allows relative propositions, so that the function π gives a mapping from proposition symbols, worlds, and agents, to $\{\mathtt{true}, \mathtt{false}\}$. For example, we can have a proposition *leader* which would be true of the pair (w, a) (i.e., $\pi(w, a)(\mathit{leader}) = \mathtt{true}$) exactly if a is the group leader in world w.

The approach of considering world/agent pairs, which is perhaps the most natural way to give semantics to relative names, also provides a natural way of dealing with knowledge about one's identity. We would say pair (w', a') is considered possible from (w, a) if, at world w, agent a thinks w' is possible (i.e., it is consistent with his knowledge) and also that a might be agent a' in w'. This last clause lets us capture directly a's knowledge or uncertainty about who he is. Formally, instead of an accessibility relation $\mathcal{K}_a$ on possible worlds for each agent a, we have one accessibility relation $\mathcal{K}$ which is an equivalence relation on the class of all world/agent pairs. As we have suggested, the intuition for this model is that knowledge cannot be completely represented just as the set of worlds which is considered possible, but also must account for knowledge about who an agent thinks he might be within the world. In our earlier example, the two agents would agree on every "objective" assertion about the world (such as, there are two agents, exactly one of them is in state t, and so on). They differ in who they think they might be: the agent in state t knows that he is that agent in state t.

Using the ideas just presented, we extend $S5_n$ to a propositional logic that allows reasoning about relative names and self-identity. The language's syntax is changed, to allow an arbitrary collection of modal operators $K_{\mathbf{n}}$ for various names $\mathbf{n}$. The truth condition for such an operator is as follows:

$M, (w, a) \models K_{\mathbf{n}}\varphi$ iff $M, (w', b') \models \varphi$ for all (w', b') with $((w, b), (w', b')) \in \mathcal{K}$, where $b = \mu(w, \mathbf{n}, a)$.

According to this clause, any name $\mathbf{n}$ will be interpreted as a relative name. This is not restrictive, because absolute names, that should not be relative to the agent, are simply the special case where the set denoted is independent of which agent we take as a reference point.

There is one more feature in our logic for relative naming: we assume that the syntax includes a special name I that allows the agent to refer to himself (so that we can say things like "I know that the agent with name $\mathbf{n}$ knows ..."). Formally, the name I always has the interpretation of the identity function; that is, $\mu(w, I, a) = a$ always. We should emphasize that our name I is not intended to be a formalization of the natural language word ("I"). The best reading of our I depends on context; for instance, we would read $K_{\mathbf{n}}K_{\mathbf{m}}K_I\varphi$ as "$\mathbf{n}$ knows that $\mathbf{m}$ knows that *he himself* knows φ".

Note that the logic we presented in part I was somewhat more general than this because we allowed relative names that denoted groups of agents. There, instead of a modal operator $K_{\mathbf{n}}$, we allowed $E_{\mathbf{n}}$ and $S_{\mathbf{n}}$ (read as "everybody with name $\mathbf{n}$ knows..." and "someone named $\mathbf{n}$ knows..." respectively). Part I also contains a sound and complex axiomatization and a complexity analysis for this logic.

Finally, we remark that part I's semantics for knowledge about self-identity is certainly not entirely novel. Lewis [29] discusses essentially equivalent semantics, but does not discuss any specific logic. Lespérance [26] uses Lewis's ideas, and gives a rich logic, although he does not focus on the issue of naming. We compare this work with ours in more detail in Section 4.5. There are also a number of other logics with some apparent similarity to our semantics for knowledge about self-identity, in that they supplement the "possible worlds" where formulas are evaluated by an explicit *context* referring to where, when, or by whom an utterance is made. Possibly the most well-known example is Kaplan's work on demonstratives and indexicals [22]. Perhaps the less important distinction between [22] and our work is that the former does not address epistemic modalities, so does not produce anything resembling a logic of knowledge that can deal with indexical terms. What is more important is that philosophical investigations, like Kaplan's, appear almost entirely concerned with formalizing human reasoning and natural language. The principal question is to ask what the words used in such reasoning "really mean". Neither part I nor this paper is intended, in any way, to address these issues. We seek formal logics that are simple, unambiguous, and sufficiently expressive to allow us to talk about the states of simple "agents" in terms of some concept roughly understandable as "knowledge".

3. The problems: scope and multiple ways of referring

In the introduction we introduced two problems, *scope* and *multiple ways of referring*, that complicate our goal of finding an expressive multi-agent epistemic logic. In this section we re-examine these issues, principally by means of additional examples.

3.1. Scope

Non-rigid names have different denotations in different worlds. So when a non-rigid name occurs in a formula, which world do we use when determining who the name refers to? This is the question of *scope*. The technical issue is the following. Suppose formula φ contains names that are within the scope of epistemic operators, for instance, $\varphi = K_{\mathbf{n}_1} K_{\mathbf{n}_2} \ldots K_{\mathbf{n}_m} p$. To determine the truth of φ, possible-worlds semantics will cause us to consider many worlds other than the "real" one. Therefore, there can be several options to use when evaluating names.

It is easy to verify that our propositional logic for naming uses the following scheme. To determine the truth of φ, with respect to world/agent pair (w, a), we (1) determine who $\mathbf{n}_1$ denotes, relative to (w, a), then (2) find all the pairs that this agent considers possible, and finally (3) check if $K_{\mathbf{n}_2} \ldots K_{\mathbf{n}_m} p$ is true in each of these pairs, using the same procedure recursively. It is irrelevant who the names $\mathbf{n}_2, \ldots, \mathbf{n}_m$ denote relative to pair (w, a), because the third step causes us to we evaluate these names later in the process, at different pairs. We have called this possibility *innermost scope*. Innermost-scope semantics have a compositional nature, and are simple and well-defined. Another extreme possibility, outermost scope, is to evaluate every name $\mathbf{n}_1, \ldots, \mathbf{n}_m$ once and for

all, at (w, a), and remember, somehow, who these agents for later in the evaluation process.

Finally, there are intermediate solutions where a name, for example $\mathbf{n}_m$, is not evaluated at (w, a), and nor is it re-evaluated at every single world that agent $\mathbf{n}_{m-1}$ considers possible. Since there are $m-1$ preceding names there are (at least) m reasonable possible ways to evaluate $\mathbf{n}_m$. We call all, except innermost scope, types of *outer*-scope interpretation.

Our next task is to show that all these possibilities are important. Innermost scope is easiest to motivate. Suppose an agent a knows φ. Of course, this agent can have several names: it might have a "proper name" used to sign messages with, a particular physical location that can be used as a name, a role in an organization, and so on. Suppose that later, an agent (named) **m** learns about a's knowledge. If we think about how **m** might learn this, we realize that what often happens is that **m** learns that "the agent named **n**", where **n** is one of a's names, knows φ. For example, a might send a signed message to **m**. Therefore, the state of knowledge afterwards is $K_{\mathbf{m}}K_{\mathbf{n}}\varphi$ where the name **n** must be read with innermost scope. In general, the simple compositional semantics of innermost scope often parallels the process of knowledge acquisition.

Unfortunately, innermost scope alone cannot express everything. Consider how we, as users of natural language, use names in knowledge contexts. If I were to assert, for example, that "You know that I know what time it is" then the "I" refers to me and not the person I'm addressing. So if we were to try to formalize this as $K_{you}K_I \ldots$, the name I should take outermost, not innermost, scope. Many similar examples demonstrate that outer-scope interpretations are often natural to us.

Moreover, outer scopes are sometimes useful for concrete domains within computer science. Here is an example; we will see others later. An agent a sends a message M that will be received by exactly one other agent, although a doesn't know who this will be.[3] Note that the only way a can reason about this other agent is using some descriptive title like "recipient-of-M". Suppose that the information contained in M is something that was previously known only to a, and, subsequently, will only be learned by whoever receives the message M. For example, M might contain a key that a has chosen to be used for encrypting other messages. Now suppose that, in fact, an agent b was the recipient of M, and that b then sends some other message M' to a third agent c (but b does not tell c about the first message M). Finally, suppose that later c is talking with a, and reveals something that proves to a that he (c) received M' from someone who saw M. For example, this would be the case if M' was encrypted with a key that was sent in M. Although c knows that this someone is b, he neglects to tell a. What is known by a and c now? One interesting observation is that "c knows that the recipient-of-M sent M'" seems to be true, so long as name "recipient-of-M" is read with *outermost* scope, because then this name denotes agent b. Agent c doesn't actually know anything about the message M, or even that such a message was sent, so the same assertion using innermost scope is false in this situation. So we have produced an outer-scope assertion which accurately describes (one aspect of) c's knowledge.

[3] More precisely, here we mean an agent named a. This distinction is not important to the example, however.

However, the example is more subtle than this: simply demonstrating such a sentence is *not* a strong argument that outer scope is necessary or useful! This is because agent c knows who sent M' (b). Consider the sentence: "b is recipient-of-M, and c knows that b sent M'", in which everything is interpreted using innermost scope. This assertion is true and, furthermore, it is stronger than the previous statement we considered. So if we were just reasoning about c's knowledge, we would never have any need to consider the earlier, outer scope, assertion because it is subsumed by this innermost-scope sentence. However, and this is the crucial point, we can not avoid outer scope this way when we consider a's knowledge about c. Agent a doesn't know that the relevant third agent is b, so he *cannot* use the innermost scope sentence we have just seen. He has only partial information about c's knowledge. The easiest thing to do in this case is to say that "a knows that c knows that recipient-of-M sent M'" as above, where recipient-of-M takes an outer (although not outermost) scope here.

This example is quite involved, but turns out to be quite typical of when outer scopes are useful: when one agent has partial knowledge about some other agent's knowledge. In such cases, innermost scope may be unavailable because the agent does not know what the right names are. Here, a didn't know that b was the recipient of M, so we cannot easily use innermost scope to describe his knowledge of c's knowledge.

Given that—in some contexts—there seems to be a need for outer scope attributions, how should we do this in a formal theory? This question is surprisingly difficult to answer satisfactorily and occupies us for much of the remainder of the paper.

To begin with, it is clear that allowing outer scope will require our logic to be more complex syntactically. After all, every name can be interpreted in several ways so formulas need to provide more information to avoid ambiguity. Hence, formulas will be longer than in a comparable logic where one fixed scope is assumed always. One idea for an adequately expressive syntax is to mark each occurrence of a name in some way, to tell which scope should be used for evaluation (and so which agent the name actually refers to). The mark would refer back to one of the enclosing modal operators.

There are other solutions as well. For example, another recent work attacking a closely related issue, but in a very different way (using predicate abstraction) is [7]. And perhaps one of the most well known answers to our problem is implicit in Russell's work ([41–43]). Russell shows how one can treat terms—names for us—as disguised definite descriptions (although when names should be treated this way is another question). The expression of such a description involves quantification, but the position of the quantifier in the enclosing sentence is not uniquely determined. That is, the quantifier can have one of several possible scopes, and each possibility gives a different reading. This corresponds exactly to the issue of scope as we have raised it here.

We do not develop any of these suggestions here, however (although the quantificational solution to scope is one aspect of the logic presented in Section 4). This is because the scope classification as presented so far only captures part of the story. In addition to knowing *which* agents are being referred to, we also need to know *how* they are being referred to. This is the issue we consider now.

3.2. Multiple ways of referring

We generally have many ways of referring to an object or individual, and in modal contexts the choices we make can be important. In simple cases this is well known and uncontroversial. Consider, for instance, Russell's famous example. Russell [41] noted that Scott was, in fact, the author of the novel *Waverley*, and that while King George IV certainly knew that Scott was Scott, he apparently did not always know that Scott was the author of *Waverley*. The main point is, of course, that even if two names denote the same individual in fact, we cannot always replace an occurrence of one by the other. One way of understanding why this is so is to think about possible-worlds semantics with nonrigid terms: it is clearly possible that two terms agree in the "real" world and yet have differing denotations in other possible worlds. One can analyze Russell's example this way, giving both "Scott" and "the author of *Waverley*" innermost scope (perhaps inside a "King George knows" modality). This is of course well known. The purpose of this section is to show that the possibility of referring to an object in many ways, and the consequent subtleties, is not only an issue for innermost scope. Indeed, things become substantially more complex for outer scope.

We begin by considering yet another example, inspired by some recent work in [32, 40]. These papers considered a distributed system, where processes are connected by point to point communication lines. However, it is not assumed that there is necessarily a global naming scheme for these lines. So while each agent will have to assign a number (or name) to each of the communication lines it has access to, a might regard his line to b as (his) line #1, yet b might regard the same line as his #3. Such conflicts are inevitable, at least initially, because the processes can only assign channel labels arbitrarily.

[32,40] also considered a logical formalism for analyzing such systems. The syntax of this special-purpose logic fits easily into the framework we gave in part I and Section 2.2. Both logics allow relative names like I and #1. However, the semantics differ significantly. As we have discussed, our logic preserves the innermost scope reading of formulas. In particular, we leave it to the reader to check that a formula such as $K_I K_{\#1} K_I \varphi$ is interpreted as "I know that the agent at the end of line #1 knows 'I know φ' ". Thus, in the formula $K_I K_{\#1} K_I \varphi$, the inner I does not refer to the same agent as the outermost I, but rather refers to the agent at the end of line #1. This is not the reading of this formula in [32,40]. If the issue was simply that of differing scope, then we might be lead to consider an alternative where the inner I denotes the same agent as does the outer; that is, where this symbol is actually *evaluated* in the context of the first K_I. Whomever "I" denotes there, say agent a, will also be who the inner I stands for (a again).

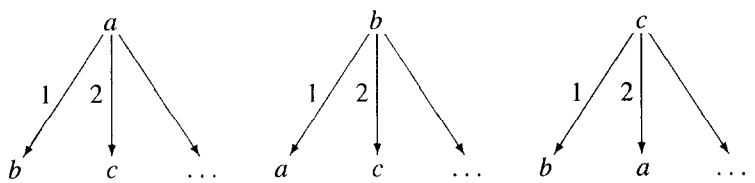
However, this seemingly plausible interpretation does not capture the reading taken by [32,40]. To understand their interpretation, suppose a sends a message saying φ to the agent at the end of his line #1, and then receives an acknowledgment. At this point, we might want to describe a's state of knowledge as "I (a) know that the agent at the end of my line #1 knows that I know φ." Suppose the agent at the end of a's line #1 is b. It is then *not* really the case that b knows that a knows φ. All that b knows is that the agent at the other end of the line that he received the message on knows φ. Although

the agent at the other end of the line that b received the message on happens to be a in this world, it may well be some other agent, c, in some other world that b considers possible. It is *this* interpretation of $K_I K_{\#1} K_I \varphi$—namely, that I know that the agent at the end of my line #1 knows that the agent at the end of the line with the name that he gives to my line #1 knows φ—which is supported by the semantics of [32, 40], and their semantics is designed specifically to enforce this.

This is certainly the appropriate reading of the formula for the protocol Moses and Roth were considering, but it is not the best interpretation for every situation. Imagine that the network is upgraded, so that in addition to the point-to-point lines, a process can broadcast a message over a radio link, and that such broadcasts may carry a signature (that is, some unforgeable identifier). If a broadcasts "I'm a, and I know φ" then everyone (including whoever is on the end of #1) hears this, so a can assert $K_I K_{\#1} K_I \varphi$ again. But in this case, the formula should be read as "a knows that the agent on the other end of his line knows that he (a) knows φ". Both I's now refer to the same agent, and now actually denote a (or more precisely, the agent with a's signature).

So even when we know the *scope* of the second I—it is to refer to the same agent as does the first, that is, it should refer to agent a *somehow*—there is ambiguity about how this reference is made. The agent a can have many names. By which of these names is he referred to by the second agent?

This difficulty with outer-scope references is even more obvious when we consider deeply nested knowledge. Consider a formula such as $K_I K_{\#1} K_{\#2} K_I \varphi$. We concentrate on the second I, and decide that it is to refer—in some manner—to the same agent as does the first. Let us suppose that this first agent, he to whom the whole sentence is relative, is a. Consider the world w shown in the following diagram:



That is, in world w, process b is at the end of a's first line and c is at the end of a's second line; process a is at the end of b's first line and c is at the end of b's second line; process b is at the end of c's first line and a is at the end of c's second line. Note that, in particular, c calls the line connecting c and b line #1, while b calls it line #2.

Now for $K_I K_{\#1} K_{\#2} K_I \varphi$ to be true at w, we must have $K_{\#1} K_{\#2} K_I \varphi$ true at all worlds that a considers possible at w, for some appropriate interpretation of the names #1, #2, and I. Since the agent at the end of a's first line is b, and the agent at the end of b's second line is c, it seems that these are reasonable interpretations of #1 and #2, and, for the purposes of this discussion, this is how we interpret #1 and #2.[4] The question

[4] This interpretation of #1 and #2 is not necessarily the right one; for instance, b might not know that c is #2. However, the example is sufficiently complicated already that we ignore this point.

still remains how to interpret the inner occurrence of the name I. We could take it to be a, or the agent at the end of c's second channel, or the agent at the end of the first channel of the agent receiving on c's first channel, or the agent at the end of b's first, etc. All of these descriptions denote a in the *actual* world w, so apparently are using the same scope. But they do *not* necessarily denote a in all the worlds that a considers that b considers that c considers possible. Thus, they result in quite different interpretations for the original formula $K_I K_{\#1} K_{\#2} K_I \varphi$. The first answer—which is perhaps the standard solution—is the least likely to be useful. The most useful, it would seem, are the second and third, where we determine a by means of its position relative to c. (We remark that Moses and Roth give yet another, somewhat unusual, interpretation to this sentence: the channel number b knows a by (here, #1) is substituted for I in *every* context—even where #1 will not be interpreted relative to b at all! It seems unlikely that this was intended. However, the analysis in their work does not require deeply nested knowledge, so the difficulties arising in such cases are not especially important to them.)

The example just discussed uses relative names, but this is only because the issue may be more apparent in this context, and their use allows us to relate our discussion to [32, 40]. But even without such names, the issue can arise. Computing installations can be known by the actual site name (company, university, whatever) as well as by means of a network address or identifier. If the latter is insufficiently mnemonic then there is a difference between knowing something about a set of machines (locations) and a set of nodes (addresses). Suppose an agent knows that every computer connected to the ARPANET received message m. In a model where we are considering many machines (including ones not connected to the network) "ARPANET" seems best treated as a group name, and the previous sentence has a clear reading if names are given the innermost-scope interpretation. However, it may not have a unique outermost-scope interpretation. (Does the agent know something about a particular set of computing sites, or about the machines named by a particular set of addresses? Depending on how the knowledge was acquired, either could be appropriate.)

All these examples illustrate the importance of the manner of reference. Any adequate formalization for names will have to take this into account. For instance, in dealing with a formula like $K_I K_{\#1} K_I \varphi$, a general logic of naming will have to identify both to whom each occurrence of I refers to and to *how* this reference is to be made, for there could be several options. Is it the "I" at the other end of the link? Is it the "I" that signed the broadcast? Or is there perhaps some other way of referring to I altogether?

We see now that the brief discussion of scope in Section 3.1 is incomplete. There, we implicitly supposed that all we needed to do was to determine which worlds or pairs to use when evaluating a name. Given the outcome of this decision, we look for the agent denoted by the name there: *this* agent is who the name stands for. This simple picture is noteworthy for the fact that no mention is made of *how* the agent selected is referred to; just identifying the agent itself is supposed to be enough. We have just seen that this is inadequate. The choice of which scope to use is indeed necessary to determine which agent is being "referred" to. But the word "referred" hides another story, and another set of decisions: there can be many ways to refer to an agent. In the following, we use the term *scope* to indicate to the first of these choices: which (world, agent) pair do we

use in order to determine the denotation of a name? But, for any outer-scope reference, the manner of reference needs to be specified as well.

In the next section, we present a powerful first-order logic that can express all possibilities. In [10] we explore another approach, which is to convert outer-scope references to innermost scope, and express these in our propositional logic for naming. This technique can handle several of the examples we have seen; in particular, it is powerful enough to capture the various possible semantics for the Moses-Roth logic for relative channel names. However, it still has many limitations. The general case simply seems far too complex to have a good solution within propositional logic.

4. A quantified modal logic

4.1. Introduction

Our overall goal in this paper is to develop a general first-order multiple-agent epistemic logic. It might seem that this is a straightforward exercise, because most of the standard propositional modal logics, like $S5_n$, have quite well-understood quantified counterparts. However, these logics are inadequate because they do not account for how agents refer to each other; i.e., they ignore the issue of naming. In part I, and in the previous sections of this paper, we have looked at several examples and problems that arise concerning naming. Here we present a logic that can cope with all the issues we have raised.

Before looking at details, let us summarize the most important aspects of this solution. First, we believe that agents (real individuals that exist in a possible world) and names (something used by one agent to refer to another, in reasoning) are quite different classes of objects. Accordingly, our logic will treat them separately. Second, we cannot see any situation where one agent reasons about another agent without the mediation of some name or description. The syntax of the logic will be designed to enforce this. Third, outer scope seems to be useful precisely when we do not care to be completely explicit about which name an agent is using. That is, we know there is a name but have partial (or conceivably, no) information about which one. Our logic uses quantification to express this. And finally, we acknowledge that the names used are often relative. Further, an individual's way of referring to *itself* seems to have special properties. So the logic will include a version of the semantics for knowledge about self-identity that we used for our propositional logic.

An example of something our logic can assert is:

$$\exists x\ (\mathit{Talking}(\mathbf{me}, x) \wedge \exists X\ (\ \mathit{Location}(X) \wedge \mathit{In}(\mathbf{me}, x, X) \wedge K_{\mathbf{me}}(\forall y : \mathit{In}(\mathbf{me}, y, X) \Rightarrow \mathit{Tall}(y)))).$$

Of course, in this section we discuss the syntax and constructions used here in detail. The *In* relation is used to say that an agent is associated with a particular (possibly relative) name. This particular assertion can be interpreted as follows: I (**me**) am talking to some agent (x), x's position relative to me is captured by some "location-type" name X (for instance, X might be "in front of"), and I know that the agent in that location is

tall. The more colloquial reading, "I know that I'm talking to someone tall" is also fine as far as it goes, but is misses the point that I know I am reasoning about the agent in a particular position. Perhaps I simply see that he is tall, without knowing his "proper name" or anything else about him. The interplay of names and agents is important in such examples, in order to convey all subtleties clearly and completely.

4.2. The logic

Our logic is many-sorted, with the two distinguished sorts *agent* and *name*. The connection between the two is captured by a distinguished a predicate symbol *In* which holds between a pair of agents and a name (pairs, because our logic will be for relative names). Intuitively, $In(a, b, \mathbf{n})$ holds at a world w iff b is in the set of agents named $\mathbf{n}$ by a in w. (The equivalent of *In* in our propositional logic would be that $\mu(w, \mathbf{n}, a) = b$, although using *In* also works if a calls several agents by name $\mathbf{n}$.)

In the following, $\mathcal{V}$ refers to some fixed first-order vocabulary of function and predicate symbols (including *In*). In defining the language, we also assume the availability an infinite supply of variables of all sorts. (In later examples, we generally use x, y, z, X, Y, Z for variables and t, u, T, U for general terms; upper case is usually used just for variables and terms of sort name.)

Intuitively, possible worlds will simply be first-order structures (interpretations) over the vocabulary $\mathcal{V}$. More formally, a *first-order possible-worlds structure with knowledge about self-identity* over $\mathcal{V}$ is a tuple $M = (\mathcal{W}, \mathcal{K}, \pi)$. Here, π is a function mapping $\mathcal{W}$ to first-order structures over $\mathcal{V}$; this π plays the analogous role in this first-order case as the propositional π did before. Formally, $\pi(w)$ is a function such that:

- $\pi(w)$ maps every sort, including agents and names, into a set of objects which is the *domain* of that sort at w. If s is a sort, we will write $\pi(w)(s)$ as s^w.
- Suppose P is an n-ary predicate in $\mathcal{V}$, which takes arguments of sorts $s_1, s_2, \ldots, s_n$. Then $\pi(w)(P)$, which we will also write as P^w, is an n-ary relation on $s_1^w \times s_2^w \times \cdots \times s_n^w$.
- Suppose f is an n-ary function symbol in $\mathcal{V}$, of sort s, which takes arguments of sorts $s_1, s_2, \ldots, s_n$. Then $\pi(w)(f)$, which we will also write as f^w, is a function from $s_1^w \times s_2^w \times \cdots \times s_n^w$ to s^w.

Note that, unlike the propositional case, the set of agents present at w is included as part of π (that is, using the notation of Section 2.2, $agent^w$ is just $\mathcal{A}_w$). Unchanged from our earlier semantics is the knowledge relation $\mathcal{K}$; this continues to be a equivalence relation on (world, agent) pairs. The only other condition on these structures is that the domain of names must be identical for all the worlds in $\mathcal{W}$. We discuss the motivation for this later in the section. Note that while the class of names is constant, the interpretation of *In* (which captures the connection between names and agents) is determined by π and so can certainly vary from world to world.

The syntax of the language is as follows:

- The class of *terms* includes all variable symbols, constant symbols (0-ary functions), a new symbol **me** of sort agent, and is closed under application of function symbols of the appropriate sorts. The symbol **me** is a special symbol, which behaves much like a constant of sort agent. As we will see shortly, **me** has a special

semantic role, so we have chosen not to regard it as part of the vocabulary $\mathcal{V}$.

- The class of legal formulas, $\mathcal{L}$, includes all atomic formulas (predicate symbol in $\mathcal{V}$ plus argument terms of the appropriate sorts, or else two terms of the same sort connected by equality "=").
- $\mathcal{L}$ is closed under the boolean connectives and quantification.
- If φ is any formula in $\mathcal{L}$ which is either closed or only has free variables of sort name, then $K_t\varphi$ is in $\mathcal{L}$ for any term t that is of sort agent.

With the exception of the knowledge modalities, this is just usual first-order syntax. The introduction of K_t is interesting because it is not quite as general as one might expect. K_t can only occur before a formula with no free variables other than variables of sort name. This means that a "formula" like $\exists x\ K_t P(x)$ is prohibited (if x is not a name variable). Such formulas are said to exhibit *quantifying-in,* because a modality separates an occurrence of a variable from its binding quantifier. Therefore, our syntax prohibits quantifying-in for sorts other than names. The reasons for this restriction, and its consequences, are complex; we discuss this issue at length later.

The final formal definitions relate the semantics to this language. We want to evaluate sentences—closed formulas—at (world, agent) pairs like w, a. But to give a satisfactory formal definition, we need to consider evaluation of formulas which do have free variables. To allow for this, we also provide a variable valuation function v which maps variable symbols into objects from the domain appropriate to the sort of the variable (i.e., if we wish to evaluate a formula at M, w, a, v, and x is a variable of sort s, we require $v(x) \in s^w$). As usual, in the case of closed formulas the function v will turn out to be irrelevant.

The semantic conditions for first-order languages are so well known that we are brief here with the standard features of the language.

- At M, w, a, v, term t denotes a domain element, say $t^{M,w,a,v}$. If t is a constant symbol, $t^{M,w,a,v}$ is t^w. If t is **me**, then $t^{M,w,a,v}$ is a. If t is a variable, $t^{M,w,a,v}$ is $v(t)$. If t is $f(t_1, \ldots, t_n)$, then $t^{M,w,a,v}$ is $f^w(t_1^{M,w,a,v}, \ldots, t_n^{M,w,a,v})$.
- $M, w, a, v \models t_1 = t_2$ iff $t_1^{M,w,a,v} = t_2^{M,w,a,v}$.
- $M, w, a, v \models \varphi \wedge \psi$ iff both $M, w, a, v \models \varphi$ and $M, w, a, v \models \psi$. Other boolean connectives are handled analogously.
- $M, w, a, v \models K_t\varphi$ iff $M, w', a', v \models \varphi$ for all (w', a') such that $((w, t^{M,w,a,v}), (w', a')) \in \mathcal{K}$.
- $M, w, a, v \models \forall x\ \varphi$ iff for all $o \in s^w$ where s is the sort of variable x, then $M, w, a, v[x/o] \models \varphi$, where $v[x/o]$ is just like v except that $v[x/o](x) = o$. Existential quantification is handled by treating $\exists x\ \varphi$ as $\neg\forall x\ \neg\varphi$.

The first clause here includes the semantic definition for **me**; the symbol denotes the agent a from whose viewpoint w is being considered, and so functions very much like the "I" we had in our propositional logic. The difference between I and **me** is minor: the former is a name that usually denotes the identity relation while the latter is of sort agent. In practice, the two can be regarded similarly. We can now see why **me** is treated specially. We wish to allow one world (like w) to be considered from different perspectives (different agents a). If the interpretation of **me** was taken to be part of $\pi(w)$ this would not be possible.

Next, look at the penultimate clause which deals with knowledge, that is, assertions

like $M, w, a, v \models K_t\varphi$. We see that this is very similar to the propositional case. In fact, the difference is really just in the syntax—K operators now refer *directly* to an agent's knowledge (because t is any term of sort agent). In our propositional logics in part I, we used symbols like $E_{\mathbf{n}}$ and $S_{\mathbf{n}}$ where **n** was a name that could denote a group of agents;[5] we can now regard $E_{\mathbf{n}}\varphi$ and $S_{\mathbf{n}}\varphi$ as alternative notation for $\forall x\ In(\mathbf{me}, x, \mathbf{n}) \Rightarrow K_x\varphi$ and $\exists x\ In(\mathbf{me}, x, \mathbf{n}) \wedge K_x\varphi$ respectively. (Where names always denote just one agent, as in our simplified logic of Section 2.2, both forms are equivalent.) We note that these translations would be even simpler if we were not looking at relative names because we could then write $In(x, \mathbf{n})$ rather than $In(\mathbf{me}, x, \mathbf{n})$.

Finally, we consider the clause for quantification. The semantics for this are, in fact, completely standard. The very fact that this is so is actually somewhat surprising and discussion of this occupies the remainder of this section. Application of our logic is discussed in the next subsection.

Traditionally, the subject of quantification in modal logic is difficult and confused. The principal concern is with *quantifying-in,* where a free variable is separated from its binding quantifier by some modality. For example, the sentence $\exists x\ K_{\mathbf{me}}P(x)$ exhibits this (we are using our syntax here, but analogous formulas exist in almost any modal language with first-order quantification). This causes problems for at least two distinct reasons. The first is more philosophical, and is raised in Section 4.3 and again, at more length, in Section 5. Briefly, the issue is similar to our concerns with outer scope. Such formulas assert knowledge about someone (some x) but they do not say anything about *how* this agent is actually being referred to (by **me**, or whoever is doing the reasoning). But in this section, let us look at the second and more technical difficulty. In general possible-worlds semantics, domains for objects can vary from world to world. Suppose we have a logic where $\exists x\ K_{\mathbf{me}}P(x)$ is legal, and where quantification is given the semantics standard to most accounts of first-order logic (including ours). Then it is conceivable that x becomes bound to an object that exists in the "actual" world w but that doesn't exist in some of the worlds that I (**me**) consider possible. If this occurs, how are we to test the truth of $M, w', a, v \models P(x)$? If the object that x is bound to (i.e., $v(x)$) doesn't exist at w', then the interpretation $\pi(w')$ will not help because $P^{w'}$ is a relation defined over a domain which does not include this object at all. As an example, consider the case where I come to believe that some particular individual, if he exists at all, lives at the North Pole. And further, let us suppose that in the real world there is such a person, even though I do not know this for certain. Then, should $\exists x\ K_{\mathbf{me}}NorthPole(x)$ be considered true? Arguments can be made for and against this. For although I consider possible some worlds where (the object denoted by) x is not even present, in all those worlds where x does exist the assertion $NorthPole(x)$ is true. Here we focus on the technical question which is this: what truth conditions should we use when evaluating the truth of formulas like $NorthPole(x)$?

This question needs to be resolved in any quantified modal logic that allows quantifying-in. Several ways of doing this have been suggested (for example, see [8] which contains a good survey of such quantified modal logics). However, we have

[5] These modalities are to be understood as "everyone named **n** knows" and "someone named **n** knows", respectively.

found that none of the well-known solutions are really appropriate for our goal, which is to model actual multi-agent computer systems using possible-worlds semantics. For instance, it is possible to avoid the quantifying-in problem completely by assuming that every knowledge-accessible worlds has the same or larger domain for objects with the sort of x. In our case, where $\mathcal{K}$ is an equivalence, this amounts to requiring the domains of objects to be constant from world to world. But while this solution is technically convenient, for us it is quite unrealistic. We imagine our "possible worlds" to be models of concrete systems; it is quite certain that agents which appear in one worlds might be absent elsewhere (after all, perhaps agents die or are created, so that the composition of the system is not commonly known). Another solution, along similar lines, is to *interpret* all the objects for the sort of x at every world, *but* not have all these objects qualify as values of bound variables everywhere.[6] In a sense, this allows us to say that an object o doesn't exist at world w, but nevertheless still ask whether o possesses property P^w.[7] However, we object to this for similar reasons to the "constant domains" case. In a concrete system, we can identify properties of agents that really exist (i.e., by just looking at the system); we do not want to speculate about the characteristics of nonexistent agents!

There is another class of solutions to the problem of quantifying in, which preserve fully general semantics and instead alter the logic. For example, we could say that $P(x)$ is automatically false at M, w, a, v whenever $x^{M,w,a,v}$ doesn't exist in the appropriate domain at w. This proposal possesses a few awkward and counterintuitive technical features (for instance, $x = x$ will not be valid; we refer the reader to [8] for more discussion). Another idea in a similar spirit is to consider situations like the one just mentioned as possessing a third, "undefined", truth value. Details of such a multivalued logic can certainly be worked out, but there is still a problem: for $K_{\mathbf{me}}P(x)$ to be true, is it necessary for $P(x)$ to be true in all relevant worlds, or simply not to be false? These are different, in general. Any one choice will limit the expressiveness of the logic.

Our logic avoids this controversy in what may seem like an unrealistically simple way. Recall that our language was defined to prohibit quantifying-in, for all sorts except names. And it was a condition of our semantics that the domain of names had to be constant from world to world. Conveniently, this is just the requirement that avoids any difficulty with quantifying-in. This solution might appear to be poor, first because it restricts the formulas that can be written (and so, it seems, will give a less expressive logic), and second, because our assumption of constant names domain seems unmotivated (after all, we rejected the constant domains approach in general, in the previous paragraph). Yet neither of these objections is really sound. We claim (1) there is philosophical justification for assuming a constant domain of names, (2) quantifying-in over names allows us to achieve the effect of quantifying-in over the sort agents, except that

[6] That is, each world effectively uses two domains for each sort. The first, which is the same for every world, is the domain over which functions and relations are defined. The second, which is a subset of the first and can vary from world to world, is the domain of objects over which $\exists x$ and $\forall x$ range. The second domain is intended to be the objects that "really exist".

[7] Such logics are sometimes said to be based on a variant of first-order logic called *free logic*.

(3) we avoid the difficult and confused consequences of such quantification. We briefly address the first of the claims now; the others will occupy us in the following sections.

Observe that names are part of the basic vocabulary we are using to describe the system. They enable us to describe similarities or differences between two possible worlds. For example, if agent a has name *leader* in one possible world, and b has this name in another, then a in the first world and b in the second have something in common. This is what allows us to make assertions like "I know that the leader knows ...". In such applications of names, it is just not useful to ask what names "exist" at a world. We can sensibly ask who has a particular name in some world, but the actual collection of names in use is part of the way we describe the possible-worlds structure as a whole.[8] This contrasts with the sort agents, because the domain of agents is a real physical feature of a possible world. It would simply be unrealistic and restrictive to assume that the set of agents is fixed.

4.3. Use of names in the quantified logic

We now look at our assertion that the absence of quantifying-in over agents is not a weakness.

The outline of this section is as follows. To begin, we briefly give one reason why quantifying-in over agents seems to be helpful. In short, this reason is that such quantification appears useful for capturing outer scope. But we have prohibited such quantification in our logic. Instead, its effect can be achieved by quantification over names, and we show how this is done in general. However, something interesting happens when we try to do this. It turns out to be impossible to "translate" quantifying-in over agents to that over names without supplying more information. We need to say *how* agents are being referred to, and this ties in with the discussion of Section 3. Quantifying-in over agents involves assumptions about how agents are referred to that are not universally valid. Our technique using names is more general (because any variety of outer scope is expressible) and more perspicuous (because all assumptions are made explicit in the formula).

If we allow quantifying-in over agents, there might seem to be a simple translation of outer scope. For example, to express "a knows that b knows that he knows φ", where the "he" is outer scope and intended to refer to a, we could write $K_a(\exists x\ x = \mathbf{me} \wedge K_b K_x \varphi)$.[9] In general, the idea is this: Let φ be of the form "$\ldots K\psi(\mathbf{n})\ldots$", where $\mathbf{n}$ is a name we want to interpret with outer scope and K is the modality that opens the scope where the name $\mathbf{n}$ should be evaluated. Suppose for simplicity that $\mathbf{n}$ is a name which denotes just a single agent. Then, roughly speaking, we would translate φ as "$\ldots K(\exists x\ In(\mathbf{me}, x, \mathbf{n}) \wedge \psi(x))\ldots$". Suppose for now that any technical problems that arise in this translation can be overcome. Then, at first glance at least, this new

[8] One way of looking at this is to realize that names are typically reifications of properties we ascribe to agents. However, we can quantify-in over names, but not over predicates; this is what give names their utility. If we moved to a second- or higher-order logic, and allowed quantifying-in for predicate variables, we would no longer require names. But our approach allows us to avoid the complexity of such a powerful logic.

[9] We intend the scope of $\exists x$ to extend to the end of the sentence. In general where, as here, the appropriate scope of quantification is clear from context, we omit extra parentheses.

formula gives as a plausibly adequate representation of the outer scope reference to **n**. The variable x is bound to the denotation of **n** *in the correct scope*, and so later, when x is used in place of the original outer scope use of **n**, this variable will denote the appropriate agent (appropriate, with respect to the scope we wished **n** to take). The difference between the various outer scopes reduces to position of a quantifier (i.e., where do we bind x to **n**?).

This was only an outline of the approach, but it seems quite straightforward. Why have we rejected this approach? We have already seen some technical problems in the previous subsection, but even in cases where these can be overcome more serious objections remain. Let us now look at why we don't really need quantifying-in over agents, and how we can use names instead. The examples we look at will reveal the deeper flaws inherent in the approach just suggested, and the advantages of the technique we adopt.

Before going into this, we need to examine one characteristic of names in our logic. In the previous subsection, we considered the possible-world structure to be supplied with an unstructured collection of names. But consider our earlier example where processes could communicate with each other along point-to-point lines, and also may broadcast messages signed with an unforgeable signature. The names here naturally fall into two groups: channel-type names, which we denoted $\#1, \ldots, \#n-1$; and signature-type names which we might list as $A_1, \ldots, A_n$. Or consider the other example in Section 3: there can be names corresponding to computers' actual sites (perhaps physical location) and names which we think of as network addresses. The point is that we often consider that there are various different families of names present. In our logic, no extra machinery is required to capture this intuition. Among the predicate symbols in the vocabulary $\mathcal{V}$ there may well be a number of unary predicates that take one argument of type name, and we can use such predicates to categorize names. For the first example, there might be a predicate *Cnum*( ) whose intended interpretation is to be true of just the channel-type names, and another *Sig*( ) for the names we think of as signatures. In this way, we can impose whatever structure we require on the collection of names.

Given this observation, we are now in a position to look at examples where we express outer scope using quantification over names. We begin with a straightforward example which shows how to achieve the affect of quantifying-in over agents using names instead.

Suppose that, for some reason, we have been tempted to write the formula $\forall x\ K_{\mathbf{me}}P(x)$ which uses quantifying-in over agents. We would read this as saying that I know, of all the agents that really exist, that they have property P. However, in reality, I refer to these agents in some particular way. Perhaps I have a list of proper names (of persons), and know that the bearers of these names are P. Perhaps I know that the agents in some (physical) locations, relative to me, are P. Or perhaps I describe the agents to myself in a way related to appearance. There are numerous possibilities for how I actually refer to this set of agents: for definiteness suppose that in this situation the appropriate manner of reference uses an agent's position, relative to myself. Then in our logic, we would assume a family of *location*-type names (i.e., we have a predicate *Loc*( ) that distinguishes them from other names that might be around). For example, "*3-Meters-in-front*" might refer to such a name, and $In(a, b, \textit{3-Meters-in-front})$ holds if

b is in the appropriate location relative to a. In deciding to use this family of names, we have decided on a *manner of referring to agents*. Once we have made such a decision, and only then, can we use quantifying-in over names:

$$\forall x\ (\exists X\ Loc(X) \wedge In(\mathbf{me}, x, X) \wedge K_{\mathbf{me}}(\forall y : In(\mathbf{me}, y, X) \Rightarrow P(y))).$$

In words, every agent existing in the current world has some location-type name, and I know that agent(s) with that name satisfy P. At the price of a more complex formula, we avoid the need for quantifying-in over agents.

Our solution seems involved, but is superior in a number of respects. First, being forced to make an explicit decision about how we refer to agents (here, this was to use relative position) is to our advantage when there are other ways this could be done. After all, the model might well contain many other names, and our formula could easily be changed to use these instead. (For example, if there is another family of absolute location names which are not relative to any agent, we could use these instead and the resulting sentence has a slightly different meaning.) The point of this is that in the example, I do not and cannot know simply about some agents, but must know about these agents with the mediation of some particular method of referring to them. This is a philosophically subtle and controversial point, and we return to it in the next section.

Another benefit of our scheme can be seen by noting that there is an alternative to the sentence presented above: the argument of $K_{\mathbf{me}}$ could have been written $(\exists y\ In(\mathbf{me}, y, X) \wedge P(y))$ instead. This differs from the formula we gave if there is no agent in position **n** in some worlds, or more than one agent among which some satisfy P. In these cases, which is the correct translation? For example, how should we treat worlds where X names the empty set? We could ignore them completely, or take the formula to be false if there are any such worlds. The original sentence has the former interpretation, the new suggestion here corresponds to the latter. We can express both possibilities, equally easily. And we are forced to decide which we want; ambiguity or vagueness is not possible.

Next, let us return to one of the examples from Section 3. As we saw there, simply writing a formula such as $K_I K_{\#1} K_I \varphi$ leaves a number of questions undecided. Our first-order logic can capture many different readings, and forces all decisions to be made explicitly. The innermost scope reading is most easily expressed; indeed, our quantified logic uses this reading automatically (because names, and other symbols such as **me**, are interpreted in the latest possible context). The direct translation of this sentence into our logic, assuming the innermost scope reading, is

$$\forall x\ In(\mathbf{me}, x, I) \Rightarrow K_x(\forall y\ In(\mathbf{me}, y, \#1) \Rightarrow K_y(\forall z\ In(\mathbf{me}, z, I) \Rightarrow K_z \varphi))).$$

This seems complex, but appropriate conventions in the notation simplify it greatly. If we use the abbreviations E and S that were suggested earlier, the above becomes just $E_I E_{\#1} E_I \varphi$. If we then adopt our earlier convention that uses K rather than E when names refer to a unique agent, we get $K_I K_{\#1} K_I \varphi$ which is the same syntax as the propositional case and is as concise as we could hope for.

So innermost scope is easy to express, as we might have expected. The situation is more interesting when we look at possible outer-scope readings. Suppose we want the

scope where the second "I" refers (somehow) to the same agent as the first. A quite general form for this is:

$$\forall x\ In(\mathbf{me}, x, I) \Rightarrow K_x(\forall y\ In(\mathbf{me}, y, \#1) \Rightarrow \exists X\ (\Psi(X, \mathbf{me}, y) \wedge K_y(\forall z\ In(\mathbf{me}, z, X) \Rightarrow K_z\varphi))))).$$

(This could be abbreviated somewhat by using E, S, and other standard simplifications.) In this sentence, $\Psi(X, \mathbf{me}, y)$ is some expression which serves to select the name X by which "y" refers to the first agent, x. (That is, we know that the first agent calls the second #1. Ψ inverts this: by what name does the second refer to the first?) This sentence says that for all agents y at the end of my first channel, then whatever name such an agent refers to me by (as determined by Ψ), the agent y knows that the agent with that name (me, the speaker) knows φ. Different possibilities for how the second agent refers to **me** are captured by different choices of Ψ. In order to express the reading of [32,40], we assume there is a family of names like #1 and #2, which correspond to channel numbers: these names are distinguished from others using predicate $Cnum()$. If we now take $\Psi(X, \mathbf{me}, y)$ to be $Cnum(X) \wedge In(y, \mathbf{me}, X)$, then X will refer to that channel which the second agent uses to communicate with the first. We leave it to the reader to check that this choice of Ψ does gives us the interpretation of [32,40]. On the other hand, if we decide that the appropriate manner of reference uses names associated with signatures then $\Psi(X, \mathbf{me}, y)$ could be $Sig(X) \wedge In(y, \mathbf{me}, X)$ (where Sig holds of exactly the signature-type names) or simply $In(y, \mathbf{me}, X)$ if the model contains these names only. Such a reading is appropriate when an agent is identified by means of a unique, unforgeable, signature.

There are some other aspects to this example that deserve comment. We have seen that Ψ depends on the two agents involved (**me** and y) as well as X. In general, Ψ can also depend on the name that the first agent uses to refer to the second. For example, if x is connected to y by several lines, the name that the first actually uses for y—#1—becomes relevant as well, and Ψ could be quite an involved formula. The important point is that there may be many reasonable, distinct, choices for Ψ. In our logic this choice is explicitly presented, and all possibilities allowed for.

Our conclusion is that our first-order language clearly exposes all the options available for interpreting a sentence, and allows any of these to be expressed. For any particular application, only a limited number of these possibilities will typically be useful; in that case, a more succinct notation or a specialized logic can be devised. But in these situations, we should not forget that we are making further assumptions about the semantics, or else are restricting the sentences that can be expressed. There seem to be few such restrictions that are universally appropriate. So it is useful to have a tool which clearly reveals these assumptions that are being made, and the ability to do this is the principal benefit of our first-order logic.

4.4. Axiomatization

The collection of all formulas valid for our semantics has a fairly simple axiomatization. Some of the necessary axioms simply express sound patterns of classical, first-order reasoning:

(Q1) $\forall x\, \varphi \Rightarrow \varphi[t/x]$ if t is substitutable for x in φ (see below for discussion).

(R3) From $\varphi \Rightarrow \psi$, infer $\varphi \Rightarrow \forall x\, \psi$ if x does not occur free in φ.

Here, x is any variable, and φ and ψ can be any formulas in $\mathcal{L}$.

In general, if t and t' are terms, by $\varphi[t/t']$ (respectively, $\varphi\langle t/t'\rangle$) we mean a formula like φ, except that all (resp., some) "substitutable" occurrences of t are replaced by t'. To ensure the validity of rules like (Q1), we must be careful about when these substitutions are performed. A similar situation occurs in classical (non-modal) logic, but our rules for substitutability are necessarily more complex. Let us say that an occurrence of a term t in formula φ is *free* if (1) no variable appearing in that occurrence of t is bound by any quantifier, and (2) unless t is simply a variable of sort name, the occurrence of t is outside the scope of all K operators in φ. Then term t' is substitutable for an occurrence of t in φ if t and t' are the same sort, and if both the original occurrence of t and the occurrence of t' that would result after replacing t by t' are free in φ. We say that t' is substitutable for t in φ just if t' is substitutable for all free occurrences of t. The restrictions on substitution to do with variables not being bound by quantifiers are required for the same reasons as in classical logic. The additional restriction, that prevents substitution within the scope of a modal operator unless both terms involved are variables of sort name, is needed because terms can be non-rigid. For example, suppose P is a predicate on names, and X is a variable of sort name. Then the sentence $(\forall X\, P(X) \Rightarrow K_t P(X)) \Rightarrow (P(N) \Rightarrow K_t P(N))$, where N is a name constant, is not sound for our semantics (and note that this is not an instance of (Q1), because N is not substitutable for the second occurrence of X). For suppose M is a model in which the same collection of names satisfies P in each world. The antecedent $(\forall X\, P(X) \Rightarrow K_t P(X))$ will be true, but $(P(N) \Rightarrow K_t P(N))$ need not be, because the denotation of N can vary from world to world.

In addition to rules for first-order reasoning, we require axioms to deal with identity:

(I1) $x = x$.

(I2) $t_1 = t_2 \Rightarrow (\varphi \Leftrightarrow \varphi\langle t_2/t_1\rangle)$.

(I3) $(x = y) \Rightarrow K_t(x = y)$ if x and y are of sort name.

(I4) $(x \neq y) \Rightarrow K_t(x \neq y)$ if x and y are of sort name.

Again, (I1) and (I2) are just like the usual, classical, axioms aside from our adoption of stronger conditions on "substitutability". Axioms (I3) and (I4) are modal principles, and express the fact that name variables are rigid (note that the corresponding axioms for other variables would not even be in the language).

Finally, we add axioms dealing with knowledge. It turns out that the necessary axioms are similar to those of the system S5:

(A1) All instances of propositional tautologies.

(M1) $K_t\varphi \wedge K_t(\varphi \Rightarrow \psi) \Rightarrow K_t\psi$.

(M2) $K_t\varphi \Rightarrow \varphi[t/\mathbf{me}]$ if t is substitutable for **me**.

(M3) $K_t\varphi \Rightarrow K_t K_{\mathbf{me}}\varphi$.

(M4) $\neg K_t\varphi \Rightarrow K_t \neg K_{\mathbf{me}}\varphi$.

(R1) From φ and $\varphi \Rightarrow \psi$, infer ψ.

(R2) From φ, infer $K_t\varphi$, if this is a well-formed formula.

In these axioms, t can be any term of sort agent. Also, φ and ψ are restricted by the requirement that the resulting instance be a well-formed formula in $\mathcal{L}$ (for example,

in (M3) formula φ must be closed or only have free variables of sort name, because otherwise $K_t\varphi$ would not be well-formed). In (M2), the definition of substitutable guarantees that only those occurrences of **me** that are outside the scope of every K modality in φ are replaced by agent term t. It is not hard to see why the replacement of **me** outside the scope of K's is needed. For suppose $M, w, a, v \models K_t\varphi$, and $t^{M,w,a,v} = b$. Our semantics ensures that $M, w, b, v \models \varphi$. But at M, w, b, v every outer occurrence of **me** in φ will denote b, not a. If we wish to evaluate φ from the point of view of agent a, we should replace each such **me** by some term which denotes b. The term t is suitable, provided it is in fact substitutable for **me**, and this substitution guarantees $M, w, a, v \models \varphi[t/\mathbf{me}]$, which explains (M2). Note that when t is just **me**, (M2) reduces to the usual "knowledge" axiom $K_{\mathbf{me}}\varphi \Rightarrow \varphi$, and so the logic of $K_{\mathbf{me}}$ alone is very close to S5.

It is easy to verify that each of these three sets of axioms and rules express principles of reasoning that are sound for our semantics. Perhaps surprisingly, the combination of the three is complete as well:

Theorem 4.1. *The axiom system consisting of* (A1), (Q1), (I1), (I2), (I3), (I4), (M1), (M2), (M3), (M4), (R1), (R2), *and* (R3) *is sound and complete with respect to the class of all first-order possible-worlds structures with knowledge about self-identity.*

Proof: See Appendix.

4.5. Comparisons with other work

The only other formal first-order logic to incorporate (what we call) knowledge about self-identity, that we know of, is [25, 26]. Since this is a quantified logic also, it is appropriate to compare this work with ours.

First, Lespérance's work is in some ways more complete than ours, as it incorporates a full theory of time and action. Our work does not address these matters, because here we have concentrated on some fundamental issues. Nevertheless, ideas from [26] would be useful in extending our logic to the temporal case.

The real difference between Lespérance's logic and ours is that ours addresses the issue of different methods of referring to an agent (i.e., the fact that there is not just one interpretation of a name even when the scope has been identified). Although [26] assumes innermost scope (just as we do), he allows quantifying-in over any sort, including agents. (He avoids technical difficulties because the domain of agents is considered fixed. We did not want to make any such assumption.) Because innermost semantics are assumed, outer scope use of names would be expressed using quantifying-in over agents in his logic. And the difficulty with using such quantification is that only one way of referring to agents is possible. Indeed, the whole question of *how* to refer to agents is not explicitly raised in this work. The assumption which allows this issue to be avoided is that there is one unique and obvious way to refer to agents. Certainly, this assumption is often reasonable and causes no difficulties for Lespérance's work. But, as

we have shown, it does not always hold.

With respect to first-order epistemic theories in general, comparisons are much more difficult to give because the literature on naming and on quantified modal logics is so large and varied (we have already mentioned [8] as a good overview of the latter). We return to some of the issues in Section 5. In particular, work by Hintikka [17, 18] and Thomason [45] is very relevant, but is better discussed there.

Here, we should begin by remarking that our analysis of quantification for (what we have called) names and agents is not entirely novel. The earliest related treatment we are aware of is [21]: the similarity is that Kaplan proposed analyzing quantifying-in for agents in terms of quantification over descriptions, and his reasons for this analysis seem close to ours. There are a number of differences, however. Perhaps most important is that [21], and apparently most subsequent work, takes seriously the question of finding the *right* set of descriptions to use, and this concern is perhaps appropriate when the task is to model natural language. But in computer science domains the question does not seem to be difficult at all: the *ways of referring* are generally obvious, and there may be none, one "best" way, or many useful possibilities. This causes a shift in focus: instead of trying to analyze quantifying-in for agents as Kaplan does (in terms of upon some special method of referring that it is our job to explicate), we instead want to allow maximum generality in how agents are referred to. One result of this is that we disallow quantifying-in for agents completely, simply because we don't want to assume that there is one privileged way of referring to agents.

Another feature of our work is *knowledge about self-identity.* It is perhaps even more obvious in the domains of interest to us than in the natural language case that the appropriate "ways of referring" are often relative. So even if a formal logic analyzes outer scope and quantifying-in for agents using names and descriptions instead (as we do), it is likely to be incomplete if it does not have knowledge about self-identity or something similar. Yet we are aware of no other fully developed logic that addresses the issue of names in this way which also contains any equivalent of knowledge about self-identity. Lewis [29] discusses the problem of referring to oneself, and arrives at semantics which are similar to ours in that they can be regarded as being based on world/agent pairs. Further, Lewis goes on to propose that *de re* reference (in essence, this is our *outer scope*) can be understood as reference using some relative relationship. Obviously, this idea lies at the heart of our work. Our emphasis on the possibility of many equally important interpretations of outer-scope reference, and our work towards a formal system for expressing these possibilities, are among the major differences between Lewis's work and ours.

And finally, we should comment on the contrast between our work with the many quantified modal logics that have been presented in the past. In general, the differences between such logics reflect differences in semantic or philosophical assumptions. In our work, we have the advantage of having quite simple "real world" applications in mind, against which we can test such assumptions. Such criteria reveal that much previous work is irrelevant to us. As one example, it seems that any work that makes (formal) sense out of the notion of quantifying-in for individuals (i.e., agents) will be able to do so only by making assumptions that we would not wish to consider. (We hope the reasons for this have become clear in the previous sections.) Moreover, even if we did

accept such quantification, there are technical difficulties having to do with nonexistent objects. As we have seen, many of the proposed solutions to this problem depend on assumptions that just are not appropriate to our applications.

5. Names and reference: a philosophical discussion

5.1. *Introduction: knowledge* de re *and* de dicto

In the previous sections, and in part I, our approach has been largely pragmatic. We considered situations that we might like to describe using a formal theory of knowledge, and then produced semantics and logics which accommodated them. But in the process, we skirted some interesting and subtle philosophical issues, and here we wish to go some way towards remedying this omission. We attempt to give a more complete explanation of several decisions taken in the previous section, and also discuss ties between our work and questions that have been raised in the philosophical literature. Of course, we cannot give an exhaustive comparison.

This section is centered around discussion of two different type of modality, *de re* and *de dicto*. In the following we will be assuming some familiarity with these concepts; our discussion will be brief. For a more complete introduction see, for example, [5,36]. Precise or formal definitions of the terms *de dicto* and *de re* do not seem to appear often, and differ from one another when they are given. We begin with *de dicto* (literally, "of words"), because it is more straightforward and less controversial. Intuitively, *de dicto* knowledge is conceptual or intensional; the knower has some *idea* about what the world is like. This idea is represented as an assertion or proposition, which the agent is said to believe. Consider this variant of Quine's example [37]: "I know that there is someone who is a spy", or $K_I(\exists x\ Spy(x))$. This is *de dicto*, because it claims that "there is someone who is a spy" is true in all worlds I consider possible. Within a formal epistemic logic, *de dicto* knowledge is generally being expressed whenever the knowledge operator occurs before a sentence (i.e., a closed formula).

The concept of *de re* knowledge ("of things") is much less clear. Unlike *de dicto*, it is not necessarily explainable simply as knowledge of some proposition. It also depends essentially on the knower's relation to other objects (see [2] for discussion). Note that statements that are simply about the world can refer to objects, but do so by using some description or name. But *de re* is supposedly knowledge of objects *themselves*, and the reference does not necessarily involve any descriptive content. One example from [2]: on vaguely seeing a man coming towards me in swirling fog, I may plausibly believe some things *about this man* but it is less certain whether I have any unique description that I can use for him. Or consider the spy example again. If I had instead claimed that "there is someone whom I know is a spy", perhaps writing this as $\exists x\ K_I Spy(x)$, then this can be seen as *de re* because I know of some particular individual who is a spy. Note that the *de re* interpretation of this example suggests that the knowledge involved is not a relation between me (the knower) and any proposition (something I know about the world), but instead expresses a relation between me, the property of being a spy, and another actual person (whom I attribute this property to).

Despite the plausibility of this example, it is open to question how much sense the concept of *de re* modality makes (see, e.g., Quine [38] for some objections). After all, is it really clear how can we refer to an object other than by means of a description? If I know of some particular spy, it is likely that I have some name or description for this person. But then the *de dicto* assertion "I know that there is someone who meets this description, and who is a spy" seems more appropriate.

We remark that, although the philosophical situation concerning *de re* is difficult, identifying *de re* usage within a formal logic can be relatively easy. In particular, formulas involving quantifying-in are often best understood in terms of *de re* knowledge. The second spy example above is an instance of this, because quantifying-in for agent variable x is used to express knowledge about some particular object, without giving any description of this object.

As we have suggested, the utility of *de dicto* versus *de re* is philosophically very controversial. But in this paper, knowledge is (almost) always *de dicto*. (Formally, this can be recognized in the early adoption of innermost scope and the restrictions on quantifying-in in the first-order logic.) We spend some time in the next subsection explaining why. We do not wish to enter too deeply into the philosophical debate, so the explanation concentrates on concrete and more technical aspects related to our logic. Finally, in the last subsection, we reconsider *de re* once more, and consider whether it ever has a useful role in this work.

5.2. *Do we need* de re *knowledge?*

As we have said, the initial work in part I only dealt with *de dicto* knowledge (although we introduced knowledge about self-identity as a slight modification to the basic *de dicto* concept). It is important to keep in mind that our goal for a theory of knowledge is to ascribe some property, called "knowledge", to agents, that can serve as an abstraction of the agent's actual internal state. This is because we want to use knowledge as a tool for explaining action and change in an agent, and the action an agent chooses, almost by definition, is dependent only on the *state* of an agent. Given this requirement it is quite plausible that all such "knowledge" is *de dicto*, because knowledge is internal to an agent's state and so consists of ideas about the world, rather than being a property essentially dependent on any relation to the world.

There is a technical argument which makes this clearer, by showing that (in a certain, rather weak, sense) *de dicto* knowledge alone always suffices to model internal (mental) states. This argument is the observation that, if the language we use to describe the world is sufficiently rich, then any single agent's knowledge can always be determined completely by some collection of sentences of the form $K_a\varphi_i$ where φ_i is closed. Such sentences are *de dicto*, and we can use what we have called innermost scope for their interpretation. (The proof of the observation here is that it is always possible to strengthen the language to talk about aspects of agents' states directly. Imagine, for instance, that the agents are simple computer processes. In this case, we could in principle include a set of proposition symbols X_i, whose intended interpretation is "the value of my variable x is i". Then, because we have knowledge about self-identity and these relative propositions, such a language can indeed be made rich enough for us

to describe any agent's state completely as some collection of sentences φ_i. But then the *de dicto* assertions $K_a\varphi_i$ will unique determine the agent a's state, and therefore, completely determine his knowledge also.) Thus, given a suitable language, *de dicto* knowledge suffices for a single agent. Of course, this is a somewhat weak argument because the necessary language can be very low-level, seemingly quite inappropriate for human knowledge for instance. But nevertheless, it adds support to the claim that *de re* knowledge is generally unnecessary in our context.

But two questions arise. Why do the standard arguments in favor of *de re* not apply to us? And second, how do we interpret our work in Sections 3 and 4? Here we very briefly look at the first of these, and address the second in the next subsection.

In the philosophical literature, there are many examples supposedly showing that *de dicto* can be either forced or insufficient. Often these are like the "man in the fog" mentioned earlier: there is not a (good) description of name for an object even though we would tend to say that the agent knows *about* the object somehow (i.e., the appropriate knowledge is apparently *de re*). Our response is to note that are far less effective if we allow the agent to refer to himself (his identity, his position, and perhaps the current time) somehow. We can *name* the man in the fog as, "the man I am looking at". In other words, once we allow enough indexicality, *de dicto* will subsume many apparent cases of *de re* knowledge. This point has been made often before, for example, in [2,29,44]. A detailed discussion of this can be found in [44, ch. 8,9], which includes an analysis of several standard philosophical examples.

This explains why we based our first-order logic of Section 4 on semantics which included relative names and knowledge about self-identity. It is certainly possible to reconstruct our logic, including the technique of using quantifying-in for names rather than agents, without these semantic features. But such a logic would be weak. I might know something about an agent, but simply have no (absolute) way of identifying or naming him; in this case, there seems little option other than to use quantification over agents directly. In the logic as we presented it, this will not occur. If I know about an agent, surely I am or have been in some particular relationship to this agent. Therefore, it is reasonable that I could find some *relative* name for him. In this way, a need for *de re* knowledge can be avoided. Our position here is almost identical to that taken by Lewis [29].

We close this section by discussing a much more technical argument which can be imagined, which appears to suggest another role for *de re* knowledge. This discussion, which can certainly be skipped by the reader, is relevant to our treatment of quantification in the logic of Section 4. We illustrate the idea as follows. Consider a first-order possible-worlds structure $M = (\mathcal{W}, \mathcal{K}, \pi)$ with the following properties: first, there an object a which belongs in the domain of agents at every world in $\mathcal{W}$, and second, there is a predicate P in the language such that that $a \in P^w$ for all w. Now consider another agent, b say. It might seems to make formal sense to assert that b knows that a has property P, because this is so in every world in the structure. But a is a semantic entity, not a syntactic one. So it is quite possible that there is no closed formula φ capturing the idea that a satisfies P. But then is b's knowledge *de dicto* or *de re*? The former seems unlikely, because we cannot write $K_b\varphi$ for any suitable φ. On the other hand, the apparently *de re* assertion "there is an agent x, such that b knows $P(x)$", which

can be expressed using quantifying-in for agents, describes the situation fairly well. The point of such an argument would be that maximum expressivity requires the *de re*-like construct of quantifying-in for agents.

But while this argument is formally correct, it has little significance in our context. The reason is somewhat subtle. The issue is whether b's "knowledge" (that a satisfies P) really should be considered to be interesting knowledge, of the type that we want to be able to express, or whether it is purely technical artifact of the model. We argue that it is best regarded as the latter. Suppose we consider another model M', which differs from M in only one detail: in one of the worlds w which b considers possible, agent a does not exist. Instead, another new object c takes its place (so that, considered as first-order structures, w in the old model and the corresponding w' in the new are completely isomorphic). Between M and M' little has changed: all worlds are isomorphic and in particular, there is no sentence in our language that can distinguish w from w'. And yet if we believe that b's knowledge is different in M and M' (which is the position considered in the previous paragraph) then b itself can distinguish w from w'. Surely we should be able to extend the language, somehow, to better reflect b's ability to make such distinctions. For instance, we might add a new term t intended to serves as name that b can use for a. The problem is, in general, we cannot extend the language enough. Note that there is nothing in the semantics to prevent t denoting some agent other than a. So after extending the language, there is *still* a difference between knowledge of $P^w(a)$ and knowledge of $P(t)$, and we can imagine models in which we have one but not the other. The point is that, however rich our language and however careful we are to make it complete enough to express all the distinctions the agents can make, it can always seems as if there is another level of knowledge (which is not *de dicto*) and which requires quantifying-in over agents to express. This extra level of "knowledge" depends on which actual collection of agents we choose at a world, and cannot be captured using $\mathcal{K}$ and the truth assignment π. Note if our language is rich, this knowledge really can have no practical purpose (otherwise, why not extend the language further?). This is why this paper regards such knowledge as nothing more than an accidental feature of the model. So our logics in part I and Section 4 both have the property that we can choose the set of agents at each world arbitrarily, yet whenever two models are isomorphic (in an obvious sense, with respect to $\mathcal{K}$ and π) then the same formulas hold in each. We simply cannot refer to aspects of the model that, quite possibly, have no "real" significance. On the other hand, quantifying-in over agents allows this to occur. Thus, while it makes *formal* sense, it is not clear how such quantification is ever necessary or even appropriate in practice.

5.3. *Cross-world identity*

Previously we noted that if the language is chosen appropriately, any single agent's knowledge can be completely described using a collection of sentences like $K_a\varphi_i$, where the φ_i are closed: *de dicto* knowledge is enough (although it is interesting that this argument fails unless we have knowledge about self-identity).

But suppose we want to give a *partial* description of this agent's knowledge (for example, we may want a concise report about the agent's state, or else wish to describe

knowledge held by a second agent about the first). Is *de dicto* in the above sense sufficient? Clearly, the answer is no. Suppose agent a knows who his neighbor is. To express this, we might be tempted to write $\exists x\ K_a(neighbor(\mathbf{me}) = x)$, which uses quantifying-in over agents. But this is dangerous. After all, agent a's knowledge is *de dicto*: there is actually some name or description N such that $K_a(neighbor(\mathbf{me}) = N)$. Quantification is an attempt to cope with *our* ignorance about N, and so it is reasonable that the quantification should still be interpreted as a disjunction over some class of names (that class we know N must belong to). The quantified sentence above fails to tell us what class to use. We note that this interpretation of quantifying-in, as really being quantification over descriptions or names, seems to have appeared first in [21]; see the discussion in Section 4.5.

We have seen this issue before, of course, in Sections 3 and 4. The solution in our logic was to allow *names* as semantic entities, and to allow quantification over them (but not agents). Since the class of all names might be divided in to groups, we can quantify over some portion of them when that is appropriate. We conclude this section by looking at how this technique fits in to the *de re/de dicto* discussion: it turns out that it can be viewed as either.

First, we could say that knowledge expressed using such quantification over names should still be regarded as *de dicto*. (Knowledge operates on sentences which are almost closed; the only free variables are *names* which have an intuitive interpretation as "ways of referring".) But on the other hand, we might observe that we do have some quantifying-in, and we can use our logic to express some varieties of outer-scope reference, so it looks somewhat like *de re* knowledge as well. This second point of view is quite reasonable if we are careful. In the remainder of this section, we will see what is needed to make this view workable. The result is another interpretation of the "different ways of referring to an agent" problem we have seen repeatedly in this paper.

In the example of a and his neighbor, the *de re* viewpoint would be to say that a has knowledge of *someone*, i.e., the person who actually is his neighbor. That is, there is someone in every one of a's possible worlds that bears the relation of neighborhood to him. But to make sense of such a claim, we need to be able to say how one individual (this "someone" a knows about) can appear in several different possible worlds. And in fact, it is not clear how we can possibly do this. For after all, given an agent which exists in possible world w, there is almost certainly not going to be another agent, identical in *all* respects, existing in any other world (if nothing else, the world it inhabits differs, although there are likely to be more concrete differences). So under what circumstances do we group two such agents together as being the same? If we want to interpret any claim of *de re* knowledge, we must answer this question. The problem is that there may be many answers (or none).

If we say two agents are the same just if they have certain physical characteristics in common (for example, where they are located) then we get one interpretation of *de re* modality. If we choose other criteria, for example, based on the agent's relative location, we get different semantics. This is what happens in our earlier $K_I K_{\#1} K_I \varphi$ example: the agent #1 probably knows something about *me*, in the sense of an agent who is in one definite position (channel number) relative to him. Alternatives would be to know about me, in the sense of the agent with the a particular signature, or in the same (absolute)

position, or in some other sense. These are all different.

Let us consider another concrete example of the difficulties that arise. Suppose we intend to design the protocol for a computer that operates in the following distributed environment. The network connects n machines, one at each of n different divisions within a company, in a ring structure. In addition, each actual machine comes with a unique hard-wired identification number that is available for the protocol's use. It is clear that the protocol we write should not depend too strongly on the precise mappings between the actual (physical machines), their order in the ring, and their location. After all, each of these may change; this should not require programs to be rewritten. In other words, we would like a protocol that does not require any initial knowledge of these mappings.

It seems straightforward to give a possible-worlds structure to model this system. The language needs names for each division, names for the possible machine identification numbers, and relative names "left" and "right" to capture the ring structure. Each possible world is simply some set of n agents (machines) together with a suitable interpretation of these names. This model is sufficient to interpret any *de dicto* knowledge we would wish to ascribe. For example, we can easily model knowledge or ignorance about the mapping between machine identification numbers and position in the ring. However, if we wish to talk about *de re* knowledge, something is missing: we need to say when two agents in different worlds are regarded as the *same* (consider again how we might interpret "machine a knows who his left-hand neighbor is"). Perhaps we identify machines across different possible worlds using the physical properties of the machine (essentially, the hard-wired identifier). Or it may be more useful to base this identification on the company division (perhaps we do not care that the hardware changes from time to time). There are other alternatives as well, and none has any automatic claim to being the single, correct answer. This is simply another decision we need to make as part of the modeling process, if we insist on using *de re* knowledge without explicitly saying how agents are being referred to.

We can summarize our main points. First, two agents in different possible worlds will *not* be identical in every respect, so there is no *a priori* correct way of verifying that two such agents are really the *same*. Thus, any claim that two agents are the same always involves use of some convention, some similarity criteria, and we must consciously decide what these criteria are. And second, any *de re* statement is making such claims about identity (the essence of *de re* is that something is known about an object, and it is implicit in this that the "same" object re-appears in different worlds). Therefore, *de re* only makes sense in conjunction with a *cross-world identity* criterion for agents or objects in general (that is, a means of determining when agents in different worlds are similar enough to be regarded as the "same"). If there are many reasonable criteria, there will by many different *de re* readings. This is precisely the same issue as the "ways of referring to an agent" we saw before. Instead of saying that we should choose a class of names to use, the *de re* viewpoint is that we need to specify the cross-world identity criteria. Classes of names are then best viewed as a way of encoding such criteria, within the model.

Of course, the question of what it means to say that two individual objects, in epistem-

ically distinct worlds, are the *same* is well known to be philosophically contentious.[10] Lewis [28] has said that in fact such objects are never the same: and that instead of identity, we should consider when one agent is the "counterpart" of another. Intuitively, we should have some (one) fixed criterion for similarity in mind, and two agents which are sufficiently close in this sense are counterparts (but definitely not identical). This is very similar to the position we are taking here and certainly compatible with our work: "counterpart" for Lewis is essentially "sharing a name" for us (although this makes most sense when we consider names that denote single agents rather than groups). Viewing things as Lewis does leads to interesting conclusions. For example, we should not expect distinct agents (in one world) to always have distinct counterparts elsewhere (something we cannot escape if "counterpart" is simply "identity").

Another particularly relevant work is that of Hintikka (for example, in [17, 18]). He looks at *world-lines* which are, essentially, like our *names* in that they refer to individual(s) in each possible world. (In fact, aside from the fact that our names are usually relative, there is no significant difference between a name and a world-line.) His interpretation of these lines is that they connect instances of the *same* individual, across the different worlds. In the formal model then, similarity (cross-world identity) is *defined* by the property of sharing a world line. There is no technical reason why these lines could not merge, split, or fail to denote anyone in some worlds, and Hintikka argues that these are useful possibilities (so there is a similar generality to Lewis's counterpart relations). The most interesting aspect of Hintikka's work for us is that he comments on the possibility of having two distinct sets of lines. That is, there may be two different ways of saying what it means for two agents (in different worlds) to be the same, and each scheme is useful. His application of this was to note that we may know about a person in a descriptive manner, but that we also sometimes identify objects in the same spatial location (relative to us, though he did not formalize this). Hintikka's work is continued by Thomason [45], who presents a logic for perception which allows for these two "modes of individuation" (i.e., two methods of cross-world identification, or "ways of referring"). It is straightforward to see how Thomason's work extends to other contexts (such as where the modality is knowledge, or where there are several other modes of individuation). Nevertheless, our logic differs from Thomason's in several significant ways. First, of course, we emphasize the possibility that there could be many different ways of referring to individuals, and give a single logic which can handle any number of these ways (in our work, the particular ways of referring and their properties are determined semantically by the collection of names in the model; in Thomason's, the language includes a new type of quantification for each possibility). Other major differences are in our use of knowledge about self-identity (because our modality was knowledge, not perception, and it seems that in epistemic contexts we frequently refer to others using relative names) and our treatment of quantifying-in (Thomason's approach seems closer to the free-logic treatment mentioned earlier).

[10] In fact, the question of whether we need cross-world identity criteria at all is controversial; [23] argues that we do not (although his concern is *necessity*, rather than knowledge, and perhaps the two cases are not directly comparable).

6. Conclusion

In this paper we have considered the significance of *naming* in epistemic logic for many agents. Our analysis has shown that a multi-agent logic should not ignore the complex question of how agents refer to each other; answering this question involves many issues that have no counterpart in more familiar theories (such as standard single-agent modal logics [4,16]). But even a single agent must refer to other objects in its environment somehow, and so most of the issues we consider should be relevant even in this case. General relative (i.e., indexical) reference (i.e., to all objects, and not necessarily just agents) is especially important in planning; consider, for instance, Agre and Chapman's Pengo system [1]. Fortunately it is not necessary to construct yet another logic for this. Almost any method for naming agents works for other objects as well, and so the logic developed in this paper needs essentially no modification.[11] In particular, any predicate or function with "**me**" as an argument is naturally relative. Lespérance's work is also very successful at this, and it is perhaps even more specifically suited to single-agent applications.

In part I we developed simple propositional logics that could express many useful varieties of names. In this paper we have identified weaknesses in these logics, in particular, the inability to handle scope and its consequences. These problems arise, to some extent, in single-agent environments, but the complexities are magnified enormously when we consider nested knowledge. In this paper we have concentrated on this, most general, case. The first-order logic we develop for this combines three powerful features: (1) we make a rigorous separation between agents and names, (2) we impose syntactic restrictions which ensure that agents only refer to each other by means of names or name-like constructs, and (3) we allow a large variety of names, including relative naming systems made possible by the semantics of knowledge about self-identity. The general philosophy, if not the actual details, behind each of these features individually is not entirely novel. But we believe that only combination of them all is effective and general.

A non-technical contribution of this paper is to demonstrate the considerable subtlety of issues involving naming. These can be quite difficult to reason about in the absence of any general theory or formal system. The primary contribution of this paper, and of the logic it presents, is to provide such a system. In any actual application there tend to be special features that reduce the need for such a powerful logic. In these cases, the far simpler logics and techniques developed in part I will be helpful. Our first-order logic can be used to give a formal specification of the assumptions made by these simpler logics.

We close with some additional comments on this work might be applied and extended. First, in [10] we consider the (small) changes that need to be made to cope with belief rather than knowledge. In [9,10] we move towards a more concrete application by examining a specific model of communicating agents. One goal of this work is to

[11] To name other objects in a similar fashion to agents, we use a new sort for each type of object (analogous to the *agents* sort), and a sort of corresponding names. The former can vary from world to world (as with agents) so long as we do quantify-in only via the appropriate names, just as in Section 4.

analyze the tight link between some names and modes of communication. The examples in Section 3.2 are one instance of this, in which each relative name corresponds to a specific communication channel.

It is, of course, desirable to show how any logic might be used as a practical reasoning tool. In our setting, we might imagine an agent being provided with a suitable epistemic logic and theorem-prover to reason about the world (and other agents) around him. We can also consider applications of epistemic logic as a verification technique for distributed systems. Both areas deserve further work. However, note that because the first-order logic in this paper incorporates classical logic, and has an axiomatization, its complexity is exactly the same as classical logic (i.e., semi-decidable). While progress in general theorem-proving continues to advance, the possibility of finding simpler logics (with faster decision procedures) is probably the more promising direction. As we mention above, the real purpose of the general logic would then be to provide a unifying framework for interpreting the restrictions which must inevitably be made to gain practicality.

Acknowledgments

I wish to thank Joseph Halpern for his invaluable advice and assistance. I am also grateful to David Israel, Daphne Koller, Yves Lespérance, Hector Levesque, Yoav Shoham, and Moshe Vardi, for their comments on earlier versions of this paper.

I was supported by an IBM graduate fellowship throughout much of this research.

Appendix A. Proof of Theorem 4.1

Theorem 4.1 states that the axiom system consisting of (A1), (Q1), (I1), (I2), (I3), (I4), (M1), (M2), (M3), (M4), (R1), (R2), and (R3) is sound and complete with respect to the class of all first-order possible-worlds structures with knowledge about self-identity.

The proof of completeness uses several quite standard techniques (see, for example, [8]). The novel features are due to knowledge about self-identity and the restriction we place on quantification for sorts other than *names*. We concentrate on the implications of these points.

Following [8], we say a set of sentences V is *ω-complete* if, whenever $V \cup \{\neg\forall x\, \varphi\}$ is consistent, there is some variable y such that y is substitutable for x in φ and $V \cup \{\neg\varphi[y/x]\}$ is consistent. (One difference between our proof and those in [8] is that we require this latter condition only for variables y, not for general terms. This, and similar differences which occur later, arise because we do not assume that terms are rigid.) If V is maximal consistent as well as ω-complete, we say it is *saturated*. Let $\mathcal{V}$ be the collection of all saturated sets. For $V \in \mathcal{V}$, let $Obj(V)$ be the subset of all formulas in V in which all occurrences of the symbol **me** are inside the scope of some K operator. Some properties of ω-complete sets are: (1) If V is ω-complete, so is $V \cup f$ for any finite set of formulas f, and (2) If V is ω-complete and consistent, it has

a saturated extension. For a proof of these, see [8]; very little modification is required for our context.

We define our model $M = (W, \mathcal{K}, \pi)$ as follows. First, W will be a set of equivalence classes of $\mathcal{V}$, where the partition is according to the relation $V \equiv V'$ iff $Obj(V) = Obj(V')$.

For $w \in W$ we define $\pi(w)$ by looking at $Obj(w)$ (where $Obj(w)$ equals $Obj(V)$ for any $V \in w$). The objects of sort s are equivalence classes of variables of sort s, where the partition is according to $x \equiv x'$ iff $x = x' \in Obj(v)$. (It is a consequence of (I1) and (I2) this relation $\equiv$ is indeed an equivalence). $P^w(x_1 \ldots, x_n)$ is true just if $P(x_1, \ldots, x_n) \in Obj(w)$. Note that (I2) ensures that this is well defined. Next, $f^w(x_1 \ldots, x_n)$ is equal to the equivalence class corresponding to y just if $f(x_1, \ldots, x_n) = y \in Obj(w)$. Because of (I2) this class is unique if it exists. Furthermore, existence follows from the provability of $\exists x\ x = f(x_1, \ldots, x_n)$ (using (I1), (Q1), (R3)) and the saturation of $V \in w$.

Finally, we define $\mathcal{K}$. Before we do this, we need to look at a property of the set of agents existing at world w. Let a be an agent in w. Recall that this means a is really an equivalence class of variables of sort *agent*. It turns out that there is a unique saturated set $V_{a,w} \in w$ such that $x = \mathbf{me} \in V_{a,w}$ for all $x \in a$. Uniqueness follows from (I2) and our construction which ensures that all $V \in w$ agree on objective formulas. For existence, consider $Obj(w) \cup \{x = \mathbf{me}\}$ for some $x \in a$. This is consistent. (For suppose there is some objective $\varphi \in Obj(w)$ with $\varphi \Rightarrow x \neq \mathbf{me}$ provable. We can argue, by induction on the length of proofs, that $\varphi \Rightarrow y \neq x$ is also provable for some new variable y, and from this a contradiction will follow by (R3) and (I1). In other words, the inductive proof shows that for any proof of $\varphi \Rightarrow x \neq \mathbf{me}$ there is a proof of $\varphi \Rightarrow y \neq x$; each line of the proof has all replacing "outside" occurrences of **me** by y. It is necessary to verify that for each inference step in the original proof, the analogous step in the new proof is also sanctioned; we omit further details.) As well as being consistent, $Obj(w) \cup \{x = \mathbf{me}\}$ is ω-complete. (It is sufficient to show that $Obj(w)$ is ω-complete, because adding one formula preserves ω-completeness. Suppose $Obj(w) \vdash \varphi[y/x]$ for all substitutable variables y. The case where φ is objective is easy: by the saturation of each $V \in w$, we know that $(\forall x\ \varphi) \in V$, but this is objective so is contained in $Obj(w)$. Otherwise, we use a similar argument as with consistency to show that $Obj(w) \vdash \varphi[z/\mathbf{me}][y/x]$ for some new variable z. This new sentences is objective so we can apply the previous argument, and then re-substitute **me** for z.) Therefore, $Obj(w) \cup \{x = \mathbf{me}\}$ has a saturated extension which can serve as $V_{a,w}$. Conversely, look at $V \in w$. There must be some variable y such that $y = \mathbf{me} \in V$ (due to the provability of $\exists x\ x = \mathbf{me}$ and saturation of V). That is, $V = V_{a,w}$ for some agent a. We have shown that there is a bijection between elements of sort *agent* at worlds w and the sets of formulas $V \in w$. Relative to any particular world w, we can regard an agent in w either as a set of variables or as a saturated set of formulas. In the following we regard agents as both, the particular usage being determined by context. Note that when we identify an agent with a saturated set, the corresponding world w is implicit (being identified by the objective formulas in the set corresponding to the agent).

Given this, we can define $\mathcal{K}$ as follows: $((w,a),(w',a')) \in \mathcal{K}$ just if $\{\varphi : K_{\mathbf{me}}\varphi \in a\} \subseteq a'$. It is a consequence of (M2), (M3), and (M4) that this re-

lation is reflexive, transitive, and symmetric as required.

To verify that this is a correctly defined first-order possible-worlds structure with knowledge about self-identity, it only remains for us to check that the domain of names is the same in every world. This is actually not quite true as things stand. Rather, it follows from (I3), and (I4) that, for any world w, all reachable worlds have the same domain for sort *names*. (Here, we say w' is *reachable* from w if there exists worlds $w_0 = w, w_1, w_2, \ldots, w_n = w'$ and agents $a_0, a_1, a_2, \ldots, a_n$ such that $((w_i, a_i), (w_{i+1}, a_{i+1})) \in \mathcal{K}$ for $i < n$.) Our ultimate goal is to show that any consistent set of sentences is true at some world w in M. As we do this, it will be clear that it is sufficient to consider only those worlds that are accessible from w. So if, having found a suitable w, we slightly redefine the model so as to consist of the accessible worlds only, then the constructed model will have a constant domain of names as required.

To complete the proof, we need to show a Truth Lemma: namely that, for some variable valuation v, then for all worlds and agents (w, a), we have $M, v, w, a \models \varphi$ if and only if $\varphi \in a$. For v we make the obvious definition, which is that $v(x)$ is (the equivalence class of) x itself. In the following, we suppress mention of M, v, and w whenever these are clear from the context.

This Truth Lemma is proved as usual, by induction on the structure of formulas. For atomic formulas, including formulas with equality, the truth lemma follows almost directly from the construction. Similarly, the case of the Boolean connectives is straightforward. Next, suppose $\forall x\, \varphi \in a$. By (Q1), $\varphi[y/x] \in a$ for all substitutable variables y of the appropriate sort, and using this and the inductive hypothesis we can show that, at a, φ is true of all objects in the domain of that sort. Conversely, suppose $\neg\forall x\, \varphi \in a$. By the saturation of a, there is some y such that $\neg\varphi[y/x] \in a$. Therefore, there is some object (in fact, the equivalence class of y) where φ is false in a, and thus $\neg\forall x\, \varphi$ is true at a as required.

The difficult case is for formulas $K_t\varphi$, where t is a term of *agent* sort. One direction is quite easy. Suppose $K_t\varphi \in a$, and that the denotation of t at a is agent b. That is, for some variable y in the equivalence class which is agent b, we have $(y = t) \in a$. But, by (I2), we conclude that $K_y\varphi \in a$ also. This is objective, so $K_y\varphi \in b$ also (where we are now regarding b as a saturated set). But we also know that $\mathbf{me} = y \in b$ (this is a consequence of the way we constructed the bijection between the two ways of viewing agents). By (I2) again, $K_{\mathbf{me}}\varphi \in b$. Finally, by the construction of $\mathcal{K}$, and induction, we see that agent b does indeed know φ.

Now consider the converse, the case where $\neg K_t\varphi \in a$. Arguing as above, we see that $\neg K_{\mathbf{me}}\varphi \in b$. We need to find an agent b' (i.e., a saturated set) that b thinks possible, such that b' does not know φ. To do this, it is sufficient to prove that $\Psi = \{\psi : K_{\mathbf{me}}\psi \in b\} \cup \{\neg\varphi\}$ has a saturated extension. First, note that this set is consistent (this follows from (M1), (R2) using the usual argument for normal modal logics.) Since any consistent ω-complete set has a saturated extension, we now only need to show that Ψ is ω-complete. The argument for this differs between the sort *names* and the other sorts.

First, suppose $\Psi \cup \{\neg\forall x\, \varphi\}$ is consistent, where x is not of sort *names*. We need to show that there is some variable y substitutable for x in φ such that $\Psi \cup \{\neg\varphi[y/x]\}$ is consistent. But in fact, we can chose any variable y other than x or those appearing

in φ (recall that in our logic, there are infinitely many variables of each sort, so some such y can certainly be chosen). For observe that every sentence ψ in Ψ must be closed (except possibly for free variables of sort *name*, which do not concern us here), because otherwise $K_{\mathbf{me}}\psi$ would not be well formed, and this is impossible given the way we defined Ψ. Using this, it is quite easy to show that $\Psi \cup \{\neg\varphi[y/x]\}$ is consistent.

Next, suppose $\Psi \cup \{\neg\forall x\, \varphi\}$ is consistent, for *name* variable x. For the case of *names*, we note that it is sufficient to show that $\Psi' = \{\psi : K_{\mathbf{me}}\psi \in b\}$ is ω-complete (because adding one additional formula never destroys ω-completeness). Suppose this is not the case, so that there is a formula ψ with $\Psi' \cup \{\neg\forall x\, \psi\}$ consistent, yet $\Psi' \cup \neg\psi[y/x]$ is inconsistent for all substitutable variables y. That is, for any such variable y, there are $\psi'_1, \ldots, \psi'_n$ with $\psi'_1 \wedge \ldots \wedge \psi'_n \Rightarrow \psi[y/x]$ provable. But then (R2), (M1), and the observation that $K_{\mathbf{me}}\psi'_i \in b$ (by the definition of Ψ') can be used to conclude that $K_{\mathbf{me}}\psi[y/x] \in b$. This is true for all y, and since b is saturated, we have $\forall x\, K_{\mathbf{me}}\psi \in b$. However, it is provable (using (M1), (M2), (M4), (M2), (Q1), (R3)) that $(\forall x\, K_{\mathbf{me}}\psi) \Rightarrow K_{\mathbf{me}}(\forall x\, \psi)$ is a theorem in our logic, for name variables x. (See [19, p. 145] for a proof. This formula is known as the *Barcan* formula.) From this, we see that $(\forall x\, \psi) \in \Psi'$, which is contrary to the consistency of Ψ' (which follows from the consistency of b). □

References

[1] P. Agre and D. Chapman, Pengi: an implementation of a theory of activity, in: *Proceedings AAAI-87* (1987) 282–272.

[2] T. Burge, Belief *de re*, *J. Philos.* **64** (6) (1977) 338–362.

[3] K.M. Chandy and J. Misra, How processes learn, *Distributed Computing*, **1** (1) (1986) 40–52.

[4] B.F. Chellas, *Modal Logic* (Cambridge University Press, Cambridge, UK, 1980).

[5] D.C. Dennett, Beyond belief, in: *The Intentional Stance* (MIT Press, Cambridge, MA, 1987).

[6] C. Dwork and Y. Moses, Knowledge and common knowledge in a Byzantine environment: crash failures, *Inform. Comput.* **88** (2) (1990) 156–186.

[7] M. Fitting, Modal logic should say more than it does, in: J.-L. Lassez and G. Plotkin, eds., *Computational Logic, Essays in Honor of Alan Robinson* (MIT Press, Cambridge, MA, 1991) 113–135.

[8] J.W. Garson, Quantification in modal logic, in: D. Gabbay and F. Guenthner, eds., *Handbook of Philosophical Logic, Vol. II* (Reidel, Dordrecht, Netherlands, 1977) 249–307.

[9] A.J. Grove, Semantics for knowledge and communication, in: B. Nebel, C. Rich, and W. Swartout, eds., *Third International Conference on Principles of Knowledge Representation and Reasoning (KR '92)*, Cambridge, MA (1992) 213–224.

[10] A.J. Grove, Topics in multi-agent logics, Ph.D. Thesis, Stanford University, Stanford, CA (1992).

[11] A.J. Grove and J.Y. Halpern, Naming and identity in propositional logics, Part I: the propositional case, *J. Logic Comput.* **3** (4) (1993) 345–378. A preliminary version appeared in: *Proceedings Second International Conference on Principles of Knowledge Representation and Reasoning (KR'91)*.

[12] J.Y. Halpern, Using reasoning about knowledge to analyze distributed systems, in: J.F. Traub, B.J. Grosz, B.W. Lampson and N.J. Nilsson, eds., *Ann. Rev. Comput. Sci.* **2** Annual Reviews Inc., Palo Alto, CA (1987) 37–68.

[13] J.Y. Halpern and Y. Moses, Knowledge and common knowledge in a distributed environment, *J. ACM* 37 (3) (1990) 549–587. A preliminary version appeared in: *Proceedings Third ACM Symposium on Principles of Distributed Computing* (1984).

[14] J.Y. Halpern and Y. Moses, A guide to completeness and complexity for modal logics of knowledge and belief, *Artif. Intell.* **54** (1992) 319–379.

[15] J.Y. Halpern and L.D. Zuck, A little knowledge goes a long way: knowledge-based derivations and correctness proofs for a family of protocols, *J. ACM* **39** (3) (1992) 449–478.
[16] J. Hintikka, *Knowledge and Belief* (Cornell University Press, Ithaca, NY, 1962).
[17] J. Hintikka, Semantics for propositional attitudes, in: *Models for Modalities* (Reidel, Dordrecht, Netherlands, 1969).
[18] J. Hintikka, Reasoning about knowledge in philosophy: the paradigm of epistemic logic, in: J.Y. Halpern, ed., *Proceedings Conference on Theoretical Aspects of Reasoning about Knowledge*, Monterey, CA (1986).
[19] G.E. Hughes and M.J. Cresswell, *An Introduction to Modal Logic* (Methuen, London, 1968).
[20] G.E. Hughes and M.J. Cresswell, *A Companion to Modal Logic* (Methuen, London, 1984).
[21] D. Kaplan, Quantifying in, *Synthese* **19** (1969) 178–214.
[22] D. Kaplan, Demonstratives, in: J. Almog, J. Perry, and H.K. Wettstein, eds., *Themes from Kaplan* (Oxford University Press, New York, 1989).
[23] S.A. Kripke, Naming and necessity, in: D. Davidson and G. Harman, eds., *Semantics of Natural Language* (Reidel, Dordrecht, Netherlands, 1972) 253–355.
[24] W. Lenzen, Recent work in epistemic logic, *Acta Philos. Fenn.* **30** (1978) 1–219.
[25] Y. Lespérance, A formal account of self-knowledge and action, in: *Proceedings IJCAI-89*, Detroit, MI (1989) 868–874.
[26] Y. Lespérance, A formal theory of indexical knowledge and action, Ph.D. Thesis, University of Toronto, Toronto, Ont. (1991).
[27] H.J. Levesque, Foundations of a functional approach to knowledge representation, *Artif. Intell.* **23** (1984) 155–212.
[28] D. Lewis, Counterpart theory and quantified modal logic, *J. Philos.* **65** (5) (1968) 113–126.
[29] D. Lewis, Attitudes *de dicto* and *de se*, *Philos. Rev.* **88** (4) (1979) 513–543.
[30] M.S. Mazer and F.H. Lochovsky, Analyzing distributed commitment by reasoning about knowledge, Tech. Report CRL 90/10, DEC-CRL (1990).
[31] R.C. Moore, A formal theory of knowledge and action, in: J. Hobbs and R.C. Moore, eds., *Formal Theories of the Commonsense World* (Ablex, Norwood, NJ, 1985) 319–358.
[32] Y. Moses and G. Roth, On reliable message diffusion, in: *Proceedings Eighth ACM Symposium on Principles of Distributed Computing* (1989) 119–128.
[33] Y. Moses and M.R. Tuttle, Programming simultaneous actions using common knowledge, *Algorithmica* **3** (1988) 121–169.
[34] G. Neiger and S. Toueg, Simulating real-time clocks and common knowledge in distributed systems, *J. ACM* **40** (2) (1993) 334–367.
[35] J. Perry, The problem of the essential indexical, *Noûs* **13** (1979) 3–21.
[36] A. Plantinga, *The Nature of Necessity* (Oxford University Press, Oxford, UK, 1974).
[37] W.V. Quine, Quantifiers and propositional attitudes, in: *The Ways of Paradox, and Other Essays* (Random House, 1966).
[38] W.V. Quine, Three grades of modal involvement, in: *The Ways of Paradox, and other essays* (Random House, 1966).
[39] S.J. Rosenschein and L.P. Kaelbling, The synthesis of digital machines with provable epistemic properties, in: J.Y. Halpern, ed., *Proceedings Conference on Theoretical Aspects of Reasoning about Knowledge*, Monterey, CA (1986) 83–97.
[40] G. Roth, Message diffusion in anonymous distributed systems, Master's Thesis, Weizmann Institute of Science (1989).
[41] B. Russell, On denoting, *Mind, N.S.* (1905).
[42] B. Russell, *Introduction to Mathematical Philosophy* (Allen & Unwin, London, 1919).
[43] N. Salmon, Reference and information content, names and descriptions, in: D. Gabbay and F. Guenthner, eds., *Handbook of Philosophical Logic. Vol. IV* (Reidel, Dordrecht, Netherlands, 1989).
[44] J.R. Searle, *Intentionality* (Cambridge University Press, Cambridge, 1983).
[45] R.H. Thomason, Perception and individuation, in: M. Munitz, ed., *Logic and Ontology* (New York University Press, New York, 1973) 261–285.