UDC: 007.52
Original scientific paper

MULTI-AGENT SYSTEMS: INCORPORATING KNOWLEDGE AND TIME

Mirko Maleković
University of Zagreb, Faculty of Organization and Informatics Varaždin, Croatia
E-mail: mmalekov@foi.hr

We consider incorporating knowledge and time in multi-agent systems. Five temporal operators o, ♥, ♦, *U, W are decribed. The following facts are proved: (a) for all formulas F in LK (propositional logic + knowledge operator K) if states s and s1 are equal, then F holds in s iff F holds in s1, (b) the same result does not hold in LKT (LK + the temporal operators). Finally, we characterize two propositions that state when the formulas* Ki(F) ⇒ ♦Kj(F) and Ki(F) ⇒ ♥Kj(F) *hold.*

Keywords: knowledge bases, knowledge theory, multi-agent systems, temporal operators, reasoning about knowledge.

1. INTRODUCTION

The idea of formal logical analysis of multi-agent systems is described in [1], [2], [3], [4], [5] and [6]. A very interesting result that states that a knowledge base can be modeled as a multi-agent system is given in [1].

In this paper, we shall characterize incorporating knowledge and time in multi-agent systems. Some new results regarding the properties of temporal operators and knowledge operators will be proved.

The paper consists of five sections and an Appendix containing some proofs. In Section 2, we introduce the basic notions of multi-agent systems. In Section 3, we characterize in detail incorporating knowledge and time in multi-agent systems. Section 4 contains the proofs of Proposition (Base): all the basic temporal operators can be defined in terms of the operators o and *U*, and Proposition (♥♦):♥♦F holds iff F holds infinitely often, and ♦♥F holds iff F holds almost everywhere. Conclusions are given in Section 5. The Appendix contains the proofs of Proposition (LK): for all formulas F in LK we have if states s and s1 are equal, then F holds in s iff F holds in s1; Proposition (LKT): the result in Proposition (LK) does not hold in LKT; Proposition (Ki ⇒ ♦Kj): if agent i knows F, then agent j eventually knows F under the condition that some premise U (defined later) holds; and Proposition (Ki ⇒ ♥Kj): if agent i knows F, then agent j always knows F under the condition that some premise U1 (defined later) hold

2. BASIC NOTIONS

In this section, we introduce the basic concepts and notations.

Suppose we have a group consisting of m agents, named 1, 2,.., m . An agent may be a man (a real agent), a software module or a communicating robot (an artificial agent). An agent may even be a component of a computer system (a wire or a message buffer). We assume these agents wish to reason about a world that can be described in terms of a nonempty set P of primitive propositions. A language is just a set of formulas, where the set of formulas LK of interest to us is defined as follows:

(1) The primitive propositions in P are formulas;

(2) If F and G are formulas, then so are $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \Rightarrow G)$, $(F \Leftrightarrow G)$, and Ki(F) for all $i \in \{1, 2, .., m\}$, where Ki is a modal operator.

A Kripke structure M for an agent group {1, 2,.., m} over P is a (m + 2)-tuple

M = (S, I, k1, k2,.., km), where S is a set of possible worlds, I is an interpretation that associates with each world in S a truth assignment to the primitive propositions in P, and k1, k2,.., km are binary relations on S, called the possibility relations for agents 1, 2,.., m, respectively.

Given $p \in P$, the expression I[w](p) = true means that p is true in a world w in a structure M. The fact that p is false, in a world v of a structure M, is indicated by the expression I[v](p) = false.

The expression $(u, v) \in ki$ means that an agent i considers a world v possible, given his information in a world u. Since ki defines what worlds an agent i considers possible in any given world, ki will be called the possibility relation of the agent i.

We now define what it means for a formula to be true at a given world in a structure.

Let $(M, w) \models F$ mean that F holds or is true at (M, w). Definition of $\models$ is as follows:

(a) $(M, w) \models p$ iff I[w](p) = true, where $p \in P$;
(b) $(M, w) \models F \wedge G$ iff $(M, w) \models F$ and $(M, w) \models G$;
(c) $(M, w) \models F \vee G$ iff $(M, w) \models F$ or $(M, w) \models G$;
(d) $(M, w) \models F \Rightarrow G$ iff $(M, w) \models F$ implies $(M, w) \models G$;
(e) $(M, w) \models F \Leftrightarrow G$ iff $(M, w) \models F \Rightarrow G$ and $(M, w) \models G \Rightarrow F$;
(f) $(M, w) \models \neg F$ iff $(M,w) \not\models F$, that is, $(M, w) \models F$ does not hold;
(g) $M \models F$ iff $(M, w) \models F$ for all $w \in S$.

Finally, we shall define a modal operator Ki, where Ki(F) is read: Agent i knows F.

(h) $(M, w) \models Ki(F)$ iff $(M, t) \models F$ for all $t \in S$ such that $(w, t) \in ki$.

In (h) we have that an agent i knows F in a world w of a structure M exactly if F holds at all worlds t that the agent i considers possible in w.

Multi-Agent Systems

A multi-agent system is any collection of interacting agents. Our key assumption is that if we look at the system at any point in time, each of the agents is in some state. We refer to this as the agent's local state. We assume that an agent's local state encapsulates all the information to which the agent has access. As each agent has a local state, it is very natural to think of the whole system as being in some (global) state. The global state includes the local states of the agents and the local state of an environment. Accordingly, we divide a system into two components: the environment and the agents, where we view the environment as everything else that is relevant. Also, the environment can be viewed as just another agent. We need to say that a given system can be modeled in many ways. How to divide the system into agents and environment depends on the system being analyzed.

Let Le be a set of possible local states for the environment and let Li be a set of possible local states for agent i , $i = 1, .., n$. We define $G = Le \times L1 \times .. \times Ln$ to be the set of global states. A global state describes the system at a given point in time. Since a system constantly changes (it is not a static entity), we are interested in how these systems change over time. We take time to range over the natural numbers, that is, the time domain is the set of the natural numbers, N.

A run over G is a function $r : N \rightarrow G$.

Thus, a run over G can be identified by a sequence of global states in G . The run r represents a complete description of how the system's global state evolves over time. Thus, r(0) describes the initial global state of the system in a possible execution, r(1) describes the next global state, and so on.

If r(m) = (se, s1,.., sn), then we define r[e](m)=se and r[i](m)=si , for i=1,.., n.

Note that r[i](m) = si is the local state of the agent i at the (global) state r(m).

A system R over G is a set of runs over G. The system R models the possible behaviors of the system being modeled.

Knowledge in Multi-Agent Systems

We assume that we have a set P of primitive propositions, which we can think of as describing basic facts about a system R. Let I be an interpretation for the propositions in P over G , which assigns truth values to the primitive propositions at the global states. Thus, for every $p \in P$ and $s \in G$, $I[s](p) \in \{true, false\}$. An interpreted system IS is a pair (R, I).

Now, we define knowledge in an interpreted system IS.

Let IS = (R, I) be an interpreted system. A Kripke structure for IS, denoted

M(IS) = (S, I, k1, .., kn), is defined in a straightforward way.

$S = \{r(m) \mid r \in R, m \in N\}$, that is, S is the set of the global states at the points (r,m) in the system R.

The possibility relations k1, k2,.., kn are defined as follows.

Let $r(m) = (se, s1,.., sn)$, $r'(m') = (se', s1',.., sn')$ be two global states in S. We say that r(m) and r'(m') are indistinguishable to an agent i iff $si = si'$.

Thus, the agent i has the same local state in both r(m) and r'(m'). We define

$ki = \{(r(m), r'(m')) \in S \times S \mid r(m)$ and $r'(m')$ are indistinguishable to the agent $i\}$, $i = 1, 2,.., n$.

Accordingly, $(r(m), r'(m')) \in ki$ iff $si = si'$, $i = 1, 2,.., n$.

There is no possibility relation ke for the environment because we are not usually interested in what the environment knows.

Now, it is evident what it means for a formula F in LK to be true at a state r(m) in an interpreted system IS. For instance, we have

$(IS, r(m)) \models p$ iff $I[r(m)](p) = true$, for all $p \in P$.

$(IS, r(m)) \models Ki(F)$ iff $(IS, r'(m')) \models F$ for all $r'(m') \in S$ such that $(r(m), r'(m')) \in ki$.

We say that a formula F in LK is valid in an interpreted system IS, denoted $IS \models F$, iff

$(IS, r(m)) \models F$ for all $r(m) \in S$.

Let us note that we do not assume that the agents compute their knowledge in any way, or that they can necessarily answer questions based on their knowledge. We interpret knowledge as an external one, ascribed to the agents by someone reasoning about the system.

To be able to make temporal statements, we extend our language LK by adding temporal operators, which are new modal operators for talking about time. This language will be denoted by LKT., and will be used for reasoning about events that happen along a single run r in the system R.

We define here five temporal operators: o (next time), ♥(always), ♦ (eventually), *U* (until), and *W* (waiting-for, or unless).

The Next Operator o
oF, read next F, is defined by $(IS, r(m)) \models oF$ iff $(IS, r(m+1)) \models F$.

Thus, oF holds at state r(m) iff F holds at the next state r(m+1).

The Always Operator ♥
♥F, read always F, is defined by $(IS, r(m)) \models$ ♥F iff $(IS, r(m')) \models F$ for all $m' \geq m$.

Accordingly, ♥F holds at state r(m) iff F holds at state r(m) (now) and at all later states.

The Eventually Operator ♦
♦F, read eventually F, is defined by (IS, r(m)) $\models$ ♦F iff (IS, r(m')) $\models$ F for some $m' \geq m$.

Thus, ♦F holds at state r(m) iff F holds at state r(m) or some state in the future.

The Until Operator U
F U F1, read F until F1, is defined by (IS, r(m)) $\models$ F U F1 iff
(IS, r(m')) $\models$ F1 for some $m' \geq m$ and (IS, r(m'')) $\models$ F for all m'' with $m \leq m'' < m'$.

The until formula F U F1 predicts the eventual occurrence of F1 and states that F holds continuously at least until the first occurrence of F1.

The Unless (Waiting-for) Operator W
F W F1, read F unless F1, has the following semantics.
(IS, r(m)) $\models$ F W F1 iff (IS, r(m)) $\models$ F U F1 or (IS, r(m)) $\models$ ♥F.

Thus, the formula F W F1 expresses the property that F holds continuously either until the next occurrence of F1 or throughout the sequence of states.

Note that our interpretation of oF makes sense because our notion of time is discrete. All the other temporal operators make perfect sense even for continuous notions of time.

2. SOME PROPERTIES OF LKT-FORMULAS

We have defined the five temporal operators: o, ♥, ♦, U, and W. In the following proposition we shall show that we can take o and U as our basic temporal operators, and define♥, ♦, and W in terms of U.

Proposition (Base)
We have
(1) IS $\models$ ♥F $\Leftrightarrow \neg$♦$\neg$F   (2) IS $\models$ ♦F $\Leftrightarrow$ True U F .

Proof (1)

Let $r(m) \in S$ be an arbitrary state.We would like to show (IS, r(m))$\models$♥F$\Leftrightarrow\neg$♦$\neg$F .

Because (IS, r(m)) $\models$ ♥F iff (for all $m' \geq m$) [(IS, r(m')) $\models$ F] iff

(for all $m' \geq m$)[(IS, r(m')) $\not\models \neg$F] iff (IS, r(m)) $\not\models$♦$\neg$F iff (IS, r(m)) $\models \neg$♦$\neg$F, we have (IS, r(m)) $\models$ ♥F $\Leftrightarrow \neg$♦$\neg$F, as desire

Proof (2)

Because IS ⊨ True , we have (for all m ∈ N)[(IS, r(m)) ⊨ True]. Now, we proceed as follows.

Let r(m) be an arbitrary state in S. We need to prove (IS, r(m)) ⊨♦F⇔ True *U* F.

Because (IS, r(m)) ⊨ ♦F iff (for some m' ≥ m)[(IS, r(m')) ⊨ F] iff (for some m' ≥ m)[(IS, r(m')) ⊨ F] and (for all m'' with m ≤ m'' < m')[(IS, r(m'') ⊨ True] iff (IS, r(m)) ⊨ True *U* F.
Consequently, (IS, r(m)) ⊨ ♦F ⇔ True *U* F.

> **Proposition(♥♦)**
> We have
> (1) (IS, r(m)) ⊨ ♥♦F iff the set {m' | (IS, r(m')) ⊨ F} is infinite.
>
> (2) (IS, r(m)) ⊨ ♦♥F iff (for some m')(for all m'' ≥ m')[(IS, r(m'')) ⊨ F]

Proposition (♥♦) says that ♥♦F holds iff F holds infinitely often, and ♦♥F holds iff F holds almost everywhere.

Proof (1)

(IS, r(m)) ⊨ ♥♦F iff (for all m' ≥ m)[(IS, r(m')) ⊨ ♦F] iff (for all m' ≥ m)(for some m'' ≥ m')[(IS, r(m'')) ⊨ F] iff {m'' | (IS, r(m'')) ⊨ F} is infinite.

Proof (2)

(IS, r(m)) ⊨ ♦♥F iff (for some m' ≥ m)[(IS, r(m')) ⊨ ♥F] iff (for some m' ≥ m)(for all m'' ≥ m')[(IS, r(m'') ⊨ F], that is, (IS,r(m)) ⊨ ♦♥F iff F holds almost everywhere.

We can see that the temporal operators defined talk about events that happen only in the present or the future, not events that have happened in the past. These operators suffice for many applications, but it is not a problem to define temporal operators for reasoning about the past. The past temporal operators will be considered in a forthcoming paper.

In the following proposition, we state that if r(m) and r'(m') are equal states, then F holds in r(m) iff F holds in r'(m'). The result is true for every formula F in LK.

> **Proposition (LK)**
> Let IS = (R, I) be an interpreted system. Then
> (for all r(m), r'(m') ∈ S)(for all F ∈ LK)[r(m) = r'(m') ⇒ (IS, r(m)) ⊨ F ⇔ (IS, r'(m')) ⊨ F].

Proposition (LK) does not hold in LKT. To show that, we shall construct (in the Appendix) an interpreted system IS = (R, I) such that r(m), r'(m') $\in$ S and r(m) = r'(m'), but (IS, r(m)) $\models$ ♦p and (IS, r'(m')) $\models \neg$♦p for some proposition p $\in$ P.

Proposition (LKT)
Proposition (LK) does not hold in LKT.

Now, we shall consider some of the important agent properties. These properties relate the knowledge of two agents, and are important if we wish to introduce an order in the set of agents. Understandably, this order can be very helpful when we analyze the respective multi-agent system.

In the following propositions, we shall use a set S[j, r](m') defined by S[j, r](m') = {ri(mi) | (r(m'), ri(mi)) $\in$ kj}. Thus, S[j, r](m') is the set of the states in S that agent j considers possible in the state r(m').

Proposition (Ki $\Rightarrow$ ♦Kj)
If U: (for some m' $\geq$ m)[r(m) $\times$ S[j, r](m') $\subseteq$ ki], then
(for all F $\in$ LKT)[(IS, r(m)) $\models$ Ki(F) $\Rightarrow$ ♦Kj(F)].

Proposition (Ki $\Rightarrow$ ♦Kj) states that if agent i knows F, then agent j eventually knows F, under the condition that U holds.

Proposition (Ki $\Rightarrow$ ♥Kj)
If U1: (for all m' $\geq$ m)[r(m) $\times$ S[j, r](m') $\subseteq$ ki], then
(for all F $\in$ LKT)[(IS, r(m)) $\models$ Ki(F) $\Rightarrow$ ♥Kj(F)].

Accordingly, agent j always knows F if agent i knows F, under the condition that U1 holds.

3. CONCLUSIONS

We have described incorporating knowledge and time in multi-agent systems. We have proved

Proposition (Base): all the basic temporal operators can be defined in terms of the operators o and U, and proposition (♥♦):♥♦F holds iff F holds infinitely often, and ♦♥F holds iff F holds almost everywhere. The proofs of Proposition (LK): for all formulas F in LK we have if states s and s1 are equal, then F holds in s iff F holds in s1; and Proposition (LKT): the result in proposition (LK) does not hold in LKT are given in the Appendix. Also in the Appendix, we have given the proofs of Proposition (Ki $\Rightarrow$ ♦Kj): if agent i knows F, then agent j eventually knows F under the condition that the premise U holds; and Proposition ((Ki $\Rightarrow$ ♥Kj): if agent i knows F, then agent j always knows F under the condition that the premise U1 holds.

Because the theory of multi-agent systems is a very important formal tool for describing and analyzing real systems, in a forthcoming paper we shall investigate the language LKT extended with the past temporal operators. Also an open problem remains: how to characterize a graphical representation of the language LKT ?

APPENDIX

Proof (Proposition (LK))

The fact is evident for a Boolean combination of propositions. Now, let F have the form F = Ki(G). We have (IS, r(m)) $\models$ Ki(G) iff (for all ri(mi) $\in$ S)[(r(m), ri(mi)) $\in$ ki $\Rightarrow$ (IS, ri(mi)) $\models$ G] iff (for all ri(mi) $\in$ S)[(r'(m'), ri(mi)) $\in$ ki $\Rightarrow$ (IS, ri(mi)) $\models$ G] iff (IS, r'(m')) $\models$ Ki(G), as desired.

Proof (Proposition (LKT))

Let r and r' be two runs as follows.

r: r(0), r(1),.., r(k),... r': r'(0), r'(1),.., r'(k),...; where r(0) = r'(0).

Next, we define the interpretation I of IS like this:

I[r(0)](p) = true and I[r'(m')](p) = false for all r'(m'). Consequently, we have (IS, r(0)) $\models$ ♦p and (IS, r'(0)) $\models$ ¬♦p

Proof (Proposition (Ki $\Rightarrow$ ♦Kj)

Assume U and V:(IS, r(m))$\models$ Ki(F). We would like to show (IS, r(m)) $\models$ ♦Kj(F).

From the assumption V, we have

(for all ri(mi) $\in$ S)[(r(m), ri(mi)) $\in$ ki $\Rightarrow$ (IS, ri(mi)) $\models$ F]. Let m' $\geq$ m be such a point that r(m) $\times$ S[j, r](m') $\subseteq$ ki. We shall prove (IS, r(m')) $\models$ Kj(F).

Let rj(mj) $\in$ S be an arbitrary state such that (r(m'), rj(mj)) $\in$ kj. It follows

rj(mj) $\in$ S[j, r](m'). Thus, from V we have (r(m), rj(mj)) $\in$ ki. Therefore, (IS, rj(mj)) $\models$ F, that is, (IS, r(m')) $\models$ Kj(F). Accordingly, we have (IS, r(m)) $\models$ ♦Kj(F), as desired.

Proof (Proposition (Ki $\Rightarrow$ ♥Kj))

Assume U1 and V1: (IS, r(m)) $\models$ Ki(F). We need to show (IS, r(m)) $\models$ ♥Kj(F).

From V1 we obtain (for all ri(mi) $\in$ S)[(r(m), ri(mi)) $\in$ ki $\Rightarrow$ (IS, ri(mi)) $\models$ F].

Let m' $\geq$ m be an arbitrary point. We have to prove (IS, r(m')) $\models$ Kj(F).

Let rj(mj) be an arbitrary state such that (r(m'), rj(mj)) $\in$ kj. It follows rj(mj) $\in$ S[j, r](m').

Thus, from U1 we obtain (r(m), rj(mj))$\in$ki. Therefore,(IS, rj(mj))$\models$ F.We conclude (IS, r(m')) $\models$ Kj(F). Because (for all m' $\geq$ m)[(IS, r(m')) $\models$ Kj(F)], we have (IS, r(m)) $\models$ ♥Kj(F).

I am grateful to the anonymous referees for their careful reading and useful comments.

REFERENCES

[1] R. Fagin et al. *Reasoning about Knowledge*. The MIT Press, London, 1995.

[2] M. J. Fischer J. and N. Immerman. Foundations of knowledge for distributed systems. In J. Y. Halpern (ed.), *Theoretical Aspects of Reasoning about Knowledge: Proc. 1986 Conference,* San Francisco, Calif.: Morgan Kaufmann, 1986, pp. 171-186.

[3] R. Fagin and J. Y. Halpern, (1989). Modelling knowledge and action in distributed systems. *Distributed Computing*, Vol. 3, No. 4, 1989, pp. 159-179.

[4] J. Y. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, Vol. 37, No. 3, 1990, pp. 549-587.

[5] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems.* Springer-Verlag, 1992.

[6] Y. Moses and B. Bloom (1994). Knowledge, timed precedence and clocks. *In Proc. 13th ACM Symp. on Principles of Distributed Computing,* 1994, pp. 294-303.

Received: 1 July 1998
Accepted: 6 September 1998

Mirko Maleković

VIŠEAGENTNI SUSTAVI: UGRAĐIVANJE ZNANJA I VREMENA

Sažetak

U ovom članku razmatrali smo znanje i vrijeme u višeagentnim sustavima. Karakterizirali smo pet temporalnih operatora: o, ♥, ♦, U i W, a zatim smo dokazali nekoliko propozicija koje utvrđuju veze između temporalnih operatora i operatora znanja.

Ključne riječi: baze znanja, operatori znanja, prosuđivanje o znanju, temporalni operatori, teorija znanja, višeagentni sustavi.