499 Modal and Temporal Logic

'Interpreted Systems' and Epistemic Logic

Marek Sergot
Department of Computing
Imperial College, London

Autumn 2008

Further reading (optional!!):
R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge, 1995.

(These notes are based on notes by Alessio Lomuscio.)

This is an example of how *epistemic logic* ('epistemic'—pertaining to knowledge) can be applied to the analysis of computer systems. It is also an example of what is sometimes called 'computationally grounded semantics'.

A common criticism of the application of (modal) logics to the modelling and analysis of computer systems is that there is *no clear correspondence* between a Kripke (relational) model and the possible states of a computational system, and no clear intuition (with the possible exception of temporal logics) about what the accessibility relation is supposed to represent. This is seen as a severe limitation. In particular, metalogical results (especially completeness) are rendered more or less useless.

Hence the study of *computationally grounded* semantics. Here one begins with a formal structure that more directly represents the features of the computer system of interest, and then one seeks to evaluate (modal) logic languages on that structure, possibly by showing how they map to familiar relational (Kripke) models.

One well-developed example is the formalism of *'interpreted systems'*—a kind of computationally grounded semantics intended to model temporal and *epistemic* properties of multi-agent systems and distributed computer systems.

In these notes the aim is just to give the basic idea of 'grounded semantics' in general and 'interpreted systems' in particular. We will only look at the epistemic ('knowledge') components in these notes. There are some brief comments at the end on how temporal aspects can be added.

Interpreted systems

Let $\{1, \ldots, n\}$ be a set of n agents (or n components of a distributed computer system).

Local states

L_i — set of local states for agent i
L_E — set of local states for the environment

The local states L_i model the instantaneous configuration of the agent i in the system. The content varies according to what we want to model: it may be the values of some (local) variables, a database of facts, ... An example follows presently.
Further assume that the agents operate in an environment whose possible states can also be modelled as a set of possible local states L_E. The environment captures relevant information that is not specific to any individual agent, such as messages in transit in a communication channel, values of sensors on some external 'world', etc.

Global states The set G of *global states* of the system is

$$\mathrm{G} \subseteq L_1 \times \cdots \times L_n \times L_E$$

A global state $g = (l_1, \ldots, l_n, l_E)$ represents a snapshot of the system in which agent 1 is in local state $l_1 \in L_1$, agent 2 is in local state $l_2 \in L_2$, ..., agent n is in local state $l_n \in L_n$, and the environment is in local state $l_E \in L_E$.
We will write $l_i(g)$ to denote the local state of agent i in global state g.

A global state $g \in \mathrm{G}$ gives a snapshot of the system. We are (often) interested in the evolution of the system over time. This is usually modelled by the notion of a *run*, which is a function $r\colon \mathbf{N} \to \mathrm{G}$ where $\mathbf{N}$ is the set of the natural numbers. We will ignore runs and time in these notes; there are some brief comments at the end.

Interpreted system Now we define a language $\mathcal{L}$ which will be used to express *global properties* of the system as a whole.
An interpreted system is

$$IS = \langle\, \mathrm{G}, h \,\rangle$$

where G is a set of *global states* and h is an interpretation function for the propositional atoms of $\mathcal{L}$. As usual $h\colon \mathit{Atoms}(\mathcal{L}) \to \wp(\mathrm{G})$.
Truth functional connectives ($\wedge$, $\vee$, $\neg$) are defined in the standard way. So now we can write (as usual) $IS, g \models \varphi$ to mean that formula φ is true at global state g in the interpreted system IS, and $IS \models \varphi$ to mean that φ is valid in IS, i.e., that φ is true at every global state g of IS.

(In these notes I am using φ to stand for an arbitrary formula and not A, B, ... as previously. There is nothing significant about this — just that I typed these at a different time.)

Epistemic modalities Interpreted systems provide a

computationally grounded semantics

that has been used to model knowledge, belief, communication, ...

Write $\mathsf{K}_i\,\varphi$ to represent that agent i 'knows' φ.

$$IS, g \models \mathsf{K}_i\,\varphi \quad \text{iff for all } g' \text{ we have that } l_i(g) = l_i(g') \text{ implies } IS, g' \models \varphi.$$

K_i is an *information-theoretic* notion of 'knows'. When $l_i(g) = l_i(g')$ then global states g and g' are *indistinguishable* for i.

Perhaps you might prefer to look at it like this. When does agent i *not* know that φ in global state g? When there is a global state g' indistinguishable for i in which φ is not true. And so:

$$IS, g \not\models \mathsf{K}_i\,\varphi \quad \text{iff } \exists\, g' \,[l_i(g) = l_i(g') \;\&\; IS, g' \not\models \varphi]$$

which is equivalently stated as

$$IS, g \models \mathsf{K}_i\,\varphi \quad \text{iff } \forall\, g' \,[l_i(g) = l_i(g') \Rightarrow IS, g' \models \varphi]$$

which is exactly the truth condition given above.

$\mathsf{K}_i\,\varphi$ means that i has enough information in its own local states to determine that φ holds globally in the system as a whole. It does not mean that i has actually performed whatever reasoning is required to determine that φ holds. Or to put it another way: K_i represents a *bird's eye* notion of knowledge: it is something that we, as external observers, attribute to agent i.

Model associated with an IS Given a set of agents $\{1, \ldots, n\}$ and an interpreted system $IS = \langle\, \mathrm{G}, h \,\rangle$, the model

$$\mathcal{M}_{IS} = \langle\, \mathrm{G}, R_1, \ldots, R_n, h \,\rangle$$

is defined as follows:

- the set of possible worlds G is the set of global states in IS
- for any $i \in 1..n$, and any $g, g' \in \mathrm{G}$, the accessibility relation R_i is defined as $g\,R_i\,g'$ iff $l_i(g) = l_i(g')$
- h is the valuation function for the atoms, as usual.

Clearly, by construction, for every global state g and formula φ we have

$$IS, g \models \varphi \quad \text{iff} \quad \mathcal{M}_{IS}, g \models \varphi$$

So now we have a way of relating the (new) notion of an interpreted system to the standard notion of a (Kripke, relational) model. R_i in $\mathcal{M}_{IS}$ is the *epistemic accessibility* relation for agent i.

Obviously every accessibility relation R_i in $\mathcal{M}_{IS}$ is an equivalence relation.

So we can expect that each K_i will be of type S5 (i.e., $KT5 = KT45 = KT4B$). This turns out to be so (see 'Axiomatisation' later).

Example: the bit transmission problem

A widely studied problem in the literature on distributed systems.

Sender S wants to communicate some message (the value of a bit, say) to receiver R over a faulty (unreliable) communication channel. The value of the bit will not be corrupted, but messages between the two may get lost.

How can sender S be sure that receiver R has received the bit?

One simple protocol:

- S sends the value of the bit to R, and continues to do so until it receives an acknowledgement.
- R does nothing until it receives the value of the bit, and then it sends acknowledgements to S, forever.
- S keeps sending the bit until it receives an acknowledgement from R, and then it stops sending.

How can we determine whether this protocol has the desired effect?

Bit transmission: Formalisation

$$\begin{aligned} L_S &= \{0,\ 1,\ 0\text{-}ack,\ 1\text{-}ack\} \\ L_R &= \{0,\ 1,\ \epsilon\} \\ L_E &= \{\cdot\} \end{aligned}$$

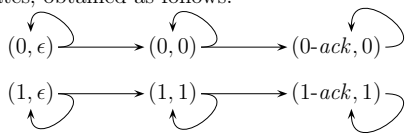
The environment will play no role in this formulation of the bit transmission problem (though it does in more complicated versions) so L_E is just $\{\cdot\}$. Moreover, we'll omit the environment's local state when writing global states to reduce clutter: I will write $(0, 0)$ when really I mean the global state $(0, 0, \cdot)$.

What is the set of global states $\mathrm{G} \subseteq L_S \times L_R \times L_E$ for this example?

We are interested in the set of global states reachable from the initial configurations

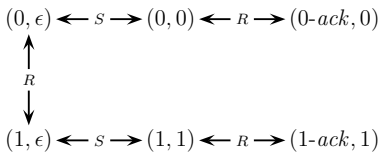
$$(0, \epsilon) \text{ and } (1, \epsilon)$$

So we have six global states, obtained as follows:



Bit transmission: Analysis

Ignoring temporal considerations, we have six global states. Here they are, showing now the epistemic accessibility relations for S and R (reflexive arcs omitted):



Now we define a suitable language. Let **bit=0**, **bit=1**, **recbit**, **recack** be propositional atoms. Define the valuation function h so that:

$$\begin{aligned} h(\mathbf{bit{=}0}) &= \{(0,\epsilon),(0,0),(0\text{-}ack,0)\} \\ h(\mathbf{bit{=}1}) &= \{(1,\epsilon),(1,1),(1\text{-}ack,1)\} \\ h(\mathbf{recbit}) &= \{(0,0),(0\text{-}ack,0),(1,1),(1\text{-}ack,1)\} \\ h(\mathbf{recack}) &= \{(0\text{-}ack,0),(1\text{-}ack,1)\} \end{aligned}$$

which gives

$$\begin{aligned} IS_{\mathrm{b}},g &\models \mathbf{bit{=}0} \quad \text{if} \quad l_S(g)=0, \text{ or } l_S(g)=0\text{-}ack \\ IS_{\mathrm{b}},g &\models \mathbf{bit{=}1} \quad \text{if} \quad l_S(g)=1, \text{ or } l_S(g)=1\text{-}ack \\ IS_{\mathrm{b}},g &\models \mathbf{recbit} \quad \text{if} \quad l_R(g)=1, \text{ or } l_R(g)=0 \\ IS_{\mathrm{b}},g &\models \mathbf{recack} \quad \text{if} \quad l_S(g)=1\text{-}ack, \text{ or } l_S(g)=0\text{-}ack. \end{aligned}$$

It is easy to check from the diagram that:

$$\begin{aligned} IS_{\mathrm{b}} &\models \mathbf{recbit} \rightarrow \big(\mathsf{K}_R(\mathbf{bit{=}0}) \vee \mathsf{K}_R(\mathbf{bit{=}1})\big) \\ IS_{\mathrm{b}} &\models \mathbf{recack} \rightarrow \mathbf{recbit} \end{aligned}$$

And also:

$$\begin{aligned} IS_{\mathrm{b}} &\models \mathbf{recack} \rightarrow \big(\mathsf{K}_R(\mathbf{bit{=}0}) \vee \mathsf{K}_R(\mathbf{bit{=}1})\big) \\ IS_{\mathrm{b}} &\models \mathbf{recack} \rightarrow \mathsf{K}_S\big(\mathsf{K}_R(\mathbf{bit{=}0}) \vee \mathsf{K}_R(\mathbf{bit{=}1})\big) \\ IS_{\mathrm{b}} &\models \mathbf{recack} \wedge (\mathbf{bit{=}0}) \rightarrow \mathsf{K}_S\,\mathsf{K}_R(\mathbf{bit{=}0}) \end{aligned}$$

This last in particular represents that communication between S and R has been established. But notice:

$$IS_{\mathrm{b}} \not\models \mathbf{recack} \wedge (\mathbf{bit{=}0}) \rightarrow \mathsf{K}_R\,\mathsf{K}_S\,\mathsf{K}_R(\mathbf{bit{=}0})$$

To get $\mathsf{K}_R\,\mathsf{K}_S\,\mathsf{K}_R(\mathbf{bit{=}0})$, the sender would have to send an acknowledgement back to the receiver, and so on.
Check these calculations (tutorial exercise).

Axiomatization

As already observed, we can expect that each individual K_i will be a normal modality of type S5.

The logic S5^n The logic S5^n is the smallest modal logic in which each K_i is normal, i.e. containing (for each $i \in 1..n$) all instances of:

$$\begin{aligned} &\text{K.} && \mathsf{K}_i(A \rightarrow B) \rightarrow (\mathsf{K}_i\,A \rightarrow \mathsf{K}_i\,B) \\ &\text{RN.} && \frac{A}{\mathsf{K}_i\,A} \end{aligned}$$

and containing (for each $i \in 1..n$) all instances of:

$$\begin{aligned} &\text{T.} && \mathsf{K}_i\,A \rightarrow A \\ &\text{4.} && \mathsf{K}_i\,A \rightarrow \mathsf{K}_i\,\mathsf{K}_i\,A \\ &\text{5.} && \neg\mathsf{K}_i\,A \rightarrow \mathsf{K}_i\,\neg\mathsf{K}_i\,A \end{aligned}$$

We have the following, which we assert without proof. (The proof is not difficult.)

Theorem *The logic S5^n is sound and complete with respect to the most general class of interpreted systems with n agents.*

Again: it is important to note that interpreted systems are intended to provide a bird's eye view of the system and so S5^n should not be seen as modelling the actual reasoning capabilities of the individual agents. $\mathsf{K}_i\,A$ means that i has enough information in its own local states to determine that A holds globally; it does not mean that i has actually performed whatever reasoning is required to determine that A holds.

Note also that because of axiom T, we have:

$$\vdash_{\mathrm{S5}^n} \mathsf{K}_i\,A \rightarrow \neg\mathsf{K}_j\,\neg A$$

so there are some interaction patterns between the 'knowledge' of the agents.
These can be strengthened. For example, to model a multi-agent system in which one agent (say agent i) knows what every other agent knows we could consider the logic S5$^n \cup \{\mathsf{K}_j\,A \rightarrow \mathsf{K}_i\,A\}$ for all $j \in 1..n$.

A currently active area of research is the study of axiomatisations of the knowledge properties of particular classes of interpreted systems. Some of the most interesting semantical classes result from interactions between knowledge and time.

Reminder: validity in a model

Let C be a class of relational (Kripke) models.

We know that Σ_{C}, the set of formulas valid on the class of models C, is not a system of modal logic according to the definition we have been using — because Σ_{C} in general is not closed under uniform substitution.

You can see that easily in the bit transmission example earlier: for example

$$IS_{\mathrm{b}} \models \mathbf{recack} \rightarrow \mathbf{recbit}$$

but (for example)

$$IS_{\mathrm{b}} \not\models \mathbf{bit{=}0} \rightarrow \mathbf{recbit}$$

So Σ_{C} in general is not closed under uniform substitution. But it *is* closed under modus ponens (propositional consequence RPL), necessitation (RN), rules RM, RK, etc.

If this is not obvious, notice that (Chellas, Theorem 3.3, p69):

- if $\models_{\mathsf{C}} A$ then $\models_{\mathsf{C}} \Box A$

from which follows e.g.

- if $\models_{\mathsf{C}} A \rightarrow B$ then $\models_{\mathsf{C}} \Box A \rightarrow \Box B$ (i.e., rule RM)

Easy to see this: if $\models_{\mathsf{C}} A \rightarrow B$ then $\models_{\mathsf{C}} \Box(A \rightarrow B)$. But schema K is valid in every class of relational models: $\models_{\mathsf{C}} \Box(A \rightarrow B) \rightarrow (\Box A \rightarrow \Box B)$; $\models_{\mathsf{C}} \Box A \rightarrow \Box B$ follows by modus ponens.

We have, by construction

$$IS, g \models \varphi \quad \text{iff} \quad \mathcal{M}_{IS}, g \models \varphi$$

which implies

$$IS \models \varphi \quad \text{iff} \quad \mathcal{M}_{IS} \models \varphi$$

We also have (soundness)

$$\vdash_{\mathrm{S5}^n} \varphi \quad \Rightarrow \quad \models_{\mathsf{Equiv}} \varphi \quad \Rightarrow \quad \mathcal{M}_{IS} \models \varphi \quad \Rightarrow \quad IS \models \varphi$$

(We have completeness as well but we won't use that here.)

So in the bit transmission example, we check in the model that, for instance:

$$IS_{\mathrm{b}} \models \mathbf{recack} \rightarrow \mathbf{recbit}$$

and it follows from this, without any need for further evaluation in the model, that (for instance):

$$IS_{\mathrm{b}} \models \mathsf{K}_S\,\mathbf{recack} \rightarrow \mathsf{K}_S\,\mathbf{recbit}$$

and that (for example)

$$IS_{\mathrm{b}} \models \mathsf{K}_R\mathsf{K}_S\,\mathbf{recack} \rightarrow \mathsf{K}_R\mathsf{K}_S\,\mathbf{recbit}$$
$$IS_{\mathrm{b}} \models \mathsf{K}_S\,\mathbf{recack} \rightarrow \mathsf{K}_S\mathsf{K}_S\,\mathbf{recbit}$$
$$\text{etc, etc}$$

(See tutorial sheet for more examples, and why they are useful.)

Interpreted systems and time

(For interest only; details are not examinable. In fact, details will not be provided.)

The global states $\mathrm{G} \subseteq L_1 \times \cdots \times L_n \times L_E$ give a snapshot of the system. To study evolution of the system over time, it is usual to look at the possible *runs* of the system:

$$\mathrm{R} \subseteq \{r \mid r\colon \mathbf{N} \rightarrow \mathrm{G}\}$$

Now one defines a language with suitably chosen temporal operators (such as F, G, P, H, 'since' and 'until', etc.) interpreted on the set of runs R.

The combination of temporal and epistemic operators gives much expressive power. For example, by nesting temporal formulas inside epistemic ones (as in e.g. $\mathsf{K}_i\,\mathsf{F}\,\varphi$) we can model *knowledge about a changing world*. By nesting epistemic formulas inside temporal ones (as in e.g. $\mathsf{G}\,\mathsf{K}_i\,\varphi$) we can model *the temporal evolution of knowledge*.

There is a multitude of complex interactions that can be studied here. Two features in particular have received much attention in distributed computing:

- synchronicity
- perfect recall

Synchronicity is an assumption commonly made when studying distributed systems. A system is synchronous if all agents/components in the system share a common clock. All systems in which agents/components/processes proceed in turns (for instance, by waiting until all agents have performed an action) are synchronous.

Perfect recall is the assumption that agents remember all information to which they have been exposed, or in other words, that the local state of every agent contains a record of all its previous local states and actions it performed. Although this is an unrealistic assumption in many practical settings, it is nevertheless very useful in many applications. In cryptography, for example, one is not interested in whether or not an agent has decoded a message, but in whether or not it has enough information in principle to decode the message given the information it has acquired so far.

Synchronicity, perfect recall (and other properties) can be studied independently and in various combinations. To give just one example, it turns out that the class of synchronous interpreted systems with perfect recall is characterised by the following schema:

$$\mathsf{K}_i\,\mathsf{X}\,A \rightarrow \mathsf{X}\,\mathsf{K}_i\,A$$

where X is the 'tomorrow/next state' operator: $\mathsf{X}\,\varphi$ is true at a time n when φ is true at time $n+1$.