Basic Epistemic Language

When we want to define the basic epistemic language, we need sets of agent symbols and sets of atomic propositions to talk about. Specifically, we have:

- a finite set \( A \) of agent symbols (often: \( a, b, a', a'', \ldots \))
- a countable set \( P \) of atomic propositions (often: \( p, q, p', p'', \ldots \))

**Definition (Basic epistemic language)**

Let \( P \) be a countable set of atomic propositions and \( A \) be a finite set of agent symbols. Then the language \( L_K \) is defined by the following BNF:

\[
\varphi ::= p \mid \neg \varphi \mid (\varphi \land \varphi) \mid K_a \varphi,
\]

where \( p \in P \) and \( a \in A \).

We use some common abbreviations and conventions:

- \((\varphi \lor \psi) = \neg(\neg \varphi \land \neg \psi)\)
- \((\varphi \rightarrow \varphi) = (\neg \varphi \lor \psi)\)
- \((\varphi \leftrightarrow \psi) = (\varphi \rightarrow \psi) \land (\psi \rightarrow \varphi)\)
- \(\top = p \lor \neg p\) for some \( p \in P \)
- \(\bot = \neg \top\)

If there is no risk of confusion, outer parentheses can be omitted.
Basic Epistemic Language

Only interesting addition compared to propositional logic: the knowledge modalities $K_a$.

- $K_a \varphi$ is read as “agent $a$ knows that $\varphi$ (is true)”.
- Its dual, $\neg K_a \neg \varphi$ is read as “agent $a$ considers $\varphi$ as possible”. Abbreviation: $\bar{K}_a \varphi$.
- For a group of agents $B \subseteq A$, we write $E_B \varphi$ to express that everybody in $B$ knows $\varphi$. I.e., $E_B \varphi \equiv \bigwedge_{b \in B} K_b \varphi$.
- Its dual is $\bar{E}_B \varphi = \neg E_B \neg \varphi \equiv \bigvee_{b \in B} \bar{K}_b \varphi$, which can be read as “some agent $b$ in $B$ considers $\varphi$ as possible”.
- Sometimes, when writing iterated operators, the following convention comes in handy: if $X$ is a modal operator, then $X^n$ is the $n$-fold application of $X$. E.g., $K_a^3 \varphi$ means $K_a K_a K_a \varphi$.

Simplified Hanabi

Example (Simplified Hanabi)

In simplified Hanabi, we have four cards ($r_1, r_2, g_1, g_2$), two players ($a, b$), and just one card per player. We write $p_c$ for the fact that player $p$ holds card $c$. Thus, for instance, $a_{r_1}$ is read as “player $a$ has card $r_1$”. Consider the situation where player $a$ has card $r_1$ and player $b$ has card $r_2$. In this situation, all of the following formulas are true:

- $a_{r_1}$ and $b_{r_2}$,
- $K_a b_{r_2}$ and $K_b a_{r_1}$,
- $K_a \neg a_{r_2}$ and $K_b \neg b_{r_1}$ (Notice that, to arrive at this conclusion, we need to make use of our background theory that contains assertions such as $\neg (a_{r_1} \land b_{r_1})$),
- $K_a (K_b a_{r_1} \lor K_b a_{g_1} \lor K_b a_{g_2})$.

Kripke Models

The semantics of the basic epistemic language is based on a special form of Kripke semantics, where we have

- states (or worlds),
- accessibility relations (or indistinguishability relations) between the worlds, and
- propositional valuations associated with the worlds.
Consider two cities, namely Groningen and Liverpool. Assume that:

- Person \( b \) lives in Groningen.
- Person \( w \) lives in Liverpool.
- “The weather in Groningen is sunny” is the atomic proposition \( g \).
- “The weather in Liverpool is sunny” is the atomic proposition \( \ell \).

States are just possible weather conditions: \( \langle g, \ell \rangle, \langle \neg g, \ell \rangle, \langle g, \neg \ell \rangle, \langle \neg g, \neg \ell \rangle \). We want to model what agent \( b \) knows. Assume that \( b \) is in state \( \langle g, \ell \rangle \). He also considers the state \( \langle g, \neg \ell \rangle \) possible.

This situation can be graphically captured by the following model \( M_1 \):

```
\[ b \xrightarrow{\langle g, \ell \rangle} \]
\[ b \xrightarrow{\langle -g, \ell \rangle} b \]
\[ b \xrightarrow{\langle g, -\ell \rangle} \]
\[ b \xrightarrow{\langle -g, -\ell \rangle} b \]
```

Definition (Kripke model)

Given a countable set of atomic propositions \( P \) and a finite set of agent names \( A \), a Kripke model \( M \) is a structure \( \mathcal{M} = (S, R_A, V_P) \) where:

- \( S \) is a set of states (also called the domain of \( \mathcal{M} \), in symbols \( D(\mathcal{M}) \)),
- \( R_A \) is a function yielding, for every \( a \in A \), an accessibility relation \( R_A(a) = R_a \subseteq S \times S \.),
- \( V_P : P \rightarrow 2^S \) is a valuation function that for all \( p \in P \) yields the set of worlds \( V_P(p) \subseteq S \) where \( p \) is true.

If \( A \) and \( P \) are not important or clear from the context, we will often drop them and write \( \mathcal{M} = (S, R, V) \).

If all accessibility relations \( R_a \) are equivalence relations (reflexive, symmetric and transitive), then we also use the symbols \( \sim \) for \( R \) and \( \sim_a \) for \( R_a \).

In that case, \( \mathcal{M} = (S, \sim, V) \) is also called an epistemic model.
Kripke Semantics

Formulas are then interpreted over states in models (aka. states, pointed models, epistemic states).

Example
- Assume we have the formula $K_b \ell$.
- This formula is not true in state $\langle \neg g, \ell \rangle$, symbolically $\langle \neg g, \ell \rangle \not\models K_b \ell$.
- Reason: In $\langle \neg g, \ell \rangle$, agent $b$ also considers world $\langle \neg g, \neg \ell \rangle$ possible, and in that world, $\ell$ does not hold.

We can define truth of an epistemic formula in an epistemic state inductively as follows.

Definition
Given a Kripke model $\mathcal{M} = (S, R, V)$ and $s \in S$, the pair $(\mathcal{M}, s)$ is called a pointed model. If $\mathcal{M}$ is an epistemic model, then $(\mathcal{M}, s)$ is called an epistemic state.

Definition
A formula $\varphi$ is true in an epistemic state $(\mathcal{M}, s)$, symbolically $\mathcal{M}, s \models \varphi$, under the following conditions:

- $\mathcal{M}, s \models p$ iff $s \in V(p)$
- $\mathcal{M}, s \models \varphi \land \psi$ iff $\mathcal{M}, s \models \varphi$ and $\mathcal{M}, s \models \psi$
- $\mathcal{M}, s \models \neg \varphi$ iff $\mathcal{M}, s \not\models \varphi$
- $\mathcal{M}, s \models K_a \varphi$ iff $\mathcal{M}, t \models \varphi$ for all $t \in S$ with $(s, t) \in R_a$

This implies, among others, that $\mathcal{M}, s \models K_a \varphi$ iff $\mathcal{M}, t \models \varphi$ for some $t \in S$ with $(s, t) \in R_a$.

Definition
If $\mathcal{M}, s \models \varphi$ for all $s \in D(\mathcal{M})$, then we say that $\varphi$ is true in $\mathcal{M}$, symbolically $\mathcal{M} \models \varphi$.

Definition
If $\mathcal{M} \models \varphi$ for all models $\mathcal{M}$ in a certain class $\mathcal{X}$ of models, then we say that $\varphi$ is valid in $\mathcal{X}$, symbolically $\mathcal{X} \models \varphi$.

Example
If $\varphi$ is valid in the class $\mathcal{K}$ of all Kripke models, then we write $\mathcal{K} \models \varphi$. 
Kripke Semantics

**Definition**
If there exists a pointed model \((\mathcal{M}, s)\) such that \(\varphi\) is true in \((\mathcal{M}, s)\), then we say \(\varphi\) is **satisfied** in \((\mathcal{M}, s)\). If \(\mathcal{M}\) belongs to a class of models \(\mathcal{X}\), then \(\varphi\) is **satisfiable** in \(\mathcal{X}\).

**Example** (Higher-order knowledge)
\[ \mathcal{M}_1, (g, \ell) \models K_b g \land \neg K_b \ell. \]

To see this, we have to verify that:
- \(\mathcal{M}_1, (g, \ell) \models K_b g \land \neg K_b \ell.\)
- \(\mathcal{M}_1, (g, \neg \ell) \models K_b g \land \neg K_b \ell.\)

In both cases, agent \(b\) considers the same states as possible, namely \((g, \ell)\) and \((g, \neg \ell)\).
- \(K_b g\) is true because in all accessible states, \(g\) is true.
- \(\neg K_b \ell\) is true because there is an accessible state, namely \((g, \neg \ell)\), where \(\ell\) is not true.

\[ \mathcal{M}_1 \models (K_b g \lor K_b \neg g) \land (\neg K_b \ell \land \neg K_b \neg \ell). \]

Easy to see that both clauses are true and thus the whole formula is true.

**Convention**
From now on: Visualizations of epistemic models use undirected edges and leave out reflexive and transitive edges.
Kripke Semantics

Example

Model $M_2$:

$$
\langle g, \ell \rangle \quad \langle g, \neg \ell \rangle \\
\langle \neg g, \ell \rangle \quad \langle \neg g, \neg \ell \rangle
$$

$w$ $b$ $b$ $w$

April 24th, 2019 B. Nebel, R. Mattmüller – DEL 21 / 63

Example

Another agent $h$ (from Otago, NZ) calls $w$ on the phone. $w$ tells $h$ that $\ell$ is true. Then $h$ tells $w$ that he will call $b$ afterwards, but he does not say whether he will tell $b$ about $\ell$. So, $w$ does not know whether $b$ knows that $\ell$ is true.

Remark: The construction of the corresponding epistemic model basically means starting with the original model and updating it with a particular action, namely $h$ calling $b$.

April 24th, 2019 B. Nebel, R. Mattmüller – DEL 23 / 63

Example

Model $M_2$:

$$
\langle g, \ell \rangle \quad \langle g, \neg \ell \rangle \\
\langle \neg g, \ell \rangle \quad \langle \neg g, \neg \ell \rangle
$$

$w$ $w$ $w$ $w$

April 24th, 2019 B. Nebel, R. Mattmüller – DEL 22 / 63

Model $M_2$:

$$
\langle g, \ell \rangle \models (K_b g \lor K_b \neg g) \land (K_w \ell \lor K_w \neg \ell)
$$

(agent $b$ knows whether $g$, and $w$ knows whether $\ell$).

$M_2, \langle g, \ell \rangle \models \neg K_w g \land \neg K_w \neg g \land (K_b g \lor K_b \neg g)$

(although agent $b$ is ignorant about $g$, he knows that agent $w$ actually knows whether $g$ holds).

Question: Can we also come up with a model that describes ignorance about what the other knows?

Answer: Yes, but to do that we need to introduce more worlds. Note that there can be distinct states with identical valuations!

April 24th, 2019 B. Nebel, R. Mattmüller – DEL 24 / 63

Model $M_3$:

$$
\langle g, \ell \rangle \models \ell \land \neg K_b \ell \land K_b (\neg K_w K_b \ell \land \neg K_w K_b \ell)
$$

April 24th, 2019 B. Nebel, R. Mattmüller – DEL 24 / 63
Kripke Semantics

Proposition

Let $\phi$ and $\psi$ be formulas of $L_K$ and let $K_a$ be an epistemic operator for some $a \in A$. Let $\mathcal{K}$ be the set of all Kripke models and $\mathcal{S}5$ be the set of all epistemic models. Then the following hold:

1. (LO1) $\mathcal{K}, K_a \phi \land K_a (\phi \to \psi) \to K_a \psi$
2. (LO2) $\mathcal{K}, \phi$ implies $\mathcal{K}, K_a \phi$
3. (LO3) $\mathcal{K}, \phi \to \psi$ implies $\mathcal{K}, K_a \phi \to K_a \psi$
4. (LO4) $\mathcal{K}, \phi \leftrightarrow \psi$ implies $\mathcal{K}, K_a \phi \leftrightarrow K_a \psi$
5. (LO5) $\mathcal{K}, (K_a \phi \land K_a \psi) \to K_a (\phi \land \psi)$
6. (LO6) $\mathcal{K}, K_a \phi \to K_a (\phi \vee \psi)$
7. (LO7) $\mathcal{S}5, \neg (K_a \phi \land K_a \neg \phi)$

Accessibility Relation Properties

Definition (Relation properties)

A relation $R$ is called

- reflexive if for all $s$, we have $(s, s) \in R$,
- symmetric if for all $s, t$, $(s, t) \in R$ implies $(t, s) \in R$,
- transitive if for all $s, t, u$, $(s, t) \in R$ and $(t, u) \in R$ implies $(s, u) \in R$,
- serial if for all $s$ there is $t$ such that $(s, t) \in R$,
- Euclidean if for all $s, t, u$, $(s, t) \in R$ and $(s, u) \in R$ implies $(t, u) \in R$, and
- an equivalence relation if it is reflexive, transitive, and symmetric (or: reflexive, transitive, and Euclidean).

Bisimulations

Definition (Bisimulation)

Let two models $\mathcal{M} = (S, R, V)$ and $\mathcal{M}' = (S', R', V')$ be given. A non-empty relation $B \subseteq S \times S'$ is a bisimulation iff for all $s \in S$ and $s' \in S'$ with $(s, s') \in B$:

- (atoms) $s \in V(p)$ iff $s' \in V'(p)$ for all $p \in P$,
- (forth) for all $a \in A$ and all $t \in S$, if $(s, t) \in R_a$, then there is a $t' \in S'$ such that $(s', t') \in R'_a$ and $(t, t') \in B$, and
- (back) for all $a \in A$ and all $t' \in S'$, if $(s', t') \in R'_a$, then there is a $t \in S$ such that $(s, t) \in R_a$ and $(t, t') \in B$.

We write $\mathcal{(\mathcal{M}, s)} \Leftrightarrow (\mathcal{M}', s')$ iff there is a bisimulation between $\mathcal{M}$ and $\mathcal{M}'$ linking $s$ and $s'$, and we then say that $\mathcal{(M, s)}$ and $\mathcal{(M', s')}$ are bisimilar.
Bisimulations

The epistemic language $L_K$ cannot distinguish between bisimilar models.

We write $(M, s) \equiv_{L_K} (M', s')$ if and only if $(M, s) \models \varphi$ iff $(M', s') \models \varphi$ for all formulas $\varphi \in L_K$.

Theorem (Bisimulation)

For all pointed models $(M, s)$ and $(M', s')$, if $(M, s) \rightarrow (M', s')$, then $(M, s) \equiv_{L_K} (M', s')$.

Proof.

By structural induction on $\varphi$. Suppose that $(M, s) \equiv (M', s')$.

- **Base case**: For atomic formulas $\varphi = p \in P$, by atoms, it must be the case that $M, s \models p$ iff $M', s' \models p$ for all $p \in P$.

- **Inductive cases**: Given formula $\varphi$, assume that the claim is already proven for all strict subformulas $\varphi'$ of $\varphi$.

  - **Negation**: Suppose that $M, s \models \neg \varphi'$. By definition, this holds iff $M, s \not\models \varphi'$. By induction hypothesis, this is equivalent to $M', s' \not\models \varphi'$, which in turn is equivalent to $M', s' \models \neg \varphi'$.

Proof (ctd.)

- **Inductive cases**: ...

  - **Conjunction**: Suppose that $M, s \models \varphi_1 \land \varphi_2$. By definition, this holds iff $M, s \models \varphi_1$ and $M, s \models \varphi_2$. By two applications of the induction hypothesis, this is equivalent to $M', s' \models \varphi_1$ and $M', s' \models \varphi_2$, which in turn is equivalent to $M', s' \models \varphi_1 \land \varphi_2$.

- **Disjunction**: Suppose that $M, s \models \varphi_1 \lor \varphi_2$. By definition, this holds iff $M, s \models \varphi_1$ or $M, s \models \varphi_2$. By two applications of the induction hypothesis, this is equivalent to $M', s' \models \varphi_1$ or $M', s' \models \varphi_2$, which in turn is equivalent to $M', s' \models \varphi_1 \lor \varphi_2$.

- **Implication**: Suppose that $M, s \models \varphi_1 \Rightarrow \varphi_2$. By definition, this holds iff $M, s \not\models \varphi_1$ or $M, s \models \varphi_2$. By two applications of the induction hypothesis, this is equivalent to $M', s' \not\models \varphi_1$ or $M', s' \models \varphi_2$, which in turn is equivalent to $M', s' \models \varphi_1 \Rightarrow \varphi_2$.
Bisimulations

Remarks:
- $(M, s) \equiv (M', s')$ implies $(M, s) \equiv \mathcal{E}_K (M', s')$, but the converse does not hold.
- The proof applies to all classes of models, not only epistemic models.

Axiomatization

Logic = set of formulas

Possible ways of characterizing a logic and reasoning in it:
- **Semantic** derivation of valid formulas via Kripke models
- **Syntactic** derivation of valid formulas via axioms and inference rules

Axioms and inference rules of minimal modal logic $K$:
- **(Prop)** all instantiations of propositional tautologies
- **(K)** $K_a(\varphi \rightarrow \psi) \rightarrow (K_a \varphi \rightarrow K_a \psi)$
  (Distribution of $K_a$ over $\rightarrow$)
- **(MP)** From $\varphi$ and $\varphi \rightarrow \psi$, infer $\psi$
  (Modus ponens)
- **(Nec)** From $\varphi$, infer $K_a \varphi$
  (Necessitation of $K_a$)
Axiomatization

**Definition (Derivation)**

Let \( X \) be an arbitrary axiomatisation with axioms \( Ax_1, \ldots, Ax_n \) and rules \( Ru_1, \ldots, Ru_k \), where each rule \( Ru_j \), \( 1 \leq j \leq k \), is of the form “From \( \varphi_1, \ldots, \varphi_{j_{ar}} \), infer \( \varphi_j \)”. We call \( j_{ar} \) the arity of the rule. Then a **derivation** of a formula \( \varphi \) within \( X \) is a finite sequence \( \varphi_1, \ldots, \varphi_m \) of formulas such that:

1. \( \varphi_m = \varphi \) and
2. every \( \varphi_i \) in the sequence is:
   1. either an instance of one of the axioms \( Ax_1, \ldots, Ax_n \),
   2. or else the result of the application of one of the rules \( Ru_1, \ldots, Ru_k \) to \( j_{ar} \) formulas in the sequence that appear before \( \varphi_i \).

If there is a derivation for \( \varphi \) in \( X \), we write \( \vdash_X \varphi \), or, if the system \( X \) is clear from the context, just \( \vdash \varphi \).

We then say that \( \varphi \) is a theorem of \( X \).

Logic \( K \) describes only (arbitrary) Kripke models, including models where \( R_a \) does not necessarily reflect knowledge.

Consider, e.g., model \( M \) below:

\[
\begin{align*}
  s_1 & : p \\
  s_2 & : \neg p \\
  & \xrightarrow{a} \quad a
\end{align*}
\]

\[
(\mathcal{M}, s_1) \models p, \text{ but } \\
(\mathcal{M}, s_1) \not\models K_a \neg p.
\]

\( \neg (p \land K_a \neg p) \) is a theorem.

Semantically, we solved this by requiring epistemic models to have reflexive accessibility relations (among other requirements).

Syntactically, we can add axiom \( K_a \varphi \rightarrow \varphi \).
Axiom system $\mathbf{K}$ is sound and complete w.r.t. the class $\mathcal{K}$ of all Kripke models, i.e., for every formula $\phi$ in $\mathcal{L}_K$, we have $\vdash_{\mathbf{K}} \phi$ iff $\mathcal{K} \models \phi$.

Axiom system $\mathbf{S5}$ is sound and complete w.r.t. the class $\mathcal{S5}$ of all epistemic models, i.e., for every formula $\phi$ in $\mathcal{L}_K$, we have $\vdash_{\mathbf{SS}} \phi$ iff $\mathcal{S5} \models \phi$. 

Example
Proof of $\vdash_{\mathbf{SS}} K_b K_a \rho \rightarrow K_a \rho$:

1. $K_b \rho \rightarrow \rho$ (axiom $T$)
2. $K_a (K_b \rho \rightarrow \rho)$ (Necessitation of $K_a$, 1)
3. $K_a (K_b \rho \rightarrow \rho) \rightarrow (K_a K_b \rho \rightarrow K_a \rho)$ (axiom $K$ with $\phi = K_b \rho$ and $\psi = \rho$)
4. $K_a K_b \rho \rightarrow K_a \rho$ (Modus ponens, 2+3)
Common Knowledge

Recall “everybody knows”: $E_B \varphi \equiv \bigwedge_{b \in B} K_b \varphi$.

- $E_B$ satisfies axiom $T$, but not (positive or negative) introspection.
- I.e., $E_B \varphi \rightarrow E_B E_B \varphi$ is not valid.
- E.g., if agents $a$ and $b$ are both (separately) told that $p$ is true, $E_{ab} p$ is true but not $E_{ab} E_{ab} p$.
- So, how to model that everybody knows that everybody knows that ... that $p$?
- \( \rightarrow \) the common knowledge operator:
- For $B \subseteq A$, $C_B \varphi \equiv \varphi \land E_B \varphi \land E_B^2 \varphi \land E_B^3 \varphi \land \ldots$, where $E_B^n \varphi = E_B E_B \ldots E_B \varphi$.

Example (Common knowledge in card games)

Agents $a$ and $b$ are dealt one card each, both (independently) either red or green. They only see their own card. The actual card deal is $rr$. Now $a$ tells $b$ that she has a red card. Next, $b$ leaves the room, giving $a$ the chance to secretly look at $b$’s card. She doesn’t have to, but she does look.

Model $M_{rg}^1$:

Notational conventions:
- Instead of $C_{(a,b)}$ or $E_{(a,b)}$, we often write $C_{ab}$ and $E_{ab}$, respectively, etc.
- Instead of $C_A$ or $E_A$, we usually write $C$ and $E$, respectively, if $A$ is the set of all agents.

Example (Common knowledge in card games, ctd.)

\[ M_{rg}^3: \]

\[ M_{rg}^2, rr \models C_{ab} \text{red}(a) \]

\[ a \text{ didn’t look} \]

\[ b \text{ not reached! remove!} \]
Common Knowledge

By language $\mathcal{L}_{KC}$, we refer to the language defined like $\mathcal{L}_K$, but with the additional common knowledge modality $C$.

**Definition (Epistemic language with common knowledge)**

Let $P$ be a countable set of atomic propositions and $A$ be a finite set of agent symbols. Then the language $\mathcal{L}_{KC}$ is defined by the following BNF:

$$\phi ::= p \mid \neg \phi \mid (\phi \land \psi) \mid K_a \phi \mid C_B \phi,$$

where $p \in P$, $a \in A$, and $B \subseteq A$.

**Semantics of common knowledge modality:** as before, using (epistemic) Kripke models.

**Definition (Accessibility relations for $E_B$ and $C_B$)**

Let $\mathcal{M} = (S, R, V)$ be a Kripke model with agents $A$ and $B \subseteq A$.

- Then $R_E_B = \bigcup_{b \in B} R_B$.
- The transitive closure of a relation $R$ is the smallest relation $R^*$ such that:
  - $R \subseteq R^*$, and
  - for all $x, y, z$, if $(x, y) \in R^*$ and $(y, z) \in R^*$ then also $(x, z) \in R^*$.

  If, additionally, $(x, x) \in R^*$ for all $x$, then $R^*$ is the reflexive-transitive closure of $R$, symbolically $R^*$.

- Then, define $R_C_B = R^*_E_B$ (Sometimes also $\sim_{C_B}$).

**Example**

$\mathcal{M}, s \models C_{ab} p$

$\mathcal{M}, s \not\models C_{abc} p$
Common Knowledge

Additional axioms and inference rules for common knowledge:

- $C_B(\varphi \rightarrow \psi) \rightarrow (C_B \varphi \rightarrow C_B \psi)$
  (Distribution of $C_B$ over $\rightarrow$)
- $C_B \varphi \rightarrow (\varphi \wedge E_B C_B \varphi)$
  (Mix)
- $C_B(\varphi \rightarrow E_B \varphi) \rightarrow (\varphi \rightarrow C_B \varphi)$
  (Induction of common knowledge)
- From $\varphi$, infer $C_B \varphi$
  (Necessitation of $C_B$)

Theorem

Together with S5 axioms and rules, the above axiomatization is sound and complete with respect to epistemic models with common knowledge.

Model Checking

Question 1 (local model checking): Given model $\mathcal{M}$, state $s$ of $\mathcal{M}$, and formula $\varphi$. How to test (algorithmically) whether $\mathcal{M}, s \models \varphi$?

Possible answer (Q1): Determine whether $\mathcal{M}, s \models \varphi$ by iteratively unraveling definition of $\models$ relation. For efficiency, cache intermediate results. This works even if $\mathcal{M}$ is only given implicitly.

Question 2 (global model checking): Given model $\mathcal{M}$ and formula $\varphi$. How to determine (algorithmically) the set of all states $s$ of $\mathcal{M}$ such that $\mathcal{M}, s \models \varphi$?

Possible answer (Q2): For all subformulas $\psi$ of $\varphi$, determine the sets of states where $\psi$ is true, inductively from small to large subformulas. Details below.
Definition (Subformula)

Let $\varphi$ be an $\mathcal{L}_{KC}$ formula. Then the set of subformulas of $\varphi$, $\text{subf}(\varphi)$, is inductively defined as follows:

$$
\text{subf}(p) = \{ p \} \text{ for } p \in P \\
\text{subf}(\neg \varphi) = \{ \neg \varphi \} \cup \text{subf}(\varphi) \\
\text{subf}(\varphi \land \psi) = \{ \varphi \land \psi \} \cup \text{subf}(\varphi) \cup \text{subf}(\psi) \\
\text{subf}(K_a \varphi) = \{ K_a \varphi \} \cup \text{subf}(\varphi) \\
\text{subf}(C_B \varphi) = \{ C_B \varphi \} \cup \text{subf}(\varphi)
$$

If $\psi \in \text{subf}(\varphi) \setminus \{ \varphi \}$, then $\psi$ is called a proper subformula of $\varphi$.

Algorithm

Let $\mathcal{M} = \langle S, R, V \rangle$ be an (epistemic) Kripke model and $\varphi \in \mathcal{L}_{KC}$ a formula. Let $\varphi_1, \ldots, \varphi_n$ be the subformulas of $\varphi$ ordered from small to large ($\varphi_n = \varphi$). For $i = 1, \ldots, n$, do:

1: switch $\varphi_i$ do
2: case $p \in P$ do
3: $[\varphi_i] := V(p)$
4: case $\neg \varphi'$ do
5: $[\varphi_i] := S \setminus [\varphi']$
6: case $\varphi' \land \varphi''$ do
7: $[\varphi_i] := [\varphi'] \cap [\varphi'']$
8: case $K_a \varphi'$ do
9: $[\varphi] := \text{preim}_a([\varphi'])$
10: case $C_B \varphi'$ do
11: $S' := [\varphi']$
12: while not fixpt($S'$) do
13: $S := S' \cap \text{preim}_B(S')$
14: end while
15: $[\varphi] := S'$

Notation:
When the model $\mathcal{M}$ and domain $S$ are clear from the context, for a given formula $\varphi$, we write $[\varphi]$ for the set of states where $\varphi$ is true, i.e., for $\{ s \in S \mid \mathcal{M}, s \models \varphi \}$.
Model Checking

Example ([C_{ab}p] = ?)

[[p]] = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7\}

S' := [[p]]

S' := S' \cap \text{preim}_{ab}(S') = \{s_1, s_2, s_3, s_4, s_5\}

S' := S' \cap \text{preim}_{ab}(S') = \{s_1, s_2, s_3, s_4, s_5\}

([C_{ab}p]] = \{s_1, s_2, s_3, s_4, s_5\}

Summary

- Basic epistemic language $L_K$: like propositional logic, plus knowledge modalities
- Kripke semantics: possible worlds, accessibility relations, propositional valuations
- $S5$ (knowledge): accessibility relations are equivalence relations
- $L_K$ formulas cannot distinguish between bisimilar models.
- Several axioms have 1-to-1 correspondence to properties of accessibility relations.
- Sound and complete axiomatizations of $K$ and $S5$
- Common knowledge = transitive closure of general knowledge ("everybody knows")
- Algorithmic aspect of epistemic logic (so far): model checking