Logic for Multiagent Systems

Master 1st Year, 1st Semester 2024/2025

Laurențiu Leuștean

Web page: https://cs.unibuc.ro//~lleustean/Teaching/2024-LMS/index.html

Agents

- The question What is an agent? does not have a definitive answer.
- Many competing, mutually inconsistent answers have been offered in the past.

Definition in Michael Wooldridge, An Introduction to MultiAgent Systems, Second Edition, John Wiley & Sons, 2009:

An agent is a system that is capable of independent (autonomous) action on behalf of its user or owner (figuring out what needs to be done to satisfy design objectives, rather than constantly being told).

Multiagent systems

Definition in Ronald Fagin, Joseph Halpern, Yoram Moses, Moshe Vardi, Reasoning about Knowledge, MIT Press, 1995:
A multiagent system is any collection of interacting agents.

Definition in Michael Wooldridge, An Introduction to MultiAgent Systems, Second Edition, John Wiley & Sons, 2009:
A multiagent system is one that consists of a number of agents, which interact with one-another.
Agents act on behalf of users with different goals and motivations.
To successfully interact, they require the ability to cooperate, coordinate, and negotiate with each other, much as people do.

Definition in Yoav Shoham, Kevin Leyton-Brown, Multiagents Systems, Cambridge University Press, 2009:
A multiagent system is a system that includes multiple autonomous entities with either diverging information or diverging interests, or both.

Multiagent systems

The motivation for studying multiagent systems stems from interest in artificial (software or hardware) agents, for example software agents living on the Internet.

Examples

- autonomous robots in a multi-robot setting
- trading agents
- game-playing agents that assist (or replace) human players in a multiplayer game
- interface agents - that facilitate the interaction between the user and various computational resources
- ...

The subject is highly interdisciplinary. Many of the ideas apply to inquiries about human individuals and institutions.

Multiagent systemse

- Consider a multiagent system, in which multiple agents autonomously perform some joint action.
- The agents need to communicate with one another.
- Problems appear when the communication is error-prone.
- One could have scenarios like the following:
  - Agent A sent the message to agent B.
  - The message may not arrive, and agent A knows this.
  - Furthemore, this is common knowledge, so agent A knows that agent B knows that A knows that if a message was sent it may not arrive.

Example

Multiagent system = distributed system; agent = processor; action = computation

We use epistemic logic to make such reasoning precise.

Epistemic logics

The field of epistemic logics or logics of knowledge has begun with the publication, in 1962, ofJaakko Hintikka's book Knowledge and Belief. An Introduction to the Logic of the Two Notions.

Epistemic logics

- are modal logics, whose language contains modal operators, which are applied to formulas.
- use a possible-worlds semantics.
- an agent's knowledge is characterized in terms of a set of possible worlds (called epistemic alternatives by Hintikka), with an accessibility relation holding between them.
- something true in all our agent's epistemic alternatives could be said to be known by the agent.

Epistemic logics

- were developed in computer science for reasoning about multiagent systems.
- are used to prove properties of these systems.
- are used to represent and reason about the information that agents posses: their knowledge.

Ronald Fagin, Joseph Halpern, Yoram Moses, Moshe Vardi, Reasoning about Knowledge, MIT Press, 1995

Propositional logic

Definition 1.1

The language of propositional logic PL consists of:

- *a countable set* $V = \{v_n \mid n \in \mathbb{N}\}$ *of variables;*
- *the logic connectives* $\neg$ *(non),* $\rightarrow$ *(implies)*
- *parantheses:* (,).

- The set *Sym* of symbols of *PL* is

$$Sym := V \cup \{\neg, \rightarrow, (,)\}.$$

- We denote variables by $u, v, x, y, z \ldots$

Definition 1.2

The set Expr of expressions of PL is the set of all finite sequences of symbols of PL.

Definition 1.3

Let $\theta = \theta_0\theta_1 \dots \theta_{k-1}$ *be an expression, where* $\theta_i \in Sym$ *for all* $i = 0, \dots, k-1$.

- *If* $0 \leq i \leq j \leq k-1$, *then the expression* $\theta_i \dots \theta_j$ *is called the* (i,j)-*subexpression of* θ.
- *We say that an expression* ψ *appears in* θ *if there exists* $0 \leq i \leq j \leq k-1$ *such that* ψ *is the* (i,j)-*subexpression of* θ.
- *We denote by* $Var(\theta)$ *the set of variables appearing in* θ.

The definition of formulas is an example of an inductive definition.

Definition 1.4

The formulas of PL are the expressions of PL defined as follows:

- *(F0) Any variable is a formula.*
- *(F1) If φ is a formula, then $(\neg\varphi)$ is a formula.*
- *(F2) If φ and ψ are formulas, then $(\varphi \to \psi)$ is a formula.*
- *(F3) Only the expressions obtained by applying rules (F0), (F1), (F2) are formulas.*

Notations

The set of formulas is denoted by *Form*. Formulas are denoted by $\varphi, \psi, \chi, \ldots$.

Proposition 1.5

The set Form is countable.

Unique readability

If φ is a formula, then exactly one of the following hold:

- $\varphi = v$, where $v \in V$.
- $\varphi = (\neg\psi)$, where ψ is a formula.
- $\varphi = (\psi \to \chi)$, where ψ, χ are formulas.

Furthermore, φ can be written in a unique way in one of these forms.

Definition 1.6

Let φ be a formula. A subformula of φ is any formula ψ that appears in φ.

Proposition 1.7 (Induction principle on formulas)

Let Γ be a set of formulas satisfying the following properties:

- $V \subseteq \Gamma$.
- Γ *is closed to* $\neg$, *that is:* $\varphi \in \Gamma$ *implies* $(\neg\varphi) \in \Gamma$.
- Γ *is closed to* $\rightarrow$, *that is:* $\varphi, \psi \in \Gamma$ *implies* $(\varphi \rightarrow \psi) \in \Gamma$.

Then $\Gamma = Form$.

It is used to prove that all formulas have a property $\mathcal{P}$: we define Γ as the set of all formulas satisfying $\mathcal{P}$ and apply induction on formulas to obtain that $\Gamma = Form$.

Language

The derived connectives $\vee$ (or), $\wedge$ (and), $\leftrightarrow$ (if and only if) are introduced by the following abbreviations:

$$\begin{aligned}
\varphi \vee \psi &:= ((\neg\varphi) \to \psi) \\
\varphi \wedge \psi &:= \neg(\varphi \to (\neg\psi))) \\
\varphi \leftrightarrow \psi &:= ((\varphi \to \psi) \wedge (\psi \to \varphi))
\end{aligned}$$

Conventions and notations

- The external parantheses are omitted, we put them only when necessary. We write $\neg\varphi$, $\varphi \to \psi$, but we write $(\varphi \to \psi) \to \chi$.
- To reduce the use of parentheses, we assume that
  - $\neg$ has higher precedence than $\to, \wedge, \vee, \leftrightarrow$;
  - $\wedge, \vee$ have higher precedence than $\to, \leftrightarrow$.
- Hence, the formula $(((\varphi \to (\psi \vee \chi)) \wedge ((\neg\psi) \leftrightarrow (\psi \vee \chi)))$ is written as $(\varphi \to \psi \vee \chi) \wedge (\neg\psi \leftrightarrow \psi \vee \chi)$.

Truth values

We use the following notations for the truth values:

1 for true and 0 for false.

Hence, the set of truth values is $\{0, 1\}$.

Define the following operations on $\{0, 1\}$ using truth tables.

$\neg : \{0, 1\} \to \{0, 1\}$,

p	$\neg p$
0	1
1	0

$\to : \{0, 1\} \times \{0, 1\} \to \{0, 1\}$,

p	q	$p \to q$
0	0	1
0	1	1
1	0	0
1	1	1

Semantics

$\vee : \{0,1\} \times \{0,1\} \to \{0,1\}$,

p	q	$p \vee q$
0	0	0
0	1	1
1	0	1
1	1	1

$\wedge : \{0,1\} \times \{0,1\} \to \{0,1\}$,

p	q	$p \wedge q$
0	0	0
0	1	0
1	0	0
1	1	1

$\leftrightarrow: \{0,1\} \times \{0,1\} \to \{0,1\}$,

p	q	$p \leftrightarrow q$
0	0	1
0	1	0
1	0	0
1	1	1

Semantics

Definition 1.8

An evaluation (or interpretation) is a function $e : V \to \{0, 1\}$.

Theorem 1.9

For any evaluation $e : V \to \{0, 1\}$ *there exists a unique function*

$$e^+ : Form \to \{0, 1\}$$

satisfying the following properties:

- $e^+(v) = e(v)$ *for all* $v \in V$.
- $e^+(\neg\varphi) = \neg e^+(\varphi)$ *for any formula* φ.
- $e^+(\varphi \to \psi) = e^+(\varphi) \to e^+(\psi)$ *for any formulas* φ, ψ.

Proposition 1.10

For any formula φ *and all evaluations* $e_1, e_2 : V \to \{0, 1\}$,

if $e_1(v) = e_2(v)$ *for all* $v \in Var(\varphi)$, *then* $e_1^+(\varphi) = e_2^+(\varphi)$.

Semantics

Let φ be a formula.

Definition 1.11

- *An evaluation $e : V \to \{0, 1\}$ is a model of φ if $e^+(\varphi) = 1$.*
  Notation: $e \models \varphi$.
- *φ is satisfiable if it has a model.*
- *If φ is not satisfiable, we also say that φ is unsatisfiable or contradictory.*
- *φ is a tautology if every evaluation is a model of φ.*
  Notation: $\models \varphi$.

Notation 1.12

The set of models of φ is denoted by $Mod(\varphi)$.

Remark 1.13

- φ *is a tautology iff* $\neg\varphi$ *is unsatisfiable.*
- φ *is unsatisfiable iff* $\neg\varphi$ *is a tautology.*

Proposition 1.14

Let $e : V \to \{0,1\}$ *be an evaluation. Then for all formulas* φ, ψ,

- $e \models \neg\varphi$ *iff* $e \not\models \varphi$.
- $e \models \varphi \to \psi$ *iff* ($e \models \varphi$ *implies* $e \models \psi$) *iff* ($e \not\models \varphi$ *or* $e \models \psi$).
- $e \models \varphi \vee \psi$ *iff* ($e \models \varphi$ *or* $e \models \psi$).
- $e \models \varphi \wedge \psi$ *iff* ($e \models \varphi$ *and* $e \models \psi$).
- $e \models \varphi \leftrightarrow \psi$ *iff* ($e \models \varphi$ *iff* $e \models \psi$).

Definition 1.15

Let φ, ψ be formulas. We say that

- φ is a semantic consequence of ψ if $Mod(\psi) \subseteq Mod(\varphi)$.
  Notation: $\psi \vDash \varphi$.
- φ and ψ are (logically) equivalent if $Mod(\psi) = Mod(\varphi)$.
  Notation: $\varphi \sim \psi$.

Remark 1.16

Let φ, ψ be formulas.

- $\psi \vDash \varphi$ iff $\vDash \psi \rightarrow \varphi$.
- $\psi \sim \varphi$ iff ($\psi \vDash \varphi$ and $\varphi \vDash \psi$) iff $\vDash \psi \leftrightarrow \varphi$.

Semantics

For all formulas φ, ψ, χ,

$$\models \quad \varphi \vee \neg\varphi$$
$$\models \quad \neg(\varphi \wedge \neg\varphi)$$
$$\models \quad \varphi \wedge \psi \to \varphi$$
$$\models \quad \varphi \to \varphi \vee \psi$$
$$\models \quad \varphi \to (\psi \to \varphi)$$
$$\models \quad (\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$$
$$\models \quad (\varphi \to \psi) \to ((\psi \to \chi) \to (\varphi \to \chi))$$
$$\models \quad (\varphi \to \psi) \vee (\neg\varphi \to \psi)$$
$$\models \quad (\varphi \to \psi) \vee (\varphi \to \neg\psi)$$
$$\models \quad \neg\varphi \to (\neg\psi \leftrightarrow (\psi \to \varphi))$$
$$\models \quad (\varphi \to \psi) \to (((\varphi \to \chi) \to \psi) \to \psi)$$
$$\models \quad \neg\psi \to (\psi \to \varphi)$$

Semantics

$$
\begin{array}{rcl}
 & \models & \psi \to (\neg\psi \to \varphi) \\
 & \models & (\varphi \to \neg\varphi) \to \neg\varphi \\
 & \models & (\neg\varphi \to \varphi) \to \varphi \\
\psi & \models & \varphi \to \psi \\
\neg\varphi & \models & \varphi \to \psi \\
\neg\psi \wedge (\varphi \to \psi) & \models & \neg\varphi \\
(\varphi \to \psi) \wedge (\psi \to \chi) & \models & \varphi \to \chi \\
\varphi \wedge (\varphi \to \psi) & \models & \psi
\end{array}
$$

Semantics

$$
\begin{aligned}
\varphi \vee \psi &\sim \neg(\neg\varphi \wedge \neg\psi) \\
\varphi \wedge \psi &\sim \neg(\neg\varphi \vee \neg\psi) \\
\varphi \to (\psi \to \chi) &\sim \varphi \wedge \psi \to \chi \\
\varphi \sim \varphi \wedge \varphi &\sim \varphi \vee \varphi \\
\varphi \wedge \psi &\sim \psi \wedge \varphi \\
\varphi \vee \psi &\sim \psi \vee \varphi \\
\varphi \wedge (\psi \wedge \chi) &\sim (\varphi \wedge \psi) \wedge \chi \\
\varphi \vee (\psi \vee \chi) &\sim (\varphi \vee \psi) \vee \chi \\
\varphi \vee (\varphi \wedge \psi) &\sim \varphi \\
\varphi \wedge (\varphi \vee \psi) &\sim \varphi
\end{aligned}
$$

Semantics

$$
\begin{aligned}
\varphi \wedge (\psi \vee \chi) &\sim (\varphi \wedge \psi) \vee (\varphi \wedge \chi) \\
\varphi \vee (\psi \wedge \chi) &\sim (\varphi \vee \psi) \wedge (\varphi \vee \chi) \\
\varphi \to \psi \wedge \chi &\sim (\varphi \to \psi) \wedge (\varphi \to \chi) \\
\varphi \to \psi \vee \chi &\sim (\varphi \to \psi) \vee (\varphi \to \chi) \\
\varphi \wedge \psi \to \chi &\sim (\varphi \to \chi) \vee (\psi \to \chi) \\
\varphi \vee \psi \to \chi &\sim (\varphi \to \chi) \wedge (\psi \to \chi) \\
\varphi \to (\psi \to \chi) &\sim \psi \to (\varphi \to \chi) \\
&\sim (\varphi \to \psi) \to (\varphi \to \chi) \\
\neg\varphi \sim \varphi \to \neg\varphi &\sim (\varphi \to \psi) \wedge (\varphi \to \neg\psi) \\
\varphi \to \psi \sim \neg\varphi \vee \psi &\sim \neg(\varphi \wedge \neg\psi) \\
\varphi \vee \psi \sim \varphi \vee (\neg\varphi \wedge \psi) &\sim (\varphi \to \psi) \to \psi \\
\varphi \leftrightarrow (\psi \leftrightarrow \chi) &\sim (\varphi \leftrightarrow \psi) \leftrightarrow \chi
\end{aligned}
$$

It is often useful to have a canonical tautology and a canonical unsatisfiable formula.

Remark 1.17

$v_0 \to v_0$ is a tautology and $\neg(v_0 \to v_0)$ is unsatisfiable.

Notation 1.18

Denote $v_0 \to v_0$ by $\top$ and call it the truth.
Denote $\neg(v_0 \to v_0)$ by $\bot$ and call it the false.

Remark 1.19

- *φ is a tautology iff $\varphi \sim \top$.*
- *φ is unsatisfiable iff $\varphi \sim \bot$.*

Let Γ be a set of formulas.

Definition 1.20

An evaluation $e : V \to \{0,1\}$ is a model of Γ if it is a model of every formula from Γ.
Notation: $e \models \Gamma$.

Notation 1.21

The set of models of Γ is denoted by $Mod(\Gamma)$.

Definition 1.22

A formula φ is a semantic consequence of Γ if $Mod(\Gamma) \subseteq Mod(\varphi)$.
Notation: $\Gamma \models \varphi$.

Definition 1.23

- Γ is satisfiable if it has a model.
- Γ is finitely satisfiable if every finite subset of Γ is satisfiable.
- If Γ is not satisfiable, we say also that Γ is unsatisfiable or contradictory.

Proposition 1.24

The following are equivalent:

- Γ is unsatisfiable.
- $\Gamma \models \bot$.

Theorem 1.25 (Compactness Theorem)

Γ is satisfiable iff Γ is finitely satisfiable.

Syntax

We use a deductive system of Hilbert type for LP.

Logical axioms

The set Axm of (logical) axioms of LP consists of:

$(A1)$ $\varphi \to (\psi \to \varphi)$

$(A2)$ $(\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$

$(A3)$ $(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)$,

where φ, ψ and χ are formulas.

The deduction rule

For any formulas φ, ψ, from φ and $\varphi \to \psi$ infer ψ (modus ponens or (MP)):

$$\frac{\varphi,\ \varphi \to \psi}{\psi}$$

Syntax

Let Γ be a set of formulas. The definition of Γ-theorems is another example of an inductive definition.

Definition 1.26

The Γ-theorems of PL are the formulas defined as follows:

- *(T0) Every logical axiom is a Γ-theorem.*
- *(T1) Every formula of Γ is a Γ-theorem.*
- *(T2) If φ and $\varphi \to \psi$ are Γ-theorems, then ψ is a Γ-theorem.*
- *(T3) Only the formulas obtained by applying rules (T0), (T1), (T2) are Γ-theorems.*

If φ is a Γ-theorem, then we also say that φ is deduced from the hypotheses Γ.

Syntax

Notations

$\Gamma \vdash \varphi \quad :\Leftrightarrow \quad \varphi$ is a Γ-theorem

$\vdash \varphi \quad :\Leftrightarrow \quad \emptyset \vdash \varphi$.

Definition 1.27

A formula φ is called a theorem of LP if $\vdash \varphi$.

By a reformulation of the conditions (T0), (T1), (T2) using the notation $\vdash$, we get

Remark 1.28

- *If φ is an axiom, then $\Gamma \vdash \varphi$.*
- *If $\varphi \in \Gamma$, then $\Gamma \vdash \varphi$.*
- *If $\Gamma \vdash \varphi$ and $\Gamma \vdash \varphi \rightarrow \psi$, then $\Gamma \vdash \psi$.*

Definition 1.29

A Γ-proof (or proof from the hypotheses Γ) is a sequence of formulas $\theta_1, \ldots, \theta_n$ such that for all $i \in \{1, \ldots, n\}$, one of the following holds:

- θ_i *is an axiom.*
- $\theta_i \in \Gamma$.
- *there exist* $k, j < i$ *such that* $\theta_k = \theta_j \to \theta_i$.

Definition 1.30

Let φ be a formula. A Γ-proof of φ or a proof of φ from the hypotheses Γ is a Γ-proof $\theta_1, \ldots, \theta_n$ such that $\theta_n = \varphi$.

Proposition 1.31

For any formula φ,

$$\Gamma \vdash \varphi \quad \text{iff} \quad \text{there exists a } \Gamma\text{-proof of } \varphi.$$

Theorem 1.32 (Deduction Theorem)

Let $\Gamma \cup \{\varphi, \psi\}$ *be a set of formulas. Then*

$$\Gamma \cup \{\varphi\} \vdash \psi \quad \text{iff} \quad \Gamma \vdash \varphi \to \psi.$$

Proposition 1.33

For any formulas φ, ψ, χ,

$$\vdash (\varphi \to \psi) \to ((\psi \to \chi) \to (\varphi \to \chi))$$

$$\vdash (\varphi \to (\psi \to \chi)) \to (\psi \to (\varphi \to \chi))$$

Proposition 1.34

Let $\Gamma \cup \{\varphi, \psi, \chi\}$ *be a set of formulas.*

$$\Gamma \vdash \varphi \to \psi \text{ and } \Gamma \vdash \psi \to \chi \quad \Rightarrow \quad \Gamma \vdash \varphi \to \chi$$

$$\Gamma \cup \{\neg\psi\} \vdash \neg(\varphi \to \varphi) \quad \Rightarrow \quad \Gamma \vdash \psi$$

$$\Gamma \cup \{\psi\} \vdash \varphi \text{ and } \Gamma \cup \{\neg\psi\} \vdash \varphi \quad \Rightarrow \quad \Gamma \vdash \varphi.$$

Consistent sets

Let Γ be a set of formulas.

Definition 1.35

Γ is called consistent if there exists a formula φ such that $\Gamma \not\vdash \varphi$. Γ is said to be inconsistent if it is not consistent, that is $\Gamma \vdash \varphi$ for any formula φ.

Proposition 1.36

- *$\emptyset$ is consistent.*
- *The set of theorems is consistent.*

Proposition 1.37

The following are equivalent:

- *Γ is inconsistent.*
- *$\Gamma \vdash \bot$.*

Completeness Theorem

Theorem 1.38 (Completeness Theorem (version 1))

Let Γ be a set of formulas. Then

$$\Gamma \textit{ is consistent} \iff \Gamma \textit{ is satisfiable.}$$

Theorem 1.39 (Completeness Theorem (version 2))

Let Γ be a set of formulas. For any formula φ,

$$\Gamma \vdash \varphi \iff \Gamma \models \varphi.$$

Modal Logics

Textbook:

P. Blackburn, M. de Rijke, Y. Venema, Modal logic, Cambridge Tracts in Theoretical Computer Science 53, Cambridge University Press, 2001

Definition 2.1

The basic modal language ML_0 consists of:

- *a set PROP of atomic propositions (denoted $p, q, r, \ldots$);*
- *the propositional connectives: $\neg$, $\rightarrow$;*
- *parentheses: (,);*
- *the modal operator $\Box$ (box).*

The set $Sym(ML_0)$ of symbols of ML_0 is

$$Sym(ML_0) := PROP \cup \{\neg, \rightarrow, (,), \Box\}.$$

The expressions of ML_0 are the finite sequences of symbols of ML_0.

Basic modal language

Definition 2.2

The formulas of the basic modal language ML_0 are the expressions inductively defined as follows:

(F0) *Every atomic proposition is a formula.*

(F1) *If φ is a formula, then $(\neg\varphi)$ is a formula.*

(F2) *If φ and ψ are formulas, then $(\varphi \to \psi)$ is a formula.*

(F3) *If φ is a formula, then $(\Box\varphi)$ is a formula.*

(F4) *Only the expressions obtained by applying rules (F0), (F1), (F2), (F3) are formulas.*

Notation: The set of formulas is denoted by $Form(ML_0)$.

Basic modal language

Formulas of ML_0 are defined, using the Backus-Naur notation, as follows:

$$\varphi ::= p \mid (\neg\varphi) \mid (\varphi \to \psi) \mid (\Box\varphi), \quad \text{where } p \in PROP.$$

Proposition 2.3 (Induction principle on formulas)

Let Γ *be a set of formulas satisfying the following properties:*

- $V \subseteq \Gamma$.
- Γ *is closed to* $\neg$, *that is:* $\varphi \in \Gamma$ *implies* $(\neg\varphi) \in \Gamma$.
- Γ *is closed to* $\to$, *that is:* $\varphi, \psi \in \Gamma$ *implies* $(\varphi \to \psi) \in \Gamma$.
- Γ *is closed to* $\Box$, *that is:* $\varphi \in \Gamma$ *implies* $(\Box\varphi) \in \Gamma$.

Then $\Gamma = Form$.

It is used to prove that all formulas have a property $\mathcal{P}$: we define Γ as the set of all formulas satisfying $\mathcal{P}$ and apply induction on formulas to obtain that $\Gamma = Form$.

Derived connectives

Connectives $\vee$, $\wedge$, $\leftrightarrow$ and the constants $\top$ (true), $\bot$ (false) are introduced as in classical propositional logic:

$$\varphi \vee \psi := ((\neg\varphi) \to \psi) \qquad \varphi \wedge \psi := \neg(\varphi \to (\neg\psi))$$

$$\varphi \leftrightarrow \psi := ((\varphi \to \psi) \wedge (\psi \to \varphi))$$

$$\top := p \to p, \text{ where } p \in PROP, \qquad \bot := \neg\top$$

Dual modal operator

The dual of $\Box$ is denoted by $\Diamond$ (diamond) and is defined as:

$$\Diamond\varphi := (\neg(\Box(\neg\varphi)))$$

for every formula φ.

Basic modal language

Usually the external parantheses are omitted, we write them only when necessary. We write $\neg\varphi, \varphi \to \psi$, $\Box\varphi$.

To reduce the use of parentheses, we assume that

- modal operators $\Diamond$ and $\Box$ have higher precedence than the other connectives.
- $\neg$ has higher precedence than $\to, \wedge, \vee, \leftrightarrow$;
- $\wedge, \vee$ have higher precedence than $\to, \leftrightarrow$.

Classical modal logic

In classical modal logic,

- $\Box\varphi$ is read as is necessarily φ.
- $\Diamond\varphi$ means it is not necessary that not φ, that is it is possible the case that φ.

Examples of formulas we would probably regard as correct principles

- $\Box\varphi \to \Diamond\varphi$ (whatever is necessary is possible)
- $\varphi \to \Diamond\varphi$ (whatever is, is possible).

The status of other formulas is harder to decide. What can we say about $\varphi \to \Box\Diamond\varphi$ (whatever is, is necessarily possible) or $\Diamond\varphi \to \Box\Diamond\varphi$ (whatever is possible, is necessarily possible)? Can we consider them as general truths? In order to give an answer to such questions, one has to define a semantics for the classical modal logic.

Definition 2.4

A relational structure is a tuple $\mathcal{F}$ consisting of:

- *a nonempty set W, called the universe (or domain) of $\mathcal{F}$, and*
- *a set of relations on W.*

We assume that every relational structure contains at least one relation. The elements of W are called points, nodes, states, worlds, times, instances or situations.

Example 2.5

A partially ordered set $\mathcal{F} = (W, R)$, where R is a partial order relation on W.

Labeled Transition Systems (LTSs), or more simply, transition systems, are very simple relational structures widely used in computer science.

Definition 2.6

An LTS is a pair $(W, \{R_a \mid a \in A\})$*, where* W *is a nonempty set of states,* A *is a nonempty set of labels and, for every* $a \in A$*,*

$$R_a \subseteq W \times W$$

is a binary relation on W*.*

LTSs can be viewed as an abstract model of computation: the states are the possible states of a computer, the labels stand for programs, and $(u, v) \in R_a$ means that there is an execution of the program a starting in state u and terminating in state v.

Relational structures

Let W be a nonempty set and $R \subseteq W \times W$ be a binary relation.

We write usually Rwv or wRv instead of $(w, v) \in R$. If Rwv, then we say that v is R-accessible from w.

The inverse of R, denoted by R^{-1}, is defined as follows:

$$R^{-1}vw \quad \text{iff} \quad Rwv.$$

We define $R^n (n \geq 0)$ inductively:

$$R^0 = \{(w, w) \mid w \in R\}, \quad R^1 = R, \quad R^{n+1} = R \circ R^n.$$

Thus, for any $n \geq 2$, we have that $R^n wv$ iff there exists $u_1, \ldots, u_{n-1}$ such that $Rwu_1, Ru_1u_2, \ldots, Ru_{n-1}v$.

Frames and models

In the sequel we give the semantics of the basic modal language ML_0.

We will do this in two distinct ways:

- at the level of models, where the fundamental notion of satisfaction (or truth) is defined.
- at the level of frames, where the key notion of validity is defined.

Frames and models

Definition 2.7

A frame for ML_0 is a pair $\mathcal{F} = (W, R)$ such that

- *W is a nonempty set;*
- *R is a binary relation on W.*

That is, a frame for the basic modal language is simply a relational structure with a single binary relation.

Interpretation using agents

Rwv holds iff the agent considers the world v possible according to the informations available in the world w. We think of R as a possibility relation, as R defines worlds that are considered possible by the agent.

Definition 2.8

A model for ML_0 is a pair $\mathcal{M} = (\mathcal{F}, V)$, where

- $\mathcal{F} = (W, R)$ *is a frame for* ML_0;
- $V : PROP \to 2^W$ *is a function called valuation.*

Thus, V assigns to each atomic proposition $p \in PROP$ a subset $V(p)$ of W. Informally, we think of $V(p)$ as the set of points in the model $\mathcal{M}$ where p is true.
Note that models for ML_0 can also be viewed as relational structures in a natural way:

$$\mathcal{M} = (W, R, \{V(p) \mid p \in PROP\}).$$

Thus, a model is a relational structure consisting of a domain, a single binary relation R and the unary relations $V(p), p \in PROP$.
A frame $\mathcal{F}$ and a model $\mathcal{M}$ are two relational structures based on the same universe. However, as we shall see, frames and models are used very differently.

Frames and models

Let $\mathcal{F} = (W, R)$ be a frame and $\mathcal{M} = (\mathcal{F}, V)$ be a model. We also write $\mathcal{M} = (W, R, V)$.

We say that the model $\mathcal{M} = (\mathcal{F}, V)$ is based on the frame $\mathcal{F} = (W, R)$ or that $\mathcal{F}$ is the frame underlying $\mathcal{M}$. Elements of W are called states in $\mathcal{F}$ or in $\mathcal{M}$. We often write $w \in \mathcal{F}$ or $w \in \mathcal{M}$.

Remark

Elements of W are also called worlds or possible worlds, having as inspiration Leibniz's philosophy and the reading of basic modal language in which

$\Box\varphi$ means necessarily φ and $\Diamond\varphi$ means possibly φ.

In Leibniz's view, necessity means truth in all possible worlds and possibility means truth in some possible world.

Frames and models

We define now the notion of satisfaction.

Definition 2.9

Let $\mathcal{M} = (W, R, V)$ be a model and w a state in $\mathcal{M}$. We define inductively the notion

formula φ is satisfied (or true) in $\mathcal{M}$ at state w,
Notation $\mathcal{M}, w \Vdash \varphi$

$$
\begin{array}{rll}
\mathcal{M}, w \Vdash p & \text{iff} & w \in V(p), \quad \text{where } p \in PROP \\
\mathcal{M}, w \Vdash \neg\varphi & \text{iff} & \text{it is not true that } \mathcal{M}, w \Vdash \varphi \\
\mathcal{M}, w \Vdash \varphi \to \psi & \text{iff} & \mathcal{M}, w \Vdash \varphi \text{ implies } \mathcal{M}, w \Vdash \psi \\
\mathcal{M}, w \Vdash \Box\varphi & \text{iff} & \text{for every } v \in W, Rwv \text{ implies } \mathcal{M}, v \Vdash \varphi.
\end{array}
$$

Frames and models

Let $\mathcal{M} = (W, R, V)$ be a model.

Notation

If $\mathcal{M}$ does not satisfy φ at w, we write $\mathcal{M}, w \nVdash \varphi$ and we say that φ is false in $\mathcal{M}$ at state w.

It follows from Definition 2.9 that for every state $w \in W$,

- $\mathcal{M}, w \Vdash \neg\varphi$ iff $\mathcal{M}, w \nVdash \varphi$.

Notation

We can extend the valuation V from atomic propositions to arbitrary formulas φ so that $V(\varphi)$ is the set of all states in $\mathcal{M}$ at which φ is true:

$$V(\varphi) = \{w \mid \mathcal{M}, w \Vdash \varphi\}.$$

Frames and models

Let $\mathcal{M} = (W, R, V)$ be a model and w a state in $\mathcal{M}$.

Proposition 2.10

For every formulas φ, ψ,

$$\begin{array}{lll} \mathcal{M}, w \Vdash \varphi \vee \psi & \textit{iff} & \mathcal{M}, w \Vdash \varphi \textit{ or } \mathcal{M}, w \Vdash \psi \\ \mathcal{M}, w \Vdash \varphi \wedge \psi & \textit{iff} & \mathcal{M}, w \Vdash \varphi \textit{ and } \mathcal{M}, w \Vdash \psi \end{array}$$

Proposition 2.11

For every formula φ,

$\mathcal{M}, w \Vdash \Diamond\varphi$ *iff there exists* $v \in W$ *such that* Rwv *and* $\mathcal{M}, v \Vdash \varphi$.

Frames and models

Let $\mathcal{M} = (W, R, V)$ be a model and w a state in $\mathcal{M}$.

Proposition 2.12

For every $n \geq 1$ and every formula φ, define

$$\Diamond^n \varphi := \underbrace{\Diamond\Diamond \ldots \Diamond}_{n \text{ times}} \varphi, \qquad \Box^n \varphi := \underbrace{\Box\Box \ldots \Box}_{n \text{ times}} \varphi.$$

Then

$$\begin{array}{lll} \mathcal{M}, w \Vdash \Diamond^n \varphi & \text{iff} & \text{there exists } v \in W \text{ s.t. } R^n wv \text{ and } \mathcal{M}, v \Vdash \varphi \\ \mathcal{M}, w \Vdash \Box^n \varphi & \text{iff} & \text{for every } v \in W, R^n wv \text{ implies } \mathcal{M}, v \Vdash \varphi. \end{array}$$

Frames and models

Let $\mathcal{M} = (W, R, V)$ be a model.

Definition 2.13

- *A formula φ is globally true or simply true in $\mathcal{M}$ if $\mathcal{M}, w \Vdash \varphi$ for every $w \in W$. Notation: $\mathcal{M} \Vdash \varphi$*
- *A formula φ is satisfiable in $\mathcal{M}$ if there exists a state $w \in W$ such that $\mathcal{M}, w \Vdash \varphi$.*

Definition 2.14

Let Σ be a set of formulas.

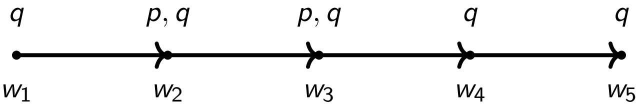
- *Σ is true at state w in $\mathcal{M}$ if $\mathcal{M}, w \Vdash \varphi$ for every $\varphi \in \Sigma$. Notation: $\mathcal{M}, w \Vdash \Sigma$*
- *Σ is globally true or simply true in $\mathcal{M}$ if $\mathcal{M}, w \Vdash \Sigma$ for every state w in $\mathcal{M}$. Notation: $\mathcal{M} \Vdash \Sigma$*
- *Σ is satisfiable in $\mathcal{M}$ if there exists a state $w \in W$ such that $\mathcal{M}, w \Vdash \Sigma$.*

Frames and models

A model $M = (W, R, V)$ is represented as a labeled directed graph:

- the nodes of the graph are the states of the model;
- the label of each node $w \in W$ describes which atomic propositions are true at state w;
- there exists an edge from node w to node v iff Rwv holds.

Example

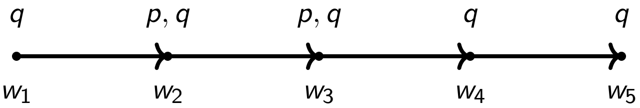


We know that $PROP = \{p, q, r\}$. Then $\mathcal{M} = (W, R, V)$, where $W = \{w_1, w_2, w_3, w_4, w_5\}$; Rw_iw_j iff $j = i + 1$; $V(p) = \{w_2, w_3\}$, $V(q) = \{w_1, w_2, w_3, w_4, w_5\}$ and $V(r) = \emptyset$.

Frames and models

Example

Let $\mathcal{M} = (W, R, V)$ be the model represented by:



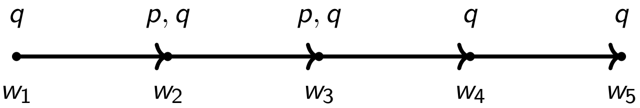
(i) $\mathcal{M}, w_1 \Vdash \Diamond\Box p$.
(ii) $\mathcal{M}, w_1 \nVdash \Diamond\Box p \to p$.
(iii) $\mathcal{M}, w_2 \Vdash \Diamond(p \wedge \neg r)$.
(iv) $\mathcal{M}, w_1 \Vdash q \wedge \Diamond(q \wedge \Diamond(q \wedge \Diamond(q \wedge \Diamond q)))$.
(v) $\mathcal{M} \Vdash \Box q$.

Proof: (i) $\mathcal{M}, w_1 \Vdash \Diamond\Box p$ iff there exists $v \in W$ such that $Rw_1 v$ and $\mathcal{M}, v \Vdash \Box p$. Take $v := w_2$. As $Rw_1 w_2$, it remains to prove that $\mathcal{M}, w_2 \Vdash \Box p$. We have that $\mathcal{M}, w_2 \Vdash \Box p \iff$ for all $u \in W$, $Rw_2 u$ implies $\mathcal{M}, u \Vdash p \iff \mathcal{M}, w_3 \Vdash p$ (as w_3 is the unique $u \in W$ s.t. $Rw_2 u$) $\iff w_3 \in V(p)$, which is true.

Frames and models

Example

Let $\mathcal{M} = (W, R, V)$ be the model represented by:



Proof: (ii) We have that $\mathcal{M}, w_1 \not\Vdash \Diamond\Box p \to p \iff \mathcal{M}, w_1 \Vdash \Diamond\Box p$ and $\mathcal{M}, w_1 \not\Vdash p$. Apply (i) and the fact that $w_1 \notin V(p)$.
(iii), (iv) Exercise.
(v) Let $w \in W$ be arbitray. Then $\mathcal{M}, w \Vdash \Box q \iff$ for all $v \in W$, Rwv implies $\mathcal{M}, v \Vdash q \iff$ for all $v \in W$, Rwv implies $v \in V(q)$, which is true, as $V(q) = W$.

Frames and models

The notion of satisfaction is internal and local. We evaluate formulas inside models, at some particular state w (the current state). Modal operators $\Diamond, \Box$ work locally: we verify the truth of φ only in the states that are R-accesibile from the current one.

At first sight this may seem a weakness of the satisfaction definition. In fact, it is its greatest source of strength, as it gives us great flexibility.
For example, if we take $R = W \times W$, then all states are accessible from the current state; this corresponds to the Leibnizian idea in its purest form.
Going to the other extreme, if we take $R = \{(v, v) \mid v \in W\}$, then no state has access to any other.
Between these extremes there is a wide range of options to explore.

Frames and models

We can ask ourselves the following natural questions:

- What happens if we impose some conditions on R (for example, reflexivity, symmetry, transitivity, etc.)?
- What is the impact of these conditions on the notions of necessity and possibility?
- What principles or rules are justified by these conditions?

Frames and validity

Validity in a frame is one of the key concepts in modal logic.

Definition 2.15

Let $\mathcal{F}$ be a frame and φ be a formula.

- φ *is valid at a state* w *in* $\mathcal{F}$ *if* φ *is true at* w *in every model* $\mathcal{M} = (\mathcal{F}, V)$ *based on* $\mathcal{F}$*.*
- φ *is valid in* $\mathcal{F}$ *if it is valid at every state* w *in* $\mathcal{F}$*.*
  Notation: $\mathcal{F} \Vdash \varphi$

Hence, a formula is valid in a frame if it is true at every state in every model based on the frame.

Validity in a frame differs in an essential way from the truth in a model. Let us give a simple example.

Example 2.16

If $\varphi \vee \psi$ is true in a model $\mathcal{M}$ at w, then φ is true in $\mathcal{M}$ at w or ψ is true in $\mathcal{M}$ at w (by Proposition 2.10).

On the other hand, if $\varphi \vee \psi$ is valid in a frame $\mathcal{F}$ at w, it does not follow that φ is valid in $\mathcal{F}$ at w or ψ is valid in $\mathcal{F}$ at w ($p \vee \neg p$ is a counterexample).

Definition 2.17

Let $\mathbf{M}$ *be a class of models,* $\mathbf{F}$ *be a class of frames and* φ *be a formula. We say that*

- φ *is true in* $\mathbf{M}$ *if it is true in every model in* $\mathbf{M}$*.*
  Notation: $\mathbf{M} \Vdash \varphi$
- φ *is valid in* $\mathbf{F}$ *if it is valid in every frame in* $\mathbf{F}$*.*
  Notation: $\mathbf{F} \Vdash \varphi$

Definition 2.18

The set of all formulas of ML_0 *that are valid in a class of frames* $\mathbf{F}$ *is called the logic of* $\mathbf{F}$ *and is denoted by* $\Lambda_{\mathbf{F}}$*.*

Example 2.19

Formulas $\Diamond(p \vee q) \to (\Diamond p \vee \Diamond q)$ and $\Box(p \to q) \to (\Box p \to \Box q)$ are valid in the class of all frames.

Proof: Let $\mathcal{F} = (W, R)$ be an arbitrary frame, w a state in $\mathcal{F}$ and $\mathcal{M} = (\mathcal{F}, V)$ be a model based on $\mathcal{F}$. We have to show that

$$\mathcal{M}, w \Vdash \Diamond(p \vee q) \to (\Diamond p \vee \Diamond q).$$

Suppose that $\mathcal{M}, w \Vdash \Diamond(p \vee q)$. Then there exists $v \in W$ such that Rwv and $\mathcal{M}, v \Vdash p \vee q$. We have two cases:

- $\mathcal{M}, v \Vdash p$. Then $\mathcal{M}, w \Vdash \Diamond p$, so $\mathcal{M}, w \Vdash \Diamond p \vee \Diamond q$.
- $\mathcal{M}, v \Vdash q$. Then $\mathcal{M}, w \Vdash \Diamond q$, so $\mathcal{M}, w \Vdash \Diamond p \vee \Diamond q$.

We let as an exercise to prove that $\Box(p \to q) \to (\Box p \to \Box q)$ is valid in the class of all frames. □

Example 2.20

Formula $\Box p \to \Box\Box p$ is not valid in the class of all frames.

Proof: We have to find a frame $\mathcal{F} = (W, R)$, a state w in $\mathcal{F}$ and a model $\mathcal{M} = (\mathcal{F}, V)$ such that

$$\mathcal{M}, w \nVdash \Box p \to \Box\Box p.$$

Consider the following frame: $\mathcal{F} = (W, R)$, where

$$W = \{0, 1, 2\}, \quad R = \{(0, 1), (1, 2)\}$$

and take a valuation V such that $V(p) = \{1\}$. Then $\mathcal{M}, 0 \Vdash \Box p$, since 1 is the only state R-accesible from 0 and $\mathcal{M}, 1 \Vdash p$, as $1 \in V(p)$.
On the other hand, $\mathcal{M}, 0 \nVdash \Box\Box p$, since $R^2 02$ and $\mathcal{M}, 2 \nVdash p$, as $2 \notin V(p)$. $\Box$

Definition 2.21

We say that a frame $\mathcal{F} = (W, R)$ is transitive if R is transitive.

Example 2.22

Formula $\Box p \to \Box\Box p$ is valid in the class of all transitive frames.

Proof: Let $\mathcal{F} = (W, R)$ be a transitive frame, w a state in $\mathcal{F}$ and $\mathcal{M} = (\mathcal{F}, V)$ be a model based on $\mathcal{F}$. Assume that $\mathcal{M}, w \Vdash \Box p$. Then for all $v \in W$,

$$(*) \qquad Rwv \text{ implies } \mathcal{M}, v \Vdash p.$$

Let us prove that $\mathcal{M}, w \Vdash \Box\Box p$. Let $u, u' \in W$ be such that Rwu' and $Ru'u$. We have to prove that $\mathcal{M}, u \Vdash p$. Since R is transitive, it follows that Rwu. Applying (*) with $v := u$ we get that $\mathcal{M}, u \Vdash p$. $\Box$

Modal consequence

We introduce the consequence relation.

The basic ideas are the following;

- A relation of semantic consequence holds when the truth of the premises guarantees the truth of the conclusion.
- The inferences depend on the class of structures we are working with. (For example, inferences for transitive frames must be different than the ones for intransitive frames.)

Thus, the definition of the consequence relation must make reference to a class of structures $\boldsymbol{S}$.

Modal consequence

Let $\boldsymbol{S}$ be a class of structures (frames or models) for ML_0.
If $\boldsymbol{S}$ is a class of models, then a model from $\boldsymbol{S}$ is simply an element $\mathcal{M}$ of $\boldsymbol{S}$. If $\boldsymbol{S}$ is a class of frames, then a model from $\boldsymbol{S}$ is a model based on a frame in $\boldsymbol{S}$.

Definition 2.23

Let Σ be a set of formulas and φ be a formula. We say that φ is a semantic consequence of Σ over $\boldsymbol{S}$ if for all models $\mathcal{M}$ from $\boldsymbol{S}$ and all states w in $\mathcal{M}$,

$$\mathcal{M}, w \Vdash \Sigma \quad \textit{implies} \quad \mathcal{M}, w \Vdash \varphi.$$

Notation: $\Sigma \Vdash_{\boldsymbol{S}} \varphi$

Thus, if Σ is true at a state of the model, then φ must be true at the same state.

Remark 2.24

$$\{\psi\} \Vdash_{\textbf{\textit{S}}} \varphi \textit{ iff } \textbf{\textit{S}} \Vdash \psi \to \varphi.$$

Example 2.25

Let *Tran* be the class of transitive frames. Then

$$\{\Box\varphi\} \Vdash_{\textit{Tran}} \Box\Box\varphi.$$

But $\Box\Box\varphi$ is NOT a semantic consequence of $\Box\varphi$ over the class of all frames.

Definition 2.26

A normal modal logic is a set Λ of formulas of ML_0 satisfying the following properties:

- *Λ contains the following axioms:*

$$\begin{array}{rl} (Taut) & \text{all propositional tautologies}, \\ (K) & \Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi), \end{array}$$

  where φ, ψ are formulas of ML_0.

- *Λ is closed under the following deduction rules:*
  - *modus ponens (MP):* $\dfrac{\varphi,\ \varphi \to \psi}{\psi}$.
    Hence, if $\varphi \in \Lambda$ and $\varphi \to \psi \in \Lambda$, then $\psi \in \Lambda$.
  - *generalization or necessitation (GEN):* $\dfrac{\varphi}{\Box\varphi}$.
    Hence, if $\varphi \in \Lambda$, then $\Box\varphi \in \Lambda$.

Normal modal logics - tautologies

We add all propositional tautologies as axioms for simplicity, it is not necessary. We could add a small number of tautologies, which generates all of them. For example,

$$\begin{array}{ll} (A1) & \varphi \to (\psi \to \varphi) \\ (A2) & (\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi)) \\ (A3) & (\neg\psi \to \neg\varphi) \to (\varphi \to \psi). \end{array}$$

Proposition 2.27

Any propositional tautology is valid in the class of all frames for ML_0.

Remark 2.28

Tautologies may contain modalities, too. For example, $\Diamond\psi \vee \neg\Diamond\psi$ is a tautology, since it has the same form as $\varphi \vee \neg\varphi$.

Normal modal logics - axiom (K)

Axiom (K) is sometimes called the distribution axiom and it is important because it allows us to transform $\Box(\varphi \to \psi)$ (a boxed formula) in an implication $\Box\varphi \to \Box\psi$, enabling further pure propositional reasoning to take place.
For example, assume that we want to prove $\Box\psi$ and we already have a proof that contains both $\Box(\varphi \to \psi)$ and $\Box\varphi$. Applying (K) and modus ponens, we get $\Box\varphi \to \Box\psi$. Applying again modus ponens, we obtain $\Box\psi$.

By Example 2.19,

Proposition 2.29

(K) *is valid in the class of all frames for* ML_0.

Theorem 2.30

*For any class **F** of frames, $\Lambda_{\mathbf{F}}$, the logic of **F**, is a normal modal logic.*

Lemma 2.31

- *The collection of all formulas is a normal modal logic, called the inconsistent logic.*
- *If $\{\Lambda_i \mid i \in I\}$ is a collection of normal modal logics, then $\bigcap_{i \in I} \Lambda_i$ is a normal modal logic.*

Definition 2.32

***K** is the intersection of all normal modal logics.*

Hence, ***K*** is the smallest normal modal logic.

$\boldsymbol{K}$

Definition 2.33

A ***K****-proof is a sequence of formulas* $\theta_1, \ldots, \theta_n$ *such that for any* $i \in \{1, \ldots, n\}$*, one of the following conditions is satisfied:*

- θ_i *is an axiom (that is, a tautology or* (K)*);*
- θ_i *is obtained from previous formulas by applying modus ponens or generalization.*

Definition 2.34

Let φ *be a formula. A* ***K****-proof of* φ *is a* ***K****-proof* $\theta_1, \ldots, \theta_n$ *such that* $\theta_n = \varphi$*.*
If φ *has a* ***K****-proof, we say that* φ *is* ***K****-provable.*
Notation: $\vdash_{\boldsymbol{K}} \varphi$*.*

Theorem 2.35

$$\boldsymbol{K} = \{\varphi \mid \vdash_{\boldsymbol{K}} \varphi\}.$$

Definition 2.36

Let $\varphi, \psi_1, \ldots, \psi_n\, (n \geq 1)$ be formulas. We say that φ is deducible in propositional logic from $\psi_1, \ldots, \psi_n$ if

$$\psi_1 \wedge \ldots \wedge \psi_n \to \varphi \text{ is a tautology.}$$

Lemma 2.37

Let $\varphi, \psi_1, \ldots, \psi_n\, (n \geq 1)$ be formulas. The following are equivalent:

- *φ is deducible in propositional logic from $\psi_1, \ldots, \psi_n$.*
- *$\psi_1 \to (\psi_2 \to \ldots \to (\psi_n \to \varphi))$ is a tautology.*

Proof: Use the fact that

$$\big(\psi_1 \wedge \ldots \wedge \psi_n \to \varphi\big) \leftrightarrow \big(\psi_1 \to (\psi_2 \to \ldots \to (\psi_n \to \varphi))\big)$$

is a tautology.

Proposition 2.38

K *is closed under propositional deduction: if φ is deducible in propositional logic from assumptions $\psi_1, \ldots, \psi_n$, then*

$$\vdash_{\boldsymbol{K}} \psi_1, \ldots, \vdash_{\boldsymbol{K}} \psi_n \quad \textit{implies} \quad \vdash_{\boldsymbol{K}} \varphi.$$

Proof:

(1)	$\vdash_{\boldsymbol{K}} \psi_1$	hypothesis
	$\vdots$	
(n)	$\vdash_{\boldsymbol{K}} \psi_n$	hypothesis
(n+1)	$\vdash_{\boldsymbol{K}} \psi_1 \to (\psi_2 \to \ldots \to (\psi_n \to \varphi))$	(Taut)
(n+2)	$\vdash_{\boldsymbol{K}} \psi_2 \to \ldots \to (\psi_{n-1} \to (\psi_n \to \varphi))$	(MP): (1), (n+1)
	$\vdots$	
(2n-1)	$\vdash_{\boldsymbol{K}} \psi_{n-1} \to (\psi_n \to \varphi)$	(MP): (n-2), (2n-2)
(2n)	$\vdash_{\boldsymbol{K}} \psi_n \to \varphi$	(MP): (n-1), (2n-1)
(2n+1)	$\vdash_{\boldsymbol{K}} \varphi$	(MP): (n), (2n) $\square$

Proposition 2.39

Assume that $\vdash_{\mathbf{K}} \varphi \to \psi$ *and that* $\vdash_{\mathbf{K}} \psi \to \chi$*. Then* $\vdash_{\mathbf{K}} \varphi \to \chi$*.*

Proof: Apply Proposition 2.38 and the fact that $\varphi \to \chi$ is deducible in propositional logic from assumptions $\varphi \to \psi$, $\psi \to \chi$, as $(\varphi \to \psi) \wedge (\psi \to \chi) \to (\varphi \to \chi)$ is a tautology. □

Proposition 2.40

Assume that $\vdash_{\mathbf{K}} \varphi \to \psi$ *and that* $\vdash_{\mathbf{K}} \varphi \to \chi$*. Then* $\vdash_{\mathbf{K}} \varphi \to \psi \wedge \chi$*.*

Proof: Apply Proposition 2.38 and the fact that $\varphi \to \psi \wedge \chi$ is deducible in propositional logic from assumptions $\varphi \to \psi$, $\varphi \to \chi$, as $(\varphi \to \psi) \wedge (\varphi \to \chi) \to (\varphi \to \psi \wedge \chi)$ is a tautology. □

$\mathbf{K}$

Proposition 2.41

$\vdash_{\mathbf{K}} \varphi \to (\psi \to \chi)$ iff $\vdash_{\mathbf{K}} \varphi \wedge \psi \to \chi$.

Proof: Apply Proposition 2.38 and the fact that

$$(\varphi \to (\psi \to \chi)) \sim (\varphi \wedge \psi \to \chi),$$

hence

$$(\varphi \to (\psi \to \chi)) \to (\varphi \wedge \psi \to \chi), (\varphi \wedge \psi \to \chi) \to (\varphi \to (\psi \to \chi))$$

are tautologies. □

Proposition 2.42

Assume that $\vdash_{\mathbf{K}} \varphi \to \psi$ and $\vdash_{\mathbf{K}} \psi \to \varphi$. Then $\vdash_{\mathbf{K}} \varphi \leftrightarrow \psi$.

Proof: Apply Proposition 2.38 and the fact that $\varphi \leftrightarrow \psi$ is deducible in propositional logic from assumptions $\varphi \to \psi$, $\psi \to \varphi$, as $(\varphi \to \psi) \wedge (\psi \to \varphi) \to (\varphi \leftrightarrow \psi)$ is a tautology. □

Example 2.43

$$\vdash_{\boldsymbol{K}} \varphi \to \psi \text{ implies } \vdash_{\boldsymbol{K}} \Box\varphi \to \Box\psi.$$

Proof: We give the following ***K***-proof:

(1)	$\vdash_{\boldsymbol{K}} \varphi \to \psi$	hypothesis
(2)	$\vdash_{\boldsymbol{K}} \Box(\varphi \to \psi)$	(GEN): (1)
(3)	$\vdash_{\boldsymbol{K}} \Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi)$	(K)
(4)	$\vdash_{\boldsymbol{K}} \Box\varphi \to \Box\psi$	(MP): (2), (3).

□

Example 2.44

$$\vdash_{\boldsymbol{K}} \varphi \to \psi \text{ implies } \vdash_{\boldsymbol{K}} \Diamond\varphi \to \Diamond\psi.$$

Proof: We give the following ***K***-proof:

(1)	$\vdash_{\boldsymbol{K}} \varphi \to \psi$	hypothesis
(2)	$\vdash_{\boldsymbol{K}} (\varphi \to \psi) \to (\neg\psi \to \neg\varphi)$	(Taut)
(3)	$\vdash_{\boldsymbol{K}} \neg\psi \to \neg\varphi$	(MP): (1), (2)
(4)	$\vdash_{\boldsymbol{K}} \Box\neg\psi \to \Box\neg\varphi$	Example 2.43: (3)
(5)	$\vdash_{\boldsymbol{K}} (\Box\neg\psi \to \Box\neg\varphi) \to (\neg\Box\neg\varphi \to \neg\Box\neg\psi)$	(Taut)
(6)	$\vdash_{\boldsymbol{K}} \neg\Box\neg\varphi \to \neg\Box\neg\psi$	(MP): (4), (5)
(7)	$\vdash_{\boldsymbol{K}} \Diamond\varphi \to \Diamond\psi$	definition of $\Diamond$

$\Box$

Example 2.45

$$\vdash_{\boldsymbol{K}} \Box(\varphi \wedge \psi) \to \Box\varphi \wedge \Box\psi.$$

Proof: We give the following ***K***-proof:

(1)	$\vdash_{\boldsymbol{K}} \varphi \wedge \psi \to \varphi$	(Taut)
(2)	$\vdash_{\boldsymbol{K}} \Box(\varphi \wedge \psi) \to \Box\varphi$	Example 2.43: (1)
(3)	$\vdash_{\boldsymbol{K}} \varphi \wedge \psi \to \psi$	(Taut)
(4)	$\vdash_{\boldsymbol{K}} \Box(\varphi \wedge \psi) \to \Box\psi$	Example 2.43: (3)
(5)	$\vdash_{\boldsymbol{K}} \Box(\varphi \wedge \psi) \to \Box\varphi \wedge \Box\psi$	Proposition 2.40, (2) and (4)

□

$\boldsymbol{K}$

Example 2.46

$$\vdash_{\boldsymbol{K}} \Box\varphi \wedge \Box\psi \to \Box(\varphi \wedge \psi).$$

Proof: We give the following $\boldsymbol{K}$-proof:

(1)	$\vdash_{\boldsymbol{K}} \varphi \to (\psi \to (\varphi \wedge \psi))$	(Taut)
(2)	$\vdash_{\boldsymbol{K}} \Box\varphi \to \Box(\psi \to (\varphi \wedge \psi))$	Ex. 2.43: (1)
(3)	$\vdash_{\boldsymbol{K}} \Box(\psi \to (\varphi \wedge \psi)) \to (\Box\psi \to \Box(\varphi \wedge \psi))$	(K)
(4)	$\vdash_{\boldsymbol{K}} \Box\varphi \to (\Box\psi \to \Box(\varphi \wedge \psi))$	Prop. 2.39, (2), (3)
(5)	$\vdash_{\boldsymbol{K}} \Box\varphi \wedge \Box\psi \to \Box(\varphi \wedge \psi)$	Prop. 2.41, (4)

□

Example 2.47

$$\vdash_{\boldsymbol{K}} \Box\varphi \land \Box\psi \leftrightarrow \Box(\varphi \land \psi).$$

Proof: We give the following $\boldsymbol{K}$-proof:

(1)	$\vdash_{\boldsymbol{K}} \Box\varphi \land \Box\psi \to \Box(\varphi \land \psi)$	Example 2.46
(2)	$\vdash_{\boldsymbol{K}} \Box(\varphi \land \psi) \to \Box\varphi \land \Box\psi$	Example 2.45
(3)	$\vdash_{\boldsymbol{K}} \Box\varphi \land \Box\psi \leftrightarrow \Box(\varphi \land \psi)$	Proposition 2.42, (1), (2)

□

$\boldsymbol{K}$

The logic $\boldsymbol{K}$ is very weak. If we are interested in transitive frames, we would like a proof system which reflects this. For example, we know that $\Box\varphi \to \Box\Box\varphi$ is valid in the class of all transitive frames, so we would want a proof system that generates this formula. $\boldsymbol{K}$ does not do this, since $\Box\varphi \to \Box\Box\varphi$ is not valid in the class of all frames.

The idea is to extend $\boldsymbol{K}$ with additional axioms.

$\boldsymbol{K}\Gamma$

By Lemma 2.31, for any set Γ of formulas, there exists the smallest normal modal logic that contains Γ.

Definition 2.48

K**Γ** *is the smallest normal modal logic that contains Γ. We say that* ***K*****Γ** *is generated by Γ or axiomatized by Γ.*

Definition 2.49

A ***K*****Γ***-proof is a sequence of formulas $\theta_1, \ldots, \theta_n$ such that for any $i \in \{1, \ldots, n\}$, one of the following conditions is satisfied:*

- *θ_i is an axiom (that is, a tautology or (K));*
- *$\theta_i \in \Gamma$;*
- *θ_i is obtained from previous formulas by applying modus ponens or generalization.*

$\boldsymbol{K}\Gamma$

Definition 2.50

Let φ be a formula. A $\boldsymbol{K}\Gamma$-proof of φ is a $\boldsymbol{K}\Gamma$-proof $\theta_1, \ldots, \theta_n$ such that $\theta_n = \varphi$.
If φ has a $\boldsymbol{K}\Gamma$-proof, we say that φ is $\boldsymbol{K}\Gamma$-provable.
Notation: $\vdash_{\boldsymbol{K}\Gamma} \varphi$.

Theorem 2.51

$$\boldsymbol{K}\Gamma = \{\varphi \mid \vdash_{\boldsymbol{K}\Gamma} \varphi\}.$$

Let Λ be a normal modal logic.

Definition 2.52

If $\varphi \in \Lambda$, we also say that φ is a Λ-theorem or a theorem of Λ and write $\vdash_\Lambda \varphi$. If $\varphi \notin \Lambda$, we write $\nvdash_\Lambda \varphi$.

With these notations, the conditions from the definition of a normal modal logic are written as follows:

For any formulas φ, ψ, the following hold:

(i) If φ is a tautology, then $\vdash_\Lambda \varphi$.

(ii) $\vdash_\Lambda (K)$.

(iii) If $\vdash_\Lambda \varphi$ and $\vdash_\Lambda \varphi \to \psi$, then $\vdash_\Lambda \psi$.

(iv) If $\vdash_\Lambda \varphi$, then $\vdash_\Lambda \Box\varphi$.

Remark 2.53

- $\vdash_{\mathbf{K}} \varphi$ *implies* $\vdash_{\Lambda} \varphi$.
- *If* $\Gamma \subseteq \Lambda$, *then* $\vdash_{\mathbf{K}\Gamma} \varphi$ *implies* $\vdash_{\Lambda} \varphi$.

Proposition 2.54

Λ is closed under propositional deduction: if φ is deducible in propositional logic from assumptions $\psi_1, \ldots, \psi_n$, then

$$\vdash_{\Lambda} \psi_1, \ldots, \vdash_{\Lambda} \psi_n \quad \textit{implies} \quad \vdash_{\Lambda} \varphi.$$

Proof: Exercise.

Definition 2.55

Let $\Gamma \cup \{\varphi\}$ be a set of formulas. We say that φ is deducible in Λ from Γ or that φ is Λ-deducible from Γ if there exist formulas $\psi_1, \dots, \psi_n \in \Gamma$ ($n \geq 0$) such that

$$\vdash_\Lambda (\psi_1 \wedge \ldots \wedge \psi_n) \to \varphi.$$

(When $n = 0$, this means that $\vdash_\Lambda \varphi$).
Notation: $\Gamma \vdash_\Lambda \varphi$ We write $\Gamma \not\vdash_\Lambda \varphi$ if φ is not Λ-deducible from Γ.

Remark 2.56

The following are equivalent:

(i) $\Gamma \vdash_\Lambda \varphi$.

(ii) *There exist formulas $\psi_1, \dots, \psi_n \in \Gamma$ ($n \geq 0$) such that*
$$\vdash_\Lambda \psi_1 \to (\psi_2 \to \ldots \to (\psi_n \to \varphi)).$$

Proposition 2.57 (Basic properties)

Let φ be a formula and Γ, Δ be sets of formulas.

(i) $\emptyset \vdash_\Lambda \varphi$ *iff* $\vdash_\Lambda \varphi$.

(ii) $\vdash_\Lambda \varphi$ *implies* $\Gamma \vdash_\Lambda \varphi$.

(iii) $\varphi \in \Gamma$ *implies* $\Gamma \vdash_\Lambda \varphi$.

(iv) *If* $\Gamma \vdash_\Lambda \varphi$ *and* $\Gamma \subseteq \Delta$, *then* $\Delta \vdash_\Lambda \varphi$.

Proof: Exercise.

Normal modal logics

Let φ, ψ be formulas and Γ be a set of formulas,

Proposition 2.58

$\Gamma \vdash_\Lambda \varphi$ iff there exists a finite subset Σ of Γ such that $\Sigma \vdash_\Lambda \varphi$.

Proposition 2.59

(i) *If $\Gamma \vdash_\Lambda \varphi$ and ψ is deducible in propositional logic from φ, then $\Gamma \vdash_\Lambda \psi$.*

(ii) *If $\Gamma \vdash_\Lambda \varphi$ and $\Gamma \vdash_\Lambda \varphi \to \psi$, then $\Gamma \vdash_\Lambda \psi$.*

(iii) *If $\Gamma \vdash_\Lambda \varphi$ and $\{\varphi\} \vdash_\Lambda \psi$, then $\Gamma \vdash_\Lambda \psi$.*

Proposition 2.60 (Deduction Theorem)

For any set of formulas Γ and any formulas φ, ψ,

$$\Gamma \vdash_\Lambda \varphi \to \psi \quad \textit{iff} \quad \Gamma \cup \{\varphi\} \vdash_\Lambda \psi.$$

Definition 2.61

A set Γ *of formulas is* *Λ-consistent* *if there exists a formula* φ *such that* $\Gamma \nvdash_\Lambda \varphi$.
Γ *is said to be* *Λ-inconsistent* *if it is not* Λ*-consistent, that is* $\Gamma \vdash_\Lambda \varphi$ *for any formula* φ.

Proposition 2.62

Let Γ *be a set of formulas. The following are equivalent:*

(i) Γ *is* Λ*-inconsistent.*

(ii) *There exists a formula* ψ *such that* $\Gamma \vdash_\Lambda \psi$ *and* $\Gamma \vdash_\Lambda \neg\psi$.

(iii) $\Gamma \vdash_\Lambda \bot$.

Proposition 2.63

Γ *is* Λ*-consistent iff any finite subset of* Γ *is* Λ*-consistent.*

Normal logics - soundness

In the following, we say "normal logic " instead of "normal modal logic".

Let $\boldsymbol{S}$ be a class of structures (frames or models) for ML_0.

Notation:

$$\Lambda_{\boldsymbol{S}} := \{\varphi \mid \mathcal{S} \Vdash \varphi \text{ for any structure } \mathcal{S} \text{ from } \boldsymbol{S}\}.$$

Definition 2.64

A normal logic Λ is sound with respect to $\boldsymbol{S}$ if $\Lambda \subseteq \Lambda_{\boldsymbol{S}}$.

Thus, Λ is sound with respect to $\boldsymbol{S}$ iff for any formula φ and for any structure $\mathcal{S}$ in $\boldsymbol{S}$,

$$\vdash_\Lambda \varphi \quad \text{implies} \quad \mathcal{S} \Vdash \varphi.$$

If Λ is sound with respect to $\boldsymbol{S}$, we say also that $\boldsymbol{S}$ is a class of frames (or models) for Λ.

Normal logics - soundness

*Theorem 2.65 (Soundness theorem for **K**)*

***K** is sound with respect to the class of all frames.*

Proof: We apply Theorem 2.30 and the fact that ***K*** is the least normal logic. □

Definition 2.66

A normal logic Λ *is*

(i) *strongly complete with respect to* $\mathbf{S}$ *if for any set of formulas* $\Gamma \cup \{\varphi\}$,

$$\Gamma \Vdash_{\mathbf{S}} \varphi \quad \textit{implies} \quad \Gamma \vdash_{\Lambda} \varphi.$$

(ii) *weakly complete with respect to* $\mathbf{S}$ *if for any formula* φ,

$$\mathbf{S} \Vdash \varphi \quad \textit{implies} \quad \vdash_{\Lambda} \varphi.$$

Obviously, weak completeness is a particular case of strong completeness; just take $\Gamma = \emptyset$ in Definition 2.66.(i).

Remark 2.67

Λ is weakly complete with respect to $\boldsymbol{S}$ iff $\Lambda_{\boldsymbol{S}} \subseteq \Lambda$.

If a normal logic Λ is both sound and weakly complete with respect to a class of structures $\boldsymbol{S}$, then there is a perfect match between the syntactic and semantic perspectives: $\Lambda = \Lambda_{\boldsymbol{S}}$.

Given a semantically specified normal logic $\Lambda_{\boldsymbol{S}}$ (that is, the logic of some class of structures of interest), a very important problem is to find a simple set of formulas Γ such that $\Lambda_{\boldsymbol{S}}$ is the logic generated by Γ; we say that Γ axiomatizes $\boldsymbol{S}$.

Completeness theorem for ***K***

Theorem 2.68

K *is sound and strongly complete with respect to the class of all frames for* ML_0.

*Logic **K4***

Let

$$(4) \quad \Box\varphi \to \Box\Box\varphi$$

We use the notation ***K4*** for the normal logic generated by (4). Thus, ***K4*** is the smallest normal logic that contains (4).

Theorem 2.69

K4 *is sound and strongly complete with respect to the class of transitive frames.*

Logic *T*

Let

$$(T) \quad \Box\varphi \to \varphi$$

We use the notation ***T*** for the normal logic generated by (T).

Definition 2.70

We say that a frame $\mathcal{F} = (W, R)$ is reflexive if R is reflexive.

Theorem 2.71

T *is sound and strongly complete with respect to the class of reflexive frames.*

*Logic **B***

Let

$$(B) \quad \varphi \to \Box\Diamond\varphi$$

We use the notation ***B*** for the normal logic ***KB*** generated by (B).

Definition 2.72

We say that a frame $\mathcal{F} = (W, R)$ is symmetric if R is symmetric.

Theorem 2.73

***B** is sound and strongly complete with respect to the class of symmetric frames.*

Logic ***KD***

Let

$$\begin{array}{ll} (D) & \Box\varphi \to \Diamond\varphi \\ (D') & \neg\Box(\varphi \wedge \neg\varphi) \end{array}$$

One can prove that $\vdash_{\boldsymbol{K}} (D) \leftrightarrow (D')$ (exercise).

Let ***KD*** be the normal logic generated by (D) (or, equivalently, by (D')).

Definition 2.74

We say that a frame $\mathcal{F} = (W, R)$ is serial if for all $w \in W$ there exists $v \in W$ such that Rwv.

Theorem 2.75

KD *is sound and strongly complete with respect to the class of serial frames.*

Logic ***K*5**

Let

$$(5) \quad \Diamond\varphi \to \Box\Diamond\varphi$$
$$(5') \quad \neg\Box\varphi \to \Box\neg\Box\varphi$$

One can prove that $\vdash_{\boldsymbol{K}} (5) \leftrightarrow (5')$ (exercise).

Let ***K*5** be the normal logic generated by (5) (or, equivalently, by (5′)).

Definition 2.76

We say that a frame $\mathcal{F} = (W, R)$ is Euclidean if for all $w, v, u \in W$,

$$\textit{if } Rwv \textit{ and } Rwu\textit{, then } Rvu.$$

Theorem 2.77

***K*5** *is sound and strongly complete with respect to the class of Euclidean frames.*

Logic **S4**

We use the notation ***S*4** for the normal logic ***KT*4** generated by (T) and (4).

Theorem 2.78

***S*4** *is sound and strongly complete with respect to the class of reflexive and transitive frames.*

Logic **S5**

We use the notation **S5** for the normal logic **KT4B** generated by (T), (4) and (B).

Proposition 2.79

$\mathbf{S5} = \mathbf{KDB4} = \mathbf{KDB5} = \mathbf{KT5}$.

Theorem 2.80

S5 *is sound and strongly complete with respect to the class of frames whose relation is an equivalence relation.*

Multimodal logics

The whole theory presented so far adapts easily to languages with more modal operators.

Let I be a nonempty set.

- The multimodal language ML_I consists of: a set $PROP$ of atomic propositions, $\neg$, $\rightarrow$, the parentheses (,) and a set of modal operators $\{\Box_i \mid i \in I\}$.
- Formulas of ML_I are defined, using the Backus-Naur notation, as follows:

$$\varphi ::= p \mid (\neg\varphi) \mid (\varphi \rightarrow \varphi) \mid (\Box_i\varphi),$$

  where $p \in PROP$ and $i \in I$.
- The dual of $\Box_i$ is denoted by $\Diamond_i$ and is defined as:

$$\Diamond_i\varphi := \neg\Box_i\neg\varphi$$

- A frame for ML_I is a relational structure $\mathcal{F} = (W, \{R_i \mid i \in I\})$, where R_i is a binary relation on W for every $i \in I$.
- A model for ML_I is, as previously, a pair $\mathcal{M} = (\mathcal{F}, V)$, where $\mathcal{F}$ is a frame and $V : PROP \to 2^W$ is a valuation.
- The last clause from the definition of the satisfaction relation $\mathcal{M}, w \Vdash \varphi$ is changed to: for all $i \in I$,

  $\mathcal{M}, w \Vdash \Box_i\varphi$ iff for every $v \in W$, $R_i wv$ implies $\mathcal{M}, v \Vdash \varphi$.
- It follows that

  $\mathcal{M}, w \Vdash \Diamond_i\varphi$ iff there exists $v \in W$ s.t. $R_i wv$ and $\mathcal{M}, v \Vdash \varphi$.
- The definitions of truth in a model ($\mathcal{M} \Vdash \varphi$), of validity in a frame ($\mathcal{F} \Vdash \varphi$) and of the consequence relation are unchanged.

Definition 2.81

A normal multimodal logic is a set Λ of formulas of ML_I satisfying the following properties:

- *Λ contains all propositional tautologies and is closed under modus ponens.*
- *Λ contains all formulas*

$$(K_i) \quad \Box_i(\varphi \to \psi) \to (\Box_i\varphi \to \Box_i\psi),$$

  where φ, ψ are formulas and $i \in I$.
- *Λ is closed under generalization: for any formula φ and all $i \in I$,*

$$\frac{\varphi}{\Box_i\varphi}.$$

Multimodal logics

- We use the same notation, $\boldsymbol{K}$, for the smallest normal multimodal logic.
- We define similarly $\boldsymbol{K}$-proofs and we also have that $\boldsymbol{K} = \{\varphi \mid \vdash_{\boldsymbol{K}} \varphi\}$.
- The multimodal logic generated by a set of formulas Γ is also denoted by $\boldsymbol{K}\Gamma$. Furthermore, $\boldsymbol{K}\Gamma = \{\varphi \mid \vdash_{\boldsymbol{K}\Gamma} \varphi\}$.

- The definitions of Λ-deducibility, Λ-consistence, soundness and weak and strong completeness are unchanged.

Epistemic Logics

Textbook:

R. Fagin, J.Y. Halpern, Y. Moses, M. Vardi, Reasoning About Knowledge, MIT Press, 2004

Reasoning about knowledge

- Consider a multiagent system, in which multiple agents autonomously perform some joint action.
- The agents need to communicate with one another.
- Problems appear when the communication is error-prone.
- One could have scenarios like the following:
  - Agent A sent the message to agent B.
  - The message may not arrive, and agent A knows this.
  - Furthemore, this is common knowledge, so agent A knows that agent B knows that A knows that if a message was sent it may not arrive.

Multiagent system = distributed system; agent = processor; action = computation

We use epistemic logic to make such reasoning precise.

Epistemic logics

In epistemic logics, the multimodal language is used to reason about knowledge. Let $n \geq 1$ and $AG = \{1, \ldots, n\}$ be the set of agents.

- We consider the multimodal language ML_{Ag}.
- We write, for every $i = 1, \ldots, n$, $K_i\varphi$ instead of $\Box_i\varphi$.
- $K_i\varphi$ is read as the agent i knows (that) φ.
- We denote by $\hat{K}_i$ the dual operator: $\hat{K}_i\varphi = \neg K_i\neg\varphi$.
- Then $\hat{K}_i\varphi$ is read as the agent i considers possible (that) φ.

Definition 3.1

An epistemic logic is a set Λ of formulas of ML_{Ag} satisfying the following properties:

- *Λ contains all propositional tautologies and is closed under modus ponens.*
- *Λ contains all formulas*

  $$K_i(\varphi \to \psi) \to (K_i\varphi \to K_i\psi),$$

  where φ, ψ are formulas and $i \in Ag$.
- *Λ is closed under generalization: for any formula φ and all $i \in Ag$,*

  $$\frac{\varphi}{K_i\varphi}.$$

We denote by ***K*** the smallest epistemic logic.

Recall the following axioms:

$$\begin{array}{rc} (T) & K_i\varphi \to \varphi \\ (D') & \neg K_i(\varphi \wedge \neg\varphi) \\ (B) & \varphi \to K_i\neg K_i\neg\varphi \end{array}$$

Properties of knowledge

- Axiom (T) is called the veridity or knowledge axiom: If an agent knows φ, then φ must hold. What is known is true. This is often taken to be the property that distinguishes knowledge from other informational attitudes, such as belief.
- Axiom (D') is the consistency axiom: an agent does not know both φ and $\neg\varphi$. An agent cannot know a contradiction.
- Axiom (B) says that if φ holds, then an agent knows that it does not know $\neg\varphi$.

Epistemic logics

Recall the following axioms:

$$\begin{array}{rl} (4) & K_i\varphi \rightarrow K_iK_i\varphi \\ (5') & \neg K_i\varphi \rightarrow K_i\neg K_i\varphi \end{array}$$

Properties of knowledge

- Axiom (4) is positive introspection: if an agent knows φ, it knows that it knows φ. An agent knows what it knows.
- Axiom (5′) is negative introspection: if an agent does not know φ, it knows that it does not know φ. An agent is aware of what it doesn't know.
- Positive and negative introspection together imply that an agent has perfect knowledge about what it does and does not know.

Let $\mathbf{S5} = \mathbf{KD'B4} = \mathbf{KD'B5'} = \mathbf{KT5'}$. $\mathbf{S5}$ is considered as the logic of idealised knowledge.

Theorem 3.2

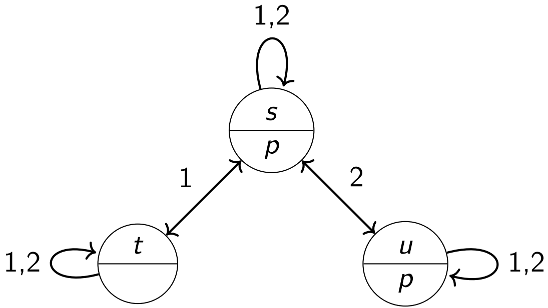
$\mathbf{S5}$ *is sound and strongly complete with respect to the class of frames whose relations are equivalence relations.*

Models for epistemic logics

A model $M = (W, \mathcal{K}_1, \ldots, \mathcal{K}_n, V)$ is represented as a labeled directed graph:

- the nodes of the graph are the states of the model;
- the label of each node $w \in W$ describes which atomic propositions are true at state w;
- we label edges by sets of agents; the label of the edge from node w to node v includes i iff $\mathcal{K}_i wv$ holds.

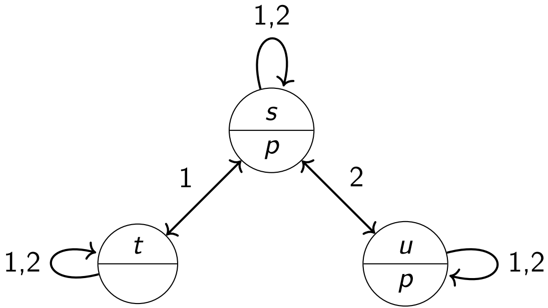
Example



We have that $Ag = \{1, 2\}$, $PROP = \{p\}$ and $\mathcal{M} = (W, \mathcal{K}_1, \mathcal{K}_2, V)$, where

- $W = \{s, t, u\}$.
- $\mathcal{K}_1 = \{(s, s), (t, t), (u, u), (s, t), (t, s)\}$.
- $\mathcal{K}_2 = \{(s, s), (t, t), (u, u), (s, u), (u, s)\}$.
- $V(p) = \{s, u\}$.

Example

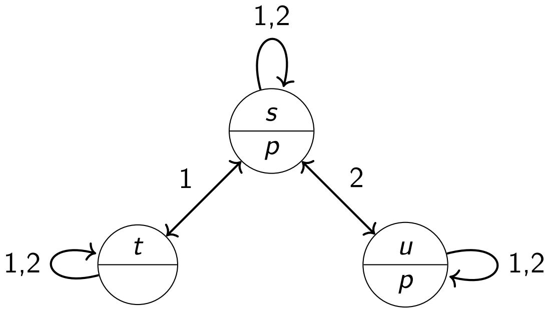


- $\mathcal{M}, s \Vdash p \wedge \neg K_1 p$.

Proof: We have that $s \in V(p)$, hence $\mathcal{M}, s \Vdash p$. Since $\mathcal{K}_1 st$ and $\mathcal{M}, t \nVdash p$, it follows that $\mathcal{M}, s \nVdash K_1 p$, hence $\mathcal{M}, s \Vdash \neg K_1 p$. Thus, $\mathcal{M}, s \Vdash p \wedge \neg K_1 p$. □

In state s, p is true, but agent 1 does not know it, since in state s it considers both s and t possible. We say that agent 1 cannot distinguish s from t. Agent 1's information is insufficient to enable it to distinguish whether the actual state is s or t.

Example

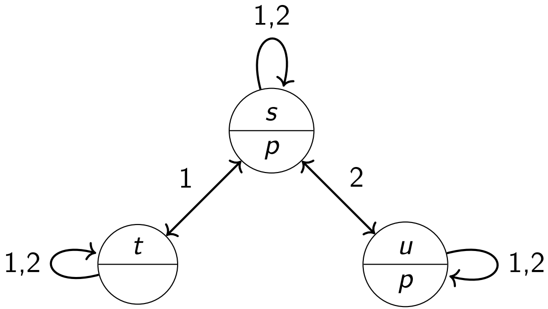


- $\mathcal{M}, s \Vdash K_2 p$.

Proof: We have that $\mathcal{M}, s \Vdash K_2 p$ iff for all $v \in W$, $\mathcal{K}_2 sv$ implies $\mathcal{M}, v \Vdash p$ iff $\mathcal{M}, s \Vdash p$ and $\mathcal{M}, u \Vdash p$ (as $\mathcal{K}_2 ss$, $\mathcal{K}_2 su$, but we don't have that $\mathcal{K}_2 st$), which is true. □

In state s, agent 2 knows that p is true, since p is true in both states that agent 2 considers possible at s (namely s and u).

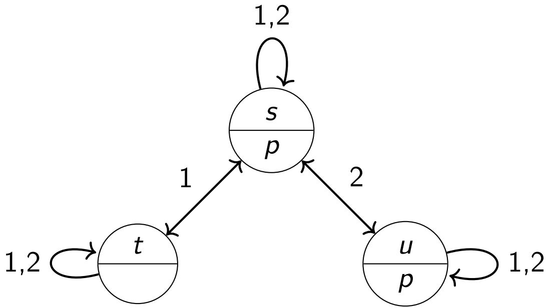
Example



- $\mathcal{M}, s \Vdash \neg K_2 \neg K_1 p$.

Proof: We have that $\mathcal{M}, s \Vdash \neg K_2 \neg K_1 p$ iff $\mathcal{M}, s \nVdash K_2 \neg K_1 p$ iff there exists $v \in W$ such that $\mathcal{K}_2 sv$ and $\mathcal{M}, v \nVdash \neg K_1 p$ iff there exists $v \in W$ such that $\mathcal{K}_2 sv$ and $\mathcal{M}, v \Vdash K_1 p$. Take $v := u$. Then $\mathcal{K}_2 su$ and $\mathcal{M}, u \Vdash K_1 p$, since $\mathcal{M}, u \Vdash p$ and $\mathcal{K}_1 uw$ iff $w = u$. □

Example



- $\mathcal{M}, s \Vdash \neg K_2 \neg K_1 p$.

Although agent 2 knows that p is true in s, it does not know that agent 1 does not know this fact. Why? Because in a state that agent 2 considers possible, namely u, agent 1 does know that p is true, while in another state agent 2 considers possible, namely s, agent 1 does not know this fact.

A simple card game

$Ag = \{1, 2\}$

- Suppose that we have a deck consisting of three cards labeled A, B, and C. Agents 1 and 2 each get one of these cards; the third card is left face down.
- A possible world is characterized by describing the cards held by each agent. For example, in the world (A, B), agent 1 holds card A and agent 2 holds card B (while card C is face down).
- The set of possible worlds is

$$W = \{(A, B), (A, C), (B, A), (B, C), (C, A), (C, B)\}.$$

- In a world such as (A, B), agent 1 thinks two worlds are possible: (A, B) and (A, C). Agent 1 knows that it has card A, but considers it possible that agent 2 could hold either card B or card C.

A simple card game

- Similarly, in world (A, B), agent 2 also considers two worlds are possible: (A, B) and (C, B).
- In general, in a world (X, Y), agent 1 considers (X, Y) and (X, Z) possible, while agent 2 considers (X, Y) and (Z, Y) possible, where Z is different from both X and Y.

- We can easily construct the $\mathcal{K}_1$ and $\mathcal{K}_2$ relations.
- It is easy to check that they are indeed equivalence relations.
- This is because an agent's possibility relation is determined by the information it has, namely, the card it is holding.

A simple card game

We describe the frame $\mathcal{F}_c = (W, \mathcal{K}_1, \mathcal{K}_2)$ for the card game as a labeled graph. Since the relations are equivalence relations, we omit the self loops and the arrows on edges for simplicity (if there is an edge from state w to state v, there must be an edge from v to w as well, by symmetry).

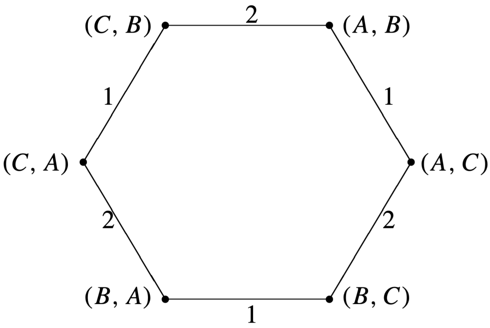
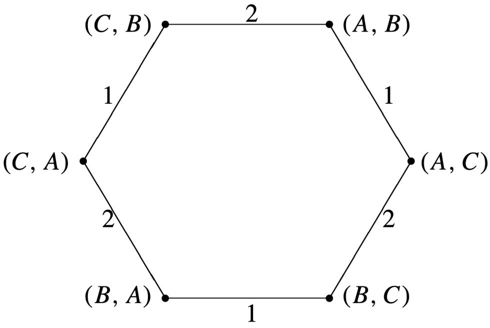


Figure 1: Frame describing a simple card game

A simple card game



- In the world (A, B), agent 1 knows that the world (B, C) cannot be the case. This is captured by the fact that there is no edge from (A, B) to (B, C) labeled 1.
- Nevertheless, agent 1 considers it possible that agent 2 considers it possible that (B, C) is the case. This is captured by the fact that there is an edge labeled 1 from (A, B) to (A, C), from which there is an edge labeled 2 to (B, C).

A simple card game

We still have not defined the model to be used in this example.

Define the set PROP of atomic propositions as

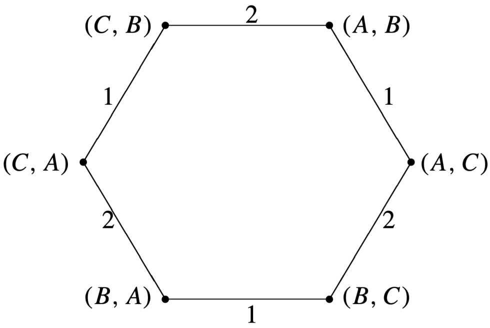
$$PROP = \{iX \mid i \in \{1,2\}, X \in \{A,B,C\}\}.$$

iX will be interpreted as agent i holds card X. Given this interpretation, we define the valuation V in the obvious way:

$$V(iX) = \begin{cases} \{(X,Z) \mid Z \in \{A,B,C\} \setminus \{X\}\} & \text{if } i = 1 \\ \{(Z,X) \mid Z \in \{A,B,C\} \setminus \{X\}\} & \text{if } i = 2. \end{cases}$$

A simple card game

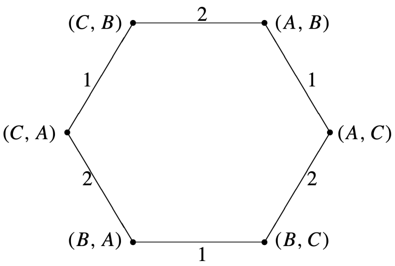
Let $\mathcal{M}_c = (\mathcal{F}_c, V)$ be the model describing this card game.



- $\mathcal{M}_c, (A, B) \models 1A \wedge 2B$.

Proof: $\mathcal{M}_c, (A, B) \models 1A \wedge 2B$ iff
$\mathcal{M}_c, (A, B) \models 1A$ and $\mathcal{M}_c, (A, B) \models 2B$ iff
$(A, B) \in V(1A)$ and $(A, B) \in V(2B)$, which is true.

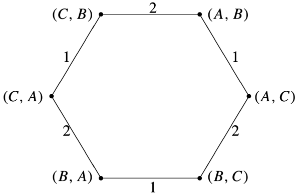
A simple card game



- $\mathcal{M}_c, (A, B) \models K_1 \neg 2A$: agent 1 knows that agent 2 does not hold an A.

Proof: $\mathcal{M}_c, (A, B) \models K_1 \neg 2A$ iff
for all $(X, Y) \in \mathcal{M}_c$, $\mathcal{K}_1(A, B)(X, Y)$ implies $\mathcal{M}_c, (X, Y) \models \neg 2A$
iff $\mathcal{M}_c, (A, B) \models \neg 2A$ and $\mathcal{M}_c, (A, C) \models \neg 2A$ iff
$\mathcal{M}_c, (A, B) \not\models 2A$ and $\mathcal{M}_c, (A, C) \not\models 2A$ iff
$(A, B) \notin V(2A)$ and $(A, C) \notin V(2A)$ iff
$(A, B), (A, C) \notin \{(B, A), (C, A)\}$, which is true. $\square$

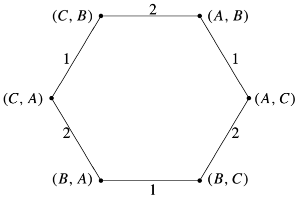
A simple card game



- $\mathcal{M}_c, (A, B) \models K_1 \neg K_2 1A$: agent 1 knows that agent 2 does not know that agent 1 holds an A.

Proof: $\mathcal{M}_c, (A, B) \models K_1 \neg K_2 1A$ iff for all $(X, Y) \in \mathcal{M}_c$, $\mathcal{K}_1(A, B)(X, Y)$ implies $\mathcal{M}_c, (X, Y) \models \neg K_2 1A$ iff $\mathcal{M}_c, (A, B) \models \neg K_2 1A$ and $\mathcal{M}_c, (A, C) \models \neg K_2 1A$ iff $\mathcal{M}_c, (A, B) \not\models K_2 1A$ and $\mathcal{M}_c, (A, C) \not\models K_2 1A$ iff $\big($there exists $(X, Y) \in \mathcal{M}_c$ such that $\mathcal{K}_2(A, B)(X, Y)$ and $\mathcal{M}_c, (X, Y) \not\models 1A\big)$ and $\big($there exists $(Y, Z) \in \mathcal{M}_c$ such that $\mathcal{K}_2(A, C)(Y, Z)$ and $\mathcal{M}_c, (Y, Z) \not\models 1A\big)$

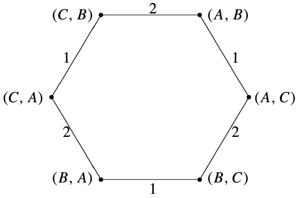
A simple card game



Proof: (continued) $\mathcal{M}_c, (A, B) \models K_1 \neg K_2 1A$ iff
$\big($there exists $(X, Y) \in \mathcal{M}_c$ such that $\mathcal{K}_2(A, B)(X, Y)$ and
$\mathcal{M}_c, (X, Y) \not\models 1A\big)$ and
$\big($there exists $(Y, Z) \in \mathcal{M}_c$ such that $\mathcal{K}_2(A, C)(Y, Z)$ and
$\mathcal{M}_c, (Y, Z) \not\models 1A\big)$

We have that $\mathcal{K}_2(A, B)(C, B)$ and $\mathcal{M}_c, (C, B) \not\models 1A$, since $(C, B) \notin V(1A)$. Thus, we can take $(X, Y) = (C, B)$.
We have that $\mathcal{K}_2(A, C)(B, C)$ and $\mathcal{M}_c, (B, C) \not\models 1A$, since $(B, C) \notin V(1A)$. Thus, we can take $(Y, Z) = (B, C)$. □

A simple card game



- $\mathcal{M}_c, (A, B) \models K_1(2B \vee 2C)$: agent 1 knows that agent 2 holds either B or C.

Proof: Exercise.

- $\mathcal{M}_c, (A, B) \models K_2 \neg K_1 2B$: agent 2 knows that agent 1 does not know that agent 2 holds a B.

Proof: Exercise.

Common and distributed knowledge

We need to reason about knowledge in a group and using this understanding to help us analyze multiagent systems.

- An agent in a group must take into account not only facts that are true about the world, but also the knowledge of other agents in the group.
- For example, in a bargaining situation, the seller of a car must consider what the potential buyer knows about the car's value. The buyer must also consider what the seller knows about what the buyer knows about the car's value, and so on.
- Such reasoning can get rather convoluted. Example: "Dean doesn't know whether Nixon knows that Dean knows that Nixon knows that McCord burgled O'Brien's office at Watergate."
- But this is precisely the type of reasoning that is needed when analyzing the knowledge of agents in a group.

Common and distributed knowledge

We are often interested in situations in which everyone in the group knows a fact.

Example

A society certainly wants all drivers to know that a red light means stop and a green light means go. Suppose we assume that every driver in the society knows this fact and follows the rules. A driver does not feel safe, unless she also knows that everyone else knows and is following the rules. Otherwise, a driver may consider it possible that, although she knows the rules, some other driver does not, and that driver may run a red light.

Common and distributed knowledge

- In some cases we also need to consider the state in which simultaneously everyone knows a fact φ, everyone knows that everyone knows φ, everyone knows that everyone knows that everyone knows φ, and so on. We say that the group has common knowledge of φ.
- The notion of common knowledge was first studied by the philosopher David Lewis in the context of conventions: in order for something to be a convention, it must be common knowledge among the members of a group.
- John McCarthy, in the context of studying common-sense reasoning, characterized common knowledge as what any fool knows.

Example

The convention that green means go and red means stop is presumably common knowledge among the drivers in our society.

Common and distributed knowledge

- Common knowledge also arises in discourse understanding.
- Suppose that Ann asks Bob “What did you think of the movie?”” referring to the movie Star Wars they have just seen. Ann and Bob must both know that the movie refers to Star Wars, Ann must know that Bob knows (so that she can be sure that Bob will give a reasonable answer to her question), Bob must know that Ann knows that Bob knows (so that Bob knows that Ann will respond appropriately to his answer), and so on.
- There must be common knowledge of what movie is meant in order for Bob to answer the question appropriately.

- Common knowledge also turns out to be a prerequisite for achieving agreement.
- That is why common knowledge is a crucial notion in the analysis of interacting groups of agents.

A group has distributed knowledge of a fact φ if the knowledge of φ is distributed among its members, so that by using their knowledge together the members of the group can deduce φ, even though it may be the case that no member of the group individually knows φ.

Example

Assume that Alice knows that Bob is in love with either Carol or Susan, and Charlie knows that Bob is not in love with Carol. Then together Alice and Charlie have distributed knowledge of the fact that Bob is in love with Susan, although neither Alice nor Charlie individually has this knowledge.

Let $\emptyset \neq G \subseteq Ag$ be a group of agents.

- Define, for every φ,

$$E_G\varphi = \bigwedge_{i \in G} K_i\varphi.$$

- $E_G\varphi$ is read as everyone in the group G knows φ.
- For every model $\mathcal{M}$ and every $w \in \mathcal{M}$,

$$\mathcal{M}, w \Vdash E_G\varphi \text{ iff } \mathcal{M}, w \Vdash K_i\varphi \text{ for all } i \in G.$$

Common and distributed knowledge

The language ML_{Ag} does not allow us to express the notions of common knowledge and distributed knowledge.

Let ML^{CD}_{Ag} be the language obtained by adding to ML_{Ag} the following modal operators for any $\emptyset \neq G \subseteq Ag$:

- C_G, read as it is common knowledge among the agents in G;
- D_G, read as it is distributed knowledge among the agents in G.

Formulas of ML^{CD}_{Ag} are defined as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \rightarrow \varphi \mid K_i\varphi \mid C_G\varphi \mid D_G\varphi,$$

where $p \in PROP$, $i \in Ag$ and $\emptyset \neq G \subseteq Ag$.

We omit the subscript G when G is the set of all agents.

Common and distributed knowledge

Let $\emptyset \neq G \subseteq Ag$ be a group of agents.

We define $E_G^k\varphi$ ($k \geq 0$) inductively:

$$E_G^0\varphi = \varphi, \qquad E_G^{k+1}\varphi = E_G E_G^k\varphi.$$

Let $\mathcal{M}$ be a model and $w \in \mathcal{M}$. We extend the definition of the satisfaction relation with the following clause:

$$\mathcal{M}, w \Vdash C_G\varphi \text{ iff } \mathcal{M}, w \Vdash E_G^k\varphi \text{ for all } k = 1, 2, \ldots.$$

Thus, the formula $C_G\varphi$ is true iff everyone in G knows φ, everyone in G knows that everyone in G knows φ, etc.

Our definition of common knowledge has a graph-theoretical interpretation.
Let $\mathcal{M}$ be a model.

Definition 3.3

Let w, v be states in $\mathcal{M}$.

- *We say that v is G-reachable from w in k steps ($k \geq 1$) if there exist states $u_0, u_1, \ldots, u_k \in \mathcal{M}$ such that $u_0 = w$, $u_k = v$ and for all $j = 0, \ldots, k-1$, there exists $i \in G$ such that $\mathcal{K}_i u_j u_{j+1}$.*
- *v is G-reachable from w if v is G-reachable from w in k steps for some $k \geq 1$.*

Thus, v is G-reachable from w iff there is a path in the graph from w to v whose edges are labeled by members of G.

Proposition 3.4

Let w be a state in $\mathcal{M}$.

- *The following are equivalent for every $k \geq 1$:*
  - $\mathcal{M}, w \Vdash E_G^k \varphi$;
  - $\mathcal{M}, v \Vdash \varphi$ *for all states v that are G-reachable from w in k steps.*
- $\mathcal{M}, w \Vdash C_G \varphi$ *iff* $\mathcal{M}, v \Vdash \varphi$ *for all states v that are G-reachable from w.*

Common and distributed knowledge

A group G has distributed knowledge of φ if the combined knowledge of the members of G implies φ.

- The question is how can we capture the idea of combining knowledge in our framework.
- The answer is: we combine the knowledge of the agents in group G by eliminating all worlds that some agent in G considers impossible.

Let $\mathcal{M}$ be a model and $w \in \mathcal{M}$. We extend the definition of the satisfaction relation with the following clause:

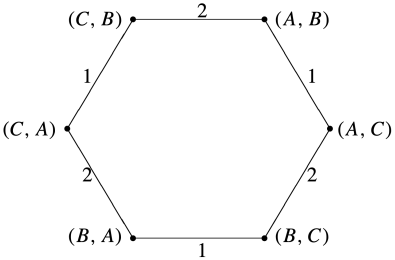
$$\begin{aligned} \mathcal{M}, w \Vdash D_G\varphi \quad & \text{iff} \quad \mathcal{M}, v \Vdash \varphi \text{ for all } v \text{ such that } (w, v) \in \textstyle\bigcap_{i \in G} \mathcal{K}_i \\ & \text{iff} \quad \mathcal{M}, v \Vdash \varphi \text{ for all } v \text{ such that } \mathcal{K}_i wv \text{ for all } i \in G. \end{aligned}$$

Example - the card game

Let $\mathcal{M}_c = (\mathcal{F}_c, V)$ be the model describing the simple card game.

- $PROP = \{iX \mid i \in \{1,2\}, X \in \{A, B, C\}\}$.
- iX read as agent i holds card X
- $V(iX) = \begin{cases} \{(X, Z) \mid Z \in \{A, B, C\} \setminus \{X\}\} & \text{if } i = 1 \\ \{(Z, X) \mid Z \in \{A, B, C\} \setminus \{X\}\} & \text{if } i = 2. \end{cases}$

$\mathcal{F}_c$ is given by



Example - the card game

Let $G = \{1, 2\}$.

- $\mathcal{M}_c \Vdash C_G(1A \vee 1B \vee 1C)$: it is common knowledge that agent 1 holds one of the cards A, B, and C.
- $\mathcal{M}_c \Vdash C_G(1B \to (2A \vee 2C))$: it is common knowledge that if agent 1 holds card B, then agent 2 holds either card A or card C.
- $\mathcal{M}_c, (A, B) \Vdash D_G(1A \wedge 2B)$: if the agents could combine their knowledge, they would know that in world (A, B), agent 1 holds card A and agent 2 holds card B.

Muddy children puzzle

- A group of n children enters their house after having played in the mud outside. They are greeted in the hallway by their father, who notices that k of the children have mud on their foreheads.
- He makes the following announcement, "At least one of you has mud on his forehead."
- The children can all see each other's foreheads, but not their own.
- The father then says, "Do any of you know that you have mud on your forehead? If you do, raise your hand now."
- No one raises his hand.
- The father repeats the question, and again no one moves.
- The father does not give up and keeps repeating the question.
- After exactly k repetitions, all the children with muddy foreheads raise their hands simultaneously.

Muddy children puzzle

For simplicity, let us call a child

- muddy if he has a muddy forehead;
- clean if he does not have a muddy forehead.

$k = 1$

- There exists only one muddy child.
- The muddy child knows the other children are clean.
- When the father says at least one is muddy, he concludes that it's him.
- None of the other children know at this point whether or not they are muddy.
- The muddy child raises his hand after the father's first question.
- After the muddy child raises his hand, the other children know that they are clean.

Muddy children puzzle

$k = 2$

- There exist two muddy children.
- Imagine that you are one of the two muddy children.
- You see that one of the other children is muddy.
- After the father's first announcement, you do not have enough information to know whether you are muddy. You might be, but it could also be that the other child is the only muddy one.
- So, you do not raise the hand after the father's first question.
- You note that the other muddy child does not raise his hand.
- You realize then that you yourself must be muddy as well, or else that child would have raised his hand.
- So, after the father's second question, you raise your hand. Of course, so does the other muddy child.

Muddy children puzzle

We shall analyse the muddy children puzzle using epistemic logic.

We assume that it is common knowledge that

- the father is truthful,
- all the children can and do hear the father,
- all the children can and do see which of the other children besides themselves have muddy foreheads,
- none of the children can see his own forehead,
- all the children are truthful and (extremely) intelligent.

Muddy children puzzle

Suppose that there are n children; we number them $1, \ldots, n$. Thus, we take $Ag = \{1, \ldots, n\}$.

- First consider the situation before the father speaks.
- Some of the children are muddy, while the rest are clean.
- We describe a possible situation by an n-tuple of 0's and 1's of the form $(x_1, \ldots, x_n)$, where $x_i = 1$ if child i is muddy, and $x_i = 0$ otherwise.
- There are 2^n possible situations.

Muddy children puzzle

$n = 3$

- Suppose that the actual situation is described by the tuple $(1, 0, 1)$, that says that child 1 and child 3 are muddy, while child 2 is clean.
- What situations does child 1 consider possible before the father speaks?
- Since child 1 can see the foreheads of all the children besides himself, his only doubt is about whether he is muddy or clean. Thus child 1 considers two situations possible: $(1, 0, 1)$ (the actual situation) and $(0, 0, 1)$. Similarly, child 2 considers two situations possible: $(1, 0, 1)$ and $(1, 1, 1)$.

In general, child i has the same information in two possible situations exactly if they agree in all components except possibly the ith component.

Muddy children puzzle

We can capture the general situation by the frame

$$\mathcal{F} = (W, \mathcal{K}_1, \ldots, \mathcal{K}_n),$$

where

- $W = \{(x_1, \ldots, x_n) \mid x_i \in \{0, 1\} \text{ for all } i = 1, \ldots, n\}$. Thus, W has 2^n states.
- For every $i = 1, \ldots, n$,

  $\mathcal{K}_i wv$ iff w and v agree in all components except possibly the ith component.
- One can easily see that $\mathcal{K}_i$'s are equivalence relations.

Thus, $\mathcal{F}$ is a frame for the epistemic logic $\boldsymbol{S5}$.

Muddy children puzzle

It remains to define $PROP$ and the valuation $V : PROP \to 2^W$.

- Since we want to reason about whether or not a given child is muddy, we take $PROP = \{p_1, \ldots, p_n, p\}$, where, intuitively, p_i stands for child i is muddy, while p stands for at least one child is muddy.
- We define V as follows:

$$V(p_i) = \{(x_1, \ldots, x_n) \in W \mid x_i = 1\},$$
$$V(p) = \{(x_1, \ldots, x_n) \in W \mid x_j = 1 \text{ for some } j = 1, \ldots, n\}.$$

- It follows that

$$\mathcal{M}, (x_1, \ldots, x_n) \Vdash p_i \text{ iff } x_i = 1,$$
$$\mathcal{M}, (x_1, \ldots, x_n) \Vdash p \text{ iff } x_j = 1 \text{ for some } j = 1, \ldots, n.$$

We have a model with 2^n nodes, each described by an n-tuple of 0's and 1's, such that two nodes are joined by an edge exactly if they differ in at most one component.

Muddy children puzzle

Recall that we omit self-loops and the arrows on edges.

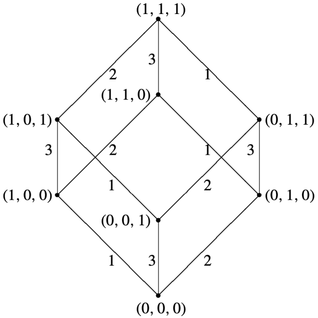
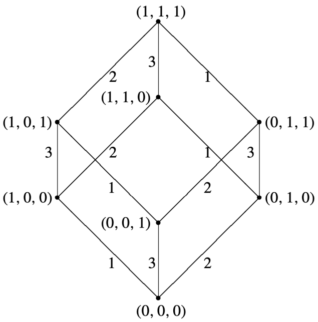


Figure 2: Frame for the muddy children puzzle with $n = 3$

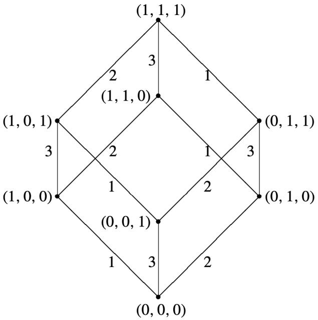
- $\mathcal{M}, (1, 0, 1) \Vdash K_1 \neg p_2$: child 1 knows that child 2 is clean
- $\mathcal{M}, (1, 0, 1) \Vdash K_1 p_3$: child 1 knows that child 3 is muddy
- $\mathcal{M}, (1, 0, 1) \Vdash \neg K_1 p_1$: child 1 does not know that he is muddy

Muddy children puzzle



- $\mathcal{M} \Vdash C(p_2 \rightarrow K_1 p_2)$: it is common knowledge that if child 2 is muddy, then child 1 knows it.
- $\mathcal{M} \Vdash C(\neg p_2 \rightarrow K_1 \neg p_2)$: it is common knowledge that if child 2 is clean, then child 1 knows it.

Muddy children puzzle



- $\mathcal{M}, (1, 0, 1) \Vdash Ep$: in $(1, 0, 1)$, every child knows that at least one child is muddy even before the father speaks;
- $\mathcal{M}, (1, 0, 1) \Vdash \neg E^2 p$: p is not true at the state $(0, 0, 0)$ that is reachable in two steps from $(1, 0, 1)$.

Muddy children puzzle

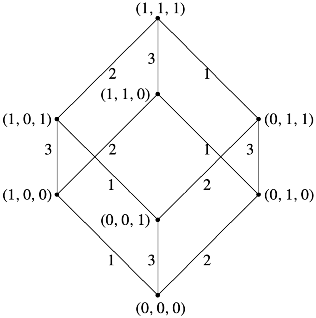
One can check that in the general case, if we have n children of whom k are muddy (so that the situation is described by an n-tuple exactly k of whose components are 1's), then $E^{k-1}p$ is true, but $E^k p$ is not, since each state reachable in $k-1$ steps has at least one 1 (and so there is at least one muddy child), but the tuple $(0, \ldots, 0)$ is reachable in k steps.

Muddy children puzzle

Let us consider what happens after the father speaks.

- The father says p, which is already known to all the children if there are two or more muddy children.
- Nevertheless, the state of knowledge changes, even if all the children already know p.

Muddy children puzzle



- In $(1, 0, 1)$, child 1 considers the situation $(0, 0, 1)$ possible and in $(0, 0, 1)$ child 3 considers $(0, 0, 0)$ possible.
- In $(1, 0, 1)$, before the father speaks, although everyone knows that at least one is muddy, child 1 thinks it possible that child 3 thinks it possible that none of the children is muddy.
- After the father speaks, it becomes common knowledge that at least one child is muddy.

Muddy children puzzle

- In the general case, we can represent the change in the group's state of knowledge graphically by simply removing the point $(0, 0, \ldots, 0)$ from the cube.
- More accurately, what happens is that the node $(0, 0, \ldots, 0)$ remains, but all the edges between$(0, 0, \ldots, 0)$ and nodes with exactly one 1 disappear, since it is common knowledge that even if only one child is muddy, after the father speaks that child will not consider it possible that no one is muddy.

Muddy children puzzle

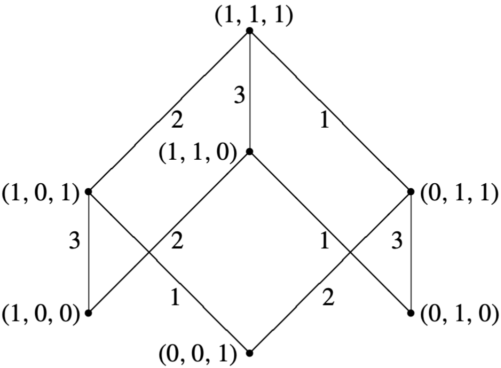


Figure 3: Frame for $n = 3$ after the father speaks

Muddy children puzzle

Let us show that each time the children respond to the father's question with a No, the group's state of knowledge changes and the cube is further truncated.

- Consider what happens after the children respond No to the father's first question.
- Now all the nodes with exactly one 1 can be eliminated. (More accurately, the edges to these nodes from nodes with exactly two 1's all disappear from the graph.)
- Nodes with one or fewer 1's are no longer reachable from nodes with two or more 1's.

Muddy children puzzle

- If the actual situation were described by, say, the tuple $(1, 0, \ldots, 0)$, then child 1 would initially consider two situations possible: $(1, 0, \ldots, 0)$, and $(0, 0, \ldots, 0)$.
- Since once the father speaks it is common knowledge that $(0, 0, \ldots, 0)$ is not possible, he would then know that the situation is described by $(1, 0, \ldots, 0)$, and thus would know that he is muddy.
- Once everyone answers No to the father's first question, it is common knowledge that the situation cannot be $(1, 0, \ldots, 0)$.
- Similar reasoning allows us to eliminate every situation with exactly one 1. Thus, after all the children have answered No to the father's first question, it is common knowledge that at least two children are muddy.

Muddy children puzzle

- Further arguments in the same spirit can be used to show that after the children answer No k times, we can eliminate all the nodes with at most k 1's (or, more accurately, disconnect these nodes from the rest of the graph).
- We thus have a sequence of frames, describing the children's knowledge at every step in the process.
- Essentially, what is going on is that if, in some node w, it becomes common knowledge that a node v is impossible, then for every node u reachable from w, the edge from u to v (if there is one) is eliminated.

Muddy children puzzle

- After k rounds of questioning, it is common knowledge that at least $k + 1$ children are muddy.
- If the true situation is described by a tuple with exactly $k + 1$ 1's, then before the father asks the question for the $(k + 1)$st time, the muddy children will know the exact situation, and in particular will know they are muddy, and consequently will answer Yes.
- Note that they could not answer Yes any earlier, since up to this point each muddy child considers it possible that he or she is clean.

Muddy children puzzle

- According to the way we are modeling knowledge in this context, a child knows a fact if the fact follows from his or her current information.
- However, if one of the children were not particularly bright, then he might not be able to figure out that he knew that he is muddy, even though in principle he had enough information to do so.
- To answer Yes to the father's question, the child must actually be aware of the consequences of his information.
- Our definition implicitly assumes that (it is common knowledge that) all reasoners are logically omniscient, that is they are smart enough to compute all the consequences of the information that they have.
- Furthermore, this logical omniscience is common knowledge.

Partition models of knowledge

Partition model of knowledge

Partition models of knowledge are defined in

Yoav Shoham, Kevin Leyton-Brown, Multiagents Systems, Cambridge University Press, 2009

Let $n \geq 1$ and $AG = \{1, \ldots, n\}$ be the set of agents.

Definition 3.5 (Partition frame)

A partition frame is a tuple $\mathcal{P}_F = (W, I_1, \ldots, I_n)$, where

- *W is a nonempty set of possible worlds.*
- *For every $i = 1, \ldots, n$, I_i is a partition of W.*

The idea is that I_i partitions W into sets of possible worlds that are indistinguishable from the point of view of agent i.

Partition model of knowledge

Recall: Let A be a nonempty set. A partition of A is a family $(A_j)_{j\in J}$ of nonempty subsets of A satisfying the following properties:

$$A = \bigcup_{j\in J} A_j \text{ and } A_j \cap A_k = \emptyset \text{ for all } j \neq k.$$

Recall: Let A be a nonempty set. There exists a bijection between the set of partitions of A and the set of equivalence relations on A:

- $(A_j)_{j\in J}$ partition of $A \mapsto$ the equivalence relation on A defined by $x \sim y \Leftrightarrow$ there exists $j \in J$ such that $x, y \in A_j$.
- $\sim$ equivalence relation on $A \mapsto$ the partition consisting of all the different equivalence classes of $\sim$.

Partition model of knowledge

- For each $i = 1, \ldots, n$, let R_{I_i} be the corresponding equivalence relation.
- Denote by $I_i(w)$ the equivalence class of w in the relation R_{I_i}.
- If the actual world is w, then $I_i(w)$ is the set of possible worlds that agent i cannot distinguish from w.
- $\mathcal{F} = (W, R_{I_1}, \ldots, R_{I_n})$ is a frame for the epistemic logic $\boldsymbol{S5}$.

Partition frame = frame for the epistemic logic $\boldsymbol{S5}$

Definition 3.6 (Partition model)

A partition model over a language Σ *is a tuple* $\mathcal{P}_M = (\mathcal{P}_F, \pi)$*, where*

- $\mathcal{P}_F = (W, I_1, \ldots, I_n)$ *is a partition frame.*
- $\pi : \Sigma \to 2^W$ *is an interpretation function.*

For every statement $\varphi \in \Sigma$, we think of $\pi(\varphi)$ as the set of possible worlds in the partition model $\mathcal{P}_M$ where φ is satisfied.

- Each possible world completely specifies the concrete state of affairs.
- We can take, for example, Σ to be a set of formulas in propositional logic over some set of atomic propositions.

Partition model of knowledge

We will use the notation $K_i\varphi$ as "agent i knows that φ".

The following defines when a statement is true in a partition model.

Definition 3.7 (Logical entailment for partition models)

Let $\mathcal{P}_M = (W, I_1, \ldots, I_n, \pi)$ be a partition model over Σ, and $w \in W$. We define the $\models$ (logical entailment) relation as follows:

- *For any $\varphi \in \Sigma$, we say that $\mathcal{P}_M, w \models \varphi$ iff $w \in \pi(\varphi)$.*
- *$\mathcal{P}_M, w \models K_i\varphi$ iff for all worlds $v \in W$, if $v \in I_i(w)$, then $\mathcal{P}_M, v \models \varphi$.*

Partition model = model for the epistemic logic ***S5***

We can reason about knowledge rigorously in terms of partition models, hence using epistemic logic.