A4M33MAS - Multiagent Systems
Agents and their behavior modeling by means of formal logic

Michal Pechoucek & Michal Jakob
Department of Computer Science
Czech Technical University in Prague

In parts based on selected graphics taken from Valentin Goranko and Wojtek Jamroga: Modal Logics for Multi-Agent Systems, 8th European Summer School in Logic Language and Information
Multi-agent systems & Logic

• Multi-agent systems
  – Complex decentralized systems whose behaviour is given by interaction among autonomous, rational entities. We study MAS so that we understand behaviour of such systems and can design such software systems.

• Logic
  – Provides a paradigm for modeling and reasoning about the complex world in a precise and exact manner
  – Provides methodology for specification and verification of complex programs

• Can be used for practical things (also in MAS):
  – automatic verification of multi-agent systems
  – and/or executable specifications of multi-agent systems
Best logic for MAS?
Modal logic is an extension of classical logic by new connectives $\square$ and $\lozenge$: necessity and possibility.

- $\square \varphi$ means that $\varphi$ is necessarily true
- $\lozenge \varphi$ means that $\varphi$ is possibly true

Independently of the precise definition, the following holds:

$$\lozenge \varphi \iff \neg \square \neg \varphi$$
Modal logic syntax

Definition 1.1 (Modal Logic with $n$ modalities)

The language of modal logic with $n$ modal operators $\square_1, \ldots, \square_n$ is the smallest set containing:

- atomic propositions $p, q, r, \ldots$;
- for formulae $\varphi$, it also contains $\neg \varphi, \square_1 \varphi, \ldots, \square_n \varphi$;
- for formulae $\varphi, \psi$, it also contains $\varphi \land \psi$.

We treat $\lor, \rightarrow, \leftrightarrow, \Box$ as macros (defined as usual).

Note that the modal operators can be nested:

$$(\square_1 \square_2 \Diamond_1 p) \lor \square_3 \neg p$$
Modal logic syntax

More precisely, necessity/possibility is interpreted as follows:

- \( p \) is necessary  \( \iff \) \( p \) is true in all possible scenarios
- \( p \) is possible  \( \iff \) \( p \) is true in at least one possible scenario

\( \rightsquigarrow \) possible worlds semantics
Modal logic semantics

Definition 1.2 (Kripke Structure)
A Kripke structure is a tuple $\langle \mathcal{W}, \mathcal{R} \rangle$, where $\mathcal{W}$ is a set of possible worlds, and $\mathcal{R}$ is a binary relation on worlds, called accessibility relation.

Definition 1.3 (Kripke model)
A possible worlds model $\mathcal{M} = \langle \mathcal{S}, \pi \rangle$ consists of a Kripke structure $\mathcal{S}$, and a valuation of propositions $\pi : \mathcal{W} \rightarrow \mathcal{P}(\{p, q, r, \ldots\})$. 
Modal logic semantics

Remarks:

- $\mathcal{R}$ indicates which worlds are relevant for each other; $w_1 \mathcal{R} w_2$ can be read as “world $w_2$ is relevant for (reachable from) world $w_1$”

- $\mathcal{R}$ can be any binary relation from $\mathcal{W} \times \mathcal{W}$; we do not require any specific properties (yet).
Modal logic semantics

Remarks:

- $\mathcal{R}$ indicates which worlds are relevant for each other; $w_1 \mathcal{R} w_2$ can be read as “world $w_2$ is relevant for (reachable from) world $w_1$”

- $\mathcal{R}$ can be any binary relation from $\mathcal{W} \times \mathcal{W}$; we do not require any specific properties (yet).

- It is natural to see the worlds from $\mathcal{W}$ as classical propositional models, i.e. valuations of propositions $\pi(w) \subseteq \{p, q, r, \ldots\}$. 
Definition 1.4 (Semantics of modal logic)

The truth of formulae is relative to a Kripke model $\mathcal{M} = \langle W, R, \pi \rangle$, and a world $w \in W$. It can be defined through the following clauses:

- $\mathcal{M}, w \models p$ iff $p \in \pi(w)$;
- $\mathcal{M}, w \models \neg \varphi$ iff not $\mathcal{M}, w \models \varphi$;
- $\mathcal{M}, w \models \varphi \land \psi$ iff $\mathcal{M}, w \models \varphi$ and $\mathcal{M}, w \models \psi$;
- $\mathcal{M}, w \models \Box \varphi$ iff, for every $w' \in W$ such that $wRw'$, we have $\mathcal{M}, w' \models \varphi$. 
Modal logic example

W₁
run

W₂
stop
Modal logic example

W₁
run

W₂
stop

run → ◊stop
Modal logic example

\[
\begin{align*}
W_1 & \quad \text{run} \\
W_2 & \quad \text{stop}
\end{align*}
\]

\[
\begin{align*}
\text{run} & \rightarrow \Diamond \text{stop} \\
\text{stop} & \rightarrow \Box \text{stop}
\end{align*}
\]
Modal logic example

\[
\begin{align*}
\text{run} & \rightarrow \Diamond \text{stop} \\
\text{stop} & \rightarrow \Box \text{stop} \\
\text{run} & \rightarrow \Diamond \Box \text{stop}
\end{align*}
\]
Modal logic

• Note:
  – most modal logics can be translated to classical logic
    
    ... but the result looks horribly ugly,
    
    ... and in most cases it is hard to automate anything
Axioms in Modal logic

Definition 1.5 (System K)
System $K$ is an extension of the propositional calculus by the axiom

$K \quad (\Box \varphi \land \Box (\varphi \rightarrow \psi)) \rightarrow \Box \psi$

Distribution axiom
Axioms in Modal logic

Definition 1.5 (System K)

System K is an extension of the propositional calculus by the axiom

\[ \text{Distribution axiom} \]

\[ K \ (\Box \varphi \land \Box (\varphi \rightarrow \psi)) \rightarrow \Box \psi \]

and the inference rule

\[ \text{Generalization axiom} \]

\[ \frac{}{\Box \varphi} \]
Theorem 1.6 (Soundness/completeness of system K)

System $K$ is sound and complete with respect to the class of all Kripke models.
Definition 1.7 (Extending K with axioms D, T, 4, 5)

System K is often extended by (a subset of) the following axioms (called as below for historical reasons):

- **T**: $\Box \varphi \rightarrow \varphi$
- **D**: $\Box \varphi \rightarrow \Diamond \varphi$
- **4**: $\Box \varphi \rightarrow \Box \Box \varphi$
- **B**: $\varphi \rightarrow \Box \Diamond \varphi$
- **5**: $\Diamond \varphi \rightarrow \Box \Diamond \varphi$
Proofs
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due [reflexivity] \( \forall w : (w, w) \in R \otimes \)

T: $\Box \varphi \rightarrow \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R$.

D: $(M_1 \models_w \varphi. \forall w' : (w, w') \in R \Rightarrow M_1 \models_w \varphi)$ and due to seriality $(M_1 \models_w \exists w' : (w, w') \in R))$ we can say that $M_1 \models_w \exists w'' : (w, w'') \in R : M_1 \models_w \varphi$.

D: $\Box \varphi \rightarrow \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to \underline{reflexivity} $\forall w : (w, w) \in R \otimes$

D: $(\mathcal{M}_1 \models_w \varphi \wedge \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi)$ and due to \underline{seriality} $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R)$ we can say that $\mathcal{M}_1 \models_w \exists w'' : (w, w'') \in R : \mathcal{M}_1 \models_w \varphi \otimes$

D: $\Box \varphi \rightarrow \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \square \varphi$ and due to reflexivity $\forall w : (w, w) \in R$ ⊗

D: $(\mathcal{M}_1 \models_w \varphi. \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi)$ and due to seriality $(\mathcal{M}_1 \models_w (\exists w' : (w, w') \in R))$

we can say that $\mathcal{M}_1 \models_w \exists w'' : (w, w'') \in R : \mathcal{M}_1 \models_{w''} \varphi$ ⊗

D: $\square \varphi \rightarrow \diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R \odot$

D: $(\mathcal{M}_1 \models_w \varphi \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi)$ and due to seriality $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R))$ we can say that $\mathcal{M}_1 \models_w \exists \forall w'' : (w, w'') \in R : \mathcal{M}_1 \models_{w''} \varphi) \odot$

4: provided that there is transitive relation on $R$ we may say that $(\mathcal{M}_1 \models_w \varphi \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} (\forall w'' : (w', w'') \in R : \mathcal{M}_1 \models_{w''} \varphi)) \odot$

4: $\Box \varphi \rightarrow \Box \Box \Box \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R \circledast$

D: $(\mathcal{M}_1 \models_w \varphi. \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \varphi)$ and due to seriality $(\mathcal{M}_1 \models_w (\exists w' : (w, w') \in R))$

we can say that $\mathcal{M}_1 \models_w \exists w'' : (w, w'') \in R : \mathcal{M}_1 \models_{w''} \varphi) \circledast$

4: provided that there is transitive relation on $R$ we may say that $(\mathcal{M}_1 \models_w \varphi. \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \varphi) \rightarrow (\mathcal{M}_1 \models_w \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} (\forall w'' : (w', w'') \in R : \mathcal{M}_1 \models_{w''} \varphi)) \circledast$

4: $\Box \varphi \rightarrow \Box \Box \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R \Diamond$

D: $(\mathcal{M}_1 \models_w \varphi \, \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi)$ and due to seriality $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R))$ we can say that $\mathcal{M}_1 \models_w \exists w'' : (w, w'') \in R : \mathcal{M}_1 \models_{w''} \varphi \Diamond$

4: provided that there is transitive relation on $R$ we may say that $(\mathcal{M}_1 \models_w \varphi \, \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \forall w'' : (w', w'') \in R : \mathcal{M}_1 \models_{w''} \varphi)) \Diamond$

4: $\Box \varphi \rightarrow \Box \Box \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R \circledast$

D: $(M_1 \models_w \varphi. \forall w' : (w, w') \in R : M_1 \models_{w'} \varphi)$ and due to seriality $(M_1 \models_w (\exists w' : (w, w') \in R))$ we can say that $M_1 \models_w \exists w'' : (w, w'') \in R : M_1 \models_{w''} \varphi \circledast$

4: provided that there is transitive relation on $R$ we may say that $(M_1 \models_w \varphi. \forall w' : (w, w') \in R : M_1 \models_{w'} \varphi) \Rightarrow (M_1 \models_w \forall w' : (w, w') \in R : M_1 \models_{w'} (\forall w'' : (w', w'') \in R : M_1 \models_{w''} \varphi)) \circledast$

B: provided that there is symmetric relation on $R$ we say that $M_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : M_1 \models_{w'} \exists w'' : (w', w'') \in R : M_1 \models_{w''} \varphi$ if $(\forall w, w', (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $M_1 \models_w \varphi \circledast$

B: $\varphi \rightarrow \Box \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \square \varphi$ and due to reflexivity $\forall w : (w, w) \in R \circ$

D: $(M_1 \models_w \varphi. \forall w' : (w, w') \in R : M_1 \models_w \varphi)$ and due to seriality $(M_1 \models_w (\exists w' : (w, w') \in R)) we can say that $M_1 \models_w \exists w'' : (w, w'') \in R : M_1 \models_w \varphi \circ$

4: provided that there is transitive relation on $R$ we may say that $(M_1 \models_w \varphi. \forall w' : (w, w') \in R : M_1 \models_w \varphi) \Rightarrow (M_1 \models_w \forall w' : (w, w') \in R : M_1 \models_w (\forall w'' : (w', w'') \in R : M_1 \models_w \varphi)) \circ$

B: provided that there is symmetric relation on $R$ we say that $M_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : M_1 \models_{w'} \exists w'' : (w', w'') \in R : M_1 \models_{w''} \varphi$ if $(\forall w, w' : (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $M_1 \models_w \varphi \circ$

B: $\varphi \rightarrow \square \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R$

D: $(M_1 \models_w \varphi. \forall w' : (w, w') \in R : M_1 \models_{w'} \varphi)$ and due to seriality $(M_1 \models_w (\exists w' : (w, w') \in R))$ we can say that $M_1 \models_w \exists w'' : (w, w'') \in R : M_1 \models_{w''} \varphi$

4: provided that there is transitive relation on $R$ we may say that $(M_1 \models_w \varphi \forall w' : (w, w') \in R : M_1 \models_{w'} \varphi) \Rightarrow (M_1 \models_w \forall w' : (w, w') \in R : M_1 \models_{w'} (\forall w'' : (w', w'') \in R : M_1 \models_{w''} \varphi))$

B: provided that there is symmetric relation on $R$ we say that $M_1 \models_w \exists w'' : (w', w'') \in R : M_1 \models_{w''} \varphi$ if $(\forall w, w', (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $M_1 \models_w \varphi$

B: $\varphi \rightarrow \Box \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \square \varphi$ and due to reflexivity $\forall w : (w, w) \in R \Rightarrow$

D: $(\mathcal{M}_1 \models_w \varphi. \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi)$ and due to seriality $(\mathcal{M}_1 \models_w (\exists w' : (w, w') \in R))$ we can say that $\mathcal{M}_1 \models_w \exists w'' : (w, w'') \in R : \mathcal{M}_1 \models_{w''} \varphi$.

4: provided that there is transitive relation on $R$ we may say that $(\mathcal{M}_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} (\forall w'' : (w', w'') \in R : \mathcal{M}_1 \models_{w''} \varphi))$.

B: provided that there is symmetric relation on $R$ we say that $\mathcal{M}_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \exists w'' : (w', w'') \in R : \mathcal{M}_1 \models_{w''} \varphi$ if $(\forall w, w', (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $\mathcal{M}_1 \models_w \varphi$.

5: $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R \models_{w'} \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w'' : (w, w'') \in R : \mathcal{M}_1 \models_{w''} \exists w'(w'', w') \in R : \mathcal{M}_1 \models_{w'} \varphi)$ due to euclidean property if $(w, w') \in R \wedge (w, w'') \in R$ then $(w', w'') \in R$.

5: $\Diamond \varphi \rightarrow \Box \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \square \varphi$ and due to **reflexivity** $\forall w : (w, w) \in R$ ⊗

D: $(\mathcal{M}_1 \models_w \varphi. \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \varphi)$ and due to **seriality** $(\mathcal{M}_1 \models_w (\exists w' : (w, w') \in R))$
we can say that $\mathcal{M}_1 \models_w \exists w'' : (w, w'') \in R : \mathcal{M}_1 \models_w \varphi$ ⊗

4: provided that there is **transitive** relation on $R$ we may say that $(\mathcal{M}_1 \models_w \varphi \ \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w (\forall w'' : (w', w'') \in R : \mathcal{M}_1 \models_w \varphi))$ ⊗

B: provided that there is **symmetric** relation on $R$ we say that $\mathcal{M}_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \exists w'' : (w', w'') \in R : \mathcal{M}_1 \models_w \varphi$ if $(\forall w, w', (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $\mathcal{M}_1 \models_w \varphi$ ⊗

5: $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R : \models_w \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w'' : (w, w'') \in R : \mathcal{M}_1 \models_w \exists w'(w'', w') \in R : \mathcal{M}_1 \models_w \varphi)$ due to **euclidean** property if $(w, w') \in R \land (w, w'') \in R$ then $(w', w'') \in R$ ⊗

5: $\Diamond \varphi \Rightarrow \square \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R \circ$

D: $(\mathcal{M}_1 \models_w \varphi. \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \varphi)$ and due to seriality $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R)$ we can say that $\mathcal{M}_1 \models_w \exists w'' : (w, w'') \in R : \mathcal{M}_1 \models_w \varphi \circ$

4: provided that there is transitive relation on $R$ we may say that $(\mathcal{M}_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} (\forall w'' : (w', w'') \in R : \mathcal{M}_1 \models_{w''} \varphi)) \circ$

B: provided that there is symmetric relation on $R$ we say that $\mathcal{M}_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : \mathcal{M}_1 \models_{w'} \varphi$ if $(\forall w, w', (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $\mathcal{M}_1 \models_w \varphi \circ$

5: $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R \models_{w'} \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w'' : (w, w'') \in R : \mathcal{M}_1 \models_{w''} \exists w'(w'', w') \in R : \mathcal{M}_1 \models_{w'} \varphi)$ due to euclidean property if $(w, w') \in R \wedge (w, w'') \in R$ then $(w', w'') \in R \circ$

5: $\Diamond \varphi \Rightarrow \Box \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \square \varphi$ and due to reflexivity $\forall w : (w, w) \in R$ ⊗

D: $(M_1 \models_w \varphi \land \forall w' : (w, w') \in R : M_1 \models_w \varphi)$ and due to seriality $(M_1 \models_w (\exists w' : (w, w') \in R))$ we can say that $M_1 \models_w \exists w'' : (w, w'') \in R : M_1 \models_{w''} \varphi$ ⊗

4: provided that there is transitive relation on $R$ we may say that $(M_1 \models_w \varphi \land \forall w' : (w, w') \in R : M_1 \models_w \varphi) \Rightarrow (M_1 \models_w \forall w' : (w, w') \in R : M_1 \models_w (\forall w'' : (w', w'') \in R : M_1 \models_{w''} \varphi))$ ⊗

B: provided that there is symmetric relation on $R$ we say that $M_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : M_1 \models_{w'} \exists w'' : (w', w'') \in R : M_1 \models_{w''} \varphi$ if $(\forall w, w', (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $M_1 \models_w \varphi$ ⊗

5: $(M_1 \models_w \exists w' : (w, w') \in R \models_{w'} \varphi) \Rightarrow (M_1 \models_w \forall w'' : (w, w'') \in R : M_1 \models_{w''} \exists w'(w'', w') \in R : M_1 \models_{w'} \varphi)$ due to euclidean property if $(w, w') \in R \land (w, w'') \in R$ then $(w', w'') \in R$ ⊗

5: $\Diamond \varphi \rightarrow \Box \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R \circledast$

D: $(\mathcal{M}_1 \models_w \varphi. \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \varphi)$ and due to seriality $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R)$
we can say that $\mathcal{M}_1 \models_w \exists w'' : (w, w'') \in R : \mathcal{M}_1 \models_w \varphi \circledast$

4: provided that there is transitive relation on $R$ we may say that $(\mathcal{M}_1 \models_w \varphi. \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w (\forall w'' : (w', w'') \in R : \mathcal{M}_1 \models_w \varphi)) \circledast$

B: provided that there is symetric relation on $R$ we say that $\mathcal{M}_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : \mathcal{M}_1 \models_w \exists w'' : (w', w'') \in R : \mathcal{M}_1 \models_w \varphi$ if $(\forall w, w', (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $\mathcal{M}_1 \models_w \varphi \circledast$

5: $(\mathcal{M}_1 \models_w \exists w' : (w, w') \in R \models_w \varphi) \Rightarrow (\mathcal{M}_1 \models_w \forall w'' : (w, w'') \in R : \mathcal{M}_1 \models_w \exists w' (w'', w') \in R : \mathcal{M}_1 \models_w \varphi)$ due to euclidean property if $(w, w') \in R \land (w, w'') \in R$ then $(w', w'') \in R \circledast$

5: $\Diamond \varphi \rightarrow \Box \Diamond \varphi$
Proofs

T: because $\models \varphi \Rightarrow \Box \varphi$ and due to reflexivity $\forall w : (w, w) \in R \circ$

D: $(M_1 \models_w \varphi. \forall w' : (w, w') \in R : M_1 \models_w \varphi)$ and due to seriality $(M_1 \models_w (\exists w' : (w, w') \in R))$
we can say that $M_1 \models_w \exists w'' : (w, w'') \in R : M_1 \models_w \varphi \circ$

4: provided that there is transitive relation on $R$ we may say that $(M_1 \models_w \varphi \forall w' : (w, w') \in R : M_1 \models_w \varphi) \Rightarrow (M_1 \models_w \forall w' : (w, w') \in R : M_1 \models_w \forall w'' : (w', w'') \in R : M_1 \models_w \varphi) \circ$

B: provided that there is symmetric relation on $R$ we say that $M_1 \models_w \varphi \Rightarrow \forall w' : (w, w') \in R : M_1 \models_w \exists w'' : (w', w'') \in R : M_1 \models_w \varphi$ if $(\forall w, w', (w, w') \in R \Rightarrow (w', w) \in R)$ then $w = w''$ and $M_1 \models_w \varphi \circ$

5: $(M_1 \models_w \exists w' : (w, w') \in R \models_w \varphi) \Rightarrow (M_1 \models_w \forall w'' : (w, w'') \in R : M_1 \models_w \exists w' : (w', w'') \in R : M_1 \models_w \varphi)$ due to euclidean property if $(w, w') \in RA \land (w, w'') \in R$ then $(w', w'') \in R \circ$

5: $\Diamond \varphi \rightarrow \Box \Diamond \varphi$
Axioms in Modal logic

- **T**: \( \Box \varphi \rightarrow \varphi \)  due to reflexivity
- **D**: \( \Box \varphi \rightarrow \Diamond \varphi \)  due to seriality
- **4**: \( \Box \varphi \rightarrow \Box \Box \varphi \)  due to transitivity
- **B**: \( \varphi \rightarrow \Box \Diamond \varphi \)  due to symmetricity
- **5**: \( \Diamond \varphi \rightarrow \Box \Diamond \varphi \)  due to euclidean property
Model of Belief & Knowledge
Model of Belief & Knowledge

• Once we are implementing an intelligent agent what do we want the program to implement e.g. its beliefs:
  – to satisfy the K axioms
  – an agent knows what it does know: positive introspection axiom (4 axiom).
  – an agent knows what it does not know: negative introsp. axiom (5 axiom).
  – it beliefs are not contradictory: if it knows something it means it does not allow the negation of its being true (D axiom).
Model of Belief & Knowledge

• Once we are implementing an intelligent agent what do we want the program to implement e.g. its beliefs:
  – to satisfy the K axioms
  – an agent knows what it does know: positive introspection axiom (4 axiom).
  – an agent knows what it does not know: negative introsp. axiom (5 axiom).
  – it beliefs are not contradictory: if it knows something it means it does not allow the negation of its being true (D axiom).

• Belief is surely a KD45 system -- modal logic system where the B relation is serial, transitive and euclidean.

- T: $\Box \varphi \rightarrow \varphi$ due to reflexivity
- D: $\Box \varphi \rightarrow \Diamond \varphi$ due to seriality
- 4: $\Box \varphi \rightarrow \Box \Box \varphi$ due to transitivity
- B: $\varphi \rightarrow \Box \Diamond \varphi$ due to symmetricity
- 5: $\Diamond \varphi \rightarrow \Box \Diamond \varphi$ due to euclidean property
Model of Belief & Knowledge

- Once we are implementing an intelligent agent what do we want the program to implement e.g. its beliefs:
  - to satisfy the K axioms
  - an agent knows what it does know: positive introspection axiom (4 axiom).
  - an agent knows what it does not know: negative introsp. axiom (5 axiom).
  - it beliefs are not contradictory: if it knows something it means it does not allow the negation of its being true (D axiom).
- Belief is surely a KD45 system -- modal logic system where the B relation is serial, transitive and euclidean.
- Knowledge is more difficult – it needs to be also true
  – this why the knowledge accessibility relation needs to be also reflexive.
Model of Belief & Knowledge

• Once we are implementing an intelligent agent what do we want the program to implement e.g. its beliefs:
  – to satisfy the K axioms
  – an agent knows what it does know: positive introspection axiom (4 axiom).
  – an agent knows what it does not know: negative introspection axiom (5 axiom).
  – it beliefs are not contradictory: if it knows something it means it does not allow the negation of its being true (D axiom).

• Belief is surely a KD45 system -- modal logic system where the B relation is serial, transitive and euclidean.

• Knowledge is more difficult – it needs to be also true
  – this why the knowledge accessibility relation needs to be also reflexive.

• Therefore knowledge is a KTD45 system.
Automated reasoning in Logic

\( \varphi \) can be true in \( \mathcal{M} \) and \( q \) (\( \mathcal{M}, q \models \varphi \))
Automated reasoning in Logic

- $\varphi$ can be true in $M$ and $q$ ($M, q \models \varphi$)
- $\varphi$ can be valid in $M$ ($M, q \models \varphi$ for all $q$)
Automated reasoning in Logic

- \( \varphi \) can be **true** in \( \mathcal{M} \) and \( q \) (\( \mathcal{M}, q \models \varphi \))
- \( \varphi \) can be **valid** in \( \mathcal{M} \) (\( \mathcal{M}, q \models \varphi \) for all \( q \))
- \( \varphi \) can be **valid** (\( \mathcal{M}, q \models \varphi \) for all \( \mathcal{M}, q \))
Automated reasoning in Logic

- \( \varphi \) can be true in \( \mathcal{M} \) and \( q \) (\( \mathcal{M}, q \models \varphi \))
- \( \varphi \) can be valid in \( \mathcal{M} \) (\( \mathcal{M}, q \models \varphi \) for all \( q \))
- \( \varphi \) can be valid (\( \mathcal{M}, q \models \varphi \) for all \( \mathcal{M}, q \))
- \( \varphi \) can be satisfiable (\( \mathcal{M}, q \models \varphi \) for some \( \mathcal{M}, q \))
Automated reasoning in Logic

- $\varphi$ can be true in $\mathcal{M}$ and $q$ ($\mathcal{M}, q \models \varphi$)
- $\varphi$ can be valid in $\mathcal{M}$ ($\mathcal{M}, q \models \varphi$ for all $q$)
- $\varphi$ can be valid ($\mathcal{M}, q \models \varphi$ for all $\mathcal{M}, q$)
- $\varphi$ can be satisfiable ($\mathcal{M}, q \models \varphi$ for some $\mathcal{M}, q$)
- $\varphi$ can be a theorem (it can be derived from the axioms via inference rules)
Automated reasoning in Logic

- model checking (local): “given $\mathcal{M}$, $q$, and $\varphi$, is $\varphi$ true in $\mathcal{M}, q$?”

- model checking (global): “given $\mathcal{M}$ and $\varphi$, what is the set of states in which $\varphi$ is true?”
Automated reasoning in Logic

- **model checking (local)**: “given $M$, $q$, and $\varphi$, is $\varphi$ true in $M$, $q$?”
- **model checking (global)**: “given $M$ and $\varphi$, what is the set of states in which $\varphi$ is true?”

**Model checking** is a technique for automatically verifying correctness properties of finite-state systems. Given a model of a system, exhaustively and automatically check whether this model meets a given specification (such as the absence of deadlocks and similar critical states that can cause the system to crash).
Automated reasoning in Logic

■ model checking (local): “given $M$, $q$, and $\varphi$, is $\varphi$ true in $M$, $q$?”
■ model checking (global): “given $M$ and $\varphi$, what is the set of states in which $\varphi$ is true?”
■ satisfiability: “given $\varphi$, is $\varphi$ true in at least one model and state?”
■ validity: “given $\varphi$, is $\varphi$ true in all models and their states?”
■ theorem proving: “given $\varphi$, is it possible to prove (derive) $\varphi$?”
Various Modal Logics

Modal logic is a **generic** framework.

Various modal logics:
- knowledge $\leadsto$ **epistemic logic**,  
- beliefs $\leadsto$ **doxastic logic**, 
- obligations $\leadsto$ **deontic logic**, 
- actions $\leadsto$ **dynamic logic**, 
- time $\leadsto$ **temporal logic**, 
- ability $\leadsto$ **strategic logic**, 
- and combinations of the above
Model of Time
Model of Time

• Modelling time as an instance of modal logic where the accessibility relation represents the relationship between the past, current and future time moments.

• Time:

- Linear

- Branching
## Typical Temporal Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\chi \varphi$</td>
<td>$\varphi$ is true in the <strong>next</strong> moment in time</td>
</tr>
<tr>
<td>$G \varphi$</td>
<td>$\varphi$ is true in all future moments</td>
</tr>
<tr>
<td>$F \varphi$</td>
<td>$\varphi$ is true in <strong>some</strong> future moment</td>
</tr>
<tr>
<td>$\varphi U \psi$</td>
<td>$\varphi$ is true <strong>until</strong> the moment when $\psi$ becomes true</td>
</tr>
</tbody>
</table>

$$G((\neg \text{passport} \lor \neg \text{ticket}) \rightarrow \chi \neg \text{board\_flight})$$

$$\text{send}(\text{msg}, \text{rcvr}) \rightarrow F \text{receive}(\text{msg}, \text{rcvr})$$
Safety Property

- *something bad will not happen*
- *something good will always hold*
Safety Property

- *something bad will not happen*
- *something good will always hold*

• Typical examples:
  \[ \mathcal{G} \neg \text{bankrupt} \]
Safety Property

- something bad will not happen
- something good will always hold

• Typical examples:

\( \neg \text{bankrupt} \)
\( G(\text{fuelOK} \lor X\text{fuelOK}) \)
and so on ...
Safety Property

- something bad will not happen
- something good will always hold

• Typical examples:

\[ G \neg \text{bankrupt} \]
\[ G(\text{fuelOK} \lor \Box \text{fuelOK}) \]

and so on . . .

Usually: \[ G \neg . . . \]
Liveness Property

- *something good will happen*
Liveness Property

- something good will happen

• Typical examples

$\mathcal{F}_{\text{rich}}$
Liveness Property

- something good will happen

• Typical examples

\[ F_{\text{rich}} \]

\text{rocketLondon} \rightarrow F_{\text{rocketParis}}

and so on ...
Liveness Property

- something good will happen

• Typical examples
  
  $\mathcal{F}$\text{rich}
  
  rocketLondon $\rightarrow$ $\mathcal{F}$rocketParis
  
  and so on ...

Usually: $\mathcal{F}$...
Fairness Property

useful when scheduling processes, responding to messages, etc.
good for specifying interaction properties of the environment

• Typical examples:
  \( G(\text{rocketLondon} \rightarrow F\text{rocketParis}) \)

• Strong Fairness:
  if something is attempted/requested, then it will be successful

• Typical examples:
  \( G(\text{attempt} \rightarrow F\text{success}) \)
  \( GF\text{attempt} \rightarrow GF\text{success} \)
Linear Temporal Logic - LTL

- Reasoning about a particular computation of a system where time is linear - just one possible future path is included.

**Definition 3.4 (Models of LTL)**

A model of LTL is a sequence of time moments. We call such models paths, and denote them by $\lambda$. Evaluation of atomic propositions at particular time moments is also needed.

**Notation:**

- $\lambda[i]$: $i$th time moment
- $\lambda[i \ldots j]$: all time moments between $i$ and $j$
- $\lambda[i \ldots \infty]$: all timepoints from $i$ on
### Definition 3.5 (Semantics of LTL)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\lambda \models p)</td>
<td>iff (p) is true at moment (\lambda[0]);</td>
</tr>
<tr>
<td>(\lambda \models X\varphi)</td>
<td>iff (\lambda[1..\infty] \models \varphi);</td>
</tr>
<tr>
<td>(\lambda \models F\varphi)</td>
<td>iff (\lambda[i..\infty] \models \varphi) for some (i \geq 0);</td>
</tr>
<tr>
<td>(\lambda \models G\varphi)</td>
<td>iff (\lambda[i..\infty] \models \varphi) for all (i \geq 0);</td>
</tr>
<tr>
<td>(\lambda \models \varphi U\psi)</td>
<td>iff (\lambda[i..\infty] \models \psi) for some (i \geq 0), and (\lambda[j..\infty] \models \varphi) for all (0 \leq j \leq i).</td>
</tr>
</tbody>
</table>
Definition 3.5 (Semantics of LTL)

\[ \lambda \models p \quad \text{iff } p \text{ is true at moment } \lambda[0]; \]
\[ \lambda \models X\varphi \quad \text{iff } \lambda[1..\infty] \models \varphi; \]
\[ \lambda \models F\varphi \quad \text{iff } \lambda[i..\infty] \models \varphi \text{ for some } i \geq 0; \]
\[ \lambda \models G\varphi \quad \text{iff } \lambda[i..\infty] \models \varphi \text{ for all } i \geq 0; \]
\[ \lambda \models \varphi U\psi \quad \text{iff } \lambda[i..\infty] \models \psi \text{ for some } i \geq 0, \text{ and } \lambda[j..\infty] \models \varphi \text{ for all } 0 \leq j \leq i. \]

Note that:
\[ G\varphi \equiv \neg F\neg \varphi \]
\[ F\varphi \equiv \neg G\neg \varphi \]
\[ F\varphi \equiv T U\varphi \]
Computational Tree Logic - CTL

• Reasoning about possible computations of a system. Time is branching – we want all paths included.

Path quantifiers: \( A \) (for all paths), \( E \) (there is a path);

Temporal operators: \( \mathcal{X} \) (nexttime), \( \mathcal{F} \) (sometime), \( \mathcal{G} \) (always) and \( \mathcal{U} \) (until);
Computational Tree Logic - CTL

- Reasoning about possible computations of a system. Time is branching – we want all alternative paths included.

**Path quantifiers:** $A$ (for all paths), $E$ (there is a path);

**Temporal operators:** $\mathcal{X}$ (nexttime), $\mathcal{F}$ (sometime), $\mathcal{G}$ (always) and $\mathcal{U}$ (until);

- **Vanilla CTL:** every temporal operator must be immediately preceded by exactly one path quantifier
- **CTL***: no syntactic restrictions
- Reasoning in Vanilla CTL can be automated.
Definition 3.8 (Semantics of CTL*: state formulae)

\[ M, q \models E\varphi \iff \text{there is a path } \lambda, \text{ starting from } q, \text{ such that } M, \lambda \models \varphi; \]

\[ M, q \models A\varphi \iff \text{for all paths } \lambda, \text{ starting from } q, \text{ we have } M, \lambda \models \varphi. \]
Definition 3.8 (Semantics of CTL*: state formulae)

$M, q \models E\varphi$ iff there is a path $\lambda$, starting from $q$, such that $M, \lambda \models \varphi$;

$M, q \models A\varphi$ iff for all paths $\lambda$, starting from $q$, we have $M, \lambda \models \varphi$.

Definition 3.9 (Semantics of CTL*: path formulae)

Exactly like for LTL!
## Definition 3.8 (Semantics of CTL*: state formulae)

\[ M, q \models E \varphi \iff \text{there is a path } \lambda, \text{ starting from } q, \text{ such that } M, \lambda \models \varphi; \]

\[ M, q \models A \varphi \iff \text{for all paths } \lambda, \text{ starting from } q, \text{ we have } M, \lambda \models \varphi. \]

## Definition 3.9 (Semantics of CTL*: path formulae)

\[ M, \lambda \models X \varphi \iff M, \lambda[1...\infty] \models \varphi; \]

\[ M, \lambda \models \varphi U \psi \iff M, \lambda[i...\infty] \models \psi \text{ for some } i \geq 0, \text{ and } M, \lambda[j...\infty] \models \varphi \text{ for all } 0 \leq j \leq i. \]
Example
Example
Example
Example
Dynamic Logic
1\textsuperscript{st} idea: Consider actions or programs $\alpha$. Each such $\alpha$ defines a transition (accessibility relation) from worlds into worlds.
Dynamic Logic

1\textsuperscript{st} idea: Consider actions or programs $\alpha$. Each such $\alpha$ defines a transition (accessibility relation) from worlds into worlds.

2\textsuperscript{nd} idea: We need statements about the outcome of actions:

- $[\alpha] \varphi$: “after every execution of $\alpha$, $\varphi$ holds,
- $\langle \alpha \rangle \varphi$: “after some executions of $\alpha$, $\varphi$ holds.”
Dynamic Logic

1\textsuperscript{st} idea: Consider actions or programs \( \alpha \). Each such \( \alpha \) defines a transition (accessibility relation) from worlds into worlds.

2\textsuperscript{nd} idea: We need statements about the outcome of actions:

\begin{itemize}
  \item \([\alpha] \varphi\): "after every execution of \( \alpha \), \( \varphi \) holds,
  \item \( \langle \alpha \rangle \varphi\): "after some executions of \( \alpha \), \( \varphi \) holds.
\end{itemize}

As usual, \( \langle \alpha \rangle \varphi \equiv \neg [\alpha] \neg \varphi \).
Dynamic Logic

3rd idea: Programs/actions can be combined (sequentially, nondeterministically, iteratively), e.g.:

$[\alpha; \beta] \varphi$

would mean “after every execution of $\alpha$ and then $\beta$, formula $\varphi$ holds”.

80
Definition 3.1 (Labelled Transition System)

A labelled transition system is a pair

$$\langle St, \{ \xrightarrow{\alpha} : \alpha \in L \} \rangle$$

where $St$ is a non-empty set of states and $L$ is a non-empty set of labels and for each $\alpha \in L$:

$$\xrightarrow{\alpha} \subseteq St \times St.$$
Definition 3.1 (Labelled Transition System)

A labelled transition system is a pair

$$\langle St, \{\xrightarrow{\alpha}: \alpha \in L\}\rangle$$

where $St$ is a non-empty set of states and $L$ is a non-empty set of labels and for each $\alpha \in L$:

$$\xrightarrow{\alpha} \subseteq St \times St.$$
Definition 3.2 (Dynamic Logic: Models)

A model of propositional dynamic logic is given by a labelled transition systems and an evaluation of propositions.
Definition 3.2 (Dynamic Logic: Models)
A model of propositional dynamic logic is given by a labelled transition systems and an evaluation of propositions.

Definition 3.3 (Semantics of DL)
$\mathcal{M}, s \models [\alpha]\varphi$ iff for every $t$ such that $s \xrightarrow{\alpha} t$, we have $\mathcal{M}, t \models \varphi$. 
Dynamic Logic
Dynamic Logic

\[
\begin{align*}
\text{start} & \rightarrow \langle \text{try} \rangle \text{halt}
\end{align*}
\]
Dynamic Logic

\[
\begin{align*}
\text{start} & \rightarrow \langle \text{try} \rangle \text{halt} \\
\text{start} & \rightarrow \neg [\text{try}] \text{halt}
\end{align*}
\]
Dynamic Logic

\[
\begin{align*}
\text{start} & \rightarrow \langle \text{try} \rangle \text{halt} \\
\text{start} & \rightarrow \neg [\text{try}] \text{halt} \\
\text{start} & \rightarrow \langle \text{try} \rangle [\text{wait}] \text{halt}
\end{align*}
\]
Concluding Remarks

• Practical Importance of Temporal and Dynamic Logics:
  – Automatic verification in principle possible (model checking).
  – Can be used for automated planning.
  – Executable specifications can be used for programming.

• Note:
  When we combine time and actions with knowledge (beliefs, desires, intentions, obligations...), we finally obtain a fairly realistic model of MAS.
Models of Practical Reasoning: BDI

Process of figuring out what to do. Practical reasoning is a matter of weighing conflicting considerations for and against competing options, where the relevant considerations are provided by what the agent desires/values/cares about and what the agent believes (Bratman).

- computational model of human decision process oriented towards an action, based on models of existing mental models of the agents

- human practical reasoning consists of two activities:
  - deliberation: deciding what state of affairs we want to achieve and
  - means-ends reasoning (planning): deciding how to achieve these states

- the outputs of deliberation process are intentions
BDI Architecture

- **BELIEFS**
  - collection of information that the agent has about its the status of the environment, peer agents, self

- **DESIRE**
  - set of long term goals the agent wants to achieve

- **INTENTIONS**
  - agents immediate commitment to executing an action, either high-level or low level (depends on agents planning horizon)

- BDI architecture connects: (i) reactive (ii) planning & (iii) logical representation. BDI architecture does not count on theorem proving

\[
\text{if } \varphi \in \mathcal{L}_{agent} \quad \text{then} \quad \varphi, (\text{Bel } A \varphi), (\text{Des } A \varphi), (\text{Int } A \varphi) \in \mathcal{L}_{bdi}
\]
BDI Inference Algorithm

- Basic algorithm:

1. initial beliefs $\rightarrow \text{Bel}$
2. while true do
3. Read(get_next_percept) $\rightarrow \text{in}$
4. Belief-revision($\text{Bel, in}$) $\rightarrow \text{Bel}$
5. Deliberate($\text{Bel, Des}$) $\rightarrow \text{Int}$
6. Plan($\text{Bel, Int}$) $\rightarrow \pi$
7. Execute($\pi$)
8. end while
BDI Modal Properties

• **BELIEFS**
  – KD45 system, modal logic where the B relation is serial, transitive and euclidean: satisfies K axioms, positive introspection axiom (4 axiom), negative introspection axiom (5 axiom), beliefs consistency axiom (D axiom).

• **DESIREs**
  – KD system, modal logic requiring desired goals not to contradict (D axiom).

\[(\text{Des} A \varphi) \rightarrow \neg(\text{Des} A \neg \varphi)\]

• **INTENTIONS**
  – KD system, modal logic requiring intentions not to contradict (D axiom).

\[(\text{Int} A \varphi) \rightarrow \neg(\text{Int} A \neg \varphi)\]
BDI Modal Properties

• **BELIEFS**
  – KD45 system, modal logic where the B relation is serial, transitive and euclidean: satisfies K axioms, positive introspection axiom (4 axiom), negative introspection axiom (5 axiom), beliefs consistency axiom (D axiom).

• **DESIREs**
  – KD system, modal logic requiring desired goals not to contradict (D axiom).

• **INTENTIONS**
  – KD system, modal logic requiring intentions not to contradict (D axiom).

\[
\begin{align*}
\text{T: } & \Box \varphi \rightarrow \varphi \\
\text{D: } & \Box \varphi \rightarrow \Diamond \varphi \\
\text{4: } & \Box \varphi \rightarrow \Box \Box \varphi \\
\text{B: } & \varphi \rightarrow \Box \Diamond \varphi \\
\text{5: } & \Diamond \varphi \rightarrow \Box \Diamond \varphi
\end{align*}
\]

\[(\text{Des } A \varphi) \rightarrow \neg(\text{Int } A \neg \varphi)\]

due to reflexivity

due to seriality

due to transitivity

due to symmetry

due to euclidean property
Properties of Intentions

• **Intention persistency:**
  – agents track the success of their intentions, and are inclined to try again if their attempts fail

\[(\text{Int } A \varphi) \land \varphi\]

• **Intention satisfiability:**
  – agents believe their intentions are possible; that is, they believe there is at least some way that the intentions could be brought about.

\[(\text{Int } A \varphi) \implies \text{EF}\varphi\]
Properties of Intentions

• **Intention-belief inconsistency:**
  - agents do not believe they will not bring about their intentions; it would be irrational of agents to adopt an intention if believed was not possible

\[(\text{Int } A \varphi) \land (\text{Bel } A \lnot \text{EF} \varphi)\]

• **Intention-belief incompleteness:**
  - agent do not believe that their intention is possible to be achieved, may be understood as rational behavior

\[(\text{Int } A \varphi) \land (\lnot \text{Bel } A \text{EF} \varphi)\]

  – agents admit that their intentions may not be implemented.

\[(\text{Int } A \varphi) \land (\text{Bel } A \text{EF} \lnot \varphi)\]
Properties of Intentions

- **Intention side-effects:**
  - Agents need not intend all the expected side effects of their intentions. Intentions are not closed under implication.

\[(\text{Bel } A \psi \Rightarrow \varphi) \land (\text{Int } A \psi) \land \neg (\text{Int } A \varphi)\]

* is thus classified as fully rational behaviour

- **Example:** I may believe that going to the dentist involves pain, and I may also intend to go to the dentist - but this does not imply that I intend to suffer pain!
## Rationality of Inevitabilities & Options

1. **inevitables:**

\[
\begin{align*}
(\text{Int } A \text{ AG} & \varphi) \Rightarrow (\text{Des } A \text{ AG} \varphi) & \quad (\text{Des } A \text{ AG} \varphi) \Rightarrow (\text{Bel } A \text{ AG} \varphi) \\
(\text{Des } A \text{ AG} \varphi) \Rightarrow (\text{Int } A \text{ AG} \varphi) & \quad (\text{Int } A \text{ AG} \varphi) \Rightarrow (\text{Bel } A \text{ AG} \varphi) \\
(\text{Bel } A \text{ AG} \varphi) \Rightarrow (\text{Des } A \text{ AG} \varphi) & \quad (\text{Bel } A \text{ AG} \varphi) \Rightarrow (\text{Int } A \text{ AG} \varphi)
\end{align*}
\]

2. **options:**

\[
\begin{align*}
(\text{Int } A \text{ EF} & \varphi) \Rightarrow (\text{Des } A \text{ EF} \varphi) & \quad (\text{Des } A \text{ EF} \varphi) \Rightarrow (\text{Bel } A \text{ EF} \varphi) \\
(\text{Des } A \text{ EF} \varphi) \Rightarrow (\text{Int } A \text{ EF} \varphi) & \quad (\text{Int } A \text{ EF} \varphi) \Rightarrow (\text{Bel } A \text{ EF} \varphi) \\
(\text{Bel } A \text{ EF} \varphi) \Rightarrow (\text{Des } A \text{ EF} \varphi) & \quad (\text{Bel } A \text{ EF} \varphi) \Rightarrow (\text{Int } A \text{ EF} \varphi)
\end{align*}
\]
Rationality of Invetibilities & Options

1. inevitables:

\( (\text{Int} \ A \ AG\varphi) \Rightarrow (\text{Des} \ A \ AG\varphi) \)
\( (\text{Des} \ A \ AG\varphi) \Rightarrow (\text{Int} \ A \ AG\varphi) \)
\( (\text{Bel} \ A \ AG\varphi) \Rightarrow (\text{Des} \ A \ AG\varphi) \)
\( (\text{Des} \ A \ AG\varphi) \Rightarrow (\text{Bel} \ A \ AG\varphi) \)
\( (\text{Int} \ A \ AG\varphi) \Rightarrow (\text{Bel} \ A \ AG\varphi) \)
\( (\text{Bel} \ A \ AG\varphi) \Rightarrow (\text{Int} \ A \ AG\varphi) \)

2. options:

\( (\text{Int} \ A \ EF\varphi) \Rightarrow (\text{Des} \ A \ EF\varphi) \)
\( (\text{Des} \ A \ EF\varphi) \Rightarrow (\text{Int} \ A \ EF\varphi) \)
\( (\text{Bel} \ A \ EF\varphi) \Rightarrow (\text{Des} \ A \ EF\varphi) \)
\( (\text{Des} \ A \ EF\varphi) \Rightarrow (\text{Bel} \ A \ EF\varphi) \)
\( (\text{Int} \ A \ EF\varphi) \Rightarrow (\text{Bel} \ A \ EF\varphi) \)
\( (\text{Bel} \ A \ EF\varphi) \Rightarrow (\text{Int} \ A \ EF\varphi) \)
Rationality of Invetibilities & Options

1. **inevitables:**

\[(\text{Int } A \ AG\varphi) \Rightarrow (\text{Des } A \ AG\varphi)\]
\[(\text{Des } A \ AG\varphi) \Rightarrow (\text{Int } A \ AG\varphi)\]
\[(\text{Bel } A \ AG\varphi) \Rightarrow (\text{Des } A \ AG\varphi)\]
\[(\text{Des } A \ AG\varphi) \Rightarrow (\text{Bel } A \ AG\varphi)\]
\[(\text{Int } A \ AG\varphi) \Rightarrow (\text{Bel } A \ AG\varphi)\]
\[(\text{Bel } A \ AG\varphi) \Rightarrow (\text{Int } A \ AG\varphi)\]

2. **options:**

\[(\text{Int } A \ EF\varphi) \Rightarrow (\text{Des } A \ EF\varphi)\]
\[(\text{Des } A \ EF\varphi) \Rightarrow (\text{Int } A \ EF\varphi)\]
\[(\text{Bel } A \ EF\varphi) \Rightarrow (\text{Des } A \ EF\varphi)\]
\[(\text{Des } A \ EF\varphi) \Rightarrow (\text{Bel } A \ EF\varphi)\]
\[(\text{Int } A \ EF\varphi) \Rightarrow (\text{Bel } A \ EF\varphi)\]
\[(\text{Bel } A \ EF\varphi) \Rightarrow (\text{Int } A \ EF\varphi)\]
Agents Individual/Social Commitments

• Commitments: knowledge structure, declarative programming concept based on intentions (intentions are special kinds of comms).
  – specify relationships among different intentional states of the agents
  – specify social relations among agents, based on their comms to joint actions

*The commitment is an agent's state of 'the mind' where it commits to adopting the single specific intention or a longer term desire.*

• We distinguish between:
  – specific, commonly used comms                 general comms
  – individual comms                              social comms
Individual Commitments

• A can get committed to its intention \( \varphi \) in several different ways:
  – **blind commitment**: also referred to as fanatical commitment, the agent is intending the intention until it believes that it has been achieved (persistent intention)

\[
(\text{Commit } A \varphi) \equiv \text{AG}( (\text{Int } A \varphi) \land (\text{Bel } A \varphi) )
\]
Individual Commitments

- $A$ can get committed to its intention $\varphi$ in several different ways:
  - blind commitment: also referred to as fanatical commitment, the agent is intending the intention until it believes that it has been achieved (persistent intention)
    \[
    (\text{Commit } A \varphi) \equiv AG((\text{Int } A \varphi) \land (\text{Bel } A \varphi))
    \]
  - single-minded commitment: besides above it intends the intention until it believes that it is no longer possible to achieve the goal
    \[
    (\text{Commit } A \varphi) \equiv AG((\text{Int } A \varphi) \land ((\text{Bel } A \varphi) \lor (\text{Bel } A \neg EF \varphi)))
    \]
Individual Commitments

- A can get committed to its intention \( \varphi \) in several different ways:
  - **blind commitment**: also referred to as fanatical commitment, the agent is intending the intention until it believes that it has been achieved (persistent intention)
    \[
    (\text{Commit } A \varphi) \equiv AG((\text{Int } A \varphi) \land (\text{Bel } A \varphi))
    \]
  - **single-minded commitment**: besides above it intends the intention until it believes that it is no longer possible to achieve the goal
    \[
    (\text{Commit } A \varphi) \equiv AG((\text{Int } A \varphi) \land ((\text{Bel } A \varphi) \lor (\text{Bel } A \neg \text{EF}\varphi)))
    \]
  - **open-minded commitment**: besides above it intends the intention as long as it is sure that the intention is achievable
    \[
    (\text{Commit } A \varphi) \equiv AG((\text{Int } A \varphi) \land ((\text{Bel } A \varphi) \lor \neg(\text{Bel } A \text{EF}\varphi)))
    \]
General Commitments

• Commitment is defined as \((\text{Commit } A \varphi \psi \lambda)\), where

• Convention is defined as \(\lambda = \{\langle \rho_k, \gamma_k \rangle \}_{k \in \{1, \ldots, l\}}\)

  – provided \(\bigcirc\) stands for until, \(A\) stands for always in the future, \(\text{Int}\) is agent’s intention and \(\text{Bel}\) is agent’s belief then for \(\lambda = \langle \rho, \gamma \rangle\) the commitment has the form:

\[
(\text{Commit } A \varphi \psi \lambda) \equiv \psi \land A((\text{Int } A \varphi) \land \text{decommitment_rule } \bigcirc \gamma)
\]

\[
(\text{Commit } A \varphi \psi \lambda) \equiv \psi \land A((\text{Int } A \varphi) \land ((\text{Bel } A \rho) \implies A(\text{Int } A \gamma)) \bigcirc \gamma) \bigcirc \gamma)
\]
General Commitments

- Commitment is defined as \((\text{Commit } A \varphi \psi \lambda)\), where
- Convention is defined as \(\lambda = \{\langle \rho_k, \gamma_k \rangle \}_{k \in \{1, \ldots, l\}}\)
  - provided \(\bigcirc\) stands for until, \(A\) stands for always in the future, \(\text{Int}\) is agent’s intention and \(\text{Bel}\) is agent’s belief then for \(\lambda = \langle \rho, \gamma \rangle\) the commitment has the form:

\[
\begin{align*}
(\text{Commit } A \varphi \psi \lambda) & \equiv \\
\psi \land A((\text{Int } A \varphi) \land \\
((\text{Bel } A \rho_1) \Rightarrow A(\text{Int } A \gamma_1)) \bigcirc \gamma_1) \\
\vdots \\
((\text{Bel } A \rho_l) \Rightarrow A(\text{Int } A \gamma_l)) \bigcirc \gamma_l) \\
\bigcirc \bigvee_{i} \gamma_i)
\end{align*}
\]
Joint (Social) Commitment

• Form of a commitment that represents how a group of agents is committed to a joint action (goal, intention, ...)

107
Joint (Social) Commitment

- Form of a commitment that represents how a group of agents is committed to a joint action (goal, intention, ...)

\[(\text{Commit} \ A \varphi \psi \lambda) \equiv \psi \land A((\text{Int} \ A \varphi) \land ((\text{Bel} \ A \rho) \Rightarrow A(\text{Int} \ A \gamma)) \land \gamma) \land \gamma)\]
Joint (Social) Commitment

- Form of a commitment that represents how a group of agents is committed to a joint action (goal, intention, ...)

\[(\text{Commit } A \varphi \psi \lambda) \equiv \psi \land A((\text{Int } A \varphi) \land ((\text{Bel } A \rho) \Rightarrow A(\text{Int } A \gamma)) \land \gamma) \land \gamma)\]

\[(\text{J-Commit } \Theta \varphi \psi \lambda) \equiv \]

\[
\forall A : (A \in \theta) \Rightarrow \\
\psi \land A((\text{Int } A \varphi) \land \\
((\text{Bel } A \rho) \Rightarrow A(\text{Int } A \gamma) \land \gamma) \\
\land \gamma)\]
Joint (Social) Commitment

• Form of a commitment that represents how a group of agents is committed to a joint action (goal, intention, ...)
  – for a convention in the form of $\lambda = \{ (\rho_k, \gamma_k) \}_{k \in \{1, \ldots, l\}}$

$$(J\text{-}\text{Commit } \Theta \varphi \psi \lambda) \equiv \forall A : (A \in \theta) \Rightarrow \psi \land A((\chi_1 \land \chi_2) \land \chi_3)$$

where

$\chi_1 = (\text{Int } A \varphi)$

$\chi_2 = ((\text{Bel } A \rho_1) \Rightarrow A((\text{Int } A \gamma_1) \land \gamma_1)) \land ((\text{Bel } A \rho_2) \Rightarrow A((\text{Int } A \gamma_2) \land \gamma_1) \land \cdots \land ((\text{Bel } A \rho_n) \Rightarrow A((\text{Int } A \gamma_n) \land \gamma_n)))$

$\chi_3 = \gamma_1 \lor \gamma_1 \lor \cdots \lor \gamma_n$
Blind Social Commitment

- each agent is trying to accomplish the commitment until achieved

$$\lambda_{blind} = \{(Bel \ A \ \varphi), (M-Bel \ \Theta \ \varphi)\}$$

$$\psi_{blind} = \neg(Bel \ A \ \varphi)$$

$$(J\text{-Commit } \Theta \ \varphi \ \psi \ \lambda) \equiv \forall A : (A \in \Theta) \Rightarrow$$

$$\neg(Bel \ A \ \varphi) \land (A((Int \ A \ \varphi) \land$$

$$((Bel \ A \ \varphi) \Rightarrow A((Int \ A (M-Bel \ \Theta \ \varphi))$$

$$\land (M-Bel \ \Theta \ \varphi))))$$

$$\land (M-Bel \ \Theta \ \varphi))).$$
Minimal Social Commitment

- minimal social commitment, also related to as joint persistent goal:
  - initially agents do not believe that goal is true but it is possible
  - every agent has the goal until termination condition is true
  - until termination: if agent believes that the goal is either true or impossible than it will want the goal that it becomes a mutually believed, but keep committed
  - the termination condition is that it is mutually believed either goal is true or impossible to be true.

\[ \psi_{soc} = \neg (Bel A \varphi) \land (Bel A EF \varphi) \]

\[ \lambda_{soc} = \left\{ \begin{array}{l}
(\langle Bel A \varphi \rangle, (M-Bel \Theta \varphi)), \\
(\langle Bel A AG \neg \varphi \rangle, (M-Bel \Theta AG \neg \varphi))
\end{array} \right\} \]
Minimal Social Commitment

\[(\text{J-Commit } \Theta \varphi \psi_{soc} \lambda_{soc}) \equiv \]

\[\forall A, A \in \Theta : [\neg(\text{Bel } A \varphi) \land (\text{Bel } A \text{ EF} \varphi)] \land\]

\[A \left\lbrack \left( (\text{Int } A \varphi) \land \right. \right.

\[\left. ((\text{Bel } A \varphi) \Rightarrow A((\text{Int } A(M-\text{Bel } \Theta \varphi))) \bowtie \chi \land \right. \]

\[\left. ((\text{Bel } A \text{ AG}\neg \varphi) \Rightarrow A((\text{Int } A(M-\text{Bel } \Theta \text{ AG}\neg \varphi))) \bowtie \chi \right) \right\rbrack \]

where \(\chi \equiv ((M-\text{Bel } \Theta \varphi) \lor (M-\text{Bel } \Theta \text{ AG}\neg \varphi)))\)
Mutual Belief ?

**Definition 1:**

\[(\text{M-Bel } \Theta \, \varphi) \equiv \forall A, \ A \in \Theta (\text{Bel } A (\text{M-Bel } \Theta \, \varphi))\]

**Definition 2:**

\[\begin{align*}
(\text{E-Bel}^0 \Theta \, \varphi) & \equiv \forall A, \ A \in \Theta (\text{Bel } A \, \varphi) \\
(\text{E-Bel}^k \Theta \, \varphi) & \equiv \forall A, \ A \in \Theta (\text{E-Bel}^{k-1} \Theta \, \varphi) \\
(\text{M-Bel } \Theta \, \varphi) & \equiv \forall m \in \mathbb{N} (\text{M-Bel}^m \Theta \, \varphi)
\end{align*}\]