Logical Aspects of Artificial Intelligence
Temporal Logics for Multi-agent Systems

Stéphane Demri  demri@lsv.fr

October 20th, 2021
Plan of the lecture

- Concurrent game structures.
- Introduction to ATL.
- Exercises session.
Breaking news

- Exam on Wednesday November 10th, 2pm-5pm.
- Room 1Z53.
- Lecture notes and exercises sheets with correction authorised.
Temporal logics for multi-agent systems
Introduction to multi-agent systems

- Multi-agent systems are transition systems in which transitions are fired when simultaneous actions are performed by different agents.

- Coalitions are made of agents that can coordinate their respective actions.

- Temporal logics for multi-agent systems contain:
  - temporal formulae to describe objectives and,
  - strategy modalities parameterised by coalitions.

- In this lecture, we present the basic ingredients in the logic ATL and variants.
Other (online) resources

- Valentin Goranko’s slides (ESSLLI’18)

- See also the proceedings of the international conferences:
  - International Conference on Autonomous Agents and Multi-Agent Systems. (AAMAS)
  - European Conference on Artificial Intelligence. (ECAI)
  - International Conference on Principles of Knowledge Representation and Reasoning. (KR)

  

Concurrent game structures
The two-robot example

- Two robots Robot₁ and Robot₂, and a carriage.
- Robot₁ can only push the carriage in clockwise direction, Robot₂ can only push it in anti-clockwise direction.
Concurrent game structure: definition

\[ \mathcal{M} = (Agt, S, Act, \text{act}, \delta, L) \]

- **Agt** is a non-empty set of *k* agents.
- **S** is a finite non-empty set of states.
- **Act**: finite set of actions.
- **L**: \( S \rightarrow \mathcal{P}(\text{PROP}) \) is a labelling specifying a truth assignment for each state.
- **act**: \( \text{Agt} \times S \rightarrow \mathcal{P}(\text{Act}) \setminus \{\emptyset\} \) is the action manager, function assigning to each agent \( a \in \text{Agt} \) and each state \( s \in S \), a non-empty set of actions.
- **Transition function** \( \delta : S \times (\text{Agt} \rightarrow \text{Act}) \rightarrow S \). \( \delta(s, f) \) undefined if there is some agent \( a \) such that \( f(a) \notin \text{act}(a, s) \).

Differences with transition systems?
An example

Action manager $\text{act} : \text{Agt} \times \text{S} \rightarrow \mathcal{P}(\text{Act}) \setminus \{\emptyset\}$.

$\text{act}(1, s_3) = \{c\}; \text{act}(2, s_3) = \{c\}.$

Transition function $\delta : \text{S} \times (\text{Agt} \rightarrow \text{Act}) \rightarrow \text{S}$.

$\delta(s_4, [1 \mapsto c, 2 \mapsto c]) = s_3 \quad \text{— undef.} \quad \delta(s_4, [1 \mapsto c, 2 \mapsto a]).$

Labelling $L : \text{S} \rightarrow \mathcal{P}(\text{PROP})$.

$L(s_1) = \{p\}.$
Another concurrent game structure

- Two agents share a file in a cyberspace,
- Each agent can apply the action Update (U) if she is enabled to do so, or Skip (N).
- State $P$ is reached when both agents have processed the file.
- Action Reset (R) allows to move to the initial state $E$. 
Turn-based CGS

- Turn-based CGS: only one agent at a time is executing an action.

- Turn-based CGS $\mathcal{M}$: for all $s \in S$, there is at most one agent $a \in \text{Agt}$ such that $\text{card}(\text{act}(a, s)) > 1$. 
ATL and its variants
Basic concepts: joint action

- **Coalition** $A \subseteq \text{Agt}$ with **opponent coalition** $\bar{A} = \text{Agt} \setminus A$.

- $f: A \rightarrow \text{Act}$: **joint action** by $A \subseteq \text{Agt}$ in $s$.
  Proviso: for all $a \in A$, we have $f(a) \in \text{act}(a, s)$.
  $f$ can be viewed as a tuple of actions of length $\text{card}(A)$.

- $f': A' \rightarrow \text{Act}$ \makebox[.95\textwidth]{$\subseteq f': A' \rightarrow \text{Act}$ \text{def} $\iff A \subseteq A'$ and $f$ is the restriction of $f'$ to $A$.}

  \[ (a_1, a_2, -, -) \subseteq (a_1, a_2, a_3, a_4) \]

  ('-' indicates undefinedness)

- $D_A(s)$: set of joint actions by $A$ in $s$. 
Basic concepts: outcome set

- \( f : A \rightarrow \text{Act.} \)

- \( \text{out}(s, f) \) defined as the set of states reachable from \( s \) in one step when the actions performed by the agents in \( A \) are determined by \( f \).

- Set of outcomes:

\[
\text{out}(s, f) \overset{\text{def}}{=} \{ s' \in S \mid \exists g \in D_{Agt}(s) \text{ s.t. } f \sqsubseteq g \land s' = \delta(s, g) \}
\]
Basic concepts: strategies

- \[ \text{card}(\text{out}(s, f)) = 1 \text{ if } f \in D_{Agt}(s). \]

- **Computation** \( \lambda = s_0 \xrightarrow{f_0} s_1 \xrightarrow{f_1} s_2 \ldots \) such that for all \( i \), we have \( s_{i+1} \in \delta(s_i, f_i) \). (**history** = finite computation)

- Herein, computations can be also written \( s_0 s_1 s_2 \ldots \) (without joint actions).

- Linear model \( L(s_0) \Rightarrow L(s_1) \Rightarrow L(s_2) \ldots \) (sequence of propositional valuations)

- **Strategy** \( F_A \) for \( A \) is a map from the set of finite computations (histories) to the set of joint actions by \( A \) such that

\[
F_A(s_0 \xrightarrow{f_0} s_1 \cdots \xrightarrow{f_{n-1}} s_n) \in D_A(s_n)
\]
Positional strategies

- Memory-based strategies vs. positional strategies.

- $F_A$ is a **positional strategy** $\iff$ for all $s_0 \xrightarrow{f_0} s_1 \cdots \xrightarrow{f_{n-1}} s_n$ and $s'_0 \xrightarrow{f'_0} s'_1 \cdots \xrightarrow{f'_{m-1}} s'_m$ with $s_n = s'_m$, we have

$$F_A(s_0 \xrightarrow{f_0} s_1 \cdots \xrightarrow{f_{n-1}} s_n) = F_A(s'_0 \xrightarrow{f'_0} s'_1 \cdots \xrightarrow{f'_{m-1}} s'_m)$$

(only the value of the last state matters)

- **Memoryless strategy** $\overset{\text{def}}{=} \text{positional strategy.}$

$$F_A : s \in S \mapsto f \in D_A(s)$$
Computations respecting a strategy

\[ \lambda = s_0 \xrightarrow{f_0} s_1 \xrightarrow{f_1} s_2 \cdots \text{ respects } F_A \iff \forall i < |\lambda|, \]

\[ s_{i+1} \in \text{out}(s_i, F_A(s_0 \xrightarrow{f_0} s_1 \ldots \xrightarrow{f_{i-1}} s_i)) \cap D_A(s_i) \]

\[ \lambda \text{ respecting } F_A \text{ is maximal whenever } \lambda \text{ cannot be extended further while respecting the strategy.} \]

\[ \text{Comp}(s, F_A) : \text{ set of maximal computations from } s \text{ respecting the strategy } F_A. \]
Computation tree given a strategy

- Positional $F_{\{1\}}$: select $a$ on $s_1$, $b$ on $s_2$, otherwise $c$.

- $F_{\{1\}}$ generates a set of computations whose linear models can be defined by a Büchi automaton (BA).
Trimming a CGS

- CGS $\mathcal{M} = (\text{Agt, } S, \text{Act}, \text{act}, \delta, L)$.

- Coalition $A \subseteq \text{Agt}$.

- Memoryless strategy $F : s \in S \mapsto f \in D_A(s)$.

- Underlying transition system $(S, R, L)$ such that for all $s, s' \in S$, we have
  $$ (s, s') \in R \iff s' \in \text{out}(s, F(s)) $$

- $R$ represents the set of moves allowed by the opponent coalition $(\text{Agt} \setminus A)$ when $A$ has the positional strategy $F$. 
Strategies as infinite trees

- For non-positional strategies, computations organised as a tree not necessarily generated from a BA.

\[ \text{Agt} = \{1, 2\} \] ; Strategy for \{1\}
Examples of strategies

- Positional strategy for Robot 1: \( F(s_0) = \text{push}, \ F(s_1) = \text{push}, \ F(s_2) = \text{wait}. \)

- The set of maximal computations respecting \( F \) from \( s_0 \) (projected on \( S \) only):
  \[
  \{ s_0^\omega \} \cup s_0^+ ((s_1^+ s_2^+)\omega \cup (s_1^+ s_2^+) s_1^\omega \cup (s_1^+ s_2^+) s_2^\omega )
  \]

- Which temporal properties are satisfied by such computations respecting \( F \)?
Specifying properties on $\omega$-sequences

- **LTL**: linear-time temporal logic.

- **LTL formulae:**
  
  $$\varphi, \psi ::= p \mid \neg \varphi \mid \varphi \land \psi \mid \varphi \lor \psi \mid X \varphi \mid \varphi U \psi$$

- Atomic formulae are propositional variables.

- **LTL models $\lambda$** are $\omega$-sequences of propositional valuations of the form $\lambda : \mathbb{N} \to \mathcal{P}(\text{PROP})$.

  ($\approx$ linear models from infinite computations)

- **$X \varphi$** states that the next state satisfies $\varphi$:

  $$\begin{array}{c}
  X\varphi \\
  \varphi
  \end{array}$$

  ![Diagram of a sequence with $X \varphi$ and $\varphi$]
Semantics of the linear-time temporal operators

- $F\varphi$ states that some future (or possibly, the current) state satisfies $\varphi$ without specifying explicitly which one that is.

- $G\varphi$ states that $\varphi$ is always satisfied.

- $\varphi U \psi$ states that $\varphi$ is true until $\psi$ is true.
Satisfaction relation

\[ \begin{align*}
\lambda, i \models p & \overset{\text{def}}{\iff} p \in \lambda(i), \\
\lambda, i \models \neg \varphi & \overset{\text{def}}{\iff} \lambda, i \not\models \varphi, \\
\lambda, i \models \varphi_1 \land \varphi_2 & \overset{\text{def}}{\iff} \lambda, i \models \varphi_1 \text{ and } \lambda, i \models \varphi_2, \\
\lambda, i \models X\varphi & \overset{\text{def}}{\iff} \lambda, i + 1 \models \varphi, \\
\lambda, i \models \varphi_1 \mathcal{U} \varphi_2 & \overset{\text{def}}{\iff} \text{there is } j \geq i \text{ such that } \lambda, j \models \varphi_2 \text{ and } \lambda, k \models \varphi_1 \text{ for all } i \leq k < j.
\end{align*} \]

\[ \begin{align*}
F\varphi & \overset{\text{def}}{=} \top \mathcal{U} \varphi, \\
G\varphi & \overset{\text{def}}{=} \neg F\neg \varphi, \\
\varphi \Rightarrow \psi & \overset{\text{def}}{=} \neg \varphi \lor \psi . . .
\end{align*} \]
About LTL

- Models(\(\varphi\)): set of models \(\lambda\) such that \(\lambda, 0 \models \varphi\).

- Models can be viewed as \(\omega\)-words over the alphabet \(\mathcal{P}(\text{PROP})\).

- Models(\(\varphi\)) can be effectively represented by a Büchi automaton \(A_\varphi\). (automata-based approach)

- LTL satisfiability problem is PSPACE-complete.
The logic ATL
(Alternating-time Temporal Logic)

- $\langle A \rangle \Phi$: the agents are divided into proponents in $A$ and opponents in $Agt \setminus A$.

- $\Phi$: property on computations ("objective").

- $M, s \models \langle A \rangle \Phi$ equivalent to solving a game with winning condition $\Phi$. ($A$ versus $Agt \setminus A$)

\[
\phi ::= p \mid \neg \phi \mid \phi \land \phi \mid \langle A \rangle X\phi \mid \langle A \rangle G\phi \mid \langle A \rangle \phi U\phi \\
p \in \text{PROP} \quad A \subseteq Agt
\]
ATL modalities, informally

- $\langle A \rangle X \varphi$: “The coalition $A$ has a collective action ensuring that any outcome (state) satisfies $\varphi$.”

- $\langle A \rangle G \varphi$: “The coalition $A$ has a collective strategy to maintain forever outcomes satisfying $\varphi$ on every computation respecting that strategy”.

- $\langle A \rangle \psi U \varphi$: “The coalition $A$ has a collective strategy to eventually reach an outcome satisfying $\varphi$, while maintaining in the meantime the truth of $\psi$, on every computation respecting that strategy”.
Satisfaction relation, formally

\[ M, s \models p \quad \iff \quad p \in L(s) \]

\[ M, s \models \langle A \rangle X \varphi \quad \iff \quad \text{there is a strategy } F_A \text{ s.t. for all } s_0 \xrightarrow{f_0} s_1 \ldots \in \text{Comp}(s, F_A), \text{ we have } M, s_1 \models \varphi \]

\[ M, s \models \langle A \rangle \varphi_1 U \varphi_2 \quad \iff \quad \text{there is a strategy } F_A \text{ s.t. for all } \lambda = s_0 \xrightarrow{f_0} s_1 \ldots \in \text{Comp}(s, F_A), \text{ there is some } i \text{ s.t. } M, s_i \models \varphi_2 \text{ and for all } j \in [0, i - 1], \text{ we have } M, s_j \models \varphi_1. \]

\[ M, s \models \langle A \rangle G \varphi \quad \iff \quad \text{there is a strategy } F_A \text{ s.t. for all } \lambda = s_0 \xrightarrow{f_0} s_1 \ldots \in \text{Comp}(s, F_A), \text{ for all } i, \text{ we have } M, s_i \models \varphi. \]
The semantics for \( \langle A \rangle G \) involves an existential quantification followed by two universal quantifications.

\[ \langle A \rangle F \varphi \overset{\text{def}}{=} \langle A \rangle (\top U \varphi). \]

The coalition \( A \) has a joint strategy to eventually reach an outcome satisfying \( \varphi \).

\[ [\varphi]^M \overset{\text{def}}{=} \{ s \in S \mid M, s \models \varphi \}. \]
Playing with formulae

(pos\textsubscript{i} holds only on s\textsubscript{i})

- \( M, s_0 \not\models \langle \langle 1 \rangle \rangle X pos_1 \) and \( M, s_0 \not\models \langle \langle 2 \rangle \rangle X pos_1 \).
- \( M, s_0 \models \langle \langle 1, 2 \rangle \rangle X pos_0 \wedge \langle \langle 1, 2 \rangle \rangle X pos_1 \wedge \langle \langle 1, 2 \rangle \rangle X pos_2 \).
- \( M, s_0 \not\models \langle \langle 1 \rangle \rangle F pos_1 \) and \( M, s_1 \models \langle \langle 1 \rangle \rangle F (pos_1 \lor pos_2) \).
- \( M, s_0 \models \langle \langle 1 \rangle \rangle G \neg pos_1 \) and \( M \models \langle \langle 1, 2 \rangle \rangle X \langle \langle 1 \rangle \rangle (pos_0 U pos_2) \).
Decision problems

- **Model-checking problem for ATL:**
  
  **Input:** $\varphi$ in ATL, a finite CGS $\mathcal{M}$ and a state $s$,
  
  **Question:** $\mathcal{M}, s \models \varphi$?

- **Satisfiability problem for ATL:**
  
  **Input:** $\varphi$ in ATL,
  
  **Question:** Is there a CGS $\mathcal{M}$ and $s$ in $\mathcal{M}$ such that $\mathcal{M}, s \models \varphi$?

- **Validity problem for ATL:**
  
  **Input:** $\varphi$ in ATL,
  
  **Question:** Is it true that for all CGS $\mathcal{M}$ and $s$ in $\mathcal{M}$, we have $\mathcal{M}, s \models \varphi$?
Model-checking problem for ATL is $\text{PTIME}$-complete. Labeling algorithm. (Positional strategies are sufficient)

Satisfiability and validity problems are $\text{EXPTIME}$-complete.
Positional strategies are sufficient for ATL!

- $\models_{pos}$: variant of $\models$ in which only positional strategies are legitimate.

- Positional strategies are sufficient for ATL:
  $$M, s \models \varphi \iff M, s \models_{pos} \varphi$$

- Positional strategies amount to remove transitions in the CGS (and keep only the ones related to the positional strategy of $A$).

- This property does not hold for the extension ATL*. (see next lecture)
“Proof”: positional strategies are sufficient for ATL

Memoryless $F^*$

s.t.

$F^*(s) = f$

and

for all $s' \neq s$

$F^*(s')$ is arbitrary

Strategy $F$ witnessing $s \Vdash \langle A \rangle X p$

Construction of memoryless $F^*$ also witnessing $s \Vdash \langle \text{can}\rangle X p$
Formula $\langle A \rangle (q U p)$
Formulae \( \langle A \rangle Gp \)
Relationships between ATL and CTL

- Computation Tree Logic CTL: branching-time temporal logic well-known to perform model-checking.

- A CGS without transitions labelled by action tuples defines a model for CTL (or with 1 agent and 1 action).

- Existential path quantifier $E$ in CTL corresponds to $\langle \text{Agt} \rangle$.
- Universal path quantifier $A$ in CTL corresponds to $\langle \emptyset \rangle$. 
CTL formulae

\[ \varphi ::= p \mid \bot \mid \neg \varphi \mid \varphi \land \varphi \mid \text{EX} \varphi \mid \text{E}(\varphi \cup \varphi) \mid \text{A}(\varphi \cup \varphi). \]

CTL models of the form \( \mathcal{T} = (S, R, L) \).
Informal semantics for $A(\varphi U \psi)$

\[
\begin{align*}
AF\varphi & \overset{\text{def}}{=} A\top U \varphi \\
EG\varphi & \overset{\text{def}}{=} \neg AF\neg \varphi
\end{align*}
\]

Tree diagrams illustrating the semantics of $AF\varphi$ and $A(\varphi U \psi)$.
CTL semantics

- Path $\pi$ in $T$: sequence of states in the graph $(S, R)$.
- A path is maximal if it is either infinite, or is finite and ends in a state with no successors.
- We assume that in CTL models no deadlock states.

$T, s \models EX \varphi$ iff there is $s'$ such that $(s, s') \in R$ and $T, s' \models \varphi$

$T, s \models E(\varphi_1 U \varphi_2)$ iff there is a path $\pi$ starting at $s$ and an $i \geq 0$ such that $\pi(0) = s$, $T, \pi(i) \models \varphi_2$ and for every $j \in [0, i - 1]$, we have $T, \pi(j) \models \varphi_1$

$T, s \models A(\varphi_1 U \varphi_2)$ iff for all paths $\pi$ such that $\pi(0) = s$, there is $i \geq 0$ such that $T, \pi(i) \models \varphi_2$ and for every $j \in [0, i - 1]$, we have $T, \pi(j) \models \varphi_1$
Relating CTL and ATL

- CTL model-checking problem is $\text{PTIME}$-complete.

- CTL satisfiability problem is $\text{EXPTIME}$-complete.

- Reduction from CTL satisfiability (resp. model-checking) to ATL satisfiability (resp. model-checking).

(E corresponds to $\langle\langle \text{Agt} \rangle\rangle$ and $A$ corresponds to $\langle\langle \emptyset \rangle\rangle$.)
Introducing a predecessor operator \texttt{pre}

- CGS $\mathcal{M} = (\text{Agt}, S, \text{Act}, \text{act}, \delta, L)$, $A \subseteq \text{Agt}$, and $Z \subseteq S$.

- $\text{pre}(\mathcal{M}, A, Z)$: set of states from which $A$ has a collective move that guarantees that the outcome to be in $Z$.

- Definition of $\text{pre}(\mathcal{M}, A, \cdot)$: $\mathcal{P}(S) \rightarrow \mathcal{P}(S)$$\text{pre}(\mathcal{M}, A, Z) \overset{\text{def}}{=} \{ s \in S \mid \text{there is } f \in D_A(s) \text{ such that } \text{out}(s, f) \subseteq Z \}$

- $\llbracket \langle A \rangle X \varphi \rrbracket^\mathcal{M} = \text{pre}(\mathcal{M}, A, [\varphi]^\mathcal{M})$.
  So, easy to compute $\llbracket \langle A \rangle X \varphi \rrbracket^\mathcal{M}$ from $[\varphi]^\mathcal{M}$. 
Proof of $[\llangle A \rrangle X \varphi]^m = \text{pre}(\mathcal{M}, A, [\varphi]^m)$

- By definition, $\text{pre}(\mathcal{M}, A, [\varphi]^m)$ is equal to
  $$\{s \in S \mid \text{there is } f \in D_A(s) \text{ such that } \text{out}(s, f) \subseteq [\varphi]^m\}$$

- Let $s \in \text{pre}(\mathcal{M}, A, [\varphi]^m)$. There is $f \in D_A(s)$ such that $\text{out}(s, f) \subseteq [\varphi]^m$.

- Let $F$ be a strategy such that $F(s) = f$.

- The strategy $F$ witnesses satisfaction of $\mathcal{M}, s \models \llangle A \rrangle X \varphi$.

- Conversely, if $\mathcal{M}, s \models \llangle A \rrangle X \varphi$ witnessed by $F$, then $s \in \text{pre}(\mathcal{M}, A, [\varphi]^m)$ as $\text{out}(s, F(s)) \subseteq [\varphi]^m$.
Equivalences based on fixpoint characterisations

\[
\langle A \rangle G \varphi \iff \varphi \land \langle A \rangle X \langle A \rangle G \varphi
\]

\[
\langle A \rangle (\varphi U \psi) \iff (\psi \lor (\varphi \land \langle A \rangle X \langle A \rangle (\varphi U \psi)))
\]
Fixpoint theory

- \( G : \mathcal{P}(X) \rightarrow \mathcal{P}(X) \) is **monotone** if for all \( Y_1, Y_2 \subseteq X \), \( Y_1 \subseteq Y_2 \) implies \( G(Y_1) \subseteq G(Y_2) \).

- Given \( G : \mathcal{P}(X) \rightarrow \mathcal{P}(X) \), a set \( Y \subseteq X \) is

  - a **fixpoint** of \( G \) if \( G(Y) = Y \),
  - a **least fixpoint** if \( Y \) is a fixpoint and \( Y \subseteq Z \) for every fixpoint \( Z \),
  - a **greatest fixpoint** if \( Y \) is a fixpoint and \( Y \supseteq Z \) for every fixpoint \( Z \).
Knaster-Tarski Theorem: a restricted form

- Knaster-Tarski Theorem (a restricted form).
  Let \( G : \mathcal{P}(X) \rightarrow \mathcal{P}(X) \) be a monotone operator. Then \( G \) has
  - a least fixpoint \( \mu G \) and,
  - a greatest fixpoint \( \nu G \).

- Moreover, \( \mu G \) obtained by applying the successive iterations of \( G \) beginning with \( \emptyset \) until a fixpoint is reached.
  \[
  \emptyset \subseteq G(\emptyset) \subseteq G^2(\emptyset) \subseteq G^3(\emptyset) \ldots
  \]

- \( \nu G \) obtained by applying the successive iterations of \( G \), beginning with \( X \), until a fixpoint is reached.
  \[
  X \supseteq G(X) \supseteq G^2(X) \supseteq G^3(X) \ldots
  \]
Given $A \subseteq \text{Agt}$, a formula $\varphi$, and a CGS $\mathcal{M}$, we define $G_{A,\varphi}: \mathcal{P}(S) \rightarrow \mathcal{P}(S)$:

$$G_{A,\varphi}(Z) \overset{\text{def}}{=} \mathcal{[\varphi]}^\mathcal{M} \cap \text{pre}(\mathcal{M}, A, Z).$$

$G_{A,\varphi}(S)$ contains all the states satisfying $\varphi$.

$$\text{pre}(\mathcal{M}, A, S) = S$$

$G_{A,\varphi}(G_{A,\varphi}(S))$ contains all the states satisfying $\varphi$ and $A$ has a strategy such that in one step all the states satisfy $\varphi$.

$G^n_{A,\varphi}(S)$ contains all the states satisfying $\varphi$ and $A$ has a strategy such that in the steps $0, \ldots, n - 1$ all the states satisfy $\varphi$.

$$G^n_{A,\varphi}(S) \subseteq G^{n-1}_{A,\varphi}(S) \subseteq \cdots \subseteq G^1_{A,\varphi}(S))$$

$G_{A,\varphi}^n = \nu Z. (G_{A,\varphi} \cap \text{pre}(\mathcal{M}, A, Z))$ (greatest fixpoint)
About $G_{A,\varphi}$

- $G_{A,\varphi}$ is monotone as $\text{pre}$ is monotone.

- Computing $\nu Z. ([\varphi]^m \cap \text{pre}(M, A, Z))$.
  - $X_0 = S$.
  - $X_1 = [\varphi]^m \cap \text{pre}(M, A, X_0)$.
  - $X_2 = [\varphi]^m \cap \text{pre}(M, A, X_1)$.
  - $\ldots$
  - $X_{i+1} = [\varphi]^m \cap \text{pre}(M, A, X_i)$.
  - $\ldots$

- For all $i$, $X_{i+1} \subseteq X_i$.  
  \textbf{(proof left as an exercise)}

- There is $N \leq \text{card}(S)$ such that $X_N = X_{N+1} = X_{N+2} = \ldots$. 
\[ \lbrack \lbrack A \rbrack \varphi U \psi \rbrack^m \] is a least fixpoint

- Given \( A \subseteq \text{Agt} \), formulae \( \varphi, \psi \), and a CGS \( M \), we define \( O_{A,\varphi,\psi} : \mathcal{P}(S) \rightarrow \mathcal{P}(S) \):

\[
O_{A,\varphi,\psi}(Z) \overset{\text{def}}{=} \lbrack \psi \rbrack^m \cup \left( \lbrack \varphi \rbrack^m \cap \text{pre}(M, A, Z) \right)
\]

- \( O_{A,\varphi,\psi}(\emptyset) \) contains all the states satisfying \( \psi \).
  \[ \text{pre}(M, A, \emptyset) = \emptyset \]

- \( O_{A,\varphi,\psi}(O_{A,\varphi,\psi}(\emptyset)) \) contains all the states satisfying \( \psi \) or those satisfying \( \varphi \) and such that \( A \) has a strategy such that in one step all the states satisfy \( \psi \).
\( [[\langle A \rangle \varphi U \psi]]^m \) is a least fixpoint (bis)

- \( O^n_{A,\varphi,\psi}(\emptyset) \) contains all the states satisfying \( \psi \) or those satisfying \( \varphi \) and such that \( A \) has a strategy such that in at most \( n \) steps, a state satisfying \( \psi \) is reached and in between all the states satisfy \( \varphi \).

- \( O^1_{A,\varphi,\psi}(\emptyset) \subseteq O^2_{A,\varphi,\psi}(\emptyset) \subseteq \cdots \subseteq O^n_{A,\varphi,\psi}(\emptyset) \).

- \( [[\langle A \rangle \varphi U \psi]]^m = \mu Z.([\psi]^m \cup ([\varphi]^m \cap \text{pre}(M, A, Z))). \) (least fixpoint)

- Valid formula

\[
\langle A \rangle \varphi U \psi \iff \psi \vee (\varphi \land \langle A \rangle X \langle A \rangle \varphi U \psi)
\]
About $O_{A, \varphi, \psi}$

- $O_{A, \varphi, \psi}$ is monotone as $\text{pre}$ is monotone.

- Computing $\mu Z.([\psi]^m \cup ([\varphi]^m \cap \text{pre}(M, A, Z)))$.
  - $X_0 = \emptyset$.
  - $X_1 = ([\psi]^m \cup ([\varphi]^m \cap \text{pre}(M, A, X_0))$.
  - $X_2 = ([\psi]^m \cup ([\varphi]^m \cap \text{pre}(M, A, X_1))$.
  - $\ldots$
  - $X_{i+1} = ([\psi]^m \cup ([\varphi]^m \cap \text{pre}(M, A, X_i))$.
  - $\ldots$

- For all $i$, $X_i \subseteq X_{i+1}$. \textit{(proof left as an exercise)}

- There is $N \leq \text{card}(S)$ such that $X_N = X_{N+1} = X_{N+2} = \ldots$. 
Conclusion

- Today lecture.
  - Concurrent game structures (CGS).
  - Introduction to ATL.

- Next week lecture.
  - Correction of the exercises.
  - Model-checking problem for ATL in PTIME and other variants from ATL (≈ 1h).
  - More exercises to prepare the exam (on Nov. 10th).