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

Stéphane Demri
demi@lmf.cn*s.f*

November 29th, 2023
Plan of the lecture

- Concurrent game structures.
- Introduction to ATL.
- Exercises session.
Exam on Wednesday January 10th 2024, 2pm-5pm.

Room 1E14 (usual room).
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

- 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 Robot_1 and Robot_2, and a carriage.
- Robot_1 can only push the carriage in clockwise direction, Robot_2 can only push it in anti-clockwise direction.
Concurrent game structure: definition

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

- **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. \( \text{act}(a, s) \approx \) “set of actions that can be executed by the agent \( a \) from the control state \( s \)”.
- **Transition function** \( \delta : S \times (\text{Agt} \rightarrow \text{Act}) \rightarrow S \). \( \delta(s, f) \) undefined if there is some agent \( a \) such that \( f(a) \not\in \text{act}(a, s) \).
An example

Action manager $\text{act} : \text{Agt} \times S \rightarrow \mathcal{P}(\text{Act}) \setminus \{\emptyset\}$.
$\text{act}(1, s_3) = \{c\}; \text{act}(2, s_3) = \{c\}$.

Transition function $\delta : S \times (\text{Agt} \rightarrow \text{Act}) \rightarrow S$.
$\delta(s_4, [1 \mapsto c, 2 \mapsto c]) = s_3 \quad \text{— undef.} \quad \delta(s_4, [1 \mapsto c, 2 \mapsto a])$.

Labelling $L : S \rightarrow \mathcal{P}(\text{PROP})$.
$L(s_1) = \{p\}$. 

$\text{Agt} = \{1, 2\}$
$S = \{s_1, s_2, s_3, s_4\}$
$\text{Act} = \{a, b, c\}$
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 $\text{card}(\text{act}(a, s)) > 1$. 
The Logic ATL and Variants
Logics of strategic ability

- To express that a coalition of agents has a collective strategy to enforce some property and to reason on it.
- A strategy is a conditional plan intended to work whatever the other agents do.

- Well-known specimens.
  - Coalition Logic CL. (one-step strategies)
  - Alternating-time temporal logic ATL. (generalisation of temporal logics)
  - Strategy Logic SL. (explicit quantification over strategies)
Basic concepts: joint action

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

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

- $g : A \rightarrow Act \sqsubseteq g' : A' \rightarrow Act \overset{\text{def}}{\iff} A \subseteq A'$ and $g$ is the restriction of $g'$ to $A$.

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

(-' indicates undefinedness)

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


Basic concepts: outcome set

- Joint action $g : A \rightarrow \text{Act}$ in $s$.

- $\text{out}(s, g) \overset{\text{def}}{=} \text{set of states reachable from } s \text{ in one step when the actions performed by the agents in } A \text{ are determined by } g$.

- Set of outcomes:

  $$\text{out}(s, g) \overset{\text{def}}{=} \{ s' \in S \mid \exists f \in D_{Agt}(s) \text{ s.t. } g \sqsubseteq f \text{ and } s' = \delta(s, f) \}$$

```
out(s_0, [1 \mapsto a]) = \{s_2, s_3, s_4\}
out(s_0, [1 \mapsto b, 2 \mapsto a]) = \{s_1\}
```
Basic concepts: computations

- \( \text{card} (\text{out}(s, f)) = 1 \) 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 \( s_{i+1} \in \delta(s_i, f_i) \). \( \text{(history} = \text{finite computation}) \)

- 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)
Basic concepts: strategies

▶ A strategy is a condition plan intended to fulfill the objectives whatever the agents of the opponent coalition perform.

▶ In a strategy, the agents of the proponent coalition select actions depending on the sequence of states already visited.

▶ **Strategy** \( \sigma \) for \( A \): map from the set of finite computations (histories) to the set of joint actions by \( A \) such that

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

▶ The domain of a strategy is potentially infinite.
Positional strategies

- Memory-based strategies vs. positional strategies.

- $\sigma$ is a **positional strategy** $\overset{\text{def}}{\iff}$ 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
  \[
  \sigma(s_0 \xrightarrow{f_0} s_1 \cdots \xrightarrow{f_{n-1}} s_n) = \sigma(s'_0 \xrightarrow{f'_0} s'_1 \cdots \xrightarrow{f'_{m-1}} s'_m)
  \]
  \[
  \text{(only the value of the last state matters)}
  \]

- **Memoryless strategy** $\overset{\text{def}}{=} \text{positional strategy}.$
  \[
  \sigma : s \in S \mapsto g \in D_A(s)
  \]
Computations respecting a strategy

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

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

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

\[ \text{comp}(s, \sigma): \text{ set of maximal computations from the state } s \text{ respecting the strategy } \sigma. \]
Strategy tree (for agent 1)

\[(a, a) \xrightarrow{p_1} s_0 \xrightarrow{(a, b)} s_1 \xrightarrow{(b, a)} s_2 \xrightarrow{p_2} (a, a)\]

\[\ldots \xrightarrow{(a, a)} s_0 \xrightarrow{(a, b)} s_1 \xrightarrow{(b, a)} s_2 \xrightarrow{(a, a)} s_0 \xrightarrow{(a, b)} s_1 \xrightarrow{(b, a)} s_2 \xrightarrow{(a, a)} \ldots\]
Strategy tree (bis)

Positional $\sigma$ for the agent 1:
select $a$ on $s_1$, $b$ on $s_2$, otherwise $c$.

$\sigma$ generates a set of computations whose linear models can be defined by a Büchi automaton (BA).

(keep only transitions compatible with $\sigma$)
Trimming a CGS

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

- Coalition \( A \subseteq \text{Agt} \).

- Memoryless strategy \( \sigma : s \in S \mapsto g \in D_A(s) \).

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

- \( R \) represents the set of moves allowed by the opponent coalition \((\text{Agt} \setminus A)\) when \( A \) has the positional strategy \( \sigma \).
Examples of strategies

Positional strategy for Robot_1: \( \sigma(s_0) = \text{push}, \sigma(s_1) = \text{push}, \sigma(s_2) = \text{wait}. \)

The set of maximal computations respecting \( \sigma \) 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 \( \sigma \)?
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 \( \lambda : \mathbb{N} \to \mathcal{P}(PROP) \).

  (≈ linear models from infinite computations)

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

  \[\begin{align*}
  X \varphi & \quad \varphi \\
  \sigma & \rightarrow \rho & \rightarrow \sigma & \rightarrow \rho & \rightarrow \sigma & \rightarrow \sigma & \cdots
  \end{align*}\]
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.

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

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

- \( \lambda, i \models p \overset{\text{def}}{\iff} p \in \lambda(i) \),

- \( \lambda, i \models \neg \varphi \overset{\text{def}}{\iff} \lambda, i \not\models \varphi \),

- \( \lambda, i \models \varphi_1 \land \varphi_2 \overset{\text{def}}{\iff} \lambda, i \models \varphi_1 \text{ and } \lambda, i \models \varphi_2 \),

- \( \lambda, i \models X \varphi \overset{\text{def}}{\iff} \lambda, i + 1 \models \varphi \),

- \( \lambda, i \models \varphi_1 \mathcal{U} \varphi_2 \overset{\text{def}}{\iff} \text{there is } j \geq i \text{ such that } \lambda, j \models \varphi_2 \text{ and } \lambda, k \models \varphi_1 \text{ for all } i \leq k < j \).

\[
F \varphi \overset{\text{def}}{=} \top \mathcal{U} \varphi \quad G \varphi \overset{\text{def}}{=} \neg F \neg \varphi \quad \varphi \Rightarrow \psi \overset{\text{def}}{=} \neg \varphi \lor \psi \ldots
\]
About LTL

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

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

- Models(φ) 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 $\Phi$.  
  
  \[
  \varphi ::= p \mid \neg \varphi \mid \varphi \land \varphi \mid \langle A \rangle X \varphi \mid \langle A \rangle G \varphi \mid \langle A \rangle \varphi U \varphi \\
  p \in \text{PROP} \quad A \subseteq \text{Agt}
  \]
ATL modalities, informally

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

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

- $\langle A \rangle \text{U}\psi \varphi$: “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

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

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

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

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

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

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

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

▷ $M, s_0 \not\models \langle 1 \rangle X pos_1$ because if the agent 2 performs the action $push$, no transition leads to $s_1$ from $s_0$.

▷ $M, s_0 \not\models \langle 2 \rangle X pos_1$ because if the agent 1 performs the action $wait$, no transition leads to $s_1$ from $s_0$.

▷ $M, s_0 \models \langle 1, 2 \rangle X pos_0 \land \langle 1, 2 \rangle X pos_1 \land \langle 1, 2 \rangle X pos_2$ because each conjunct can be satisfied using a distinct (positional) strategy.

By way of example, for the satisfaction of $\langle 1, 2 \rangle X pos_0$, the strategy $\sigma$ verifies $\sigma(s_0) = [1 \mapsto push, 2 \mapsto push]$. 
Playing with formulae (II)

$\langle 1 \rangle F \langle 1 \rangle F pos_1$ because the positional strategy for the agent 2 that consists in performing $\text{push}$ on $s_0$ and $\text{wait}$ on $s_2$ never leads to the state $s_1$ from the state $s_0$.

$\langle 1 \rangle F (pos_1 \lor pos_2)$ because any strategy for the agent 1 would be fine. Indeed, $s_1$ already satisfies $pos_1$ by definition of the CGS $M$.

$\langle 1, 2 \rangle X \langle 1 \rangle (pos_0 \cup pos_2)$ because $\langle 1 \rangle (pos_0 \cup pos_2)$ (any strategy is fine as $s_2$ satisfies $pos_2$) and the coalition $\{1, 2\}$ has a strategy with $\sigma(s_0) = [1 \mapsto \text{wait}, 2 \mapsto \text{push}]$, leading to $s_2$ in one step.
Decision problems

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

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

- **Validity problem for ATL:**
  - Input: $\phi$ in ATL,
  - Question: Is it true that for all CGS $\mathcal{M}$ and $s$ in $\mathcal{M}$, we have $\mathcal{M}, s \models \phi$?
Computational complexity

- Model-checking problem for ATL is $\text{PTIME}$-complete. Labeling algorithm presented during the next lecture. (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.
  (alternative notation: $\models_R$ for $\models$ versus $\models_r$ for $\models_{pos}$)

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

- Positional strategies are sufficient for ATL:
  $$\mathcal{M}, s \models \varphi \iff \mathcal{M}, s \models_{pos} \varphi$$
  (Great! Quantifications over finite sets)

- This property does not hold for the extension ATL*. (see next lecture)
“Proof”: positional strategies are sufficient for ATL
Formulae $\langle A \rangle_{Gp}$
Formula $\langle A \rangle (\varphi_1 U \varphi_2)$
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 a CGS with only one agent).

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

▶ 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 $A(\varphi U \psi)$

$$AF\varphi \overset{\text{def}}{=} A\top U \varphi \quad EG\varphi \overset{\text{def}}{=} \neg AF\neg \varphi$$
CTL semantics

- Path $\pi$ in $\mathcal{T}$: 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.

\[ \mathcal{T}, s \models \text{EX} \varphi \quad \text{iff} \quad \text{there is } s' \text{ such that } (s, s') \in R \text{ and } \mathcal{T}, s' \models \varphi \]

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

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

- CTL model-checking problem is PTIME-complete.
- CTL satisfiability problem is EXPTIME-complete.
- Reduction from CTL satisfiability (resp. model-checking) to ATL satisfiability (resp. model-checking).

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

- **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 $\text{pre}(\mathcal{M}, A, \cdot)$**: $\mathcal{P}(S) \rightarrow \mathcal{P}(S)$

  \[
  \text{pre}(\mathcal{M}, A, Z) \overset{\text{def}}{=} \{ s \in S \mid \text{there is } g \in D_A(s) \text{ such that } \text{out}(s, g) \subseteq Z \} 
  \]
Example

\[
\text{pre}(\mathcal{M}, \{1\}, \{D, U_1, P\}) = \quad ??
\]
Proof of $\llbracket \langle A \rangle X \varphi \rrbracket^m = \mathsf{pre}(M, A, \llbracket \varphi \rrbracket^m)$

- By definition, $\mathsf{pre}(M, A, \llbracket \varphi \rrbracket^m)$ is equal to
  $$\{ s \in S \mid \text{there is } g \in D_A(s) \text{ such that } \text{out}(s, g) \subseteq \llbracket \varphi \rrbracket^m \}$$

- Let $s \in \mathsf{pre}(M, A, \llbracket \varphi \rrbracket^m)$. There is $g \in D_A(s)$ such that $\text{out}(s, g) \subseteq \llbracket \varphi \rrbracket^m$.

- Let $\sigma$ be a strategy such that $\sigma(s) = g$.

- The strategy $\sigma$ witnesses satisfaction of $M, s \models \langle A \rangle X \varphi$.

- Conversely, if $M, s \models \langle A \rangle X \varphi$ witnessed by $\sigma$, then $s \in \mathsf{pre}(M, A, \llbracket \varphi \rrbracket^m)$ as $\text{out}(s, \sigma(s)) \subseteq \llbracket \varphi \rrbracket^m$
Equivalences based on fixpoint characterisations

- How to compute $[[A] \varphi U \psi]^m$ and $[[A] G \varphi]^m$?

- Validity of the equivalences:
  
  $\langle A \rangle G \varphi \iff \varphi \land \langle A \rangle X \langle A \rangle G \varphi$

  $\langle A \rangle (\varphi U \psi) \iff (\psi \lor (\varphi \land \langle A \rangle X \langle A \rangle (\varphi U \psi)))$

- $[[A] G \varphi]^m$ and $[[A] (\varphi U \psi)]^m$ are fixpoints.
  
  (but in which sense?)
Fixpoint theory (rudiments)

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

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

Let $\mathcal{G} : \mathcal{P}(X) \to \mathcal{P}(X)$ be a monotone operator. Then $\mathcal{G}$ has

- a least fixpoint $\mu \mathcal{G}$ and,
- a greatest fixpoint $\nu \mathcal{G}$.

Moreover, $\mu \mathcal{G}$ obtained by applying the successive iterations of $\mathcal{G}$ beginning with $\emptyset$ until a fixpoint is reached.

$$\emptyset \subseteq \mathcal{G}(\emptyset) \subseteq \mathcal{G}^{2}(\emptyset) \subseteq \mathcal{G}^{3}(\emptyset) \cdots$$

$\nu \mathcal{G}$ obtained by applying the successive iterations of $\mathcal{G}$, beginning with $X$, until a fixpoint is reached.

$$X \supseteq \mathcal{G}(X) \supseteq \mathcal{G}^{2}(X) \supseteq \mathcal{G}^{3}(X) \cdots$$

If $X$ is finite, the fixpoints $\mu \mathcal{G}$ and $\nu \mathcal{G}$ can be obtained in a number of steps bounded by $\text{card}(X)$. 
$[[A]G\varphi]^m$ is a greatest fixpoint

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

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

- $G_{A,\varphi}(S)$ 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 $\varphi$.

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

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

- $[[A]G\varphi]^m = \nu Z.([\varphi]^m \cap \text{pre}(\mathcal{M}, A, Z))$ (greatest fixpoint)
About $G_{A,\Phi}$

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

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

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

- There is $N \leq \text{card}(S)$ such that $X_N = X_{N+1} = X_{N+2} = \cdots$. 
\[[\langle A \rangle \varphi \cup \psi]^M\] is a least fixpoint

- Given \( A \subseteq Agt \), 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 \( \psi \).
$\lbrack \langle A \rangle \varphi U \psi \rbrack^m$ 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 $\varphi$.

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

- $\lbrack \langle A \rangle \varphi U \psi \rbrack^m = \mu Z. (\lbrack \psi \rbrack^m \cup (\lbrack \varphi \rbrack^m \cap \text{pre}(\mathcal{M}, A, Z)))$.

- (least fixpoint)

- Valid formula

$$\langle A \rangle \varphi U \psi \iff \psi \lor (\varphi \land \langle A \rangle X \langle A \rangle \varphi U \psi)$$
About $\mathcal{O}_{A,\varphi,\psi}$

- $\mathcal{O}_{A,\varphi,\psi}$ is monotone as $\text{pre}$ is monotone.

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

- For all $i$, $X_i \subseteq X_{i+1}$. (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.
   - Fixpoints and operators.

▶ Next week lecture.
   - Correction of the exercises.
   - Model-checking problem for ATL in PTIME and other variants from ATL.
   - ATL with incomplete information
   - $\text{ATL}^+$: between ATL and $\text{ATL}^*$, PSPACE-hardness.