ESSAI 2024 course: Logic-based specification and verification of multi-agent systems

Lecture 1: Introduction. Multi-agent transition systems and concurrent game models. The alternating time temporal logic ATL

Valentin Goranko
Stockholm University

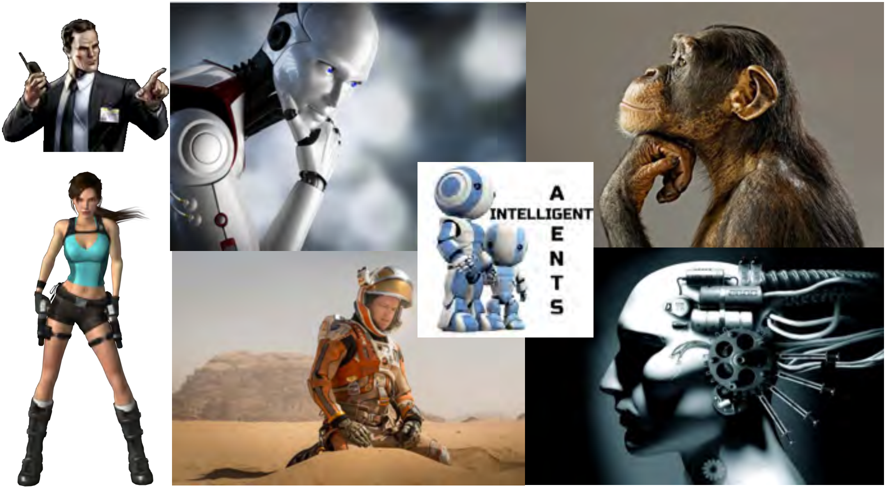
2nd European Summer School on Artificial Intelligence
ESSAI 2024
Athens, July 15-19, 2024

Overview of the lecture

- Introduction: agents and multi-agent systems (MAS),
- Multi-agent transition systems and concurrent game models
- The temporal logic ATL for reasoning about strategic abilities in multi-agent systems
- Logical decision problems for ATL and their algorithmic solutions.
- Solving the model checking problem for ATL.

Introduction: agents and multi-agent systems

Introduction: (intelligent) agents



Introduction: multi-agent systems (MAS)

Introduction: Agents and multi-agent systems

- Agents:

▷ relatively autonomous.

▷ have knowledge/information: about the system, themselves, and the other agents (incl. the environment).

▷ have abilities to perform certain actions.

▷ have goals, and can act in their pursuit.

▷ can plan their actions ahead and can execute plans (strategies).

▷ Can communicate, i.e. exchange information and cooperate with other agents.

- Multi-agent system (MAS): a set of agents acting in a common framework ('system'), in pursuit of their goals, by following individual or collective strategies.

Examples: open systems, distributed systems, concurrent processes, computer networks, social networks, stock markets, etc.

Why using logic for multi-agent systems?

Formal logic provides a generic and uniform framework for:

- Formal representation and modelling of multi-agent systems.
- Formal specification of properties of MAS in logical languages.
- Conceptual analysis of multi-agent systems and the interaction of rational agents in them.
- Formal logical reasoning about multi-agent systems, using systems of deduction and logical decision procedures.
- Formal verification of properties of MAS by model checking. Applications e.g. to automated design of agents' strategies.
- Applications of constructive satisfiability testing to synthesis of agents, communication protocols, controllers, or entire multi-agent systems satisfying formally specified behavior or objectives.

Modelling multi-agent strategic interaction:

Multi-agent transition systems / concurrent game models

Multi-agent transition systems intuitively

▷ Agents (players) act in a common environment (the "system") by taking actions in a discrete succession of rounds.

▷ At any moment the system is in a current state.

▷ At the current state all players take simultaneously actions, each choosing from a set of available actions.

▷ The resulting collective action effects a transition to a successor state, where the same happens, resulting in a new transition, etc.

This dynamics is captured by a multi-player transition system.

Concurrent Game Models formally

$$\langle \mathbb{A}, \text{States}, \text{Act}, \text{act}, \text{out}, \mathrm{Prop}, L \rangle$$

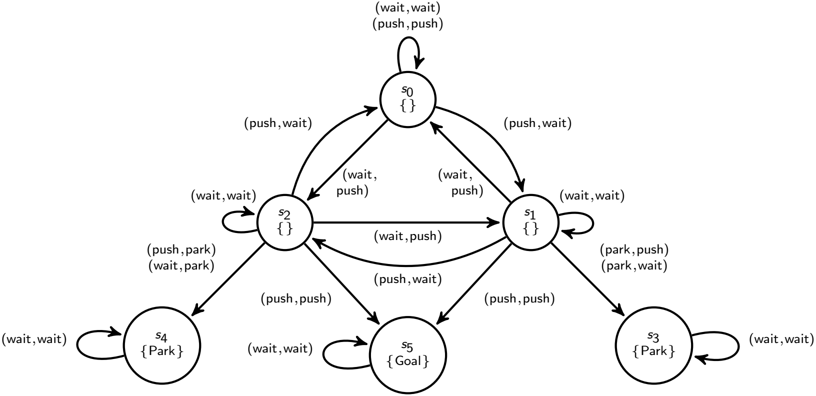
where:

- $\mathbb{A}$ is a finite set of agents (players);
- States is a set of system states;
- Act is a set of possible actions. An action profile is a mapping $\sigma : \mathbb{A} \to \text{Act}$, i.e. a tuple of actions, one for each agent.
- $\text{act} : \mathbb{A} \times \text{States} \to \mathcal{P}(\text{Act})$ – mapping assigning to every agent $\mathbf{i}$ and state s a non-empty set $\text{act}(\mathbf{i}, s)$ of actions available to $\mathbf{i}$ at s.

  An action profile σ is available at s if $\sigma(\mathbf{i}) \in \text{act}(\mathbf{i}, s)$, for each $\mathbf{i} \in \mathbb{A}$.
- $\text{out} : \text{States} \to (\text{Act}^{\mathbb{A}} - \to \text{States})$ is a global outcome (partial) function, assigning for every $s \in \text{States}$ and an available action profile σ the successor (outcome) state $\text{out}(s, \sigma)$.
- Prop is the set of atomic propositions;
- $L : \text{States} \to \mathcal{P}(\mathrm{Prop})$ is the labeling (state description) function.

Example: a two-agent transition system

Two robots, **Yin** and **Yang**, are pushing a trolley along tracks.
Usually Yin pushes clockwise and Yang pushes anticlockwise, with the same force.
Exception: when both push at either state s_1 or s_2 the trolley moves to s_5.



- $\mathbb{A} = \{$**Yin**, **Yang**$\}$; States $= \{s_0, s_1, s_2, s_3, s_4, s_5\}$; Act $= \{$push, wait, park$\}$.
- Action function: as on the figure. Outcome function: as on the figure.
- $\mathrm{Prop} = \{$Goal, Park$\}$. L : States $\rightarrow \mathcal{P}(\mathrm{Prop})$ defined as on the figure: $L(s_0) = L(s_1) = L(s_2) = \emptyset$, $L(s_5) = \{$Goal$\}$, $L(s_3) = L(s_4) = \{$Park$\}$.

Plays and strategies in concurrent game models

Given a CGM $\mathcal{M} = \langle \mathbb{A}, \mathsf{States}, \mathsf{Act}, \mathsf{act}, \mathsf{out}, \mathrm{Prop}, L \rangle$ and a state $s \in \mathsf{States}$:

- A state s' in $\mathcal{M}$ is a successor of the state s if there is an available action profile $(\sigma_{\mathbf{1}}, ..., \sigma_{\mathbf{n}}) \in \Sigma_s$ such that $s' = \mathsf{out}(s; \sigma_{\mathbf{1}}, ..., \sigma_{\mathbf{n}})$.

  The set of successors of s: $\mathbf{succ}(s)$.
- A play in $\mathcal{M}$: an infinite sequence $s_0, s_1, ...,$ such that $s_{i+1} \in \mathbf{succ}(s_i)$.
- A (perfect recall) strategy in $\mathcal{M}$ for an agent $\mathbf{i} \in \mathbb{A}$: a mapping $f_{\mathbf{i}} : \mathsf{States}^+ \to \mathsf{Act}$ that assigns to every finite sequence of states $s_0, ..., s_n$ an action $f_{\mathbf{i}}(\langle s_0, ..., s_n \rangle) \in \mathsf{act}(s_n, \mathbf{i})$.

  A no recall (memoryless, positional) strategy is one that prescribes actions only depending on the current state.
- A collective strategy in $\mathcal{M}$ for a set (coalition) of agents C: a family of strategies $f_{\mathrm{C}} = \{f_{\mathbf{i}}\}_{\mathbf{i} \in \mathrm{C}}$.
- A collective strategy f_{C} enables a play λ if that play can occur as a result of the players in C following their strategies in f_{C}.

The multi-agent logic of strategic reasoning ATL(*)

The multi-agent logic of strategic reasoning ATL(*)

Alternating-time Temporal Logic ATL(*): introduced by Alur, Henzinger, and Kupferman, during 1997-2002. Extends propositional logic PL with:

- *Temporal operators:* $\mathcal{X}$ (next time), $\mathcal{G}$ (forever), $\mathcal{U}$ (until)
- *Coalitional strategic path operators:* $\langle\langle A \rangle\rangle$ for any group of agents A. We will write $\langle\langle \mathbf{i} \rangle\rangle$ instead of $\langle\langle \{\mathbf{i}\} \rangle\rangle$.

Syntax of the full version ATL*:

$$\varphi := p \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \langle\langle A \rangle\rangle\varphi \mid \mathcal{X}\varphi \mid \mathcal{G}\varphi \mid \varphi_1 \mathcal{U} \varphi_2$$

Syntax of the restricted version ATL:

$$\varphi := p \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \langle\langle A \rangle\rangle\mathcal{X}\varphi \mid \langle\langle A \rangle\rangle\mathcal{G}\varphi \mid \langle\langle A \rangle\rangle\varphi_1 \mathcal{U} \varphi_2$$

Remark: the computation tree logic CTL(*) can be regarded as a fragment of ATL(*), where:
- the existential path quantifier E is identified with $\langle\langle \mathbb{A} \rangle\rangle$,
- the universal path quantifier A is identified with $\langle\langle \emptyset \rangle\rangle$.
One agent suffices.

Semantics of ATL intuitively

$\langle\langle A \rangle\rangle\varphi$: *"The coalition A has a collective strategy to guarantee the satisfaction of the goal φ on every play enabled by that strategy."*

In particular:

- $\langle\langle A \rangle\rangle \mathcal{X}\varphi$: 'The coalition A *has a collective action that ensures an outcome (state) satisfying* φ',
- $\langle\langle A \rangle\rangle \mathcal{G}\varphi$: 'The coalition A *has a collective strategy to maintain forever outcomes satisfying* φ',
- $\langle\langle A \rangle\rangle \psi \mathcal{U} \varphi$: 'The coalition A *has a collective strategy to eventually reach an outcome satisfying* φ, *while meanwhile maintaining the truth of* ψ'.

Definable operators:

- $\langle\langle A \rangle\rangle \mathcal{F}\varphi := \langle\langle A \rangle\rangle \top \mathcal{U} \varphi$, meaning *'The coalition A has a collective strategy to eventually reach an outcome satisfying* φ*'.*
- $[[A]]\,\varphi := \neg\langle\langle A \rangle\rangle\neg\varphi$, meaning:
  'The coalition A cannot prevent the satisfaction of φ'.

Expressing properties in ATL: some examples

$$\langle\langle \textbf{Yin} \rangle\rangle \mathcal{F}\ \mathrm{Park} \rightarrow [[\textbf{Yang}]]\ \mathcal{F}\ \mathrm{Park}$$

If **Yin** *has a strategy to eventually park the trolley,*
then **Yang** *cannot prevent the parking of the trolley.*

$$\neg\langle\langle \textbf{Yin} \rangle\rangle \mathcal{X}\ \mathrm{Goal} \wedge \neg\langle\langle \textbf{Yang} \rangle\rangle \mathcal{X}\ \mathrm{Goal} \wedge \langle\langle \{\textbf{Yin}, \textbf{Yang}\} \rangle\rangle \mathcal{X}\ \mathrm{Goal}$$

Neither **Yin** *nor* **Yang** *has has an action ensuring an outcome satisfying* Goal*,*
but they both have a collective action ensuring such outcome.
(True at states s_1 and s_2 in the example.)

$$(\langle\langle \textbf{Yin} \rangle\rangle \mathcal{G}\ \mathrm{Safe} \wedge \langle\langle \textbf{Yin} \rangle\rangle \mathcal{F}\ \mathrm{Goal}) \rightarrow \langle\langle \textbf{Yin} \rangle\rangle (\mathrm{Safe}\ \mathcal{U}\ \mathrm{Goal})$$

If **Yin** *has a strategy to keep the system in safe states forever and has a strategy to eventually achieve its goal, then* **Yin** *has a strategy to keep the system in safe states until it achieves its goal.*

$$(\langle\langle \textbf{Yin} \rangle\rangle \mathcal{G}\ \mathrm{Safe} \wedge \langle\langle \textbf{Yang} \rangle\rangle \mathcal{F}\ \mathrm{Goal}) \rightarrow \langle\langle \textbf{Yin}, \textbf{Yang} \rangle\rangle (\mathrm{Safe}\ \mathcal{U}\ \mathrm{Goal})$$

If **Yin** *has a strategy to keep the system in safe states forever and* **Yang** *has a strategy to eventually reach a goal state, then* **Yin** *and* **Yang** *together have a strategy to stay in safe states until a goal state is reached.*

ATL semantics: formally

Truth of a formula ψ at a state s of a CGM $\mathcal{M}$:

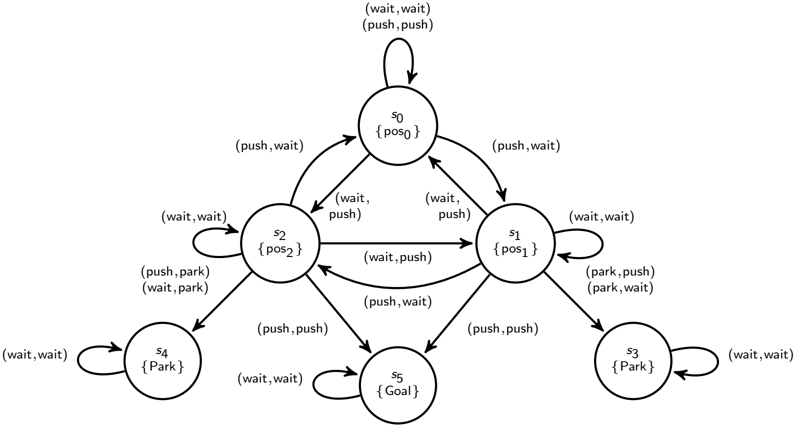
$$\mathcal{M}, s \models \psi$$

Defined by structural induction on formulae, via the clauses:

- $\mathcal{M}, s \models \langle\langle A\rangle\rangle \mathcal{X}\varphi$ iff there exists a collective strategy $F_A = \{f_\mathbf{i}\}_{\mathbf{i}\in \mathbf{A}}$ such that $\mathcal{M}, s_1 \models \varphi$ for every s-play $s, s_1, \ldots$ enabled by F_A.
- $\mathcal{M}, s \models \langle\langle A\rangle\rangle \mathcal{G}\varphi$ iff there exists a collective strategy $F_A = \{f_\mathbf{i}\}_{\mathbf{i}\in \mathbf{A}}$ such that $\mathcal{M}, s_i \models \varphi$ for every s-play $s, s_1, \ldots$ enabled by F_A and $i \geq 0$.
- $\mathcal{M}, s \models \langle\langle A\rangle\rangle \varphi \mathcal{U} \psi$ iff there exists a collective strategy $F_A = \{f_\mathbf{i}\}_{\mathbf{i}\in \mathbf{A}}$ such that for every s-play $s, s_1, \ldots$ enabled by F_A there is $i \geq 0$ for which $\mathcal{M}, s_i \models \psi$ and for all j such that $0 \leq j < i$, $\mathcal{M}, s_j \models \varphi$.

For the semantics of ATL memoryless strategies suffice.

Deciding the truth of ATL formulae in a CGM: examples



$\mathcal{M}, s_0 \overset{?}{\models} \langle\!\langle \mathbf{Yin} \rangle\!\rangle \mathcal{X}\, pos_1$ N

$\mathcal{M}, s_0 \overset{?}{\models} \langle\!\langle \mathbf{Yin}, \mathbf{Yang} \rangle\!\rangle \mathcal{X}\, pos_1$ Y

$\mathcal{M}, s_0 \overset{?}{\models} \langle\!\langle \mathbf{Yin}, \mathbf{Yang} \rangle\!\rangle \mathcal{F}\, \mathrm{Goal}$ Y

$\mathcal{M}, s_0 \overset{?}{\models} \langle\!\langle \mathbf{Yin} \rangle\!\rangle \mathcal{G}\, \neg \mathrm{Park}$ Y

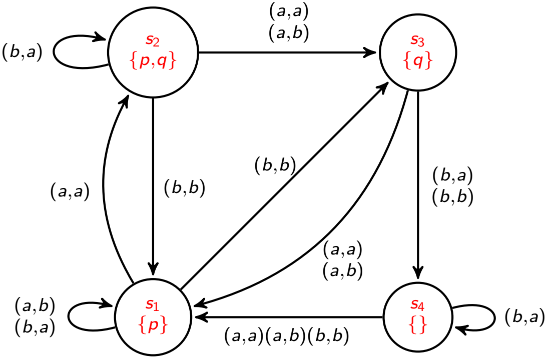
$\mathcal{M}, s_0 \overset{?}{\models} \langle\!\langle \mathbf{Yin}, \mathbf{Yang} \rangle\!\rangle ((\neg pos_1)\, \mathcal{U}\, \mathrm{Park})$ Y;

$\mathcal{M}, s_0 \overset{?}{\models} \langle\!\langle \mathbf{Yin} \rangle\!\rangle \mathcal{F}\, \langle\!\langle \mathbf{Yang} \rangle\!\rangle \mathcal{F}\, \mathrm{Park}$ N

$\mathcal{M}, s_0 \overset{?}{\models} \langle\!\langle \mathbf{Yin}, \mathbf{Yang} \rangle\!\rangle \mathcal{G}(\neg pos_1 \wedge \langle\!\langle \mathbf{Yin}, \mathbf{Yang} \rangle\!\rangle \mathcal{X} pos_1)$ Y

Deciding the truth of ATL formulae: exercises

Two agents: 1 and 2. Two types of actions: a, b.



$\mathcal{M}, s_1 \overset{?}{\models} \langle\langle 1 \rangle\rangle \mathcal{F} q \vee \langle\langle 2 \rangle\rangle \mathcal{G} \neg q$ No

$\mathcal{M}, s_1 \overset{?}{\models} \langle\langle 1 \rangle\rangle \mathcal{G} p \wedge \langle\langle 2 \rangle\rangle \mathcal{G} p$ No

$\mathcal{M}, s_3 \overset{?}{\models} \langle\langle \emptyset \rangle\rangle \mathcal{F} \langle\langle 2 \rangle\rangle \mathcal{X} p$ Yes

$\mathcal{M}, s_2 \overset{?}{\models} \langle\langle 1 \rangle\rangle \mathcal{G} \langle\langle 1, 2 \rangle\rangle (\neg q \mathrm{U} p)$ Yes

Extending the semantics of ATL*

Two types of formulae in ATL*:

State formulae $\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\!\langle A \rangle\!\rangle\gamma$, where $A \subseteq \mathbb{A}$ and $p \in \mathrm{Prop}$.

Path formulae: $\gamma ::= \varphi \mid \neg\gamma \mid \gamma \wedge \gamma \mid \mathcal{X}\gamma \mid \mathcal{G}\gamma \mid \gamma\,\mathcal{U}\,\gamma$

The semantics of state formulae: as in ATL.

The semantics of path formulae: defined on paths (plays), as in LTL.

ATL* is much more expressive and has more complex semantics.

Strategies generally need memory. Example: $\langle\!\langle \mathbf{a} \rangle\!\rangle(\mathcal{F}p \wedge \mathcal{F}q)$.
(Exercise: find a simple model where this is true at some state if memory-based strategies are used, but false if only positional strategies are allowed.)

Nesting of strategic operators causes higher complexity and also some problems with the semantics.

Logical decision problems in ATL

Validity and satisfiability in ATL

An ATL formula ϕ is:

- (logically) valid if $\mathcal{M}, s \models \phi$ for every CGM $\mathcal{M}$ and a state $s \in \mathcal{M}$.
- satisfiable if $\mathcal{M}, s \models \phi$ for some CGM $\mathcal{M}$ and a state $s \in \mathcal{M}$.

Axiomatizing the validities of ATL: local axioms

Pauly (2000) introduced the *Coalition Logic* CL, which is essentially the $\langle\!\langle\rangle\!\rangle\mathcal{X}$-fragment of ATL. Pauly's complete axiomatization of CL extends the classical propositional logic with the following axioms and rule:

- $\mathbb{A}$-Maximality: $\neg\langle\!\langle\emptyset\rangle\!\rangle\mathcal{X}\neg\varphi \to \langle\!\langle\mathbb{A}\rangle\!\rangle\mathcal{X}\varphi$
- Safety: $\neg\langle\!\langle C\rangle\!\rangle\mathcal{X}\bot$
- Liveness: $\langle\!\langle C\rangle\!\rangle\mathcal{X}\top$
- Superadditivity: for any $C_1, C_2 \subseteq \mathbb{A}$ such that $C_1 \cap C_2 = \emptyset$:

$$(\langle\!\langle C_1\rangle\!\rangle\mathcal{X}\varphi_1 \wedge \langle\!\langle C_2\rangle\!\rangle\mathcal{X}\varphi_2) \to \langle\!\langle C_1 \cup C_2\rangle\!\rangle\mathcal{X}(\varphi_1 \wedge \varphi_2)$$

- $\langle\!\langle C\rangle\!\rangle\mathcal{X}$-Monotonicity Rule:

$$\frac{\varphi_1 \to \varphi_2}{\langle\!\langle C\rangle\!\rangle\mathcal{X}\varphi_1 \to \langle\!\langle C\rangle\!\rangle\mathcal{X}\varphi_2}$$

Axiomatizing the validities of ATL: fixpoint axioms

The axiomatization of CL extends to one for ATL with the following fixed point axioms and rules for $\mathcal{G}$ and $\mathcal{U}$:

(FP$_\mathcal{G}$) $\langle\langle C\rangle\rangle\mathcal{G}\varphi \leftrightarrow \varphi \wedge \langle\langle C\rangle\rangle\mathcal{X}\langle\langle C\rangle\rangle\mathcal{G}\varphi$.

(GFP$_\mathcal{G}$) $\langle\langle\emptyset\rangle\rangle\mathcal{G}(\theta \rightarrow (\varphi \wedge \langle\langle C\rangle\rangle\mathcal{X}\theta)) \rightarrow \langle\langle\emptyset\rangle\rangle\mathcal{G}(\theta \rightarrow \langle\langle C\rangle\rangle\mathcal{G}\varphi)$,

(FP$_\mathcal{U}$) $\langle\langle C\rangle\rangle\psi\,\mathcal{U}\,\varphi \leftrightarrow \varphi \vee (\psi \wedge \langle\langle C\rangle\rangle\mathcal{X}\langle\langle C\rangle\rangle\psi\,\mathcal{U}\,\varphi)$,

(LFP$_\mathcal{U}$) $\langle\langle\emptyset\rangle\rangle\mathcal{G}((\varphi \vee (\psi \wedge \langle\langle C\rangle\rangle\mathcal{X}\theta)) \rightarrow \theta) \rightarrow \langle\langle\emptyset\rangle\rangle\mathcal{G}(\langle\langle C\rangle\rangle\psi\,\mathcal{U}\,\varphi \rightarrow \theta)$,

plus the rule $\langle\langle\emptyset\rangle\rangle\mathcal{G}$-Necessitation:

$$\frac{\varphi}{\langle\langle\emptyset\rangle\rangle\mathcal{G}\varphi}.$$

Completeness: VG and G. van Drimmelen (TCS'2006).

Logical decision problems in ATL

- Local model checking: given an ATL formula ψ, a finite CGM $\mathcal{M}$ and a state $s \in \mathcal{M}$, determine whether $\mathcal{M}, s \models \psi$.

- Global model checking: given an ATL formula ψ and a finite CGM $\mathcal{M}$, determine the set $\|\psi\|_{\mathcal{M}}$ of states in $\mathcal{M}$ where ψ is true.

Used for automated verification of formal specifications in open and multi-agent systems and synthesis of strategies and protocols.

- Satisfiability testing: given an ATL formula ψ, determine whether ψ is satisfiable, i.e., whether $\mathcal{M}, s \models \psi$ for some CGM $\mathcal{M}$ and a state $s \in \mathcal{M}$.

- Constructive satisfiability testing: given an ATL formula ψ, determine whether ψ is satisfiable, and if so, construct a CGM $\mathcal{M}$ and a state $s \in \mathcal{M}$ such that $\mathcal{M}, s \models \psi$.

Used for synthesis of multi-agent systems and controllers from formal specifications.

Solving the model checking problem for ATL

- Alur, Henzinger, and Kupferman [JACM'2002] extend the labeling algorithm for model checking for CTL to ATL and show that the model checking of ATL is PTIME-complete.

- They also extend the method to Fair ATL (ATL with fairness constraints) and to the full ATL* and show that:

- model checking of Fair ATL is PSPACE-complete

- model-checking ATL* is 2EXPTIME-complete
(even in the special case of turn-based synchronous models).

- Furthermore, under assumptions of incomplete information and perfect memory, model checking of ATL becomes undecidable.

Solving the satisfiability problem for ATL

VG and G. van Drimmelen [TCS'2006]: an algorithm for deciding SAT, using alternating tree automata and *bounding-branching model property*.

- VG and D. Shkatov [ToCL'2010]: constructive and practically usable tableau-based method for deciding for ATL in EXPTIME.

- VG, S. Cerrito, and A. David [ToCL'2014]: extended to ATL^+ (with goals being boolean combinations of ATL goals).

Extended to ATL and implemented in 2013-2015 by Amélie David (Univ. d'Evry Val d'Essonne)*. Links:

for ATL: `http://atila.ibisc.univ-evry.fr/tableau_ATL`

for ATL*: `https://atila.ibisc.univ-evry.fr/tableau_ATL_star`

Sven Schewe [ICALP'2008]: SAT for ATL^* is 2EXPTIME-complete. Uses automata on infinite trees. Implementation?

Addendum:
Solving the model checking problem for ATL

The operator Pre

Given a CGM $\mathcal{M} = \langle \mathbb{A}, S, Act, d, \mathsf{out}, \mathrm{Prop}, L \rangle$ a coalition $C \subseteq \mathbb{A}$ and a set $X \subseteq S$, we define $\mathrm{Pre}(\mathcal{M}, C, X)$ as the set of states from which the coalition C has a collective action that guarantees the outcome to be in X, no matter how the remaining agents act.

Formally:

$$\mathrm{Pre}(\mathcal{M}, C, X) := \{s \in S \mid \exists \alpha_C \forall \alpha_{\mathbb{A} \setminus C} \mathsf{out}(s, \alpha_C, \alpha_{\mathbb{A} \setminus C}) \in X\}$$

where α_C denotes a vector of moves for the set of agents C.

In particular, $\mathrm{Pre}(\mathcal{M}, C, \|\varphi_{\mathcal{M}}\|)$ is precisely the set of states in $\mathcal{M}$ where the formula $\langle\langle C \rangle\rangle \mathcal{X} \varphi$ is true.

The temporal operators as fixed points: $\langle\langle C \rangle\rangle \mathcal{G}$

The validity $\langle\langle C \rangle\rangle \mathcal{G}\varphi \leftrightarrow \varphi \wedge \langle\langle C \rangle\rangle \mathcal{X} \langle\langle C \rangle\rangle \mathcal{G}\varphi$

means that $\| \langle\langle C \rangle\rangle \mathcal{G}\varphi \|_{\mathcal{M}}$ is a fixed point of the operator

$$\mathbf{G}_{C,\varphi}(Z) := \|\varphi\|_{\mathcal{M}} \cap \mathrm{Pre}(\mathcal{M}, C, Z)$$

The validity $\langle\langle \emptyset \rangle\rangle \mathcal{G}(\theta \rightarrow (\varphi \wedge \langle\langle C \rangle\rangle \mathcal{X}\theta)) \rightarrow \langle\langle \emptyset \rangle\rangle \mathcal{G}(\theta \rightarrow \langle\langle C \rangle\rangle \mathcal{G}\varphi)$

means that $\| \langle\langle C \rangle\rangle \mathcal{G}\varphi \|_{\mathcal{M}}$ is the greatest (post)-fixed point of $\mathbf{G}_{C,\varphi}$.

Therefore: $\| \langle\langle C \rangle\rangle \mathcal{G}\varphi \|_{\mathcal{M}}$ **can be computed by starting from** $Z = \text{States}$ **and iteratively applying** $\mathbf{G}_{C,\varphi}$ **until stabilization.**

It suffices to reach a stage where $Z \subseteq \mathbf{G}_{C,\varphi}(Z)$.

Then $\mathbf{G}_{C,\varphi}(Z) = Z$ will hold.

The temporal operators as fixed points: $\langle\langle C \rangle\rangle \mathcal{U}$

The validity $\langle\langle C \rangle\rangle \psi \mathcal{U} \varphi \leftrightarrow \varphi \vee (\psi \wedge \langle\langle C \rangle\rangle \mathcal{X} \langle\langle C \rangle\rangle \psi \mathcal{U} \varphi)$

means that $\|\langle\langle C \rangle\rangle \psi \mathcal{U} \varphi\|_{\mathcal{M}}$ is a fixed point of the operator

$$\mathbf{U}_{C,\varphi,\psi}(Z) := \|\varphi\|_{\mathcal{M}} \cup (\|\psi\|_{\mathcal{M}} \cap \mathrm{Pre}(\mathcal{M}, C, Z))$$

The validity $\langle\langle \emptyset \rangle\rangle \mathcal{G}((\varphi \vee (\psi \wedge \langle\langle C \rangle\rangle \mathcal{X} \theta)) \to \theta) \to \langle\langle \emptyset \rangle\rangle \mathcal{G}(\langle\langle C \rangle\rangle \psi \mathcal{U} \varphi \to \theta)$

means that $\|\langle\langle C \rangle\rangle \psi \mathcal{U} \varphi\|_{\mathcal{M}}$ is the least (pre)-fixed point of $\mathbf{U}_{C,\varphi,\psi}$.

Therefore: $\|\langle\langle C \rangle\rangle \psi \mathcal{U} \varphi\|_{\mathcal{M}}$ **can be computed by starting from** $Z = \emptyset$ **and iteratively applying** $\mathbf{U}_{C,\varphi,\psi}$ **until stabilization.**

It suffices to reach a stage where $\mathbf{U}_{C,\varphi,\psi}(Z) \subseteq Z$.

Then $\mathbf{U}_{C,\varphi,\psi}(Z) = Z$ will hold.

Algorithm for global model checking of ATL formulae

procedure $\textsc{GlobalMC}(\mathrm{ATL})(\mathcal{M}, \varphi)$

  case $\varphi = p \in \mathrm{Prop}$ **: return** $\{s \in \mathsf{States} \mid p \in L(s)\}$

  case $\varphi = \neg\psi$ **: return** $S \setminus \|\psi\|_{\mathcal{M}}$

  case $\varphi = \psi_1 \vee \psi_2$ **: return** $\|\psi_1\|_{\mathcal{M}} \cup \|\psi_2\|_{\mathcal{M}}$

  case $\varphi = \langle\langle A\rangle\rangle \mathcal{X}\psi$ **: return** $\mathrm{Pre}(\mathcal{M}, A, \|\psi\|_{\mathcal{M}})$

  case $\varphi = \langle\langle A\rangle\rangle \mathcal{G}\psi$**:** $\rho \leftarrow \mathsf{States}$; $\tau \leftarrow \|\psi\|_{\mathcal{M}}$;

  while $\rho \not\subseteq \tau$ **do**

    $\rho \leftarrow \tau$; $\tau \leftarrow \mathrm{Pre}(\mathcal{M}, A, \rho) \cap \|\psi\|_{\mathcal{M}}$

  end while; **return** ρ

  end case

  case $\varphi = \langle\langle A\rangle\rangle \psi_1 \,\mathcal{U}\, \psi_2$**:** $\rho \leftarrow \emptyset$; $\tau \leftarrow \|\psi_2\|_{\mathcal{M}}$;

  while $\tau \not\subseteq \rho$ **do**

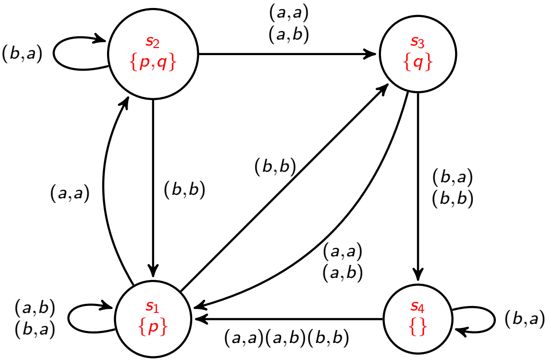
    $\rho \leftarrow \tau$; $\tau \leftarrow \|\psi_2\|_{\mathcal{M}} \cup (\mathrm{Pre}(\mathcal{M}, A, \rho) \cap \|\psi_1\|_{\mathcal{M}})$

  end while; **return** ρ

  end case

end procedure

Global model checking of ATL formulae: exercises



$\|\langle\langle 1\rangle\rangle \mathcal{G} p\|_{\mathcal{M}} = \{s_1, s_2\}$ $\|\langle\langle 2\rangle\rangle \mathcal{G} p\|_{\mathcal{M}} = \emptyset$

$\|\langle\langle \emptyset\rangle\rangle (\neg q \mathsf{U} p)\|_{\mathcal{M}} = \{s_1, s_2\}$ $\|\langle\langle 2\rangle\rangle (\neg q \mathsf{U} p)\|_{\mathcal{M}} = \{s_1, s_2, s_4\}$

$\|\langle\langle 1\rangle\rangle \mathcal{G} \langle\langle 2\rangle\rangle (\neg q \mathsf{U} p)\|_{\mathcal{M}} = \{s_1, s_2, s_4\}$

Lecture 1: concluding remarks

- Concurrent game models and the logic ATL provides a general framework for modelling, specification, formal verification, and synthesis strategies and of entire multi-agent systems.
- Various potential applications, to distributed computing, concurrency, networks, robotic systems, AI, etc.
- Many variations and extensions,
  and many challenges, conceptual and technical.
- Great potential for new research and contributions.

END OF LECTURE 1