ATL, Game Theory and Argumentation

Jürgen Dix
(joint work with N. Bulling, C. Chesnevar, W. Jamroga)

Department of Computer Science
Clausthal University of Technology

23rd April, Luxembourg

ATL: What Agents Can Achieve

- ATL: *Agent Temporal Logic* [Alur et al. 1997]
- Extension of CTL: **Temporal logic meets game theory**

ATL: What Agents Can Achieve

- ATL: *Agent Temporal Logic* [Alur et al. 1997]
- Extension of CTL: **Temporal logic meets game theory**
- Main idea: usual **temporal operators:** $\bigcirc, \Box, \mathcal{U}$ plus *cooperation modalities* $\langle\!\langle A \rangle\!\rangle$

$\langle\!\langle A \rangle\!\rangle \Phi$
stands for
coalition A has a *collective strategy*
to *enforce* Φ

ATL: What Agents Can Achieve

- ATL: *Agent Temporal Logic* [Alur et al. 1997]
- Extension of CTL: **Temporal logic meets game theory**
- Main idea: usual **temporal operators:** $\bigcirc, \Box, \mathcal{U}$ plus *cooperation modalities* $\langle\langle A \rangle\rangle$

$$\langle\langle A \rangle\rangle \Phi$$

stands for

coalition A **has a** *collective strategy* **to** *enforce* Φ

- What exactly is a *strategy*?

Vanilla ATL

- Full ATL (denoted by ATL^*) is too complex:
  - Strategies **with memory**: **2EXPTIME**
  - Strategies **without memory**: **PSPACE**

- "Vanilla" ATL: temporal operators are preceded by exactly one cooperation modality: $\langle\langle A\rangle\rangle \bigcirc \Phi$, $\langle\langle B\rangle\rangle \Box \Phi$, $\langle\langle B\rangle\rangle\, \mathcal{U}\, \Phi$.

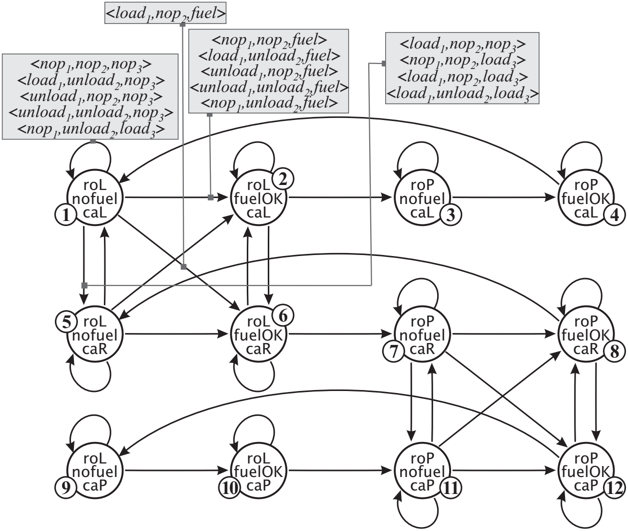
- Vanilla ATL suffices for most purposes.

Example: Rocket and Cargo

- Rocket: moves between London (roL) and Paris (roP),
- Cargo: in London (caL), in Paris (caP), or in rocket (caR).
- Rocket can be moved only if it has its fuel tank full ($fuelOK$),
- When it moves, it consumes fuel, and $nofuel$ holds after each flight.
- 3 agents,
- x can **load** the cargo, **unload** it, and **move** the rocket,
- y can **unload** the cargo and **move** the rocket,
- z can **load** the cargo and tank the rocket (action **fuel**),
- Each agent can also decide to do nothing (**nop**: "no-operation");

Overview

Rocket Example: Adding Agents and Actions



Models: Concurrent Game Structures (CGS)

- $M = \langle \mathbb{A}\mathrm{gt}, Q, \Pi, \pi, Act, d, \mathbf{o} \rangle$,
- $\mathbb{A}\mathrm{gt}$: a finite set of all agents, $|\mathbb{A}\mathrm{gt}| = k$,
- Q: a set of states, $|Q| = n$,
- Π: a set of atomic propositions,
- $\pi : Q \to \mathcal{P}(\Pi)$: a valuation of propositions,
- Act: a finite set of (atomic) actions,
- $d : \mathbb{A}\mathrm{gt} \times Q \to \mathcal{P}(Act)$ defines actions available to an agent,
- $\mathbf{o}$: a det. transition function that assigns outcome states $q' = \mathbf{o}(q, \alpha_1, \ldots, \alpha_k)$ to states and tuples of actions.

What is the size of a CGS?

- # transitions= m vs. # states = n.

Models: Concurrent Game Structures (CGS)

- $M = \langle \mathbb{A}\text{gt}, Q, \Pi, \pi, Act, d, \mathbf{o} \rangle$,
- $\mathbb{A}\text{gt}$: a finite set of all agents, $|\mathbb{A}\text{gt}| = k$,
- Q: a set of states, $|Q| = n$,
- Π: a set of atomic propositions,
- $\pi : Q \to \mathcal{P}(\Pi)$: a valuation of propositions,
- Act: a finite set of (atomic) actions,
- $d : \mathbb{A}\text{gt} \times Q \to \mathcal{P}(Act)$ defines actions available to an agent,
- $\mathbf{o}$: a det. transition function that assigns outcome states $q' = \mathbf{o}(q, \alpha_1, \ldots, \alpha_k)$ to states and tuples of actions.

What is the size of a CGS?

- # transitions= m vs. # states = n.

Strategies and Paths

- A strategy is a **conditional plan**
  - $s_a : Q \rightarrow Act$ (imperfect recall: ATL$_{Ir}$)
  - $s_a : Q^* \rightarrow Act$ (perfect recall: ATL$_{IR}$)

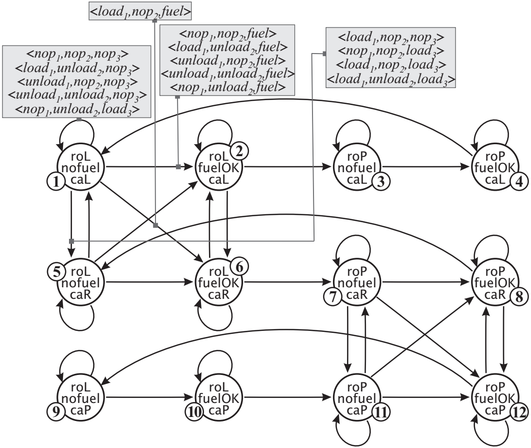
For vanilla ATL: ATL$_{Ir}$ = ATL$_{IR}$.
For full ATL: ATL$_{Ir} \neq$ ATL$_{IR}$.

Strategies and Paths

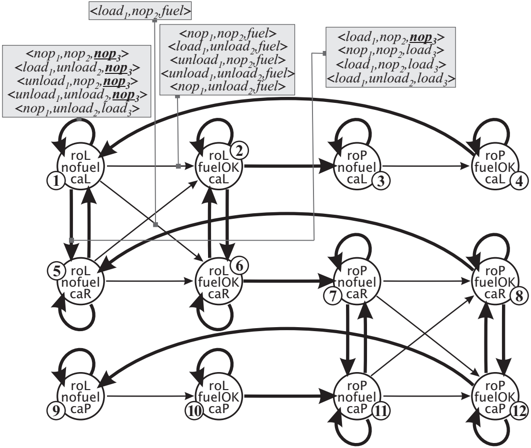
- A **strategy** is a **conditional plan**
  - $s_a : Q \to Act$ (imperfect recall: ATL$_{Ir}$)
  - $s_a : Q^* \to Act$ (perfect recall: ATL$_{IR}$)

  For vanilla ATL: ATL$_{Ir}$ = ATL$_{IR}$.
  For full ATL: ATL$_{Ir} \neq$ ATL$_{IR}$.
- A **path** is an infinite sequence of states that can be affected by subsequent transitions.
- Paths refer to **possible courses of action**.
- $S_A = \langle s_{a_1}, s_{a_2}, \ldots, s_{a_r} \rangle$ is a **collective strategy** for $A = \{a_1, a_2, \ldots, a_r\}$.
- Function $out(q, S_A)$ returns the set of all paths that may result from agents A executing strategy S_A from state q onward.

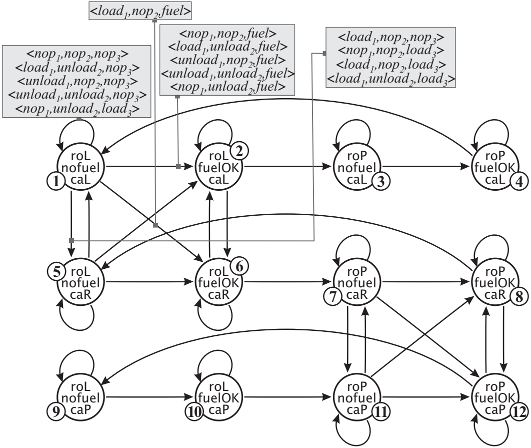
Rocket Agents



$$nofuel \rightarrow \langle\langle 3 \rangle\rangle \Box nofuel$$



$$nofuel \rightarrow \langle\langle 3 \rangle\rangle \Box nofuel$$



$$caL \rightarrow \langle\langle 1,3 \rangle\rangle \Diamond caP$$

$$caL \rightarrow \neg\langle\langle 1 \rangle\rangle \Diamond caP$$

Overview

Model Checking ATL$_I$: wrt m=# of transitions.

- Model checking:
  Does φ hold in model M (CGS) and state q?
- Nice results: model checking ATL is tractable!
- **Perfect** = **imperfect recall**: ATL$_{Ir}$ = ATL$_{IR}$.

Model Checking ATL$_I$: wrt m=# of transitions.

- Model checking:
  Does φ hold in model M (CGS) and state q?
- Nice results: model checking ATL is tractable!
- **Perfect** = **imperfect recall**: ATL$_{Ir}$ = ATL$_{IR}$.

Theorem (Alur, Kupferman & Henzinger 1998)

ATL$_{IR}$ (resp. ATL$_{Ir}$) model checking is P-complete, and can be done in time **linear in the size of the model and the length of the formula**.

Complexity wrt n

Model Checking ATL (wrt n= # states)

- m: transitions, n: **states**,
  d: actions, k: agents.

  How does m depend on n and k?

Model Checking ATL (wrt n= # states)

- m: transitions, n: **states**,
  d: actions, k: agents.

  How does m depend on n and k?

  $m = O(nd^k)$

Model Checking ATL (wrt n= # states)

- m: transitions, n: **states**,
  d: actions, k: agents.

  How does m depend on n and k?

  $m = O(nd^k)$
- m is not polynomially bounded in n when agents are present.

Model Checking ATL (wrt n= # states)

- m: transitions, n: **states**,
  d: actions, k: agents.
  How does m depend on n and k?
  $m = O(nd^k)$
- m is not polynomially bounded in n when agents are present.
- Agents make models explode!

Model Checking ATL (wrt n= # states)

- m: transitions, n: **states**,
  d: actions, k: agents.

  How does m depend on n and k?

  $m = O(nd^k)$
- m is not polynomially bounded in n when agents are present.
- Agents make models explode!
- Do agents make model checking explode?

First Result

- $\Sigma_i^P = \mathbf{NP}^{\Sigma_{\mathbf{i-1}}^{\mathbf{P}}}$: problems solvable in pol. time by a **non-deterministic** TM making queries to a $\mathbf{\Sigma_{i-1}^P}$ oracle
- $\Delta_i^P = \mathbf{P}^{\Sigma_{\mathbf{i-1}}^{\mathbf{P}}}$: problems solvable in pol. time by a **deterministic** TM making adaptive queries to a $\mathbf{\Sigma_{i-1}^P}$ oracle
- $\Sigma_2^P = \mathbf{NP}^{\mathbf{NP}}$
- $\Delta_3^P = \mathbf{P}^{[\mathbf{NP}^{\mathbf{NP}}]}$

First Result

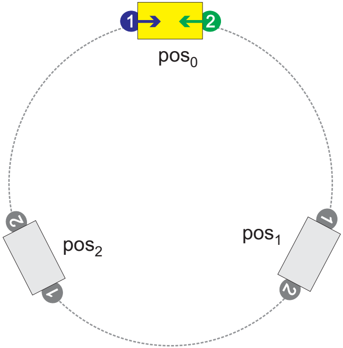
- Σ_i^P= $\mathbf{NP}^{\Sigma^{\mathbf{P}}_{\mathbf{i-1}}}$: problems solvable in pol. time by a **non-deterministic** TM making queries to a $\Sigma^{\mathbf{P}}_{\mathbf{i-1}}$ oracle
- Δ_i^P= $\mathbf{P}^{\Sigma^{\mathbf{P}}_{\mathbf{i-1}}}$: problems solvable in pol. time by a **deterministic** TM making adaptive queries to a $\Sigma^{\mathbf{P}}_{\mathbf{i-1}}$ oracle
- Σ_2^P = $\mathbf{NP}^{\mathbf{NP}}$
- Δ_3^P = $\mathbf{P}^{[\mathbf{NP}^{\mathbf{NP}}]}$

Proposition

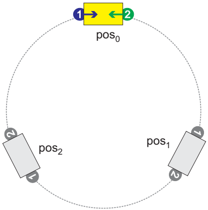
Model checking ATL_{IR} is Δ_3^P-complete wrt the number of states (n), decisions (d) and agents (k) in the model, and the length of the formula (l).
For positive ATL_{IR}, model checking is $\Sigma^{\mathbf{P}}_{\mathbf{2}}$-complete.

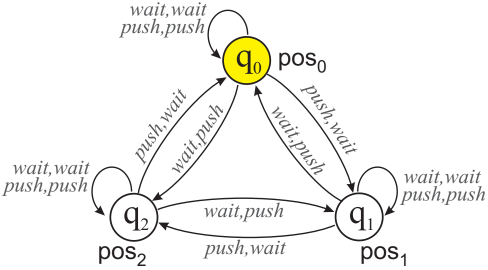
Overview

Example: Robots and Carriage

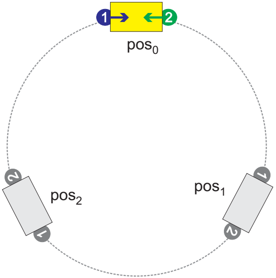


The CGS model

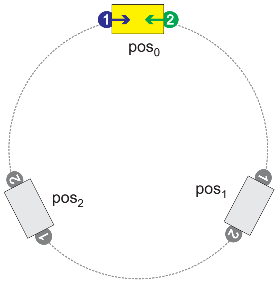


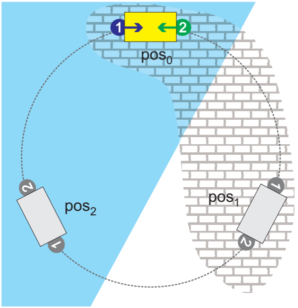


ATL with perfect Information

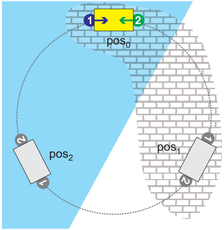


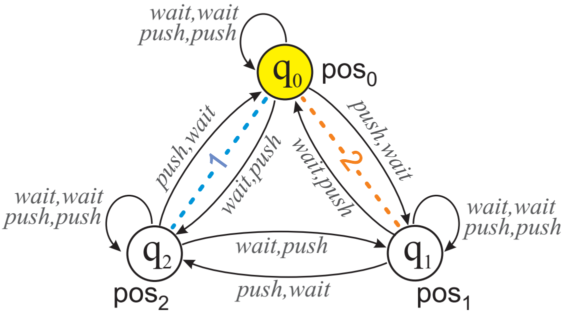
ATL with *imperfect* Information





ATL$_{ir}$: ATL with Imperfect Information





From CGS to CEGS

Memoryless strategies:

- We extend CGS by epistemic relations $\sim_a$, one per agent: we obtain CEGS.
- Uniform strategies per agent: $q \sim_a q' \Rightarrow s_a(q) = s_a(q')$
- Uniform strategies for group of agents:
  $q \sim_A q' \Rightarrow s_a(q) = s_a(q')$, where $q \sim_A q'$ is defined by
  there is an agent $a \in A$ with $q \sim_a q'$

From CGS to CEGS

Memoryless strategies:

- We extend CGS by epistemic relations $\sim_a$, one per agent: we obtain CEGS.
- Uniform strategies per agent: $q \sim_a q' \Rightarrow s_a(q) = s_a(q')$
- Uniform strategies for group of agents:
  $q \sim_A q' \Rightarrow s_a(q) = s_a(q')$, where $q \sim_A q'$ is defined by
  there is an agent $a \in A$ with $q \sim_a q'$

Strategies with memory

For $Q^* \rightarrow Act$, definitions above are appropriately modified.

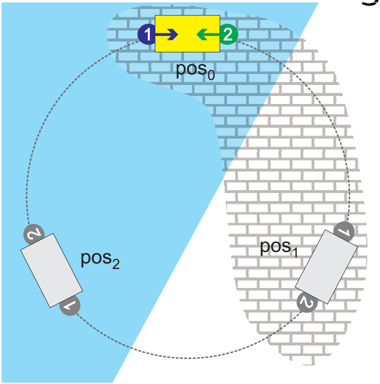
ATL$_{ir}$ and ATL$_{iR}$

- $M, q \models \langle\langle A\rangle\rangle_{ir} \bigcirc \varphi$ iff there exists uniform S_A such that, for every path $\Lambda \in \bigcup_{q' \sim_A q} out(q', S_A)$, we have $M, \Lambda[1] \models \varphi$;
- $M, q \models \langle\langle A\rangle\rangle_{ir} \Box \varphi$ iff there exists uniform S_A such that, for every $\Lambda \in \bigcup_{q' \sim_A q} out(q', S_A)$, we have $M, \Lambda[i] \models \varphi$ for every $i \geq 0$;
- $M, q \models \langle\langle A\rangle\rangle_{ir} \varphi \mathcal{U} \psi$ iff there exists uniform S_A such that, for every $\Lambda \in \bigcup_{q' \sim_A q} out(q', S_A)$, we have $M, \Lambda[i] \models \psi$ for some $i \geq 0$, and $M, \Lambda[j] \models \varphi$ for every $0 \leq j < i$.

What about strategies with memory (ATL$_{iR}$)?

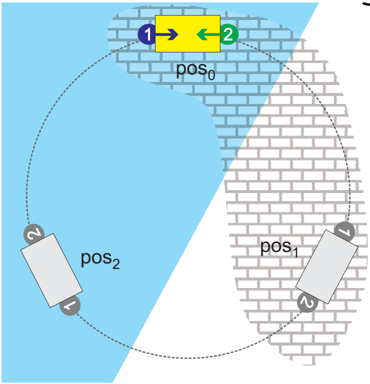
Instead of equivalences $q' \sim_A q$, one has to consider sequences $\overline{q'} \sim_A \overline{q}$.

Example: Robots and Carriage



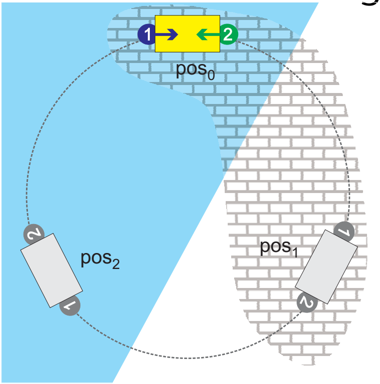
$\langle\langle 1 \rangle\rangle_{ir} \Box \neg pos_1$ **?**

Example: Robots and Carriage



$$\neg\langle\langle 1\rangle\rangle_{ir}\Box\neg pos_1$$

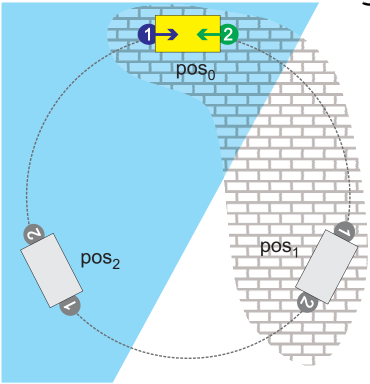
Example: Robots and Carriage



$\neg\langle\langle 1\rangle\rangle_{ir}\Box\neg pos_1$

$\langle\langle 2\rangle\rangle_{ir}\Box\neg pos_1$ **?**

Example: Robots and Carriage



$\neg\langle\langle 1\rangle\rangle_{ir}\Box\neg pos_1$

$\neg\langle\langle 2\rangle\rangle_{ir}\Box\neg pos_1$

Why not $\langle\langle 2\rangle\rangle_{ir}\Box\neg pos_1$ by using the following strategy for agent 2: “push” when in $\overline{q_0}$ and “wait” when in $\overline{q_2}$?

This not a feasible strategy, because it does not work in q_1.

Overview

Complexity Results for Strategic Logics

	m, l	n, k, l	
CTL	P-complete [1]	P-complete [1]	
ATL$_{Ir}$	P-complete [3]		
ATL$_{ir}$			

Complexity Results for Strategic Logics

	m, l	n, k, l	
CTL	P-complete [1]	P-complete [1]	
ATL$_{Ir}$	P-complete [3]		
ATL$_{ir}$			

[1] Clarke, Emerson & Sistla (1986). *Automatic verification of finite-state concurrent systems ....* **ACM Prog. Lang. Syst**

[3] Alur, Henzinger & Kupferman (2002). *Alternating-time Temporal Logic*. **J. ACM**

Complexity Results for Strategic Logics

	m, l	n, k, l	
CTL	P-complete [1]	P-complete [1]	
ATL$_{Ir}$	P-complete [3]	Δ_3^P**-complete** [5,6]	
ATL$_{ir}$			

[1] Clarke, Emerson & Sistla (1986). *Automatic verification of finite-state concurrent systems ....* **ACM Prog. Lang. Syst**

[3] Alur, Henzinger & Kupferman (2002). *Alternating-time Temporal Logic.* **J. ACM**

[5] **Jamroga & Dix (CEEMAS 2005).** ***Do agents make model checking explode?***
[6] **Laroussinie, Markey & Oreiby (FORMATS 2006).** ***Model-Checking Timed.***

Complexity Results for Strategic Logics

	m, l	n, k, l	n_{local}, k, l
CTL	P-complete [1]	P-complete [1]	
ATL$_{Ir}$	P-complete [3]	Δ_3^P-**complete** [5,6]	
ATL$_{ir}$	Δ_2^P-**complete** [4,7]	Δ_3^P-**complete** [7]	

[1] Clarke, Emerson & Sistla (1986). *Automatic verification of finite-state concurrent systems ....* **ACM Prog. Lang. Syst**

[3] Alur, Henzinger & Kupferman (2002). *Alternating-time Temporal Logic.* **J. ACM**

[4] Schobbens (2004). *Alternating-time logic with imperfect recall.* **ENTCS**

[5] **Jamroga & Dix (CEEMAS 2005).** ***Do agents make model checking explode?***

[6] **Laroussinie, Markey & Oreiby (FORMATS 2006).** ***Model-Checking Timed.***

[7] **Jamroga & Dix (2008).** ***Model checking abilities of agents.*** **Theory of Computing systems**

Complexity Results for Strategic Logics

	m, l	n, k, l	n_{local}, k, l
CTL	P-complete [1]	P-complete [1]	
ATL_{Ir}	P-complete [3]	**Δ_3^P-complete** [5,6]	
ATL_{ir}	**Δ_2^P-complete** [4,7]	**Δ_3^P-complete** [7]	

[1] Clarke, Emerson & Sistla (1986). *Automatic verification of finite-state concurrent systems ...*. **ACM Prog. Lang. Syst**

[3] Alur, Henzinger & Kupferman (2002). *Alternating-time Temporal Logic*. **J. ACM**

[4] Schobbens (2004). *Alternating-time logic with imperfect recall*. **ENTCS**

[5] **Jamroga & Dix (CEEMAS 2005).** ***Do agents make model checking explode?***

[6] **Laroussinie, Markey & Oreiby (FORMATS 2006).** ***Model-Checking Timed.***

[7] **Jamroga & Dix (2008).** ***Model checking abilities of agents*. Theory of Computing systems**

Complexity Results for Strategic Logics

	m, l	n, k, l	n_{local}, k, l
CTL	P-complete [1]	P-complete [1]	PSPACE-complete [2]
ATL$_{Ir}$	P-complete [3]	Δ_3^P**-complete** [5,6]	
ATL$_{ir}$	Δ_2^P**-complete** [4,7]	Δ_3^P**-complete** [7]	

[1] Clarke, Emerson & Sistla (1986). *Automatic verification of finite-state concurrent systems ...*. **ACM Prog. Lang. Syst**

[2] Kupferman, Vardi & Wolper (2000). *An automata-theoretic approach to ...*. **J. ACM**

[3] Alur, Henzinger & Kupferman (2002). *Alternating-time Temporal Logic*. **J. ACM**

[4] Schobbens (2004). *Alternating-time logic with imperfect recall*. **ENTCS**

[5] **Jamroga & Dix (CEEMAS 2005).** ***Do agents make model checking explode?***

[6] **Laroussinie, Markey & Oreiby (FORMATS 2006).** ***Model-Checking Timed.***

[7] **Jamroga & Dix (2008).** ***Model checking abilities of agents*****. Theory of Computing systems**

Complexity Results for Strategic Logics

	m, l	n, k, l	n_{local}, k, l
CTL	P-complete [1]	P-complete [1]	PSPACE-complete [2]
ATL_{Ir}	P-complete [3]	Δ_3^P**-complete** [5,6]	**EXPTIME-complete** [8]
ATL_{ir}	Δ_2^P**-complete** [4,7]	Δ_3^P**-complete** [7]	

[1] Clarke, Emerson & Sistla (1986). *Automatic verification of finite-state concurrent systems ....* **ACM Prog. Lang. Syst**

[2] Kupferman, Vardi & Wolper (2000). *An automata-theoretic approach to ....* **J. ACM**

[3] Alur, Henzinger & Kupferman (2002). *Alternating-time Temporal Logic.* **J. ACM**

[4] Schobbens (2004). *Alternating-time logic with imperfect recall.* **ENTCS**

[5] **Jamroga & Dix (CEEMAS 2005).** ***Do agents make model checking explode?***

[6] **Laroussinie, Markey & Oreiby (FORMATS 2006).** ***Model-Checking Timed.***

[7] **Jamroga & Dix (2008).** ***Model checking abilities of agents.*** **Theory of Computing systems**

[8] **Hoek, Lomuscio & Wooldridge (AAMAS 2006).** ***On the complexity of practical ATL model checking.***

Complexity Results for Strategic Logics

	m, l	n, k, l	n_{local}, k, l
CTL	P-complete [1]	P-complete [1]	PSPACE-complete [2]
ATL$_{Ir}$	P-complete [3]	**Δ_3^P-complete** [5,6]	**EXPTIME-complete** [8]
ATL$_{ir}$	**Δ_2^P-complete** [4,7]	**Δ_3^P-complete** [7]	**PSPACE-complete** [9]

[1] Clarke, Emerson & Sistla (1986). *Automatic verification of finite-state concurrent systems ....* **ACM Prog. Lang. Syst**
[2] Kupferman, Vardi & Wolper (2000). *An automata-theoretic approach to ....* **J. ACM**
[3] Alur, Henzinger & Kupferman (2002). *Alternating-time Temporal Logic.* **J. ACM**
[4] Schobbens (2004). *Alternating-time logic with imperfect recall.* **ENTCS**
[5] **Jamroga & Dix (CEEMAS 2005).** ***Do agents make model checking explode?***
[6] **Laroussinie, Markey & Oreiby (FORMATS 2006).** ***Model-Checking Timed.***
[7] **Jamroga & Dix (2008).** ***Model checking abilities of agents.*** **Theory of Computing systems**
[8] **Hoek, Lomuscio & Wooldridge (AAMAS 2006).** ***On the complexity of practical ATL model checking.***
[9] **Jamroga & Ågotnes (AAMAS 2007).** ***Modular Interpreted Systems***

Last Column: meaning of n_{local}

MIS

A modular interpreted system (MIS) is of the form $\langle \mathbb{A}\mathrm{gt}, Act, \mathrm{In} \rangle$ where each agent a_i has the following internal structure $a_i = \langle St_i, d_i, \mathbf{out_i}, \mathbf{in_i}, o_i, \Pi_i, \pi_i \rangle$.

Set of global states is the cartesian product of the St_i: $n = n_1 \cdot \ldots \cdot n_k$.

Last Column: meaning of n_{local}

MIS

A modular interpreted system (MIS) is of the form $\langle \mathbb{A}\mathrm{gt}, Act, \mathrm{In}\rangle$ where each agent a_i has the following internal structure $a_i = \langle St_i, d_i, \mathbf{out_i}, \mathbf{in_i}, o_i, \Pi_i, \pi_i\rangle$.

Set of global states is the cartesian product of the St_i: $n = n_1 \cdot \ldots \cdot n_k$.

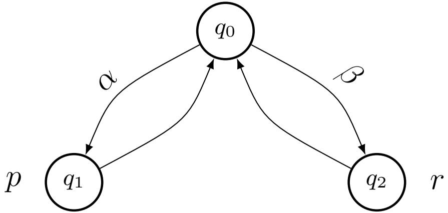
- A MIS viewed as a CGS (for ATL) is very succinct.
- For ATL$_{ir}$, we have in addition to CGS all the local epistemic relations $\sim_1, \ldots, \sim_k$ (CEGS).
- A MIS viewed as a CEGS (for ATL$_{ir}$) does not compress that much.

Overview

What about ATL_{iR}?

ATL_{iR}: Imperfect information, perfect recall.

- Undecidable: we do not yet have a formal proof!
- We have not found any result in the literature which directly implies the undecidability of ATL_{iR}.
- Why is it undecidable?



$$\mathcal{M}, q_0 \models \langle\langle 1 \rangle\rangle_{IR}(\Box\Diamond p \wedge \Box\Diamond r)$$

$$\mathcal{M}, q_0 \not\models \langle\langle 1 \rangle\rangle_{Ir}(\Box\Diamond p \wedge \Box\Diamond r)$$

References

Joint work with Wojtek Jamroga.

W.Jamroga and J.Dix
Model checking abilities of agents.
Theory of Computing Systems, 42 (3), 366–410, 2008.

W. Jamroga and J. Dix.
Do agents make model checking explode?
In *Proceedings of CEEMAS'05*, LNAI 3690, pages 398–407, 2005.

ATL + Plausibility

Agents usually act rationally!

We would like to

- extend ATL with a notion of plausibility,
- reason about **rational behavior** of agents,
- have a logic that can *express* any solution concept,
- compare **different** game theoretic solution concepts.

Plausibility concept

ATL: Reasoning about all possible behaviors.

ATLP: Reasoning about plausible behaviors.

Plausibility concept

ATL: Reasoning about all possible behaviors.

$\langle\langle A\rangle\rangle\varphi$: Agents A have *a* collective strategy to enforce φ against **any** response of their opponents.

ATLP: Reasoning about plausible behaviors.

Plausibility concept

ATL: Reasoning about all possible behaviors.

$\langle\langle A\rangle\rangle\varphi$: Agents A have *a* collective strategy to enforce φ against **any** response of their opponents.

ATLP: Reasoning about plausible behaviors.

$\mathbf{Pl}\,\langle\langle A\rangle\rangle\varphi$: Agents A have a plausible collective strategy to enforce φ against any plausible response of their opponents.

Example: Playing **undominated** strategies is often plausible,...

Overview

The Base Logic: $\mathcal{L}_{ATLP}^{\text{base}}$

Definition ($\mathcal{L}_{ATLP}^{\text{base}}$)

The language $\mathcal{L}_{ATLP}^{\text{base}}$ is defined over nonempty sets:

- Π of propositions, $p \in \Pi$,
- $\mathbb{A}\text{gt}$ of agents, $a \in \mathbb{A}\text{gt}$, $\quad A \subseteq \mathbb{A}\text{gt}$, and
- Ω of basic plausibility terms, $\omega \in \Omega$.

$$\varphi ::= \quad p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\langle A\rangle\rangle \bigcirc \varphi \mid \langle\langle A\rangle\rangle \Box \varphi \mid \langle\langle A\rangle\rangle \varphi \mathcal{U} \varphi \mid \mathbf{Pl}_{A}\varphi \mid (\text{set-pl } \omega)\varphi$$

The Base Logic: $\mathcal{L}_{ATLP}^{\text{base}}$

Definition ($\mathcal{L}_{ATLP}^{\text{base}}$)

The language $\mathcal{L}_{ATLP}^{\text{base}}$ is defined over nonempty sets:

- Π of propositions, $p \in \Pi$,
- $\mathbb{A}\mathrm{gt}$ of agents, $a \in \mathbb{A}\mathrm{gt}$, $A \subseteq \mathbb{A}\mathrm{gt}$, and
- Ω of basic plausibility terms, $\omega \in \Omega$.

$$\varphi ::= \; p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\langle A\rangle\rangle \bigcirc \varphi \mid \langle\langle A\rangle\rangle \Box \varphi \mid \langle\langle A\rangle\rangle \varphi \mathcal{U} \varphi \mid \mathbf{Pl}_A \varphi \mid (\textbf{set-pl } \omega)\varphi$$

$$\mathcal{M}, q \models \mathbf{Pl}_B \langle\langle A\rangle\rangle \gamma$$

if

A can **enforce** γ,

when agents in B play only plausible strategies

Plausibility Terms

Ω: Set of basic plausibility terms, $\omega \in \Omega$

Hard-wired sets of strategies:

- ω_{NE}: Nash equilibria
- ω_{PO} : Pareto optimal strategies

How to activate?

Plausibility Terms

Ω: Set of basic plausibility terms, $\omega \in \Omega$

Hard-wired sets of strategies:

ω_{NE}: Nash equilibria
ω_{PO} : Pareto optimal strategies

How to activate?

(**set-pl** ω) : Sets plausible strategies to $[\![\omega]\!] \subseteq \Sigma$

Plausibility Terms

Ω: Set of basic plausibility terms, $\omega \in \Omega$

Hard-wired sets of strategies:

- ω_{NE}: Nash equilibria
- ω_{PO} : Pareto optimal strategies

How to activate?

($\mathbf{set\text{-}pl}$ ω) : Sets plausible strategies to $[\![\omega]\!] \subseteq \Sigma$

And where do the terms come from?

How to describe strategies?

Plausibility terms: **abstract labels**, **no structure**!

Idea: Formulas that describe plausible strategies!

Select all s such that s is better than any other strategy s'

Complex plausibility terms:

$$\omega = \sigma. \quad \varphi(\sigma)$$

How to describe strategies?

Plausibility terms: **abstract labels, no structure**!

Idea: Formulas that describe plausible strategies!

Complex plausibility terms:

$$\omega = \sigma. \quad \varphi(\sigma)$$

How to describe strategies?

Plausibility terms: **abstract labels**, **no structure**!

Idea: Formulas that describe plausible strategies!

Complex plausibility terms:

$$\omega = \sigma. \underbrace{\varphi(\sigma)}_{\text{Property that } \sigma \text{ should fulfill}}$$

How to describe strategies?

Plausibility terms: **abstract labels, no structure**!

Idea: Formulas that describe plausible strategies!

Complex plausibility terms:

$$\omega = \sigma.\forall\sigma_1\exists\sigma_2\ldots\forall\sigma_n \qquad \underbrace{\varphi(\sigma,\sigma_1,\ldots\sigma_n)}_{\in\mathcal{L}_{ATLP}^{\text{base}}(\Omega\cup\{\sigma,\sigma_1,\ldots,\sigma_n\})}$$

How to describe strategies?

Plausibility terms: **abstract labels, no structure**!

Idea: Formulas that describe plausible strategies!

Complex plausibility terms:

$$\omega = \sigma.\forall\sigma_1\exists\sigma_2\ldots\forall\sigma_n \qquad \underbrace{\varphi(\sigma,\sigma_1,\ldots\sigma_n)}_{\in\mathcal{L}^{\mathsf{base}}_{ATLP}(\Omega\cup\{\sigma,\sigma_1,\ldots,\sigma_n\})}$$

Example: $\omega_{\mathsf{DOM}} = \sigma.\forall\sigma' \quad (\sigma$ better than $\sigma')$

How to describe strategies?

Plausibility terms: **abstract labels, no structure**!

Idea: Formulas that describe plausible strategies!

Complex plausibility terms:

$$\omega = \sigma.\forall\sigma_1\exists\sigma_2\ldots\forall\sigma_n \qquad \underbrace{\varphi(\sigma,\sigma_1,\ldots\sigma_n)}_{\in\mathcal{L}_{ATLP}^{\text{base}}(\Omega\cup\{\sigma,\sigma_1,\ldots,\sigma_n\})}$$

Example: $\omega_{\text{DOM}} = \sigma.\forall\sigma' \quad (\sigma$ better than $\sigma')$

How to determine whether a strategy is good?

Overview

Some Game Theory

NF games: **Normal Form**: Players move **simultanously**.

EF games: **Extensive Form**: Alternate moves. Moves can depend on the whole history. This is a tree.

Some Game Theory

NF games: **Normal Form**: Players move **simultanously**.

EF games: **Extensive Form**: Alternate moves. Moves can depend on the whole history. This is a tree.

EF $\rightsquigarrow$ NF: One can easily transform a EF game into a NF game.

EF $\rightsquigarrow$ CGS: Each EF game can be modelled as a CGS.

CGS $\not\rightsquigarrow$ EF: CGS can have cycles or simultaneous moves.

Some Game Theory

NF games: **Normal Form**: Players move **simultanously**.

EF games: **Extensive Form**: Alternate moves. Moves can depend on the whole history. This is a tree.

EF $\rightsquigarrow$ NF: One can easily transform a EF game into a NF game.

EF $\rightsquigarrow$ CGS: Each EF game can be modelled as a CGS.

CGS $\not\rightsquigarrow$ EF: CGS can have cycles or simultaneous moves.

We want to define *CGSP* that correspond to NF games.

Concurrent game structures with plausibility

$\mathcal{M} = (\mathbb{A}\mathrm{gt}, Q, \Pi, \pi, Act, d, \delta, \Upsilon, \Omega, [\![\cdot]\!])$

- $\Upsilon \subseteq \Sigma$: set of (plausible) strategy profiles

  Example: $\Upsilon = \{(head, head)\}$

- $\Omega = \{\omega_1, \omega_2, \dots\}$: set of plausibility terms

  Example: ω_{NE} stands for all Nash equilibria

- $[\![\cdot]\!] : Q \to (\Omega \to \mathcal{P}(\Sigma))$: plausibility mapping, it assigns a set of strategy profiles to each state and plausibility term

  Example: $[\![\omega_{NE}]\!]_q = \{(head, head), (tail, tail)\}$

Concurrent game structures with plausibility

$\mathcal{M} = (\mathbb{A}\mathrm{gt}, Q, \Pi, \pi, Act, d, \delta, \Upsilon, \Omega, [\![\cdot]\!])$

- $\Upsilon \subseteq \Sigma$: set of (plausible) strategy profiles

  Example: $\Upsilon = \{(head, head)\}$

- $\Omega = \{\omega_1, \omega_2, \dots\}$: set of plausibility terms

  Example: ω_{NE} stands for all Nash equilibria

- $[\![\cdot]\!] : Q \to (\Omega \to \mathcal{P}(\Sigma))$: plausibility mapping, it assigns a set of strategy profiles to each state and plausibility term

  Example: $[\![\omega_{NE}]\!]_q = \{(head, head), (tail, tail)\}$

Concurrent game structures with plausibility

$\mathcal{M} = (\mathbb{A}\mathrm{gt}, Q, \Pi, \pi, Act, d, \delta, \Upsilon, \Omega, [\![\cdot]\!])$

- $\Upsilon \subseteq \Sigma$: set of (plausible) strategy profiles

  Example: $\Upsilon = \{(head, head)\}$

- $\Omega = \{\omega_1, \omega_2, \dots\}$: set of plausibility terms

  Example: ω_{NE} stands for all Nash equilibria

- $[\![\cdot]\!] : Q \to (\Omega \to \mathcal{P}(\Sigma))$: plausibility mapping, it assigns a set of strategy profiles to each state and plausibility term

  Example: $[\![\omega_{NE}]\!]_q = \{(head, head), (tail, tail)\}$

Concurrent game structures with plausibility

$\mathcal{M} = (\mathbb{A}\mathrm{gt}, Q, \Pi, \pi, Act, d, \delta, \Upsilon, \Omega, [\![\cdot]\!])$

- $\Upsilon \subseteq \Sigma$: set of (plausible) strategy profiles

  Example: $\Upsilon = \{(head, head)\}$

- $\Omega = \{\omega_1, \omega_2, \dots\}$: set of plausibility terms

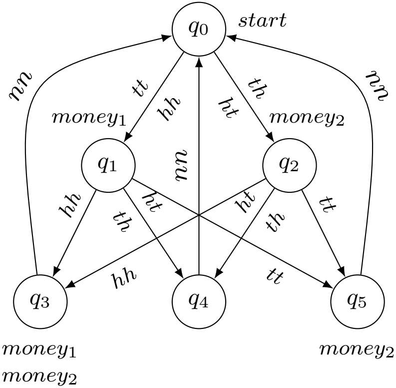
  Example: ω_{NE} stands for all Nash equilibria

- $[\![\cdot]\!] : Q \to (\Omega \to \mathcal{P}(\Sigma))$: plausibility mapping, it assigns a set of strategy profiles to each state and plausibility term

  Example: $[\![\omega_{NE}]\!]_q = \{(head, head), (tail, tail)\}$

General Solution Concepts: CGSP

Idea: Agents have **preferences**: $\vec{\eta} = \langle \eta_1, \ldots, \eta_k \rangle$
η_i: ATL path formulæ (**payoff**) Example: $\eta_2 = \Diamond money_2$

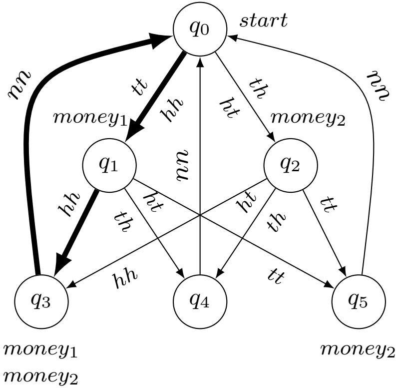


No payoffs needed as for classical solution concepts!

General Solution Concepts: CGSP

Idea: Agents have **preferences**: $\vec{\eta} = \langle \eta_1, \ldots, \eta_k \rangle$

η_i: ATL path formulæ (**payoff**) Example: $\eta_2 = \Diamond money_2$



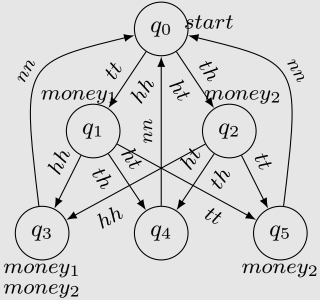
No payoffs needed as for classical solution concepts!

General Solution Concepts: CGSP

Idea: Agents have **preferences**: $\vec{\eta} = \langle \eta_1, \ldots, \eta_k \rangle$
η_i: ATL path formulæ (**payoff**) Example: $\eta_2 = \Diamond money_2$

CGSP +preferences $\rightsquigarrow$ normal form game

Each **CGSP** M **with** $\vec{\eta}$ corresponds to a **normal form game** $\mathcal{S}$.



$\rightsquigarrow$

$\eta_1 \backslash \eta_2$	s_{hh}	s_{ht}	s_{th}	s_{tt}
s_{hh}	**1, 1**	0, 0	0, 1	0, 1
s_{ht}	0, 0	0, 1	0, 1	0, 1
s_{th}	0, 1	0, 1	**1, 1**	0, 0
s_{tt}	0, 1	0, 1	0, 0	0, 1

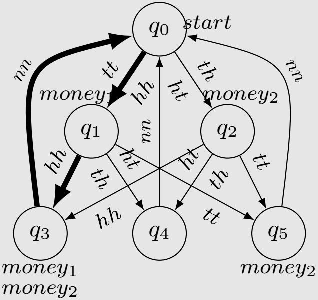
General Solution Concepts: CGSP

Idea: Agents have **preferences**: $\vec{\eta} = \langle \eta_1, \dots, \eta_k \rangle$

η_i: ATL path formulæ (**payoff**) Example: $\eta_2 = \Diamond money_2$

CGSP +preferences $\rightsquigarrow$ normal form game

Each **CGSP M with $\vec{\eta}$** corresponds to a **normal form game $\mathcal{S}$**.



$\rightsquigarrow$

$\eta_1 \backslash \eta_2$	s_{hh}	s_{ht}	s_{th}	s_{tt}
s_{hh}	**1, 1**	0, 0	0, 1	0, 1
s_{ht}	0, 0	0, 1	0, 1	0, 1
s_{th}	0, 1	0, 1	**1, 1**	0, 0
s_{tt}	0, 1	0, 1	0, 0	0, 1

Characterizing Solution Concepts

$NE^{\vec{\eta}}(\sigma)$: $\bigwedge_{a\in\mathbb{A}\mathrm{gt}} BR_a^{\vec{\eta}}(\sigma)$

Characterizing Solution Concepts

$BR_a^{\vec{\eta}}(\sigma)$: $(\textbf{set-pl}\ \sigma[\mathbb{A}\mathrm{gt}\setminus\{a\}])\mathbf{Pl}_{\mathbb{A}\mathrm{gt}}\big(\langle\langle a\rangle\rangle\eta_a \rightarrow (\textbf{set-pl}\ \sigma)\langle\langle\emptyset\rangle\rangle\eta_a\big)$

$NE^{\vec{\eta}}(\sigma)$: $\bigwedge_{a\in\mathbb{A}\mathrm{gt}} BR_a^{\vec{\eta}}(\sigma)$

Characterizing Solution Concepts

$BR_a^{\vec{\eta}}(\sigma)$: $(\textbf{set-pl}\ \sigma[\mathbb{A}\mathrm{gt}\setminus\{a\}])\mathbf{Pl}_{\mathbb{A}\mathrm{gt}}\big(\langle\langle a\rangle\rangle\eta_a \rightarrow (\textbf{set-pl}\ \sigma)\langle\langle\emptyset\rangle\rangle\eta_a\big)$

$NE^{\vec{\eta}}(\sigma)$: $\bigwedge_{a\in\mathbb{A}\mathrm{gt}} BR_a^{\vec{\eta}}(\sigma)$

$SPN^{\vec{\eta}}(\sigma)$: $\langle\langle\emptyset\rangle\rangle\Box NE^{\vec{\eta}}(\sigma)$

Characterizing Solution Concepts

$$BR_a^{\vec{\eta}}(\sigma): \quad (\textbf{set-pl } \sigma[\mathbb{A}\mathrm{gt}\setminus\{a\}])\mathbf{Pl}_{\mathbb{A}\mathrm{gt}}\big(\langle\!\langle a\rangle\!\rangle \eta_a \to (\textbf{set-pl } \sigma)\langle\!\langle \emptyset\rangle\!\rangle \eta_a\big)$$

$$NE^{\vec{\eta}}(\sigma): \quad \bigwedge_{a\in\mathbb{A}\mathrm{gt}} BR_a^{\vec{\eta}}(\sigma)$$

$$SPN^{\vec{\eta}}(\sigma): \quad \langle\!\langle \emptyset\rangle\!\rangle \Box NE^{\vec{\eta}}(\sigma)$$

$$PO^{\vec{\eta}}(\sigma): \quad \forall\sigma'\ \mathbf{Pl}_{\mathbb{A}\mathrm{gt}}\Big(\bigwedge_{a\in\mathbb{A}\mathrm{gt}}((\textbf{set-pl } \sigma')\langle\!\langle \emptyset\rangle\!\rangle \eta_a \to (\textbf{set-pl } \sigma)\langle\!\langle \emptyset\rangle\!\rangle \eta_a)\vee$$
$$\bigvee_{a\in\mathbb{A}\mathrm{gt}}((\textbf{set-pl } \sigma)\langle\!\langle \emptyset\rangle\!\rangle \eta_a \wedge \neg(\textbf{set-pl } \sigma')\langle\!\langle \emptyset\rangle\!\rangle \eta_a\Big).$$

Characterizing Solution Concepts

$$BR_a^{\vec{\eta}}(\sigma): \quad (\textbf{set-pl}\ \sigma[\mathbb{A}\mathrm{gt} \setminus \{a\}])\mathbf{Pl}_{\mathbb{A}\mathrm{gt}}\big(\langle\langle a\rangle\rangle \eta_a \to (\textbf{set-pl}\ \sigma)\langle\langle \emptyset\rangle\rangle \eta_a\big)$$

$$NE^{\vec{\eta}}(\sigma): \quad \bigwedge_{a\in\mathbb{A}\mathrm{gt}} BR_a^{\vec{\eta}}(\sigma)$$

$$SPN^{\vec{\eta}}(\sigma): \quad \langle\langle \emptyset\rangle\rangle \Box NE^{\vec{\eta}}(\sigma)$$

$$PO^{\vec{\eta}}(\sigma): \quad \forall\sigma'\ \mathbf{Pl}_{\mathbb{A}\mathrm{gt}}\Big(\bigwedge_{a\in\mathbb{A}\mathrm{gt}}((\textbf{set-pl}\ \sigma')\langle\langle \emptyset\rangle\rangle \eta_a \to (\textbf{set-pl}\ \sigma)\langle\langle \emptyset\rangle\rangle \eta_a) \vee \bigvee_{a\in\mathbb{A}\mathrm{gt}}((\textbf{set-pl}\ \sigma)\langle\langle \emptyset\rangle\rangle \eta_a \wedge \neg(\textbf{set-pl}\ \sigma')\langle\langle \emptyset\rangle\rangle \eta_a\Big).$$

Theorem

These notions correspond to those in game theory:
$[\![\sigma.NE^{\vec{\eta}}(\sigma)]\!]_M^q$ = NE strategies in $\mathcal{S}(M,\vec{\eta},q)$.

Similarly for SPN and PO and UNDOM.

Example

Both agents play **without restrictions:**

$$\mathcal{M}, q_0 \models \neg\langle\langle a_2 \rangle\rangle \Diamond money_2$$

Both agents play a **Nash equilibrium** strategy:

$$M, q_0 \models (\textbf{set-pl}\ \sigma.\mathrm{NE}^{\eta}(\sigma))\mathbf{Pl}_{\mathbb{A}\mathrm{gt}}\langle\langle a_2 \rangle\rangle \Diamond money_2$$

The Full Language: $\mathcal{L}_{ATLP}$

Plausibility terms:

$$\sigma.\forall\sigma_1\exists\sigma_2\ldots\forall\sigma_n \quad \varphi$$

where

$$\varphi \in \mathcal{L}_{ATLP}^{\text{base}}$$

What about nesting (**set-pl** $\cdot$) operators?

(**set-pl** ... (**set-pl** ... (**set-pl** ...) ...) ...)

The Full Language: $\mathcal{L}_{ATLP}$

Plausibility terms:

$$\sigma.\forall\sigma_1\exists\sigma_2\ldots\forall\sigma_n \quad \varphi$$

where

$$\varphi \in \mathcal{L}_{ATLP}^{\text{base}}$$

What about nesting (**set-pl** $\cdot$) operators?

$$(\textbf{set-pl} \ldots (\textbf{set-pl} \ldots (\textbf{set-pl} \ldots) \ldots) \ldots)$$

We get a hierarchy of logics:

$\mathcal{L}_{ATLP}^{k}$: k nestings

$\mathcal{L}_{ATLP} := \lim_{k\to\infty} \mathcal{L}_{ATLP}^{k}$

Overview

Model Checking Complexity

Theorem

Model checking $\mathcal{L}_{ATLP}^{base}$ is $\Delta_3^{\mathbf{P}}$-complete.

This is in the line with game theoretical results!

Model Checking Complexity

Expressivity vs. Complexity

	0	1	2	...	i	...	unbounded
$\mathcal{L}^1_{ATLP}$	$\mathbf{\Delta^P_3}$	$\mathbf{\Delta^P_4}$	$\mathbf{\Delta^P_5}$	...	$\mathbf{\Delta^P_{i+3}}$	...	**PSPACE**
$\mathcal{L}^2_{ATLP}$	$\mathbf{\Delta^P_4}$	$\mathbf{\Delta^P_6}$	$\mathbf{\Delta^P_7}$	...	$\mathbf{\Delta^P_{5+i-\max\{0,1-i\}}}$	...	**PSPACE**
$\vdots$						...	$\vdots$
$\mathcal{L}^k_{ATLP}$ $i > k+1$	$\mathbf{\Delta^P_{k+2}}$	$\mathbf{\Delta^P_{k+4}}$	$\mathbf{\Delta^P_{k+6}}$	...	$\mathbf{\Delta^P_{i+2k+1-\max\{0,k-i-1\}}}$	...	**PSPACE**

For particular well-behaved models, model checking is even polynomial.

References

Joint work with Wojtek Jamroga and Nils Bulling

- N. Bulling and W.Jamroga and J.Dix
  Reasoning about Temporal Properties of Rational Play
  Annals of Mathematics and AI, 53 (1), 2009.
- W. Jamroga and N. Bulling.
  A framework for reasoning about rational agents.
  In *Proceedings of AAMAS'07*, pages 592–594, 2007.

ATL + Coalition Formation

Motivation

ATL: $\langle\langle A\rangle\rangle\gamma$

Group A of agents can **enforce** property γ.

Where does A come from?

Is it reasonable to assume that these agents work together?

Idea: Focus on reasonable coalitions

ATL^c: $\langle\!| A |\!\rangle\gamma$

A is able to form a reasonable coalition which enforces γ.

Motivation

ATL: $\langle\langle A\rangle\rangle\gamma$

Group A of agents can **enforce** property γ.

Where does A come from?

Is it *reasonable* to assume that these agents work together?

Idea: *Focus on reasonable coalitions*

ATL^c: $\langle\!| A |\!\rangle\gamma$

A is *able to form a reasonable* coalition which enforces γ.

Overview

How to model conflicts?

Based on Amgoud (2005):

Definition (Coalitional framework)

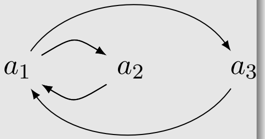
A coalitional framework is a tuple

$$\mathsf{cf} = (\mathfrak{C}, \mathcal{A})$$

where

- $\mathfrak{C}$: non-empty set of elements
- $\mathcal{A} \subseteq \mathfrak{C} \times \mathfrak{C}$: attack or defeat relation

$\mathbb{CF}(\mathbb{A}\mathrm{gt})$: coalitional frameworks over $\mathbb{A}\mathrm{gt}$



How to determine coalitions?

Definition (Coalitional framework semantics)

A **semantics** is a mapping

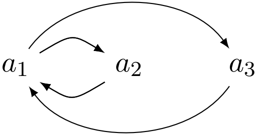
$$\mathfrak{sem} : \mathbb{CF}(\mathfrak{C}) \rightarrow \mathcal{P}(\mathcal{P}(\mathfrak{C}))$$

How to determine coalitions?

Definition (Coalitional framework semantics)

A **semantics** is a mapping

$$\mathfrak{sem} : \mathbb{CF}(\mathfrak{C}) \rightarrow \mathcal{P}(\mathcal{P}(\mathfrak{C}))$$



$\rightsquigarrow \quad \mathfrak{sem} : \{\{a_1, a_2\}, \{a_1\}\}$

Is $\{a_1, a_2\}$ sensible?

How to determine coalitions?

Definition (Coalitional framework semantics)

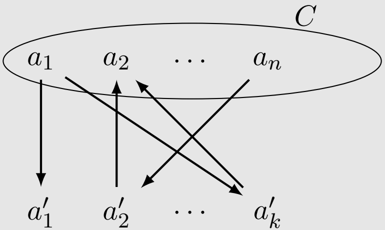
A **semantics** is a mapping

$$\mathfrak{sem} : \mathbb{CF}(\mathfrak{C}) \to \mathcal{P}(\mathcal{P}(\mathfrak{C}))$$

Consider for example **Stable Coalitions**:

Let $C \subseteq \mathfrak{C}$.
C is called **stable coalition** iff

- C is conflict-free and
- it defeats all elements not in C.



Coalitional ATL: Syntax

Definition (ATL^c)

The language $\mathcal{L}_{\mathsf{ATL}^c}$ is defined over nonempty sets:

- Π of propositions
- $\mathbb{A}\mathrm{gt}$ of agents

$$\varphi ::= \; p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\langle A \rangle\rangle \bigcirc \varphi \mid \langle\langle A \rangle\rangle \Box \varphi \mid \langle\langle A \rangle\rangle \varphi \mathcal{U} \varphi \mid \langle\!| A |\!\rangle \bigcirc \varphi \mid \langle\!| A |\!\rangle \Box \varphi \mid \langle\!| A |\!\rangle \varphi \mathcal{U} \varphi$$

Semantics

$\mathcal{M}, q \models \langle\!| A |\!\rangle \gamma$ iff
there is a reasonable coalition B (wrt A) which can enforce γ

Coalitional ATL: Syntax

Definition (ATL^c)

The language $\mathcal{L}_{\mathsf{ATL}^c}$ is defined over nonempty sets:

- Π of propositions
- $\mathbb{A}\mathrm{gt}$ of agents

$$\varphi ::= \quad p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\langle A\rangle\rangle \bigcirc \varphi \mid \langle\langle A\rangle\rangle \Box \varphi \mid \langle\langle A\rangle\rangle \varphi \,\mathcal{U}\, \varphi \mid \langle\!| A |\!\rangle \bigcirc \varphi \mid \langle\!| A |\!\rangle \Box \varphi \mid \langle\!| A |\!\rangle \varphi \,\mathcal{U}\, \varphi$$

Semantics

$\mathcal{M}, q \models \langle\!| A |\!\rangle \gamma$ iff
there is a reasonable coalition B (wrt A) which can enforce γ

Overview

Coalitional concurrent game structures

$$\mathcal{M} = \langle \mathbb{A}\text{gt}, Q, \Pi, \pi, Act, d, o, \zeta, \mathfrak{sem} \rangle$$

- $\zeta : Q \rightarrow \mathbb{CF}(\mathbb{A}\text{gt})$
- $\mathfrak{sem}$: (argumentation) semantics

Coalitional concurrent game structures

$\mathcal{M} = \langle \mathbb{A}\mathrm{gt}, Q, \Pi, \pi, Act, d, o, \zeta, \mathfrak{sem} \rangle$

- $\zeta : Q \rightarrow \mathbb{CF}(\mathbb{A}\mathrm{gt})$
- $\mathfrak{sem}$: (argumentation) semantics

Overview

Model Checking Complexity

Theorem

Model checking ATL^c is in $\Delta_2^{\mathrm{P}} = \mathrm{P}^{\mathrm{NP}}$ for standard argumentation semantics *(stable, preferred, admissible, ...).*

References

Joint work with Nils Bulling and Carlos Chesñevar.

N. Bulling, C. Chesñevar, and J. Dix.
Modelling coalitions: ATL+argumentation.
In *Proceedings of AAMAS'08*, Estoril, Portugal, May 2008. ACM Press.
Revised Version in LNAI 5384, pp.190-211, 2009.

N. Bulling and J. Dix.
A Finer grained Modelling of Rational Coalitions using GOALS.
In *Proceedings of the 14th Argentinean Conference on Computer Science CACIC'08*, September 2008.

Thank you.