(Bi)simulations for Multi-agent Systems

F. Belardinelli

Laboratoire IBISC – Université d'Evry
IRIT Toulouse

joint work with A. Lomuscio and R. Condurache, C. Dima, W. Jamroga, A. V. Jones

FMAI, Napoli – 23 February 2017

Outline

1. Background: (Bi)simulations for Modal Logics
   - Key notion to assess the expressivity of a modal language [vB76]
   - Abstraction-based techniques for system verification [CGJ+00, FV99] (Michael's talk)
2. The Problem: (Bi)simulations for Logics of Strategies
   - Relatively well-understood in the perfect information setting [AHKV98, ÅGJ07]
   - . . . less so under imperfect information
   - but imperfect information is crucial as well as difficult (Bastien's talk)
3. Today:

   [BCD+17]: (Bi)simulations for ATL_{ir} $\Rightarrow$ Verification of the ThreeBallot voting protocol

   [BL17]: Agent-based (bi)simulations and three-valued abstractions
4. Future Work:
   - More expressive languages (Strategy Logics, . . .)
   - Decidability of finding (bi)simulations
   - Abstraction refinement

Definition (ATL)

Formulas ϕ in **ATL** are defined by the following BNF:

$$\phi \quad ::= \quad p \mid \neg\phi \mid \phi \to \phi \mid \langle\langle A \rangle\rangle X\phi \mid \langle\langle A \rangle\rangle \phi U \phi \mid \langle\langle A \rangle\rangle \phi R \phi$$

ATL is interpreted on **Concurrent Game Structure with imperfect information**:

Definition (iCGS)

An **iCGS** is a CGS $\mathcal{G} = \langle Ag, AP, S, s_0, \{\sim_i\}_{i \in Ag}, Act, d, \to, \pi \rangle$ such that

- for every agent $i \in Ag$, $\sim_i$ is an **equivalence relation** on S

Semantical setup:

- we consider **uniform, memoryless strategies** $\sigma : S \to Act$
  - in particular, $s \sim_i s' \Rightarrow \sigma_i(s) = \sigma_i(s')$
- imperfect recall $\Rightarrow$ state-based semantics
- we consider both the **objective** and **subjective** interpretation of ATL

Definition (Semantics)

Given an iCGS $\mathcal{G}$, the *subjective* (resp. *objective*) interpretation $\models_x$ of an ATL formula ϕ at state s (for $x = subj$ (resp. $x = obj$)) is defined as

$$
\begin{array}{lll}
(\mathcal{G}, s) \models_x p & \text{iff} & p \in \pi(s) \\
(\mathcal{G}, s) \models_x \neg\phi & \text{iff} & (\mathcal{G}, s) \not\models_x \phi \\
(\mathcal{G}, s) \models_x \phi \wedge \phi' & \text{iff} & (\mathcal{G}, s) \models_x \phi \text{ and } (\mathcal{G}, s) \models_x \phi' \\
(\mathcal{G}, s) \models_x \langle\langle A \rangle\rangle X\phi & \text{iff} & \exists\sigma_A\ \forall\lambda \in out_x^{\mathcal{G}}(s, \sigma_A), (\mathcal{G}, \lambda[1]) \models_x \phi \\
(\mathcal{G}, s) \models_x \langle\langle A \rangle\rangle \phi U \phi' & \text{iff} & \exists\sigma_A\ \forall\lambda \in out_x^{\mathcal{G}}(s, \sigma_A), \exists j \geq 0 \text{ with } (\mathcal{G}, \lambda[j]) \models_x \phi' \text{ and } \forall 0 \leq k < j, (\mathcal{G}, \lambda[k]) \models_x \phi \\
(\mathcal{G}, s) \models_x \langle\langle A \rangle\rangle \phi R \phi' & \text{iff} & \exists\sigma_A\ \forall\lambda \in out_x^{\mathcal{G}}(s, \sigma_A), \text{ either } \forall j \geq 0, (\mathcal{G}, \lambda[j]) \models_x \phi, \text{ or} \\
& & \qquad \exists k \geq 0 \text{ with } (\mathcal{G}, \lambda[k]) \models_x \phi' \text{ and } \forall 0 \leq l \leq k, (\mathcal{G}, \lambda[l]) \models_x \phi
\end{array}
$$

The epistemic operator K_i is definable in the *subjective* interpretation of ATL:

- $K_i\phi ::= \langle\langle i \rangle\rangle \phi U \phi$

(Bi)simulations for ATL$_{ir}$

- **Partial strategies** are defined on subsets of S.
- $C_A(q) = \{q' \in S \mid q' \sim_A^C q\}$ is the **common knowledge neighbourhood** of q.

Definition (Simulation)

Consider two iCGS $\mathcal{G}$ and $\mathcal{G}'$ (on the same sets Ag and AP), and a group $A \subseteq Ag$ of agents.
A relation $\rightharpoonup_A \subseteq S \times S'$ is a **simulation for** A iff $q \rightharpoonup_A q'$ implies that

1. $\pi(q) = \pi'(q')$
2. for every $i \in A$ and $r' \in S'$, if $q' \sim_i' r'$ then for some $r \in S$, $q \sim_i r$ and $r \rightharpoonup_A r'$
3. there exists a mapping $ST = ST_{C_A(q), C_A(q')}$ with $ST : PStr_A(C_A(q)) \to PStr_A(C_A(q'))$ such that for any two states $r \in C_A(q)$, $r' \in C_A(q')$, if $r \rightharpoonup_A r'$ then
   1. for every partial strategy $\sigma_A \in PStr_A(C_A(q))$ and state $s' \in S'$, if $r' \xrightarrow{ST(\sigma_A)(r')} s'$ then there exists some state s such that $r \xrightarrow{\sigma_A(r)} s$ and $s \rightharpoonup_A s'$
   2. $ST_{C_A(q), C_A(q')} = ST_{C_A(r), C_A(r')}$

Bisimulations are defined in the standard way.

Remark

Checking the existence of a (bi)simulation between iCGS is in PSPACE.

Preservation Result

Theorem

Consider iCGS $\mathcal{G}$ and $\mathcal{G}'$ and A-bisimilar states $q \in S$, $q' \in S'$. Then, for every A-formula φ,

$$(\mathcal{G}, q) \models \varphi \quad \textit{if and only if} \quad (\mathcal{G}', q') \models \varphi$$

The proof makes use of the following lemma:

Lemma

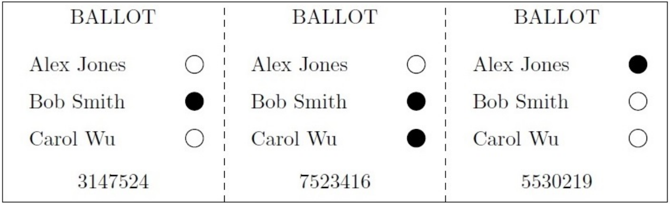
If $q \rightharpoonup_A q'$ then for every uniform strategy σ_A, there exists a uniform strategy σ'_A such that

- *for every run $\lambda' \in out_x^{\mathcal{G}'}(q', \sigma'_A)$, for $x \in \{subj, obj\}$, there exists a run $\lambda \in out_x^{\mathcal{G}}(q, \sigma_A)$ such that $\lambda(i) \rightharpoonup_A \lambda'(i)$ for every $i \geq 0$.*

Applications: the Three-Ballot Voting Protocol

ThreeBallot is a voting protocol **without cryptography** [RR07].

1. Each voter gets a paper "multi-ballot" to vote with.



2. The voter fills in the multi-ballot, separates the three parts and casts them in the ballot box.
   - to vote for a candidate, one must mark exactly two (arbitrary) bubbles on her row;
   - to not vote for a candidate, one must mark exactly one of the bubbles on her row;
   - in all the other cases the vote is invalid.
3. The voter also receives a copy of one of her three ballots.
4. The ballots are tallied by counting the number of bubbles marked for each candidate, and then subtracting the number of voters from the count.
5. All ballots are scanned and published on the web bulletin board (BB).
6. The voter can check if her receipt matches a ballot listed on the BB.
7. If no ballot matches the receipt, the voter can file a complaint.

iCGS for the Three-Ballot Voting Protocol

- The ThreeBallot voting protocol can be represented as iCGS

iCGS for the Three-Ballot Voting Protocol

- The ThreeBallot voting protocol can be represented as iCGS

  . . . but these are large

iCGS for the Three-Ballot Voting Protocol

- The ThreeBallot voting protocol can be represented as iCGS

  . . . but these are large

Several possible formalisations:

- $\mathcal{G}_{tot}$: for each agent, any configuration of the three ribbons (compatible with the agent's choice) is allowed.
- $\mathcal{G}_{lex}$: for each agent, a single representative of her choice is produced.
- $\mathcal{G}_{count}$: the environment no longer copies ribbons on the ballot board, but rather counts the votes for each candidate by "peeping" at the ballot of each voter.

Proposition

All $\mathcal{G}_{tot}$, $\mathcal{G}_{lex}$, $\mathcal{G}_{count}$ are bisimilar (for the attacker), but with increasingly smaller state spaces.

Verification of ThreeBallot

The attacker has a strategy whereby she knows how some of the agents have voted (for $i \neq att$):

$$\varphi_i = \langle\langle att \rangle\rangle F \big(pub \wedge (v_i \rightarrow \bigvee_{1 \leq j \leq nc} K_{att} \; p_{ch_i = j}) \big)$$

- statistics for $\mathcal{G}_{tot}$:

		# voters		
		2v	3v	4v
# candid.	2c	0.93 s $\lvert S\rvert = 3.49091e{+}06$	7.765 s $\lvert S\rvert = 1.46625e{+}10$	NA
	3c	23.61 s $\lvert S\rvert = 2.44048e{+}08$	NA	NA

- statistics for $\mathcal{G}_{lex}$:

		# voters		
		2v	3v	4v
# candid.	2c	0.38 s $\lvert S\rvert = 196388$	3.42 s $\lvert S\rvert = 1.92068e{+}08$	823.12 s $\lvert S\rvert = 2.26211e{+}11$
	3c	15.32 s $\lvert S\rvert = 8.09895e{+}06$	4807.79 s $\lvert S\rvert = 1.03982e{+}11$	NA

- statistics for $\mathcal{G}_{count}$:

		# voters			
		2v	3v	4v	5v
# candid.	2c	0.15 s $\lvert S\rvert = 4406$	0.72 s $\lvert S\rvert = 39201$	2.39 s $\lvert S\rvert = 3.08043e{+}06$	17.03 s $\lvert S\rvert = 6.57133e{+}07$
	3c	0.44 s $\lvert S\rvert = 101993$	4.29 s $\lvert S\rvert = 3.81446e{+}06$	44.18 s $\lvert S\rvert = 2.17425e{+}09$	NA

Verification of ThreeBallot

The attacker has a strategy whereby she knows how some of the agents have voted (for $i \neq att$):

$$\varphi_i = \langle\langle att \rangle\rangle F\big(pub \wedge (v_i \rightarrow \bigvee_{1 \leq j \leq nc} K_{att}\ p_{ch_i = j})\big)$$

- statistics for $\mathcal{G}_{tot}$:

		# voters		
		2v	3v	4v
# candid.	2c	0.93 s $\|S\| = 3.49091e{+}06$	7.765 s $\|S\| = 1.46625e{+}10$	NA
	3c	23.61 s $\|S\| = 2.44048e{+}08$	NA	NA

- statistics for $\mathcal{G}_{lex}$:

		# voters		
		2v	3v	4v
# candid.	2c	0.38 s $\|S\| = 196388$	3.42 s $\|S\| = 1.92068e{+}08$	823.12 s $\|S\| = 2.26211e{+}11$
	3c	15.32 s $\|S\| = 8.09895e{+}06$	4807.79 s $\|S\| = 1.03982e{+}11$	NA

- statistics for $\mathcal{G}_{count}$:

		# voters			
		2v	3v	4v	5v
# candid.	2c	0.15 s $\|S\| = 4406$	0.72 s $\|S\| = 39201$	2.39 s $\|S\| = 3.08043e{+}06$	17.03 s $\|S\| = 6.57133e{+}07$
	3c	0.44 s $\|S\| = 101993$	4.29 s $\|S\| = 3.81446e{+}06$	44.18 s $\|S\| = 2.17425e{+}09$	NA

Smaller state space $\Rightarrow$ Faster verification

Summary of [BCD+17]

Results:

- A novel notion of (bi)simulation on iCGS that preserves the interpretation of ATL_{ir}
- A (rather preliminary) application to the verification of the ThreeBallot voting protocol

Future work:

- Bisimulations for iCGS with perfect and bounded recall: in many applications agents do have some memory of past states and actions.
- For the verification of voting protocols, it is key to extend ATL with epistemic modalities to express properties of secrecy, anonymity and confidentiality.
- Automating and implementing the procedure in a model checking tool for the formal verification of (electronic) voting protocols.

Three-value Simulations and Abstractions

- Three-value abstractions for temporal logics:
  - understood in terms of *over-* and *under-approximations* of the system's transitions [BG99]
  - $\exists\exists$-transitions as *may*-transitions
  - $\forall\exists$-transitions as *must*-transitions
- Extended to ATL (with perfect information) [SG04, BK06]
- Here we consider the imperfect information case
- Even more interestingly, we consider agent-based *simulations* and *abstractions* (kind of ...)
  - compact representation of multi-agent systems (Hector's talk)

Three-value Semantics

We assume the notion of agent as primitive [FHMV95]

Definition (Generalised Agent)

A **(generalised) agent** is a tuple $i = \langle L, Act, P^{may}, P^{must}, t^{may}, t^{must} \rangle$ such that

- L is the (possibly infinite) set of **local states**
- Act is the (finite) set of **individual actions**
- P^{may} and P^{must} are **protocol functions** from L to 2^{Act}.
  - for every $l \in L$, $P^{must}(l) \subseteq P^{may}(l)$
- t^{may} and t^{must} are **local transition relations** defined on $L \times ACT \times L$.
  1. for $x \in \{may, must\}$, transition $t^x(l, a, l')$ holds for some $l' \in L$ iff $a_i \in P^x(l)$
  2. $t^{must} \subseteq t^{may}$

- Definition motivated by abstractions
- **Standard** agents [FHMV95] have
  - $P^{must}(l) = P^{may}(l)$
  - $t^{must} = t^{may}$

Three-value Semantics

Agents interact, thus generating Interpreted Systems (iCGS in disguise).

Definition (Generalised IS)

A **(generalised) interpreted system** is a tuple $M = \langle Ag, I, T, \Pi \rangle$ such that

- every $i \in Ag$ is an **agent**
- $I \subseteq \mathcal{G}$ is the set of **(global) initial states**
- $T : \mathcal{G} \times ACT \to \mathcal{G}$ is the **global transition function**
  - $s' = T(s, a)$ iff for all $i \in Ag$, $s'_i = t_i^x(s_i, a)$ for $x \in \{may, must\}$
- $\Pi : \mathcal{G} \times AP \to \{\mathrm{tt}, \mathrm{ff}, \mathrm{uu}\}$ is the **labelling function**

In **standard** IS [FHMV95] we have

- all agents are standard
- the value of atoms is always defined ($\neq \mathrm{uu}$)

Three-value Semantics

We have *must* and *may* strategies.

Definition (Uniform x-Strategy)

For $x \in \{may, must\}$, a (**uniform, memoryless**) x-**strategy** for $i \in Ag$ is a function $\sigma_i^x : L_i \to Act_i$. In particular, for every local state $l \in L_i$, $\sigma_i^x(l) \in P_i^x(l)$.

Strategies are uniform.

Definition (Satisfaction)

The 3-valued satisfaction relation $\models^3$ for an IS M, state $s \in \mathcal{S}$, and ATL formula ϕ is defined as

$$
\begin{array}{lll}
((M,s) \models^3 q) = \tau & \text{iff} & \Pi(s,q) = \tau, \text{ for } \tau \in \{\mathrm{tt}, \mathrm{ff}\} \\
((M,s) \models^3 \neg\varphi) = \mathrm{tt} & \text{iff} & ((M,s) \models^3 \varphi) = \mathrm{ff} \\
((M,s) \models^3 \neg\varphi) = \mathrm{ff} & \text{iff} & ((M,s) \models^3 \varphi) = \mathrm{tt} \\
((M,s) \models^3 \varphi \wedge \varphi') = \mathrm{tt} & \text{iff} & ((M,s) \models^3 \varphi) = \mathrm{tt} \text{ and } ((M,s) \models^3 \varphi') = \mathrm{tt} \\
((M,s) \models^3 \varphi \wedge \varphi') = \mathrm{ff} & \text{iff} & ((M,s) \models^3 \varphi) = \mathrm{ff} \text{ or } ((M,s) \models^3 \varphi') = \mathrm{ff} \\
((M,s) \models^3 \langle\langle A \rangle\rangle X \varphi) = \mathrm{tt} & \text{iff} & \text{for some } \sigma_A^{must}, \text{ for all } \lambda \in out(s, \sigma_A^{must}), ((M, \lambda[1]) \models^3 \varphi) = \mathrm{tt} \\
((M,s) \models^3 \langle\langle A \rangle\rangle X \varphi) = \mathrm{ff} & \text{iff} & \text{for every } \sigma_A^{may}, \text{ for some } \lambda \in out(s, \sigma_A^{may}), ((M, \lambda[1]) \models^3 \varphi) = \mathrm{ff} \\
 & \vdots &
\end{array}
$$

In all other cases the value of ϕ is undefined (uu).

Three-value Semantics

The three-value semantics is a conservative extension of the standard two-value semantics:

Proposition

In every standard IS M, for every state $s \in \mathcal{S}$ and ATL formula ϕ,

$$((M,s) \models^3 \phi) = \mathrm{tt} \quad \textit{iff} \quad (M,s) \models \phi$$
$$((M,s) \models^3 \phi) = \mathrm{ff} \quad \textit{iff} \quad (M,s) \not\models \phi$$

In particular, the truth value $((M,s) \models^3 \phi)$ is always defined.

Agent-based Simulations

First, we define simulation on local states.

- hereafter we assume the same actions for simulation and simulator
- no such limitation in the paper

Definition (Local Simulation)

A **local simulation** for agent i is a relation $\Sigma_i \subseteq L_i \times L'_i$ such that $\Sigma_i(l_1, l'_1)$ implies

1. $P_i^{must}(l_1) \subseteq P_i'^{must}(l'_1)$
2. $P_i'^{may}(l'_1) \subseteq P_i^{may}(l_1)$

   Moreover,
3. for all $l_2 \in L_i$, if $t_i^{must}(l_1, a, l_2)$ then for some $l'_2 \in L'_i$, $t_i'^{must}(l'_1, a, l'_2)$ and $\Sigma_i(l_2, l'_2)$
4. for all $l'_2 \in L'_i$, if $t_i'^{may}(l'_1, a, l'_2)$ then for some $l_2 \in L_i$, $t_i^{may}(l_1, a, l_2)$ and $\Sigma_i(l_2, l'_2)$

Intuition: If $l \preceq l'$ then

- l' 'simulates' *must*-transitions from l
- l 'simulates' *may*-transitions from l'

Agent-based Simulations

Second, we define simulation on agents.

Definition (Agent Simulation)

The primed agent i' *must*-**simulates** agent $i \in Ag$, or $i \preceq^{must} i'$, iff

- for every $l \in L$, $l \preceq l'$ for some $l' \in L'$.

Agent i' *may*-**simulates** i, or $i \preceq^{may} i'$, iff

- for every $l \in L$, $l' \preceq l$ for some $l' \in L'$.

Intuition: agent i' *must*-simulates agent i iff

- i' has 'more' *must*-transitions than i
- i' has 'less' *may*-transitions than i.

Symmetrically for *may*-simulations.

Given a set $A \subseteq Ag$ of agents, $Ag'_A = \{i' \mid i \preceq^{must} i', i \in A\} \cup \{j' \mid j \preceq^{may} j', j \in \overline{A}\}$

Definition (State Simulation)

A global state s' defined on Ag'_A **simulates** s on Ag, or $s \preceq_A s'$, iff

1. for every $i \in A$, $s_i \preceq s'_i$
2. for every $i \in \overline{A}$, $s'_i \preceq s_i$

Agent-based Simulations

Finally, we define simulation on IS.

Definition (IS Simulation)

Given a set $A \subseteq Ag$ of agents, an IS M' A-**simulates** an IS M, or $M \preceq_A M'$, iff

1. Ag'_A is the set of simulations for agents in Ag
2. for every $s \in I$, $s \preceq_A s'$ for some $s' \in I'$
3. for every $s \in \mathcal{S}$, $s' \in \mathcal{S}'$, if $s \preceq_A s'$ and $\Pi'(s', p) = t$, for $t \in \{\mathrm{tt}, \mathrm{ff}\}$, then $\Pi(s, p) = t$.

Theorem (Preservation Result)

If $M \preceq_A M'$, $s \preceq_A s'$ and $\tau \in \{\mathrm{tt}, \mathrm{ff}\}$, then for every A-formula ϕ,

$$((M', s') \models^3 \phi) = \tau \quad \textit{implies} \quad ((M, s) \models^3 \phi) = \tau$$

Agent-based Abstractions

We can introduce suitable abstractions for local states, agents, and IS:

- local states are partitioned in equivalence classes
- $\exists\exists$-transitions as *may*-transitions
- $\forall\exists$-transitions as *must*-transitions

Theorem

The abstraction M^A A-simulates the IS M.

Corollary

If M^A is the abstraction of IS M, $s \in s'$, and $\tau \in \{\mathrm{tt}, \mathrm{ff}\}$, then for every A-formula ϕ,

$$((M^{Abs}, s') \models^3 \phi) = \tau \quad \textit{implies} \quad ((M, s) \models^3 \phi) = \tau$$

In the paper we discuss an instance of the Train-Gate-Controller scenario with counters.

Summary of [BL17]

Results:

- Three-value simulations for ATL under imperfect information.
- Three-value abstractions that are similar.
- Both are based on a notion of agent $\Rightarrow$ allows for modular abstraction

Future Work:

- Counterexample-guided refinement?
- Strategy Logic?
- Tool?

Questions?

References

T. Ågotnes, V. Goranko, and W. Jamroga.
Alternating-time temporal logics with irrevocable strategies.
In *Proceedings of TARK XI*, pages 15–24, 2007.

Rajeev Alur, Thomas A. Henzinger, Orna Kupferman, and Moshe Y. Vardi.
Alternating refinement relations.
In *In Proceedings of the Ninth International Conference on Concurrency Theory (CONCUR98), volume 1466 of LNCS*, pages 163–178. Springer-Verlag, 1998.

F. Belardinelli, R. Condurache, C. Dima, W. Jamroga, and A. V. Jones.
Bisimulations for verifying strategic abilities applied to voting protocols.
In *Proceedings of the 16th International Conference onAutonomous Agents and Multi-Agent Systems (AAMAS17)*. IFAAMAS, 2017.

G. Bruns and P. Godefroid.
Model checking partial state spaces.
In *Proceedings of the 11th International Conference on Computer Aided Verification (CAV99)*, volume 1633 of *LNCS*, pages 274–287. Springer-Verlag, 1999.

Thomas Ball and Orna Kupferman.
An abstraction-refinement framework for multi-agent systems.
In *Proceedings of the 21st Annual IEEE Symposium on Logic in Computer Science (LICS06)*, pages 379–388. IEEE, 2006.

F. Belardinelli and A. Lomuscio.
Agent-based abstractions for verifying alternating-time temporal logic with imperfect information.
In *Proceedings of the 16th International Conference onAutonomous Agents and Multi-Agent Systems (AAMAS17)*. IFAAMAS, 2017.

E. M. Clarke, O. Grumberg, S. Jha, Y. Lu, and H. Veith.
Counterexample-guided abstraction refinement.
In *Proceedings of the 12th International Conference on Computer Aided Verification (CAV00)*, volume 1855 of *Lecture Notes in Computer Science*, pages 154–169. Springer, 2000.