Coalitional Games

Stéphane Airiau and Wojtek Jamroga

Stéphane: ILLC @ University of Amsterdam
Wojtek: ICR @ University of Luxembourg

European Agent Systems Summer School
Torino, Italy, September 2009
Part 2. Reasoning about Coalitions

- 2.1 Modal Logic
- 2.2 ATL
- 2.3 Rational Play (ATLP)
- 2.4 Imperfect Information
- 2.5 Model Checking
- 2.6 References
2. Reasoning about Coalitions

Outline

- In the previous chapter, we showed how coalitions can be rationally formed
- In this chapter, we show how one can use modal logic to reason about their play and their outcome.
2. Reasoning about Coalitions

1. Modal Logic

2.1 Modal Logic
Why logic at all?

- framework for thinking about systems,
- makes one realise the implicit assumptions,
- ... and then we can:
  - investigate them, accept or reject them,
  - relax some of them and still use a part of the formal and conceptual machinery;

- reasonably expressive but simpler and more rigorous than the full language of mathematics.
2. Reasoning about Coalitions

1. Modal Logic

Why logic at all?

- **Verification**: check specification against implementation
- **Executable specifications**
- **Planning as model checking**

- **Game solving, mechanism design, and reasoning about games** have natural interpretation as logical problems
Modal logic is an extension of classical logic by new connectives □ and ◊: necessity and possibility.

- “□p is true” means $p$ is necessarily true, i.e. true in every possible scenario,
- “◊p is true” means $p$ is possibly true, i.e. true in at least one possible scenario.
Various modal logics:

- knowledge $\rightarrow$ epistemic logic,
- beliefs $\rightarrow$ doxastic logic,
- obligations $\rightarrow$ deontic logic,
- actions $\rightarrow$ dynamic logic,
- time $\rightarrow$ temporal logic,

and combinations of the above

Most famous multimodal logic: BDI logic of beliefs, desires, intentions (and time)
Definition 2.1 (Kripke Semantics)

Kripke model (possible world model):

\[ M = (\mathcal{W}, R, \pi), \]

- \( \mathcal{W} \) is a set of possible worlds
- \( R \subseteq \mathcal{W} \times \mathcal{W} \) is an accessibility relation
- \( \pi : \mathcal{W} \to \mathcal{P}(\Pi) \) is a valuation of propositions.

\[ M, w \models \Box \varphi \text{ iff for every } w' \in \mathcal{W} \text{ with } wRw' \text{ we have that } M, w' \models \varphi. \]
An Example

\[ x \models 1 \rightarrow K_s x \models 1 \]
2. Reasoning about Coalitions

2.2 ATL
2. Reasoning about Coalitions

2. ATL

ATL: What Agents Can Achieve

- Temporal logic meets game theory
- Main idea: cooperation modalities

\[ \langle A \rangle \Phi: \text{ coalition } A \text{ has a collective strategy to enforce } \Phi \]
2. Reasoning about Coalitions

2. ATL

- $\langle j\text{ames}b\text{ond}\rangle \Diamond \text{win}$:
  “James Bond has an infallible plan to eventually win”

- $\langle j\text{ames}b\text{ond}, b\text{ondsgirl}\rangle \text{fun} \bigcup \text{shot}$:
  “James Bond and his girlfriend are able to have fun until someone shoots at them”

- “Vanilla” ATL: every temporal operator preceded by exactly one cooperation modality;

- ATL*: no syntactic restrictions;
ATL Models: Concurrent Game Structures

- Agents, actions, transitions, atomic propositions
- Atomic propositions + interpretation
- Actions are abstract
Definition 2.2 (Concurrent Game Structure)

A concurrent game structure is a tuple $M = \langle \text{Agt}, St, \pi, Act, d, o \rangle$, where:

- **Agt**: a finite set of all agents
- **St**: a set of states
- **$\pi$**: a valuation of propositions
- **Act**: a finite set of (atomic) actions
- **$d : \text{Agt} \times St \rightarrow \mathcal{P}(Act)$**: defines actions available to an agent in a state
- **$o$: a deterministic transition function** that assigns outcome states $q' = o(q, \alpha_1, \ldots, \alpha_k)$ to states and tuples of actions
Example: Robots and Carriage
Definition 2.3 (Strategy)

A **strategy** is a conditional plan. We represent strategies by functions $s_a : St \rightarrow Act$.

Function $\text{out}(q, S_A)$ returns the set of all paths that may result from agents $A$ executing strategy $S_A$ from state $q$ onward.
Definition 2.4 (Semantics of ATL)

\[ M, q \models p \iff p \text{ is in } \pi(q); \]
\[ M, q \models \varphi \land \psi \iff M, q \models \varphi \text{ and } M, q \models \psi; \]
\[ M, q \models \langle A \rangle \Phi \iff \text{there is a collective strategy } S_A \text{ such that, for every path } \lambda \in \text{out}(q, S_A), \text{ we have } M, \lambda \models \Phi. \]

\[ M, \lambda \models \Box \varphi \iff M, \lambda[1] \models \varphi; \]
\[ M, \lambda \models \lozenge \varphi \iff M, \lambda[i] \models \varphi \text{ for some } i \geq 0; \]
\[ M, \lambda \models \square \varphi \iff M, \lambda[i] \models \varphi \text{ for all } i \geq 0; \]
\[ M, \lambda \models \varphi \mathcal{U} \psi \iff M, \lambda[i] \models \psi \text{ for some } i \geq 0, \text{ and } M, \lambda[j] \models \varphi \text{ for all } 0 \leq j \leq i. \]
Example: Robots and Carriage

\[ \text{pos}_0 \rightarrow \langle 1 \rangle \square \neg \text{pos}_1 \]
Temporal operators allow a number of useful concepts to be formally specified

- safety properties
- liveness properties
- fairness properties
2. Reasoning about Coalitions

Safety (maintenance goals):

“something bad will not happen”
“something good will always hold”

Typical example:

□¬bankrupt

Usually: □¬....

In ATL:

⟨⟨os⟩⟩□¬crash
2. Reasoning about Coalitions

Liveness (achievement goals):

“something good will happen”

Typical example:

◊ rich

Usually: ◊ ....

In ATL:

\[\langle \text{alice, bob} \rangle \Diamond \text{paperAccepted} \]
2. Reasoning about Coalitions

Fairness (service goals):

“if something is attempted/requested, then it will be successful/allocated”

Typical examples:

\(\Box (\text{attempt} \rightarrow \Diamond \text{success})\)

\(\Box \Diamond \text{attempt} \rightarrow \Box \Diamond \text{success}\)

In ATL* (!):

\(\langle \langle \text{prod, dlr} \rangle \Box (\text{carRequested} \rightarrow \Diamond \text{carDelivered})\)
Connection to Games

- Concurrent game structure = generalized extensive game

- $\langle A \rangle \gamma: \langle A \rangle$ splits the agents into proponents and opponents

- $\gamma$ defines the winning condition
  $\leadsto$ infinite 2-player, binary, zero-sum game

- Flexible and compact specification of winning conditions
- Solving a game ≈ checking if $M, q \models \langle A \rangle \gamma$
- But: do we really want to consider all the possible plays?
2.3 Rational Play (ATLP)
Game-theoretical analysis of games:

- **Solution concepts** define rationality of players
  - maxmin
  - Nash equilibrium
  - subgame-perfect Nash
  - undominated strategies
  - Pareto optimality

- Then: we assume that players play rationally
- ...and we ask about the outcome of the game under this assumption

- Role of rationality criteria: constrain the possible game moves to “sensible” ones
2. Reasoning about Coalitions

3. Rational Play (ATLP)

start → ¬\(\langle 1 \rangle \diamond \text{money}_1\)

start → ¬\(\langle 2 \rangle \diamond \text{money}_2\)
2. Reasoning about Coalitions

3. Rational Play (ATLP)

ATL + Plausibility (ATLP)

ATL: reasoning about all possible behaviors.

$\langle \langle A \rangle \rangle \varphi$: agents $A$ have some collective strategy to enforce $\varphi$ against any response of their opponents.

ATLP: reasoning about plausible behaviors.

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

Important

The possible strategies of both $A$ and $\text{Agt} \setminus A$ are restricted.
New in ATLP:

\((\text{set-pl } \omega)\) : the set of plausible profiles is \textit{set/reset} to the strategies described by \(\omega\). Only \textit{plausible strategy profiles} are considered!

Example: \((\text{set-pl } greedy_1)\langle 2 \rangle \Diamond \text{money}_2\)
Concurrent game structures with plausibility

\[ M = (\text{Ag}t, St, \Pi, \pi, \text{Act}, d, \delta, \mathcal{Y}, \Omega, \| \cdot \|) \]

- \[ \mathcal{Y} \subseteq \Sigma \]: set of (plausible) strategy profiles

- \[ \Omega = \{ \omega_1, \omega_2, \ldots \} \]: set of plausibility terms
  
  Example: \[ \omega_{NE} \] may stand for all Nash equilibria

- \[ \| \cdot \| : St \rightarrow (\Omega \rightarrow \mathcal{P}(\Sigma)) \]: plausibility mapping
  
  Example: \[ \| \omega_{NE} \|_q = \{ (\text{confess, confess}) \} \]
Outcome = Paths that may occur when agents $A$ perform $s_A$ when only plausible strategy profiles from $\Upsilon$ are played

\[
\text{out}_\Upsilon(q, s_A) = \{ \lambda \in St^+ \mid \exists t \in \Upsilon(s_A) \forall i \in \mathbb{N} \left( \lambda[i + 1] = \delta(\lambda[i], t(\lambda[i])) \right) \}
\]

$P$: the players always show same sides of their coins

$s_1$: always show “heads”
Semantics of ATLP

\[ M, q \models \langle A \rangle \gamma \] iff there is a strategy \( s_A \) consistent with \( \gamma \) such that \( M, \lambda \models \gamma \) for all \( \lambda \in \text{out}_\gamma(q, s_A) \).

\[ M, q \models (\text{set-pl} \omega) \varphi \] iff \( M^\omega, q \models \varphi \) where the new model \( M^\omega \) is equal to \( M \) but the new set \( \gamma^\omega \) of plausible strategy profiles is set to \( \| \omega \|_q \).
Example: Pennies Game

What is a Nash equilibrium in this game? We need some kind of winning criteria!
Agent 1 “wins”, if \( \gamma_1 \equiv \Box (\neg \text{start} \rightarrow \text{money}_1) \) is satisfied.
Agent 2 “wins”, if \( \gamma_2 \equiv \diamond \text{money}_2 \) is satisfied.

Now we have a qualitative notion of success.

\[
M, q_0 \models (\text{set-pl } \omega_{NE}) \langle \langle 2 \rangle \rangle \Box (\neg \text{start} \rightarrow \text{money}_1)
\]

where \( \parallel \omega_{NE} \parallel_{q_0} = “\text{all profiles belonging to grey cells}”. \)
How to obtain plausibility terms?

**Idea**

Formulae that describe plausible strategies!

\[(\text{set-pl } \sigma.\theta) \varphi\]: “suppose that \(\theta\) characterizes rational strategy profiles, then \(\varphi\) holds”.

Sometimes quantifiers are needed...

E.g.: \((\text{set-pl } \sigma. \forall \sigma' \text{ dominates}(\sigma, \sigma'))\)
Characterization of Nash Equilibrium

σₐ is a’s best response to σ (wrt γ):

BRₐ(σ) ≜ (set-pl σ[Agt\{a}]) (⟨a⟩γₐ → (set-pl σ)⟨∅⟩γₐ)

σ is a Nash equilibrium:

NEγ(σ) ≜ ∩ₐ∈Agt BRₐ(σ)
Example: Pennies Game revisited

\[ \gamma_1 \equiv \Box(\neg \text{start} \rightarrow \text{money}_1); \quad \gamma_2 \equiv \Diamond \text{money}_2. \]

\[
\begin{array}{c|cccc}
\gamma_1 \setminus \gamma_2 & hh & ht & th & tt \\
\hline
HH & 1, 1 & 0, 0 & 0, 1 & 0, 1 \\
HT & 0, 0 & 0, 1 & 0, 1 & 0, 1 \\
TH & 0, 1 & 0, 1 & 1, 1 & 0, 0 \\
TT & 0, 1 & 0, 1 & 0, 0 & 0, 1 \\
\end{array}
\]

\[ M_1, q_0 \models (\text{set-pl} \sigma. NE^{\gamma_1, \gamma_2}(\sigma))\langle 2 \rangle \Box(\neg \text{start} \rightarrow \text{money}_1) \]

...where \( NE^{\gamma_1, \gamma_2}(\sigma) \) is defined as on the last slide.
Characterizations of Other Solution Concepts

\( \sigma \) is a \textit{subgame perfect Nash equilibrium}:
\[
SPN^{\tilde{\gamma}}(\sigma) \equiv \langle \emptyset \rangle \Box \text{NE}^{\tilde{\gamma}}(\sigma)
\]

\( \sigma \) is \textit{Pareto optimal}:
\[
PO^{\tilde{\gamma}}(\sigma) \equiv \forall \sigma' \Big( \\
\bigwedge_{a \in \text{Agt}} \left( (\text{set-pl} \ \sigma') \langle \emptyset \rangle \gamma_a \rightarrow (\text{set-pl} \ \sigma) \langle \emptyset \rangle \gamma_a \right) \lor \\
\bigvee_{a \in \text{Agt}} \left( (\text{set-pl} \ \sigma) \langle \emptyset \rangle \gamma_a \land \neg (\text{set-pl} \ \sigma') \langle \emptyset \rangle \gamma_a \right) ,
\Big)
\]
σ is undominated:

\[\text{UNDOM}^\gamma(\sigma) \equiv \forall \sigma_1 \forall \sigma_2 \exists \sigma_3 \]

\[
\left( ((\text{set-pl} \langle \sigma_1 \{a\} , \sigma_2 \mathbb{Agt} \{a\} \rangle \langle \emptyset \rangle \gamma_a \rightarrow \right.
\]

\[
(\text{set-pl} \langle \sigma_2 \{a\} , \sigma_3 \mathbb{Agt} \{a\} \rangle \langle \emptyset \rangle \gamma_a )
\]

\[
\lor ( (\text{set-pl} \langle \sigma_3 \{a\} , \sigma_3 \mathbb{Agt} \{a\} \rangle \langle \emptyset \rangle \gamma_a \land
\]

\[
\neg (\text{set-pl} \langle \sigma_1 \{a\} , \sigma_3 \mathbb{Agt} \{a\} \rangle \langle \emptyset \rangle \gamma_a ) \right).
\]
Theorem 2.5

The characterizations coincide with game-theoretical solution concepts in the class of game trees.
2. Reasoning about Coalitions

2.4 Imperfect Information
How can we reason about extensive games with imperfect information?

Let’s put ATL and epistemic logic in one box.

⇝ Problems!
Does it make sense?
Problem:
Strategic and epistemic abilities are *not* independent!

\[ \langle A \rangle \Phi = A \text{ can enforce } \Phi \]

It should at least mean that \( A \) are able to *identify* and *execute* the right strategy!

Executable strategies = uniform strategies
Definition 2.6 (Uniform strategy)

Strategy $s_a$ is uniform iff it specifies the same choices for indistinguishable situations:

- (no recall:) if $q \sim_a q'$ then $s_a(q) = s_a(q')$
- (perfect recall:) if $\lambda \approx_a \lambda'$ then $\Rightarrow s_a(\lambda) = s_a(\lambda)$, where $\lambda \approx_a \lambda'$ iff $\lambda[i] \sim_a \lambda'[i]$ for every $i$.

A collective strategy is uniform iff it consists only of uniform individual strategies.
Note:

Having a successful strategy does not imply knowing that we have it!

Knowing that a successful strategy exists does not imply knowing the strategy itself!
Levels of Strategic Ability

From now on, we restrict our discussion to uniform memoryless strategies.

Our cases for $\langle A \rangle \Phi$ under incomplete information:

1. There is $\sigma$ such that, for every execution of $\sigma$, $\Phi$ holds
2. $A$ know that there is $\sigma$ such that, for every execution of $\sigma$, $\Phi$ holds
3. There is $\sigma$ such that $A$ know that, for every execution of $\sigma$, $\Phi$ holds
Case [4]: knowing how to play

- Single agent case: we take into account the paths starting from indistinguishable states (i.e., \( \bigcup_{q' \in \text{img}(q, \sim_a)} \text{out}(q, s_A) \))

- What about coalitions?
- Question: in what sense should they know the strategy? Common knowledge \((C_A)\), mutual knowledge \((K_A)\), distributed knowledge \((D_A)\)?
Given strategy $\sigma$, agents $A$ can have:

- **Common knowledge** that $\sigma$ is a winning strategy. This requires the least amount of additional communication (agents from $A$ may agree upon a total order over their collective strategies at the beginning of the game and that they will always choose the maximal winning strategy with respect to this order)

- **Mutual knowledge** that $\sigma$ is a winning strategy: everybody in $A$ knows that $\sigma$ is winning
Distributed knowledge that $\sigma$ is a winning strategy: if the agents share their knowledge at the current state, they can identify the strategy as winning

“The leader”: the strategy can be identified by agent $a \in A$

“Headquarters’ committee”: the strategy can be identified by subgroup $A' \subseteq A$

“Consulting company”: the strategy can be identified by some other group $B$
Many subtle cases...

Solution: constructive knowledge operators
Constructive Strategic Logic (CSL)

- $\langle A \rangle \Phi$: $A$ have a uniform memoryless strategy to enforce $\Phi$
- $K_a \langle a \rangle \Phi$: $a$ has a strategy to enforce $\Phi$, and knows that he has one
- For groups of agents: $C_A, E_A, D_A, \ldots$
- $K_a \langle a \rangle \Phi$: $a$ has a strategy to enforce $\Phi$, and knows that this is a winning strategy
- For groups of agents: $C_A, E_A, D_A, \ldots$
Non-standard semantics:

- Formulae are evaluated in sets of states
- $M, Q \models \langle A \rangle \Phi$: A have a single strategy to enforce $\Phi$ from all states in $Q$

Additionally:

- $\text{out}(Q, S_A) = \bigcup_{q \in Q} \text{out}(q, S_A)$
- $\text{img}(Q, R) = \bigcup_{q \in Q} \text{img}(q, R)$
- $M, q \models \varphi$ if $M, \{q\} \models \varphi$
Definition 2.7 (Semantics of CSL)

\( M, Q \models p \) iff \( p \in \pi(q) \) for every \( q \in Q \);

\( M, Q \models \neg \varphi \) iff not \( M, Q \models \varphi \);

\( M, Q \models \varphi \land \psi \) iff \( M, Q \models \varphi \) and \( M, Q \models \psi \);

\( M, Q \models \langle A \rangle \gamma \) iff there exists \( S_A \) such that, for every \( \lambda \in out(Q, S_A) \), we have that \( M, \lambda[1] \models \varphi \).
2. Reasoning about Coalitions

$M, Q \models K_A \varphi$ iff $M, q \models \varphi$ for every $q \in \text{img}(Q, \sim_K)$ (where $K = C, E, D$);

$M, Q \models \hat{K}_A \varphi$ iff $M, \text{img}(Q, \sim_{\hat{K}}) \models \varphi$ (where $\hat{K} = C, E, D$ and $K = C, E, D$, respectively).
Example: Simple Market

@ q₁ :

¬ KC[c] ◊ success

¬ E{1,2}[c] ◊ success

¬ K₁[c] ◊ success

¬ K₂[c] ◊ success

D{1,2}[c] ◊ success
Theorem 2.8 (Expressivity)

CSL is \textit{strictly more expressive} than most previous proposals.

Theorem 2.9 (Verification complexity)

The complexity of model checking CSL is \textit{minimal}.
2.5 Model Checking
Model Checking Formulae of CTL and ATL

- **Model checking:** Does $\varphi$ hold in model $M$ and state $q$?
- Natural for verification of existing systems; also during design ("prototyping")
- Can be used for automated planning
2. Reasoning about Coalitions

function plan(ϕ).

Returns a subset of St for which formula ϕ holds, together with a (conditional) plan to achieve ϕ. The plan is sought within the context of concurrent game structure $S = \langle \text{Agt}, St, \Pi, \pi, o \rangle$.

case $\varphi \in \Pi$ : return $\{\langle q, \varphi \rangle \mid \varphi \in \pi(q)\}$
case $\varphi = \neg \psi$ : $P_1 := \text{plan}(\psi)$;
  return $\{\langle q, \varphi \rangle \mid q \notin \text{states}(P_1)\}$
case $\varphi = \psi_1 \lor \psi_2$ :
  $P_1 := \text{plan}(\psi_1)$; $P_2 := \text{plan}(\psi_2)$;
  return $\{\langle q, \varphi \rangle \mid q \in \text{states}(P_1) \cup \text{states}(P_2)\}$
case $\varphi = \langle \langle A \rangle \rangle \circ \psi$ : return pre(A, states(plan(ψ)))
case $\varphi = \langle \langle A \rangle \rangle \Box \psi$ :
  $P_1 := \text{plan}(\text{true})$; $P_2 := \text{plan}(\psi)$; $Q_3 := \text{states}(P_2)$;
  while $\text{states}(P_1) \not\subseteq \text{states}(P_2)$
    do $P_1 := P_2 |_{\text{states}(P_1)}$; $P_2 := \text{pre}(A, \text{states}(P_1)) |_{Q_3}$ od;
  return $P_2 |_{\text{states}(P_1)}$
case $\varphi = \langle \langle A \rangle \rangle \psi_1 U \psi_2$ :
  $P_1 := \emptyset$; $Q_3 := \text{states}(\text{plan}(\psi_1))$; $P_2 := \text{plan}(\text{true}) |_{\text{states}(\text{plan}(\psi_2))}$;
  while $\text{states}(P_2) \not\subseteq \text{states}(P_1)$
    do $P_1 := P_1 \oplus P_2$; $P_2 := \text{pre}(A, \text{states}(P_1)) |_{Q_3}$ od;
  return $P_1$
Complexity od Model Checking ATL

Theorem (Alur, Kupferman & Henzinger 1998)

ATL model checking is $P$-complete, and can be done in time linear in the size of the model and the length of the formula.

So, let’s model-check!

Not as easy as it seems.
Nice results: model checking ATL is tractable.
But: the result is relative to the size of the model and the formula
Well known catch: size of models is exponential wrt a higher-level description
Another problem: transitions are labeled
So: the number of transitions can be exponential in the number of agents.
3 agents/attributes, 12 states, 216 transitions
Model Checking Temporal & Strategic Logics

<table>
<thead>
<tr>
<th></th>
<th>$m, l$</th>
<th>$n, k, l$</th>
<th>$n_{\text{local}}, k, l$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CTL</strong></td>
<td>$\text{P} \ [1]$</td>
<td>$\text{P} \ [1]$</td>
<td>$\text{PSPACE} \ [2]$</td>
</tr>
<tr>
<td><strong>ATL</strong></td>
<td>$\text{P} \ [3]$</td>
<td>$\Delta^P_3 \ [5,6]$</td>
<td>$\text{EXPTIME} \ [8,9]$</td>
</tr>
<tr>
<td><strong>CSL</strong></td>
<td>$\Delta^P_2 \ [4,7]$</td>
<td>$\Delta^P_3 \ [7]$</td>
<td>$\text{PSPACE} \ [9]$</td>
</tr>
</tbody>
</table>

Main message:

- Complexity is very sensitive to the context!
- In particular, the way we define the input, and measure its size, is crucial.
Even if model checking appears very easy, it can be very hard.

Still, people do automatic model checking!

- LTL: SPIN
- CTL/ATL: MOCHA, MCMAS, VeriCS

Even if model checking is theoretically hard, it can be feasible in practice.
2.6 References
2. Reasoning about Coalitions


