



TU Clausthal

Clausthal University of Technology

Model Checking Rational Play

Wojciech Jamroga and Nils Bulling

IfI Technical Report Series

IfI-07-05



IfI



Department of Informatics
Clausthal University of Technology

Impressum

Publisher: Institut für Informatik, Technische Universität Clausthal
Julius-Albert Str. 4, 38678 Clausthal-Zellerfeld, Germany

Editor of the series: Jürgen Dix

Technical editor: Wojciech Jamroga

Contact: wjamroga@in.tu-clausthal.de

URL: <http://www.in.tu-clausthal.de/forschung/technical-reports/>

ISSN: 1860-8477

The IfI Review Board

Prof. Dr. Jürgen Dix (Theoretical Computer Science/Computational Intelligence)

Prof. Dr. Klaus Ecker (Applied Computer Science)

Prof. Dr. Barbara Hammer (Theoretical Foundations of Computer Science)

Prof. Dr. Kai Hormann (Computer Graphics)

Prof. Dr. Gerhard R. Joubert (Practical Computer Science)

apl. Prof. Dr. Günter Kemnitz (Hardware and Robotics)

Prof. Dr. Ingbert Kupka (Theoretical Computer Science)

Prof. Dr. Wilfried Lex (Mathematical Foundations of Computer Science)

Prof. Dr. Jörg Müller (Business Information Technology)

Prof. Dr. Niels Pinkwart (Business Information Technology)

Prof. Dr. Andreas Rausch (Software Systems Engineering)

apl. Prof. Dr. Matthias Reuter (Modeling and Simulation)

Prof. Dr. Harald Richter (Technical Computer Science)

Prof. Dr. Gabriel Zachmann (Computer Graphics)

Model Checking Rational Play

Wojciech Jamroga and Nils Bulling

Department of Informatics, Clausthal University of Technology, Germany
{wjamroga,bulling}@in.tu-clausthal.de

Abstract

We show that the problem of model checking “**ATL** with Plausibility” is Δ_3^P -complete. We consider two variants of the logic: one with abstract terms describing plausibility sets, and another one where plausibility assumptions are imposed through formulae of **ATLI** [19]. In both cases, the complexity results are the same.

1 Introduction

Alternating-time temporal logic (ATL) [1, 2] is a temporal logic that incorporates some basic game-theoretical notions. In [15], we extended **ATL** with a notion of *plausibility*, which can be used to model and reason about what agents can plausibly achieve. Our intuition was to use game-theoretical solution concepts (like Nash equilibrium, Pareto optimality, dominant strategies etc.) to define what it means to play rationally, and then to assume it plausible that agents behave in a rational way. Technically, some strategies (or rather *strategy profiles*) were assumed plausible in a given model, and one could reason about what can happen if only the plausible profiles are used.

The formulation of *alternating-time temporal logic with plausibility (ATLP)* from [15] was rather abstract, with unstructured terms used to address various rationality assumptions, and their denotation “hard-wired” in the model. In [16], we proposed to refine the language of terms so that it would allow to specify sets of rational strategy profiles in the object language. The idea was to build the terms on formulae of **ATLI** (*ATL with intentions*, [19]), as these can be used to describe sets of strategies and strategy profiles.

This technical report complements [16] by giving a more detailed account of the model checking complexity for the resulting logic.

2 Preliminaries

In this section, we summarize some modal logics for reasoning about agents in game-like scenarios: first, the basic logic of **ATL** [1, 2]; then, its two extensions **ATLP** [15] and **ATLI** [19].

2.1 Alternating-time Temporal Logic

Alternating-time temporal logic (ATL) [1, 2] enables reasoning about temporal properties and strategic abilities of agents. Formally, the language of **ATL** is given as follows.

Definition 1 (\mathcal{L}_{ATL} [1, 2]) Let $\mathbb{A}gt = \{1, \dots, k\}$ be a nonempty finite set of all agents, and Π be a set of propositions (with typical element p). We will use symbol a to denote a typical agent, and A to denote a typical group of agents from $\mathbb{A}gt$. The logic $\mathcal{L}_{ATL}(\mathbb{A}gt, \Pi)$ is defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\langle A \rangle\rangle \bigcirc \varphi \mid \langle\langle A \rangle\rangle \square \varphi \mid \langle\langle A \rangle\rangle \varphi \mathcal{U} \varphi.$$

Informally, $\langle\langle A \rangle\rangle \varphi$ says that agents A have a collective strategy to enforce φ . **ATL** formulae include the usual temporal operators: \bigcirc (“in the next state”), \square (“always from now on”) and \mathcal{U} (strict “until”). Additionally, \diamond (“now or sometime in the future”) can be defined as $\diamond\varphi \equiv \top \mathcal{U} \varphi$. It should be noted that the path quantifiers A, E of computation tree logic **CTL** [8] can be expressed in **ATL** with $\langle\langle \emptyset \rangle\rangle, \langle\langle \mathbb{A}gt \rangle\rangle$ respectively. The semantics of **ATL** is defined in so-called *concurrent game structures*.

Definition 2 (CGS [2]) A concurrent game structure (**CGS**) is a tuple: $M = \langle \mathbb{A}gt, St, \Pi, \pi, Act, d, o \rangle$, consisting of: a set $\mathbb{A}gt = \{1, \dots, k\}$ of agents; a nonempty set St of states; set Π of atomic propositions; valuation of propositions $\pi : St \rightarrow \mathcal{P}(\Pi)$; set Act of actions. Function $d : \mathbb{A}gt \times St \rightarrow \mathcal{P}(Act)$ indicates the actions available to agent $a \in \mathbb{A}gt$ in state $q \in St$; it is required that $d(a, q)$ is nonempty for every a, q . We will often write $d_a(q)$ instead of $d(a, q)$, and use $d(q)$ to denote the set $d_1(q) \times \dots \times d_k(q)$ of action profiles in state q . Finally, o is a transition function which maps each state $q \in St$ and action profile $\vec{\alpha} = \langle \alpha_1, \dots, \alpha_k \rangle \in d(q)$ to another state $q' = o(q, \vec{\alpha})$.

A *computation* or *path* $\lambda = q_0 q_1 \dots \in St^+$ is an infinite sequence of states such that there is a transition between each q_i, q_{i+1} . We define $\lambda[i] = q_i$ to denote the i -th state of λ . Λ_M denotes all paths in M . The set of all paths starting in q is given by $\Lambda_M(q)$.

Definition 3 (Strategy, outcome [1, 2]) A (memoryless) strategy of agent a is a function $s_a : St \rightarrow Act$ such that $s_a(q) \in d_a(q)$.¹ We denote the set of such functions by Σ_a . A collective strategy s_A for team $A \subseteq \text{Agt}$ specifies an individual strategy for each agent in A ; the set of A 's collective strategies is given by $\Sigma_A = \prod_{a \in A} \Sigma_a$. The set of all strategy profiles is given by $\Sigma = \Sigma_{\text{Agt}}$.

The outcome of strategy s_A in state q is defined as the set of all paths that may result from executing s_A from state q on: $out(q, s_A) = \{\lambda \in \Lambda_M(q) \mid \forall i \in \mathbb{N}_0 \exists \vec{\alpha} = \langle \alpha_1, \dots, \alpha_k \rangle \in d(\lambda[i]) \forall a \in A (\alpha_a = s_A^a(\lambda[i]) \wedge o(\lambda[i], \vec{\alpha}) = \lambda[i+1])\}$, where s_A^a denotes agent a 's part of the collective strategy s_A .

The semantics of **ATL** is given by the following clauses:

$$M, q \models p \text{ iff } p \in \pi(q)$$

$$M, q \models \neg\varphi \text{ iff } M, q \not\models \varphi$$

$$M, q \models \varphi \wedge \psi \text{ iff } M, q \models \varphi \text{ and } M, q \models \psi$$

$$M, q \models \langle\langle A \rangle\rangle \bigcirc \varphi \text{ iff there is } s_A \in \Sigma_A \text{ such that } M, \lambda[1] \models \varphi \text{ for all } \lambda \in out(q, s_A)$$

$$M, q \models \langle\langle A \rangle\rangle \square \varphi \text{ iff there is } s_A \in \Sigma_A \text{ such that } M, \lambda[i] \models \varphi \text{ for all } \lambda \in out(q, s_A) \text{ and } i \in \mathbb{N}_0$$

$$M, q \models \langle\langle A \rangle\rangle \varphi \mathcal{U} \psi \text{ iff there is } s_A \in \Sigma_A \text{ such that, for all } \lambda \in out(q, s_A), \text{ there is } i \in \mathbb{N}_0 \text{ with } M, \lambda[i] \models \psi, \text{ and } M, \lambda[j] \models \varphi \text{ for all } 0 \leq j < i.$$

2.2 ATL with Plausibility: Reasoning about Rational Agents

Agents usually have limited ability to predict the future. However, some lines of action seem often more sensible or realistic than others. Having defined a rationality criterion, we obtain means to determine the most plausible plays, and compute their outcome. In [15], we proposed an extension of **ATL** for reasoning about rational agents, which had in turn been inspired by the work by Van Otterloo and colleagues [31, 29, 30] and the research on social laws [24, 22, 27]. We called the logic **ATLP**, i.e., “**ATL** with plausibility”.

Definition 4 ($\mathcal{L}_{\text{ATLP}}$ [15]) Let Agt, Π be as before, and Ω be a set of plausibility terms (with typical element ω). The language $\mathcal{L}_{\text{ATLP}}(\text{Agt}, \Pi, \Omega)$ is defined recursively as:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\langle A \rangle\rangle \bigcirc \varphi \mid \langle\langle A \rangle\rangle \square \varphi \mid \langle\langle A \rangle\rangle \varphi \mathcal{U} \varphi \mid \mathbf{Pl} \varphi \mid \mathbf{Ph} \varphi \mid (\mathbf{set-pl} \ \omega) \varphi.$$

¹ This is a deviation from the original semantics of **ATL** [1, 2], where strategies assign agents' choices to *sequences* of states (which suggests that agents can recall the whole history of each game). While the choice between the two types of strategies affects the semantics of most **ATL** extensions, both yield equivalent semantics for “pure” **ATL** [23].

PI restricts the considered strategy profiles to ones that are *plausible* in the given model. **Ph** disregards plausibility assumptions, and refers to all *physically* available strategies. (**set-pl** ω) allows to define (or redefine) the set of plausible strategy profiles to the ones described by plausibility term ω (in this sense, it implements *revision* of plausibility). With **ATLP**, we can for example say that $\mathbf{PI} \langle \emptyset \rangle \square (\text{closed} \wedge \mathbf{Ph} \langle \text{guard} \rangle \bigcirc \neg \text{closed})$: “it is plausible that the emergency door will always remain closed, but the guard retains the physical ability to open them”; or (**set-pl** ω_{NE}) $\mathbf{PI} \langle a \rangle \diamond \neg \text{jail}_a$: “suppose that only playing Nash equilibria is rational; then, agent a can plausibly reach a state where he is out of prison”. To define the semantics of **ATLP**, we extend **CGS** to *concurrent game structures with plausibility* (**CGSP**). Apart from an actual plausibility set Υ , a **CGSP** specifies a *plausibility mapping* $[\cdot] : St \rightarrow (\Omega \rightarrow \mathcal{P}(\Sigma))$ that maps each term $\omega \in \Omega$ to a set of strategy profiles, depending on the current state.

Definition 5 (CGSP [15]) A concurrent game structure with plausibility (**CGSP**) is given by a tuple

$$M = \langle \mathbb{A}gt, St, \Pi, \pi, Act, d, o, \Upsilon, \Omega, [\cdot] \rangle$$

where $\langle \mathbb{A}gt, St, \Pi, \pi, Act, d, o \rangle$ is a **CGS**, $\Upsilon \subseteq \Sigma$ is a set of plausible strategy profiles; Ω is a set of plausibility terms, and $[\cdot]$ is a plausibility mapping.

When talking about the outcome of rational/plausible play (e.g., with formula $\mathbf{PI} \langle A \rangle \gamma$), the strategy profiles that can be used *by all the agents* are restricted to the ones from Υ . Thus, coalition A can only choose strategies that are *substrategies* of plausible strategy profiles. Moreover, the agents in $\mathbb{A}gt \setminus A$ can only respond in a way that yields a plausible strategy profile.

Definition 6 (Substrategy, outcome [15]) Let $A \subseteq B \subseteq \mathbb{A}gt$, and let s_B be a collective strategy for B . We use $s_B[A]$ to denote the substrategy of s_B for agents A , i.e., strategy t_A such that $t_A^a = s_B^a$ for every $a \in A$. Additionally, for a set of strategy profiles P , $P(s_A)$ denotes all strategy profiles from P that contain s_A as substrategy (i.e., $P(s_A) = \{s' \in P \mid s'[A] = s_A\}$).

Let M be a **CGSP**, $A \subseteq \mathbb{A}gt$ be a set of agents, $q \in St$ be a state, $s_A \in \Sigma_A$ be a collective strategy of A , and $P \subseteq \Sigma$ be a set of strategy profiles. The set $out(q, s_A, P)$ contains all paths which may result from agents A executing s_A , when only strategy profiles from P can be played. Formally: $out(q, s_A, P) = \{\lambda \in \Lambda_M(q) \mid \exists z \in P(s_A) \forall i (\lambda[i+1] = o(\lambda[i], z(\lambda[i])))\}$. Furthermore, $\Sigma_A(P)$ denotes all A 's strategies consistent with P , i.e., $\Sigma_A(P) = \{s_A \in \Sigma_A \mid \exists t \in P \ s_A = t[A]\}$.

Let $P \subseteq \Sigma_{\mathbb{A}gt}$ be a set of strategy profiles. The semantics of **ATLP** is given by the satisfaction relation \models_P defined as follows:

$$M, q \models_P p \text{ iff } p \in \pi(q)$$

$$\begin{aligned}
M, q \models_P \neg\varphi & \text{ iff } M, q \not\models_P \varphi \\
M, q \models_P \varphi \wedge \psi & \text{ iff } M, q \models_P \varphi \text{ and } M, q \models_P \psi \\
M, q \models_P \langle\langle A \rangle\rangle \bigcirc \varphi & \text{ iff there is } s_A \in \Sigma_A(P) \text{ with } M, \lambda[1] \models_P \varphi \text{ for all } \lambda \in \text{out}(q, s_A, P) \\
M, q \models_P \langle\langle A \rangle\rangle \square \varphi & \text{ iff there is } s_A \in \Sigma_A(P) \text{ such that } M, \lambda[i] \models_P \varphi \text{ for all } \lambda \in \\
& \text{out}(q, s_A, P) \text{ and all } i \in \mathbb{N}_0 \\
M, q \models_P \langle\langle A \rangle\rangle \varphi \mathcal{U} \psi & \text{ iff there is } s_A \in \Sigma_A(P) \text{ such that, for all } \lambda \in \text{out}(q, s_A, P), \\
& \text{there is } i \in \mathbb{N}_0 \text{ with } M, \lambda[i] \models_P \psi, \text{ and } M, \lambda[j] \models_P \varphi \text{ for all } 0 \leq j < i \\
M, q \models_P \mathbf{Pl} \varphi & \text{ iff } M, q \models_{\Upsilon} \varphi \\
M, q \models_P \mathbf{Ph} \varphi & \text{ iff } (M, q) \models \varphi \\
M, q \models_P (\mathbf{set-pl}) \varphi & \text{ iff } M^\omega, q \models_P \varphi \text{ where the new model } M^\omega \text{ is equal to } M \\
& \text{but the new set } \Upsilon^\omega \text{ of plausible strategy profiles is set to } \llbracket \omega \rrbracket_q.
\end{aligned}$$

The “absolute” satisfaction relation \models is given by \models_Σ . Note that an ordinary concurrent game structure (without plausibility) can be interpreted as a **CGSP** with all strategy profiles assumed plausible, i.e., with $\Upsilon = \Sigma$. In this way satisfaction of **ATLP** formulae can be extended to ordinary **CGS**.

2.3 ATL with Intentions

ATLI (“**ATL** with Intentions”) [19] allows to characterize some solution concepts for extensive games (and concurrent game structures).² In [16], we proposed to use **ATLI** for specification of plausibility assumptions. Here, we briefly recall the main ideas behind **ATLI**.

ATLI [19] extends **ATL** with formulae $(\text{str}_a \sigma_a) \varphi$ with the intuitive reading: “suppose that player a intends to play according to strategy σ_a , then φ holds”. Thus, it allows to refer to agents’ strategies explicitly via strategic terms $\sigma_a \in \mathfrak{Str}_a$. We assume that all \mathfrak{Str}_a are pairwise disjoint. The set of all strategic terms is denoted by $\mathfrak{Str} = \bigcup_{a \in \text{Agt}} \mathfrak{Str}_a$.

Definition 7 ($\mathcal{L}_{\text{ATLI}}$ [19]) *The language $\mathcal{L}_{\text{ATLI}}(\text{Agt}, \Pi, \mathfrak{Str})$ is defined as follows:*

$$\theta ::= p \mid \neg\theta \mid \theta \wedge \theta \mid \langle\langle A \rangle\rangle \bigcirc \theta \mid \langle\langle A \rangle\rangle \square \theta \mid \langle\langle A \rangle\rangle \theta \mathcal{U} \theta \mid (\text{str}_a \sigma_a) \theta.$$

Models of **ATLI** $M = \langle \text{Agt}, St, \Pi, \pi, Act, d, o, \mathcal{I}, \mathfrak{Str}, \|\cdot\| \rangle$ extend concurrent game structures with intention relations $\mathcal{I} \subseteq St \times \text{Agt} \times Act$ (with $q \mathcal{I}_a \alpha$ meaning that a possibly intends to do action α when in q). Moreover, strategic terms are interpreted as strategies according to function $\|\cdot\| : \mathfrak{Str} \rightarrow \bigcup_{a \in \text{Agt}} \Sigma_a$ such that $\|\sigma_a\| \in \Sigma_a$ for $\sigma_a \in \mathfrak{Str}_a$. The set of paths consistent with

² For some previous work on modal characterizations of solution concepts cf. [13, 12, 3, 25, 26].

all agents' intentions is defined as $\Lambda^{\mathcal{I}} = \{\lambda \in \Lambda_M \mid \forall i \exists \alpha \in d(\lambda[i]) (o(\lambda[i], \alpha) = \lambda[i+1] \wedge \forall a \in \mathbb{A}gt \lambda[i]\mathcal{I}_a\alpha_a)\}$. We say that strategy s_A is *consistent with A's intentions* if $q\mathcal{I}_a s_A[a](q)$ for all $q \in St, a \in A$. The *intention-consistent outcome set* is defined as: $out^{\mathcal{I}}(q, s_A) = out(q, s_A) \cap \Lambda^{\mathcal{I}}$. The semantics of strategic operators in **ATLI** is given as follows:

$M, q \models \langle\langle A \rangle\rangle \bigcirc \theta$ iff there is a collective strategy s_A consistent with A 's intentions, such that for every $\lambda \in out^{\mathcal{I}}(q, s_A)$, we have that $M, \lambda[1] \models \theta$;

$M, q \models \langle\langle A \rangle\rangle \square \theta$ and $M, q \models \langle\langle A \rangle\rangle \theta \mathcal{U} \theta'$: analogous;

$M, q \models (\mathbf{str}_a \sigma) \theta$ iff $revise(M, a, \|\sigma\|), q \models \theta$;

Function $revise(M, a, s_a)$ updates model M by setting a 's intention relation to $\mathcal{I}'_a = \{\langle q, s_a(q) \rangle \mid q \in St\}$, so that s_a and \mathcal{I}_a represent the same mapping in the resulting model. Note that a "pure" **CGS** M can be seen as a **CGS** with the "full" intention relation $\mathcal{I}^0 = \{\langle q, a, \alpha \rangle \mid q \in St, a \in \mathbb{A}gt, \alpha \in d_a(q)\}$.

Additionally, for $A = \{a_1, \dots, a_r\}$ and $\sigma_A = \langle \sigma_1, \dots, \sigma_r \rangle$, we define: $(\mathbf{str}_{A\sigma_A})\varphi \equiv (\mathbf{str}_{a_1}\sigma_1) \dots (\mathbf{str}_{a_r}\sigma_r)\varphi$.

2.4 ATLI-Based Plausibility Terms

Ideally, one would like to have a *flexible language of terms* that would allow to specify any sensible rationality assumption, and then impose it on the system. Our idea is to use **ATLI** formulae θ to specify sets of plausible strategy profiles, with the presumed meaning that Υ collects exactly the profiles for which θ holds. Then, we can embed such **ATLI**-based plausibility specifications in formulae of **ATLP** in order to reason about rational agents. We call the resulting language **ATLP**^[ATLI].

Definition 8 ($\mathcal{L}_{\mathbf{ATLP}[\mathbf{ATLI}]}$) Let $\Omega^* = \{(\sigma.\theta) \mid \theta \in \mathcal{L}_{\mathbf{ATLI}}(\mathbb{A}gt, \Pi, \{\sigma[1], \dots, \sigma[k]\})\}$. That is, Ω^* collects terms of the form $(\sigma.\theta)$, where θ is an **ATLI** formula including only references to individual agents' parts of the strategy profile σ . The language of **ATLP**^[ATLI] is defined as $\mathcal{L}_{\mathbf{ATLP}[\mathbf{ATLI}]}(\mathbb{A}gt, \Pi, \Omega^*)$.

The idea behind terms of this form is simple. We have an **ATLI** formula θ , parameterized with a variable σ that ranges over the set of strategy profiles Σ . Now, we want $(\sigma.\theta)$ to denote exactly the set of profiles from Σ , for which formula θ holds. However – as σ denotes a strategy profile, and **ATLI** allows only to refer to strategies of individual agents – we need a way of addressing substrategies of σ in θ . This can be done by using **ATLI** terms $\sigma[i]$, which will be interpreted as the i 's substrategy in σ . Below, we define the concept formally.

Definition 9 (**CGSP for** $\mathcal{L}_{\mathbf{ATLP}[\mathbf{ATLI}]}$) Let $\langle \mathbb{A}gt, St, \Pi, \pi, Act, d, o \rangle$ be a **CGS**, and let $\Upsilon \subseteq \Sigma$ be a set of plausible strategy profiles. $M = \langle \mathbb{A}gt, St, \Pi, \pi, Act, d, o, \Upsilon, \Omega^*, \llbracket \cdot \rrbracket \rangle$

is a **CGS** with plausibility iff the denotation $\llbracket \cdot \rrbracket$ of terms from Ω^* is defined as follows.

First, we define a family of **ATLI** models $M^s = \langle \text{Agt}, St, \Pi, \pi, Act, d, o, \mathcal{I}^0, \text{Str}, \|\cdot\| \rangle$, one for each strategy profile $s \in \Sigma$, with $\text{Str}_a = \{\sigma[a]\}$, and $\|\sigma[a]\| = s[a]$. Then, we define the plausibility mapping as:

$$\llbracket \sigma.\theta \rrbracket_q = \{s \in \Sigma \mid M^s, q \models \theta\}.$$

For example, we may assume that rational agents do not grant the other agents with too much control over their lives:

$$(\sigma \cdot \bigwedge_{a \in \text{Agt}} (\text{str}_a \sigma[a]) \neg \langle \langle \text{Agt} \setminus \{a\} \rangle \rangle \diamond \text{dead}_a).$$

Note that games defined by **CGS** are, in general, not determined, so the above specification does not guarantee that each rational agent can efficiently protect his life. It only requires that he should behave cautiously so that his opponents do not have complete power to kill him.

3 Model Checking ATLP and ATLP^[ATLI]

In this section we show that model checking **ATLP** is Δ_3^P -complete, which seems in line with existing results on the complexity of solving games. It is well known that determining the existence of a solution concept instance with certain natural properties (e.g., a Nash equilibrium with expected utility of at least k , or a Pareto-optimal Nash equilibrium) is NP-hard even for normal form (i.e., one-step) games in the setting of mixed strategies [10, 7]. Similar results are known for extensive turn-based games with imperfect information and recall [9, 20, 5]. Formally, mixed strategies and imperfect information are absent in **ATLP**. However, the framework turns out to be quite powerful in terms of expressiveness. In particular, imperfect information strategies (sometimes called *uniform* strategies) can be characterized in **ATLP** for a relevant subclass of models, and checking strategic properties of systems in which *all* agents must play uniform strategies is Δ_3^P -complete – which renders **ATLP** model checking also Δ_3^P -complete. This coincides with another result from game theory: if both players in a 2-player imperfect information game have imperfect recall, and chance moves are allowed, then the problem of finding a max-min pure strategy is Σ_2^P -complete [20].³

We mainly consider checking formulae of **ATLP** against “pure” concurrent game structures (i.e., we assume that plausibility assumptions will be specified explicitly in the formula), although we briefly show, too, that the

³ Note that strategic operators can be nested in an **ATLP** formula, thus specifying a sequence of games, with the outcome of each game depending on the previous ones – and solving such games requires adaptive calls to a Σ_2^P oracle.

<p>function $mcheck(M, q, \varphi, \theta_1, q_1, \theta_2, q_2)$; Returns “true” iff φ holds in M, q. The current plausibility assumptions are specified by the truth of the ATLI formula θ_1 at state q_1. The most recent plausibility specification (not necessarily incorporated into the definition of the current plausibility set Υ yet) corresponds to the truth of θ_2 at q_2.</p>
<p>cases $\varphi \equiv p, \varphi \equiv \neg\psi, \varphi \equiv \psi_1 \wedge \psi_2$: proceed as usual; case $\varphi \equiv (\mathbf{set-pl} \ \sigma.\theta')\psi$: return($mcheck(M, q, \psi, \theta_1, q_1, \theta', q)$); case $\varphi \equiv \mathbf{Pl} \ \psi$: return($mcheck(M, q, \psi, \theta_2, q_2, \theta_2, q_2)$); case $\varphi \equiv \mathbf{Ph} \ \psi$: return($mcheck(M, q, \psi, \top, q_1, \theta_2, q_2)$); case $\varphi \equiv \langle\langle A \rangle\rangle \circ \psi$, where ψ includes some $\langle\langle B \rangle\rangle$: Label all $q' \in St$, in which $mcheck(M, q, \psi, \theta_1, q_1, \theta_2, q_2)$ returns “true”, with a new proposition yes. Return $mcheck(M, q, \langle\langle A \rangle\rangle \circ \text{yes}, \theta_1, q_1, \theta_2, q_2)$; case $\varphi \equiv \langle\langle A \rangle\rangle \circ \psi$, where ψ includes no $\langle\langle B \rangle\rangle$: Remove all operators Pl, Ph, (set-pl \cdot) from ψ (they are irrelevant, as no cooperation modality comes further), yielding ψ'. Return $solve(M, q, \langle\langle A \rangle\rangle \circ \psi', \theta_1, q_1)$; cases $\langle\langle A \rangle\rangle \square \psi$ and $\langle\langle A \rangle\rangle \psi_1 \mathcal{U} \psi_2$: analogously; end case</p>

 Figure 1: Model checking **ATLP**: main function

results carry over to model checking against **CGS** with plausibility. The size of the input is measured with the number of transitions in the model (m) and the length of the formula (l). Note that the problem of checking **ATLP** with respect to the size of the *whole* **CGSP** (including the plausibility set Υ), is trivially linear in the size of the model – but the model size is exponential with respect to the number of states and transitions.

3.1 Model Checking **ATLP**^[ATLI] is in Δ_3^P

First, we consider the upper bound for complexity of model checking **ATLP** with plausibility terms expressed in **ATLI**. A detailed algorithm for model checking **ATLP**^[ATLI] formulae against concurrent game structures is presented in Figures 1 and 2. Apart from the model, the state, and the formula to be checked, the input includes *two* plausibility specifications (each represented by an **ATLI** formula and a state at which it should be evaluated). The first specification describes the current set of plausible strategy profiles Υ . The latter is the argument of the most recent (**set-pl** \cdot) operation, not necessarily incorporated into the definition of Υ yet – unless the **Pl** operator has been used since. As both **CTL** and **ATLI** model checking is linear in the number of transitions in the model and the length of the formula [6, 19], we get the following.

Proposition 1 $M, q \models \varphi$ iff $mcheck(M, q, \varphi, \top, q, \top, q)$. The algorithm runs in

<p>function <i>solve</i>($M, q, \varphi, \theta, q'$); Returns “true” iff φ holds in M, q under plausibility assumptions specified by the truth of θ at q'. We assume that $\varphi \equiv \langle\langle A \rangle\rangle \Box \psi$, where ψ is a propositional formula, i.e., it includes no $\langle\langle B \rangle\rangle, \mathbf{Pl}, \mathbf{Ph}, (\mathbf{set-pl} \cdot)$.</p> <ul style="list-style-type: none"> ■ Label all $q' \in St$, in which ψ holds, with a new proposition yes; ■ Guess a strategy profile s; ■ if <i>plausiblestrat</i>(s, M, q', θ) then return(not <i>beatable</i>($s[A], M, q, \langle\langle A \rangle\rangle \Box \text{yes}$)); else return(false);
<p>function <i>beatable</i>($s_A, M, q, \langle\langle A \rangle\rangle \gamma, q', \theta$); Returns “true” iff the opponents can beat s_A so that it does not enforce γ in M, q under plausibility assumptions specified by the ATLI formula θ at q'. The path formula γ is of the form $\bigcirc \psi, \Box \psi, \psi \mathcal{U} \psi'$ with propositional ψ, ψ'.</p> <ul style="list-style-type: none"> ■ Guess a strategy profile t; ■ if <i>plausiblestrat</i>(t, M, q', θ) and $t[A] = s_A$ then <ul style="list-style-type: none"> – $M' :=$ “trim” M, removing all transitions that cannot occur when t is executed; – return(<i>mcheck</i>_{CTL}($M', q, A\gamma$)); else return(false);
<p>function <i>plausiblestrat</i>(s, M, q, θ); Checks if strategy profile s satisfies formula θ in M, q.</p> <ul style="list-style-type: none"> ■ return(<i>mcheck</i>_{ATLI}(M^s, q, θ)); // For M^s, cf. Definition 9

Figure 2: Model checking **ATLP**: guessing strategies and counterstrategies

time Δ_3^P with respect to the number of transitions in the model and the length of the formula.

3.2 Model Checking ATLP with Arbitrary Plausibility Terms

The algorithm in Figures 1 and 2 uses the **ATLI**-based plausibility terms presented in Section 2.4. In the general case, we can think of any arbitrary implementation of terms in Ω . As long as *plausiblestrat*(s, M, q, θ) can be computed in polynomial time, it does not affect the overall complexity of *mcheck*. In fact, it is enough to require that *plausiblestrat*(s, M, q, θ) can be computed in *nondeterministic* polynomial time, as the witness for *plausiblestrat* can be guessed together with the strategy profile s in function *solve*, and with the strategy profile t in function *beatable*, respectively.

Proposition 2 *If the verification of plausibility (plausiblestrat) is in NP, then the model checking algorithm (mcheck) is in Δ_3^P with respect to m, l .*

Note that, if a list (or several alternative lists) of plausible strategy profiles is given explicitly in the model (via the plausibility set Υ and/or the denotations of abstract plausibility terms ω from Section 2.2), then the problem of guessing an appropriate strategy from such a list is in NP (memoryless strategies have polynomial size with respect to m). As a consequence, we have the following:

Corollary 3 *Model checking **ATLP** (with both abstract and **ATLI**-based plausibility terms) against **CGSP** is in Δ_3^P with respect to m, l .*

3.3 Model Checking **ATLP** is Δ_3^P -hard

We prove the Δ_3^P -hardness through a reduction of **SNSAT**₂, the typical Δ_3^P -complete variant of the Boolean satisfiability problem. The reduction follows in two steps. First, we define a modification of **ATL**_{ir} [23], in which *all* agents are required to play only uniform strategies. We call it “uniform **ATL**_{ir}” (**ATL**_{ir}^u in short), and show a polynomial reduction of **SNSAT**₂ to **ATL**_{ir}^u model checking. Then, we point out how each formula and model of **ATL**_{ir}^u can be equivalently translated (in polynomial time) to a **CGS** and a formula of **ATLP**^[ATLI], thus yielding a polynomial reduction of **SNSAT**₂ to **ATLP**^[ATLI]. Again, we consider two cases: **ATLP** with arbitrary plausibility terms, and **ATLP** with terms defined through formulae of **ATLI**. The first part of the reduction (from **SNSAT**₂ to model checking **ATL**_{ir}^u) is the same in both cases, but the second part (from model checking **ATL**_{ir}^u to **ATLP**) proceeds differently, and we discuss both variants accordingly.

Readers interested in additional technical details are referred to [17, 18, 14, 11], where important parts of our construction are described.

3.3.1 Uniform **ATL**_{ir}

First, we introduce the logic of “uniform **ATL**_{ir}” (**ATL**_{ir}^u). The idea is based on Schobbens’s **ATL**_{ir} [23], i.e., **ATL** for agents with imperfect information and imperfect recall. There, it was assumed that the coalition A in formula $\langle\langle A \rangle\rangle_{ir} \varphi$ can only use strategies that assign same choices in indistinguishable states (so called *uniform* strategies). Then, the outcome of every strategy of A was evaluated against every possible behavior of the remaining agents $\mathbb{A}_{gt} \setminus A$ (with no additional assumption with respect to that behavior).

In **ATL**_{ir}^u, we assume that the opponents ($\mathbb{A}_{gt} \setminus A$) are also required to respond *with a uniform memoryless strategy*. The syntax of **ATL**_{ir}^u is the same as that of **ATL**, only cooperation modalities are annotated with additional tags *ir* and *u* to indicate the imperfect **i**nformation and **r**ecall, and **u**niformity of all agents’ strategies.

The semantics of \mathbf{ATL}_{ir}^u can be defined as follows. First, we define models as *concurrent epistemic game structures (CEGS)*, i.e. **CGS** with epistemic relations $\sim_a \subseteq St \times St$, one per agent. (The intended meaning of $q \sim_a q'$ is that agent a cannot distinguish between between states q and q' .) Additionally, we require that agents have the same options in indistinguishable states, i.e., that $q \sim_a q'$ implies $d_a(q) = d_a(q')$. A (memoryless) strategy s_A is *uniform* if $q \sim_a q'$ implies $s_A^a(q) = s_A^a(q')$ for all $q, q' \in St, a \in A$. To simplify the notation, we define $[q]_a = \{q' \mid q \sim_a q'\}$ to be the class of states indistinguishable from q for a ; $[q]_A = \bigcup_{a \in A} [q]_a$ collects all the states that are indistinguishable from q for some member of the group A ; finally, $out(Q, s_A) = \bigcup_{q \in Q} out(q, s_A)$ collects all the execution paths of strategy s_A from states in set Q .

Now, the semantics is given by the clauses below:

$$M, q \models p \quad \text{iff } p \in \pi(q)$$

$$M, q \models \neg \varphi \quad \text{iff } M, q \not\models \varphi$$

$$M, q \models \varphi \wedge \psi \quad \text{iff } M, q \models \varphi \text{ and } M, q \models \psi$$

$$M, q \models \langle\langle A \rangle\rangle_{ir}^u \bigcirc \varphi \quad \text{iff there is a uniform strategy } s_A \text{ such that, for every uniform counterstrategy } t_{\text{Agt} \setminus A}, \text{ and } \lambda \in out([q]_A, \langle s_A, t_{\text{Agt} \setminus A} \rangle),^4 \text{ we have } M, \lambda[1] \models \varphi;$$

$$M, q \models \langle\langle A \rangle\rangle_{ir}^u \square \varphi \quad \text{iff there is a uniform strategy } s_A \text{ such that, for every uniform counterstrategy } t_{\text{Agt} \setminus A}, \text{ and } \lambda \in out([q]_A, \langle s_A, t_{\text{Agt} \setminus A} \rangle), \text{ we have } M, \lambda[i] \models \varphi \text{ for all } i = 0, 1, \dots;$$

$$M, q \models \langle\langle A \rangle\rangle_{ir} \varphi \mathcal{U} \psi \quad \text{iff there is a uniform strategy } s_A \text{ such that, for every uniform counterstrategy } t_{\text{Agt} \setminus A}, \text{ and } \lambda \in out([q]_A, \langle s_A, t_{\text{Agt} \setminus A} \rangle), \text{ there is } i \in \mathbb{N}_0 \text{ with } M, \lambda[i] \models \psi, \text{ and } M, \lambda[j] \models \varphi \text{ for all } 0 \leq j < i.$$

3.3.2 Reduction of \mathbf{SNSAT}_2 to Model Checking of \mathbf{ATL}_{ir}^u

We recall the definition of \mathbf{SNSAT}_2 after [21].

Definition 10 (\mathbf{SNSAT}_2)

Input: p sets of propositional variables $X_r = \{x_{1,r}, \dots, x_{k,r}\}$, p sets of propositional variables $Y_r = \{y_{1,r}, \dots, y_{k,r}\}$, p propositional variables z_r , and p Boolean formulae φ_r in positive normal form (i.e., negation is allowed only on the level of literals). Each φ_r involves only variables in $X_r \cup Y_r \cup \{z_1, \dots, z_{r-1}\}$, with the following requirement: $z_r \equiv \exists X_r \forall Y_r. \varphi_r(z_1, \dots, z_{r-1}, X_r, Y_r)$.

Output: The value of z_p .

Note that every non-literal formula φ_r can be written as $\chi_1 \text{ op } \chi_2$ with $\text{op} \in \{\wedge, \vee\}$. Recursively, χ_i can be written as $\chi_{i1} \text{ op}_i \chi_{i2}$ and χ_{ij} as $\chi_{ij1} \text{ op}_{ij} \chi_{ij2}$ etc.

⁴ Note that the definition of concurrent game structures, that we use after [2], implies that **CGS** are deterministic, so there is in fact exactly one such path λ .

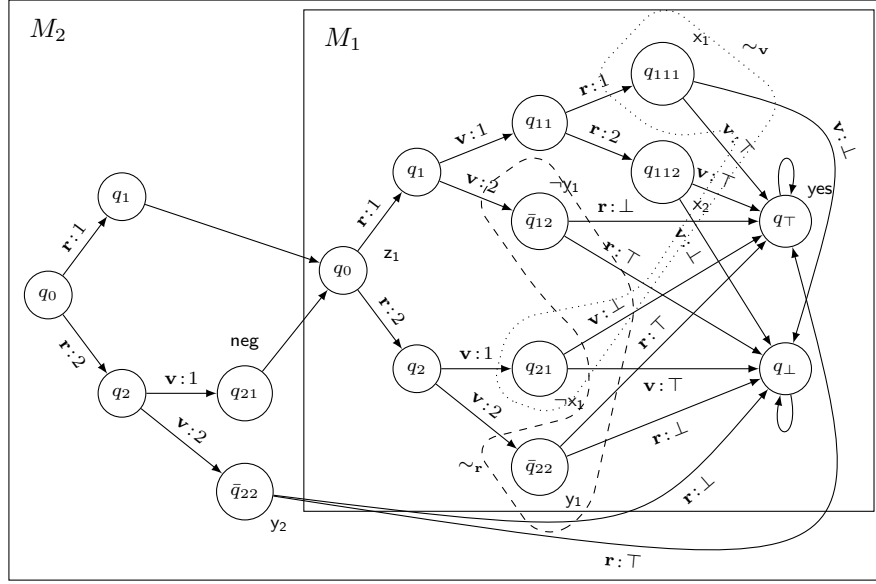


Figure 3: CEGS M_2 for $\varphi_1 \equiv ((x_1 \wedge x_2) \vee \neg y_1) \wedge (\neg x_1 \vee y_1)$, $\varphi_2 \equiv z_1 \wedge (\neg z_1 \vee y_2)$.

Our reduction of **SNSAT**₂ is an extension of the reduction of **SNSAT** presented in [17, 18]. That is, we construct the **CEGS** M_r corresponding to z_r with two players: *verifier* v and *refuter* r . The **CEGS** is turn-based, that is, every state is “governed” by a single player who determines the next transition. Each subformula $\chi_{i_1 \dots i_l}$ of φ_r has a corresponding state $q_{i_1 \dots i_l}$ in M_r . If the outermost logical connective of φ_r is \wedge , the refuter decides at q_0 which subformula χ_i of φ_r is to be satisfied, by proceeding to the “subformula” state q_i corresponding to χ_i . If the outermost connective is \vee , the verifier decides which subformula χ_i of φ_r will be attempted at q_0 . This procedure is repeated until all subformulae are single literals. The states corresponding to literals are called “proposition” states.

The difference from the construction from [17, 18] is that formulae are in positive normal form (rather than CNF) and that we have two kinds of “proposition” states now: $q_{i_1 \dots i_l}$ refers to a literal consisting of some $x \in X_r$ and is governed by v ; $\bar{q}_{i_1 \dots i_l}$ refers to some $y \in Y_r$ and will be governed by r . Now, the values of the underlying propositional variables x, y are declared at the “propositional” states, and the outcome is computed. That is, if v executes \top for a positive literal, i.e. $\chi_{i_1 \dots i_l} = x$, (or \perp for $\chi_{i_1 \dots i_l} = \neg x$) at $q_{i_1 \dots i_l}$, then the system proceeds to the “winning” state q_\top ; otherwise, the system goes to the “sink” state q_\perp . For states $\bar{q}_{i_1 \dots i_l}$ the procedure is analogous. Mod-

els corresponding to subsequent z_r are nested like in Figure 3.⁵ “Proposition” states referring to the same variable x are indistinguishable for \mathbf{v} (so that he has to declare the same value of x in all of them), and the states referring to the same y are indistinguishable for \mathbf{r} . A sole \mathbf{ATL}_{ir}^u proposition *yes* holds only in the “winning” state q_\top . As in [17, 18], we have the following result which concludes the reduction.

Proposition 4 *The above construction depicts a polynomial reduction of \mathbf{SNSAT}_2 to model checking \mathbf{ATL}_{ir}^u in the following sense. Let*

$$\begin{aligned}\Phi_1 &\equiv \langle\langle \mathbf{v} \rangle\rangle_{ir}^u (\text{-neg}) \mathcal{U} \text{yes}, \quad \text{and} \\ \Phi_r &\equiv \langle\langle \mathbf{v} \rangle\rangle_{ir}^u (\text{-neg}) \mathcal{U} (\text{yes} \vee (\text{neg} \wedge \langle\langle \emptyset \rangle\rangle_{ir}^u \bigcirc \neg \Phi_{r-1})) \quad \text{for } r = 2, \dots, p.\end{aligned}$$

Then, we have z_p iff $M_p, q_0^p \models_{\mathbf{ATL}_{ir}^u} \Phi_p$.

Note that there is a straightforward Δ_3^P algorithm that model-checks formulae of \mathbf{ATL}_{ir}^u : when checking $\langle\langle A \rangle\rangle_{ir}^u T\varphi$ in M, q , it first recursively checks φ (bottom-up), and labels the states where φ held with a special proposition *yes*. Then, the algorithm guesses a uniform strategy s_A and calls an oracle that guesses a uniform counterstrategy $t_{\text{Agt} \setminus A}$. Finally, it trims M according to $\langle s_A, t_{\text{Agt} \setminus A} \rangle$, and calls a **CTL** model checker to check formula $AT\text{yes}$ in state q of the resulting model. This gives us the following result.

Proposition 5 *Model checking \mathbf{ATL}_{ir}^u is Δ_3^P -complete with respect to the number of transitions in the model and the length of the formula. It is Δ_3^P -complete even for turn-based **CEGS** with at most two agents.*

3.3.3 From \mathbf{ATL}_{ir}^u to **ATLP** with Arbitrary Plausibility Terms

Now we show how \mathbf{ATL}_{ir}^u model checking can be reduced to model checking of **ATLP**. We are given a **CEGS** M , a state q in M , and an \mathbf{ATL}_{ir}^u formula φ . First, we sketch the reduction to model checking arbitrary **ATLP** formulae against **CGSP** (i.e., **CGS** with plausibility sets given explicitly in the model). Let Σ^u be the set of all uniform strategy profiles in M . We take **CGSP** M' as M (sans epistemic relations) extended with plausibility set $\Upsilon = \Sigma^u$. Then:

$$M, q \models_{\mathbf{ATL}_{ir}^u} \langle\langle A \rangle\rangle_{ir}^u \varphi \quad \text{iff} \quad M', q \models_{\mathbf{ATLP}} \text{Pl} \langle\langle A \rangle\rangle \varphi,$$

which completes the reduction.⁶

⁵ All states in the model for z_r are additionally indexed by r .

⁶ We note in passing that, technically, the size of the resulting model M' is not entirely polynomial. M' includes the plausibility set Υ , which is exponential in the number of states in M (since it is equal to the set of all uniform strategy profiles in M). This is of course the case when we want to store Υ explicitly. However, checking if a strategy profile is uniform can be done in time linear wrt the number of states in M , so an *implicit* representation of Υ (e.g., the checking procedure itself) requires only linear space.

We do not discuss this issue in more depth, as we focus on the other variant of **ATLP** (with **ATLI**-based terms) in this paper.

For model checking **ATLP** formulae with abstract terms ω against “pure” concurrent game structures, the reduction is similar. We take **CGS** M' as M minus epistemic relations, and plus a plausibility mapping $\llbracket \cdot \rrbracket$ such that $\llbracket \omega \rrbracket_q = \Sigma^u$. Then, again,

$$M, q \models_{\mathbf{ATL}_{ir}^u} \langle\langle A \rangle\rangle_{ir}^u \varphi \quad \text{iff} \quad M', q \models_{\mathbf{ATLP}} (\mathbf{set-pl} \ \omega) \mathbf{Pl} \langle\langle A \rangle\rangle \varphi.^7$$

3.3.4 From \mathbf{ATL}_{ir}^u to **ATLP** with **ATLI**-Based Plausibility Terms

The reduction of \mathbf{ATL}_{ir}^u model checking to model checking of **ATLP**^[ATLI] against “pure” **CGS** is more sophisticated. We do not present a reduction for full model checking of \mathbf{ATL}_{ir}^u ; it is enough to show the reduction for the kind of models that we get in Section 3.3.2 (i.e., turn-based models with two agents, two “final” states q_\top, q_\perp , no cycles except for the loops at the final states, and uncertainty appearing only in states one step before the end of the game).

First, we reconstruct the concurrent epistemic game structure M_p from Section 3.3.2 so that the last action profile is always “remembered” in the final states. Then, we show how uniformity of strategies can be characterized with a formula of **ATLI** extended with epistemic operators. Next, we show how the model and the formula can be transformed to get rid of epistemic links and operators (yielding a “pure” **CGS** and a formula of “pure” **ATLI**). Finally, we show how the resulting characterization of uniformity can be “plugged” into an **ATLP** formula to require that only uniform strategy profiles are taken into account.

Adding more final states to the model. To recall, the input of \mathbf{ATL}_{ir}^u model checking consists in our case of a concurrent epistemic game structure M_p (like the one in Figure 3) and an \mathbf{ATL}_{ir}^u formula Φ_p (cf. Proposition 4). We begin the reduction by reconstructing M_p to M'_p in which the last action profile is “remembered” in the final states. The idea is based on the construction from [11, Proposition 16] where it is applied to all states of the system, cf. Figure 4.

In our case, we first create copies of states q_\top, q_\perp , one per incoming transition. That is, the construction yields states of the form $\langle q, \alpha_1, \dots, \alpha_k \rangle$, where $q \in \{q_\top, q_\perp\}$ is a final state of the original model M_p , and $\langle \alpha_1, \dots, \alpha_k \rangle$ is the action profile executed just before the system proceeded to q . Each copy has the same valuation of propositions as the original state q , i.e., $\pi'(\langle q, \alpha_1, \dots, \alpha_k \rangle) = \pi(q)$. Then, for each action $\alpha \in Act$ and agent $i \in \mathbb{A}gt$, we add a new proposition $i : \alpha$. Moreover, we fix the valuation of $i : \alpha$ in M'_p so that it holds exactly

⁷ Cf. footnote 6.

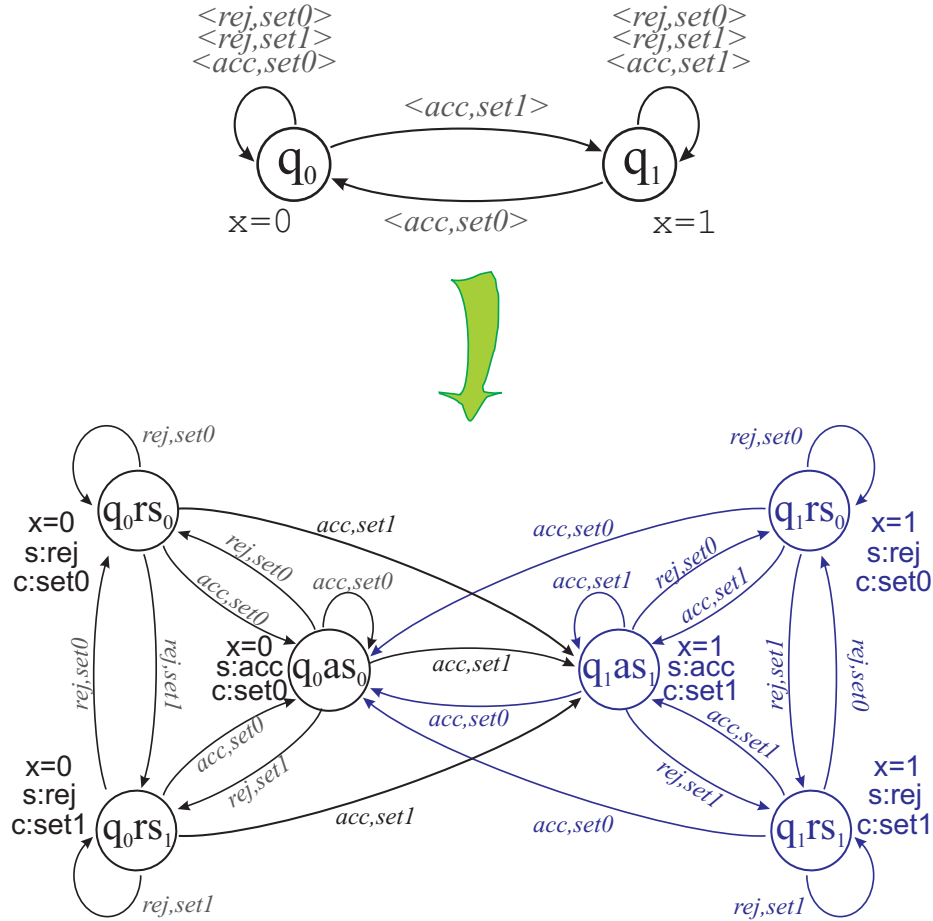


Figure 4: Memorizing the last action profile in a simple 2-agent system

in the final states that can be achieved by an action profile in which i executes α (i.e., states $\langle q, \alpha_1, \dots, \alpha_i, \dots, \alpha_k \rangle$). Note that the number of both states and transitions in M'_p is linear in the transitions of M_p .

The transformation produces model M'_p which is equivalent to M_p in the following sense: let φ be a formula of \mathbf{ATL}_{ir}^u that does not involve special propositions $i : \alpha$. Then, for all $q \in St$:

$$M_p, q \models_{\mathbf{ATL}_{ir}^u} \varphi \quad \text{iff} \quad M'_p, q \models_{\mathbf{ATL}_{ir}^u} \varphi.$$

In M'_p , agents can “recall” their actions executed at states that involved some uncertainty (i.e., states in which the image of some indistinguishabil-

ity relation \sim_i was not a singleton). Now we can use **ATLI** (with additional help of knowledge operators, see below) to characterize uniformity of strategies.

ATLI+Knowledge (ATLI+K) In the next step, we will show that uniformity of a strategy can be characterized in **ATLI** extended with epistemic operators K_a . $K_a\varphi$ reads as “agent a knows that φ ”. The semantics of **ATLI+K** extends that of **ATLI** by adding the standard semantic clause from epistemic logic:

$$M, q \models K_a\varphi \text{ iff } M, q' \models \varphi \text{ for every } q' \text{ such that } q \sim_a q'.$$

We note that **ATLI+K** can be also seen as **ATEL** [28] extended with intentions.

Characterizing uniformity in ATLI+K. Let us now consider the following formula of **ATLI+Knowledge**:

$$\text{uniform}(\sigma) \equiv (\text{str}\sigma)\langle\langle\emptyset\rangle\rangle\Box \bigwedge_{i \in \text{Agt}} \bigvee_{\alpha \in d(i,q)} K_i\langle\langle\emptyset\rangle\rangle\bigcirc i : \alpha.$$

The reading of $\text{uniform}(\sigma)$ is: suppose that profile σ is played ($\text{str}\sigma$); then, for all reachable states ($\langle\langle\emptyset\rangle\rangle\Box$), every agent has a single action ($\bigwedge_{i \in \text{Agt}} \bigvee_{\alpha \in d(i,q)}$) that is determined for execution ($\langle\langle\emptyset\rangle\rangle\bigcirc i : \alpha$) in every state indistinguishable from the current state (K_i). Thus, formula $\text{uniform}(\sigma)$ characterizes the *uniformity* of strategy profile σ . Formally, for every concurrent epistemic game structure M , we have that $M, q \models_{\text{ATLI+K}} \text{uniform}(\sigma)$ iff $\|\sigma[a]\|$ is uniform for each agent $a \in \text{Agt}$ (for all states reachable from q). Of course, only reachable states matter when we look for strategies that should enforce a temporal goal.

Note that the epistemic operator K_a refers to incomplete information, but σ is now an arbitrary (i.e., not necessarily uniform) strategy profile. We observe that the length of the formula is linear in the number of agents and actions in the model.

Translating Knowledge to Ability. To get rid of the epistemic operators from formula $\text{uniform}(\sigma)$ and epistemic relations from model M'_p , we use the construction from [14] (which refines that from [11, Section 4.4]). The construction yields a concurrent game structure $tr(M'_p)$ and an **ATLI** formula $tr(\text{uniform}(\sigma))$. The idea can be sketched as follows. The set of agents becomes extended with *epistemic agents* e_i (one per $a_i \in \text{Agt}$), yielding $\text{Agt}'' = \text{Agt} \cup \text{Agt}^e$. Similarly, the set of states is augmented with *epistemic states* q^e for every $q \in St'$ and $e \in \text{Agt}^e$; the states “governed” by the epistemic agent e_a

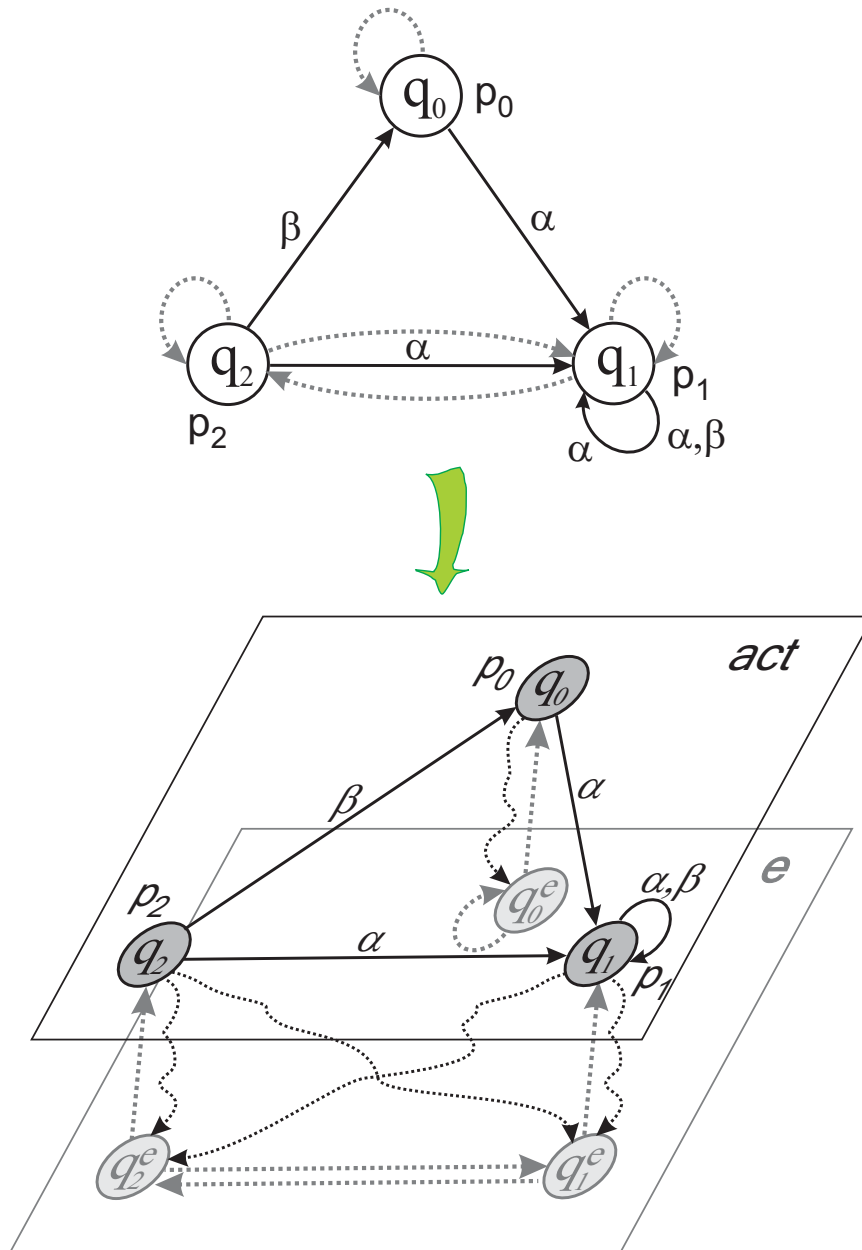


Figure 5: Getting rid of knowledge and epistemic links

are labeled with a special proposition e_a . The “real” states q from the original model are called “action” states, and are labeled with another special proposition act . Epistemic agent e_a can enforce transitions to states that are indistinguishable for agent a (see Figure 5 for an example).⁸ Then, “ a knows φ ” can be rephrased as “ e_a can only effect transitions to epistemic states where φ holds”. With some additional tricks to ensure the right interplay between actions of epistemic agents, we get the following translation of formulae:

$$\begin{aligned}
 tr(p) &= p, & \text{for } p \in \Pi \\
 tr(\neg\varphi) &= \neg tr(\varphi) \\
 tr(\varphi \vee \psi) &= tr(\varphi) \vee tr(\psi) \\
 tr(\langle\langle A \rangle\rangle \bigcirc \varphi) &= \langle\langle A \cup \text{Agt}^e \rangle\rangle \bigcirc (act \wedge tr(\varphi)) \\
 tr(\langle\langle A \rangle\rangle \square \varphi) &= \langle\langle A \cup \text{Agt}^e \rangle\rangle \square (act \wedge tr(\varphi)) \\
 tr(\langle\langle A \rangle\rangle \varphi \mathcal{U} \psi) &= \langle\langle A \cup \text{Agt}^e \rangle\rangle (act \wedge tr(\varphi)) \mathcal{U} (act \wedge tr(\psi)) \\
 tr(K_i \varphi) &= \neg \langle\langle e_1, \dots, e_i \rangle\rangle \bigcirc (e_i \wedge \langle\langle e_1, \dots, e_k \rangle\rangle \bigcirc (act \wedge \neg tr(\varphi))).
 \end{aligned}$$

Note that the length of $tr(\varphi)$ is linear in the length of φ and the number of agents k . Two important facts follow from [14, Theorem 8]:

Lemma 6 For every **CEGS** M and a formula of **ATL**_{ir}^u that does not include the special propositions act, e_1, \dots, e_k , we have

$$M, q \models_{\mathbf{ATL}_{ir}^u} \varphi \quad \text{iff} \quad tr(M), q \models_{\mathbf{ATL}_{ir}^u} tr(\varphi).$$

Lemma 7 For every **CEGS** M , we have

$$M, q \models_{\mathbf{ATLI+K}} \text{uniform}(\sigma) \quad \text{iff} \quad tr(M), q \models_{\mathbf{ATLI+K}} tr(\text{uniform}(\sigma)).$$

Putting the pieces together: the reduction. We observe that **ATL**_{ir}^u can be seen as **ATL** where only uniform strategy profiles are allowed. An **ATLI** formula that characterizes uniformity has been defined in the previous paragraphs. It can be now plugged into our “**ATL** with Plausibility” to restrict agents’ behavior in the way the semantics of **ATL**_{ir}^u does. This way, we obtain a reduction of **SNSAT**₂ to model checking of **ATLP**^[ATLI].

Proposition 8

$$z_p \quad \text{iff} \quad tr(M'_p), q_0^p \models_{\mathbf{ATLP}[\mathbf{ATLI}]} (\mathbf{set-pl} \ \sigma.tr(\text{uniform}(\sigma))) \mathbf{Pl} \ tr(\Phi_p).$$

Proof. We have z_p iff $M'_p, q_0^p \models_{\mathbf{ATL}_{ir}^u} \Phi_p$ iff $tr(M'_p), q_0^p \models_{\mathbf{ATL}_{ir}^u} tr(\Phi_p)$
 iff $tr(M'_p), q_0^p \models_{\mathbf{ATLP}[\mathbf{ATLI}]} (\mathbf{set-pl} \ \sigma.tr(\text{uniform}(\sigma))) \mathbf{Pl} \ tr(\Phi_p). \quad \blacksquare$

⁸ An interested reader is referred to [14] for the technical details of the construction.

3.4 Summary of the Results

As a result, we obtain the following theorem.

Theorem 9 *Model checking **ATLP** is Δ_3^P -complete with respect to the number of transitions in the model and the length of the formula.*

On the way, we have also proved that checking strategic abilities when *all* players are required to play uniformly is Δ_3^P -complete (that is, harder than ability against the worst line of events captured by **ATL**_{irr} formulae, which is “only” Δ_2^P -complete). We believe it is an interesting result with respect to verification of various kinds of agents’ ability under incomplete information. We note that the result from [20] for extensive games with incomplete information can be seen as a specific case of our result, at least in the class of games with binary payoffs.

4 Conclusions

In this technical report, we prove that model checking **ATLP** is Δ_3^P -complete, for abstract plausibility terms as well as terms based on formulae of “**ATL** with Intentions” (**ATLI**). On the way, we also define another interesting variant of **ATL** – where both proponents and opponents are required to use only uniform strategies – and we establish its model checking complexity.

The logics of **ATLI** and **ATLP** share many similarities. Thus, it might be even more elegant to “plug in” plausibility specifications written in **ATLP** itself. A preliminary take on this idea has been presented in [4], but the model checking complexity of the resulting language remains to be studied.

References

- [1] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time Temporal Logic. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 100–109. IEEE Computer Society Press, 1997.
- [2] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time Temporal Logic. *Journal of the ACM*, 49:672–713, 2002.
- [3] A. Baltag. A logic for suspicious players. *Bulletin of Economic Research*, 54(1):1–46, 2002.
- [4] N. Bulling and W. Jamroga. A logic for reasoning about rational agents: Yet another attempt. In L. Czaja, editor, *Proceedings of CS&P*, pages 87–99, 2007.

References

- [5] F. Chu and J. Halpern. On the NP-completeness of finding an optimal strategy in games with common payoffs. *International Journal of Game Theory*, 2001.
- [6] E.M. Clarke, E.A. Emerson, and A.P. Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM Transactions on Programming Languages and Systems*, 8(2):244–263, 1986.
- [7] V. Conitzer and T. Sandholm. Complexity results about Nash equilibria. Technical Report CMU-CS-02-135, School of Computer Science, Carnegie-Mellon University, 2002.
- [8] E. A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 995–1072. Elsevier Science Publishers, 1990.
- [9] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman: San Francisco, 1979.
- [10] I. Gilboa and E. Zemel. Nash and correlated equilibria: Some complexity considerations. *Games and Economic Behavior*, 1989.
- [11] V. Goranko and W. Jamroga. Comparing semantics of logics for multi-agent systems. *Synthese*, 139(2):241–280, 2004.
- [12] B.P. Harrenstein, W. van der Hoek, J.-J. Meyer, and C. Witteveen. A modal characterization of Nash equilibrium. *Fundamenta Informaticae*, 57(2–4):281–321, 2003.
- [13] P. Harrenstein, W. van der Hoek, J.-J. Meijer, and C. Witteveen. Subgame-perfect Nash equilibria in dynamic logic. In M. Pauly and A. Baltag, editors, *Proceedings of the ILLC Workshop on Logic and Games*, pages 29–30. University of Amsterdam, 2002. Tech. Report PP-1999-25.
- [14] W. Jamroga. Reducing knowledge operators in the context of model checking. Technical Report IfI-07-09, Clausthal University of Technology, 2007.
- [15] W. Jamroga and N. Bulling. A framework for reasoning about rational agents. In *Proceedings of AAMAS'07*, pages 592–594, 2007.
- [16] W. Jamroga and N. Bulling. A logic for reasoning about rational agents. In F. Sadri and K. Satoh, editors, *Proceedings of CLIMA '07*, pages 54–69, 2007.
- [17] W. Jamroga and J. Dix. Model checking ATL_{ir} is indeed Δ_2^P -complete. In *Proceedings of EUMAS'06*, 2006.

- [18] W. Jamroga and J. Dix. Model checking abilities of agents: A closer look. *Theory of Computing Systems*, 2008. To appear.
- [19] W. Jamroga, W. van der Hoek, and M. Wooldridge. Intentions and strategies in game-like scenarios. In Carlos Bento, Amílcar Cardoso, and Gaël Dias, editors, *Progress in Artificial Intelligence: Proceedings of EPIA 2005*, volume 3808 of *Lecture Notes in Artificial Intelligence*, pages 512–523. Springer Verlag, 2005.
- [20] D. Koller and N. Megiddo. The complexity of twoperson zero-sum games in extensive form. *Games and Economic Behavior*, 4:528–552, 1992.
- [21] F. Laroussinie, N. Markey, and Ph. Schnoebelen. Model checking CTL+ and FCTL is hard. In *Proceedings of FoSSaCS'01*, volume 2030 of *Lecture Notes in Computer Science*, pages 318–331. Springer, 2001.
- [22] Y. Moses and M. Tennenholz. Artificial social systems. *Computers and AI*, 14(6):533–562, 1995.
- [23] P. Y. Schobbens. Alternating-time logic with imperfect recall. *Electronic Notes in Theoretical Computer Science*, 85(2), 2004.
- [24] Y. Shoham and M. Tennenholz. On the synthesis of useful social laws for artificial agent societies. In *Proceedings of AAI-92*, 1992.
- [25] J. van Benthem. Rational dynamics and epistemic logic in games. In S. Vannucci, editor, *Logic, Game Theory and Social Choice III*, pages 19–23, 2003.
- [26] W. van der Hoek, W. Jamroga, and M. Wooldridge. A logic for strategic reasoning. In *Proceedings of AAMAS'05*, pages 157–164, 2005.
- [27] W. van der Hoek, M. Roberts, and M. Wooldridge. Social laws in alternating time: Effectiveness, feasibility and synthesis. *Synthese*, 156(1):1–19, 2005.
- [28] W. van der Hoek and M. Wooldridge. Tractable multiagent planning for epistemic goals. In C. Castelfranchi and W.L. Johnson, editors, *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02)*, pages 1167–1174. ACM Press, New York, 2002.
- [29] S. van Otterloo and G. Jonker. On Epistemic Temporal Strategic Logic. *Electronic Notes in Theoretical Computer Science*, XX:35–45, 2004. Proceedings of LCMAS'04.

References

- [30] S. van Otterloo and O. Roy. Verification of voting protocols. Working paper, University of Amsterdam, 2005.
- [31] S. van Otterloo, W. van der Hoek, and M. Wooldridge. Preferences in game logics. In *Proceedings of AAMAS-04*, pages 152–159, 2004.