Information and Computation 245 (2015) 98–123

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

Augmenting ATL with strategy contexts [☆]

François Laroussinie [a], Nicolas Markey [b,*]

[a] LIAFA, CNRS & U. Paris-Diderot, Case 7014, F-75205 Paris cedex 13, France
[b] LSV, CNRS & ENS Cachan, 61 av. Pdt Wilson, F-94230 Cachan, France

ARTICLE INFO

Article history:
Received 28 February 2014
Available online 26 June 2015

Keywords:
Temporal logics
Games for synthesis
Model checking
Satisfiability

ABSTRACT

We study the extension of the alternating-time temporal logic (ATL) with strategy contexts: contrary to the original semantics, in this semantics the strategy quantifiers do not reset the previously selected strategies.
We show that our extension ATL_{sc} is very expressive, but that its decision problems are quite hard: model checking is k-EXPTIME-complete when the formula has k nested strategy quantifiers; satisfiability is undecidable, but we prove that it is decidable when restricting to turn-based games. Our algorithms are obtained through a very convenient translation to QCTL (the computation-tree logic CTL extended with atomic quantification), which we show also applies to Strategy Logic, as well as when strategy quantification ranges over memoryless strategies.

1. Introduction

The alternating-time temporal logic (ATL) is a convenient extension of CTL for expressing properties of multi-agent systems. For this, it includes quantification over strategies of the agents, instead of the sole quantifiers over paths of CTL. However, in order to retain the nice algorithmic properties of CTL, the quantification over strategies in ATL is *forgetful*, in the sense that each quantifier deletes the previously selected strategies. Under this semantics, ATL model checking is PTIME-complete [3], while satisfiability is EXPTIME-complete [37].

ATL with strategy contexts (ATL_{sc} for short), introduced in [6], lifts this restriction, and stores the previously selected strategies within a context. The players keep on following their strategies until the formula says otherwise. This makes the logic very expressive, and very convenient for specifying properties of multi-agent systems, in settings where the agents are neither collaborative nor completely antagonist. For instance, considering a client/server interaction, ATL_{sc} can express the existence of a policy for the server in order to enforce mutual exclusion to the resource, and to make each client have a way of eventually accessing the resource. Such a property mixes collaboration between the server and each client, and possibly antagonism between clients. This expressive power comes with a cost: ATL_{sc} model checking was proved to be k-EXPTIME-complete, where k is the number of nested strategy quantifiers in the formula being checked [13,14]. It follows that ATL_{sc} model-checking problem is Tower-complete (Tower contains the classes k-EXPTIME for all k; see [33] for more details about the complexity class Tower). Satisfiability of ATL_{sc} is undecidable [35].

[☆] This paper is a long version of [25], whose focus was mainly on satisfiability of ATL_{sc} *via* quantified CTL [14]. We added several results from [13], with new proofs using QCTL, so that this paper contains all our expressiveness and algorithmic results about ATL_{sc} with a uniform presentation.
[*] Corresponding author.
E-mail addresses: francois.laroussinie@liafa.univ-paris-diderot.fr (F. Laroussinie), nicolas.markey@lsv.ens-cachan.fr (N. Markey).

http://dx.doi.org/10.1016/j.ic.2014.12.020
0890-5401/© 2015 Elsevier Inc. All rights reserved.

In this paper, we revisit these problems with a uniform technique: we establish a very tight link between ATL$_{sc}$ and QCTL (the extension of CTL with quantification over atomic propositions [34,21,17,24]). We prove that the model-checking problems for both logics are essentially the same problem (there are reductions both ways). Moreover, while satisfiability is undecidable for ATL$_{sc}$, we use our translation to QCTL to prove that satisfiability is decidable in restricted cases, and in particular when restricted to turn-based games.

The convenience of QCTL is not particular to ATL$_{sc}$, and we prove that our technique also applies to Strategy Logic (SL) [11,29], another extension of ATL with explicit handling of strategies. Also, using a different semantics for quantification in QCTL, we use our translation to study the variant of ATL$_{sc}$ where strategy quantifiers range over memoryless strategies. In this case, model checking is PSPACE-complete, but satisfiability is undecidable, even for finitely branching two-player turn-based games.

Related work Over the last decade, several formalisms have been proposed as improvements to ATL for specifying properties of multi-agent models. We review some of these extensions here.

Extensions of ATL *with explicit strategy restrictions* To the best of our knowledge, the first temporal logic allowing for real interaction between strategies is the logic ATL *with commitments* (a.k.a. *Counterfactual* ATL) [36]. This logic extends ATL with an operator that can force a player to play according to a given (explicit) strategy. This can be used to express *e.g.* what an agent could do if some other agent were restricted to follow a given strategy. Building on similar ideas, ATL *with explicit strategies* [38], and the more recent ATL *with explicit actions* [19], decorate the strategy quantifiers with strategies or actions, thus restricting the allowed behaviors of some of the players.

These extensions provide a way of improving the expressiveness of ATL, and involve a true interaction between strategies. However, having explicit strategies is many cases not very powerful, nor very convenient. It must be noticed that all three extensions where only studied for *memoryless* strategies (both for explicit strategies and for the range of strategy quantifiers), which weakens the expressiveness but gives rise to conceptually simple algorithms.

Extensions of ATL *with contexts* Together with Thomas Brihaye and Arnaud Da Costa, we introduced ATL$_{sc}$ in [6]. We proved that model checking is decidable in [13]. The present paper includes the results of those two papers, with a more detailed and uniform presentation using QCTL. The link between QCTL and ATL$_{sc}$ was only sketched in [14]; the present paper explains this link is more details, and how it can be used to solve model checking and satisfiability (in restricted cases) for ATL$_{sc}$.

Several variants of our semantics of ATL with strategy contexts were already proposed by other researchers. ATL *with irrevocable strategies* [1] makes strategies sticky: once a player is assigned a strategy (through a strategy quantifier), she cannot change strategy anymore (and the subsequent strategy quantifications for that player are ignored). This way, similarly to our approach, strategies are stored in a context, but they cannot be dropped from the context. The same authors then defined ATL *with strategy commitment and release* [2], which in essence is the same as our extension ATL$_{sc}$. They also proposed several variations on the semantics, and discussed their expressiveness. In terms of verification, both logics were only studied *w.r.t.* memoryless strategies, which again greatly simplifies the algorithmic questions.

Stochastic Game Logic [5] also extends ATL by storing strategies in contexts. However, this logic is studied in the context of stochastic games and strategies, which makes the setting much richer. Model checking is proven undecidable in the general case, and decidable when strategy quantification is restricted to range over memoryless (randomized or deterministic) strategies.

Finally, the logic *Strategy Interaction Logic* (SIL) introduced in [39] is equivalent to ATL$_{sc}$, with a slightly different syntax. The authors of [39] identify a fragment of SIL, which they name *Basic* SIL (BSIL), in which the strategy context is reset after each temporal modality. In this case, the authors prove that model checking can be decided in polynomial space over turn-based games. Another fragment of SIL, *Temporal Cooperation Logic* (TCL), is investigated in [20]: there, no alternation is allowed between existential and universal strategy quantification, unless the strategy context is reset. This is shown to make model-checking EXPTIME-complete over turn-based games. Practical experiments are reported in [39,20].

Strategy logics *Strategy Logic* (SL) [11,29] is another proposal for expressing complex properties of games. It builds on a different approach, where strategies can be handled explicitly: first-order quantification can be used to select strategies, which can then be assigned to players. LTL is then used to impose constraints on the resulting paths. The logic was defined and studied in [11] for two-player turn-based games. An extended version was proposed in [29] for n-player concurrent games.

In terms of expressiveness, SL embeds all the logics listed above. As regards algorithms, SL model checking was proved decidable [29,27], while satisfiability is undecidable [29]. As we explain in Section 6.2, SL actually enjoys the same algorithmic properties as ATL$_{sc}$ (in particular, we prove that satisfiability is decidable over turn-based games), and these results can also be obtained via a tight correspondence with QCTL.

Several variants of SL have also been defined and studied. In particular, the syntactic restriction SL[1G] (*One-Goal Strategy Logic*) [28] restricts formulas to be in prenex normal form, with a single assignment of strategies to players and an LTL objective. This logic can be shown to have elementary (2-EXPTIME-complete) model-checking and satisfiability problems. Other variants, *e.g.* with boolean combinations of goals in the scope of strategy quantification, have also been investigated. An algorithm for model checking SL[1G] is reported in [7], and for an epistemic variant of SL in [8].

On a different note, *Updatable Strategy Logic* (USL) [9] has been considered as an extension of SL to non-deterministic strategies: such strategies are allowed to return sets of moves, rather than a single move. In that setting, assigning a strategy to an agent does not remove the previously assigned strategy, but it refines it. While the setting is different, USL again shares similarities with SL and ATL_{sc}, and in particular could be handled via QCTL.[1]

2. Definitions

2.1. Preliminaries

Given two mappings $f: A \to B$ and $g: A' \to B'$, and a subset $C \subseteq A \cap A'$, we say that *f and g coincide on C*, written $f \simeq_C g$, when $f(c) = g(c)$ for all $c \in C$. In case $A = A'$, and $B = 2^S$ and $B' = 2^{S'}$ for some sets S and S', then given $T \subseteq S \cap S'$, we say that *f and g agree in T*, written $f \equiv_T g$, if for all $a \in A$, it holds $f(a) \cap T = g(a) \cap T$. If A is a subset of a larger set $\mathbb{A}$, then f is said to be a *partial function* (*w.r.t.* $\mathbb{A}$), and A is called its *domain* (written $\text{dom}(f)$).

For $k \in \mathbb{N}$, we define $[k] = \{i \in \mathbb{N} \mid 0 \leq i < k\}$. We also let $[\infty] = \mathbb{N}$. Let Σ be a set. A *word* over Σ is a mapping $w: [k] \to \Sigma$, for some $k \in \mathbb{N} \cup \{\infty\}$; for $n \in [k]$, we usually write w_n for $w(n)$. The word w is finite when $k \in \mathbb{N}$ (then k is the *length* of w, denoted $|w|$), otherwise it is infinite (then $|w| = +\infty$). When w is finite, we write $\text{last}(w)$ for its last element $w(|w| - 1)$. The (only) word of length zero is denoted with ε. Given a finite word v and a word w, their *concatenation* $v \cdot w$ is the sequence u s.t. $u(n) = v(n)$ when $n < |v|$ and $u(n) = w(n - |v|)$ when $n \geq |v|$. When v is a one-letter word, we sometimes write $v_0 \cdot w$ to denote $v \cdot w$. A *prefix* of a word w is a finite word p such that there exists a word s such that $w = p \cdot s$. For any $n \leq |w|$, w has a unique prefix of length n, which we denote $w_{<n}$ (or sometimes $w_{\leq n-1}$).

Let D be a set. A *D-tree* is a non-empty set T of finite words over D such that for any $t \in T$, all the prefixes of t are in T. The elements of T are called *nodes*, and the special node ε (the empty word) is the *root* of T. A *Σ-labeled D-tree* is a pair $\langle T, \ell \rangle$ where T is a D-tree and $\ell: T \to \Sigma$ labels the nodes of T with a letter in Σ.

2.2. Kripke structures

Let AP be a set of atomic propositions, and $\Sigma = 2^{\text{AP}}$.

Definition 1. A *Kripke structure over* AP is a 3-tuple $\mathcal{S} = \langle Q, R, \ell \rangle$ where Q is a finite or countably infinite set of states, $R \subseteq Q^2$ is a binary relation and $\ell: Q \to \Sigma$ is a labeling function. We always assume that the relation R be left-total, *i.e.*, for any $q \in Q$, there is a $q' \in Q$ such that $(q, q') \in R$.

Let $\mathcal{S} = \langle Q, R, \ell \rangle$ be a Kripke structure, and $q \in Q$. A *path in $\mathcal{S}$ from q* is a non-empty word π over Q such that $\pi_0 = q$ and for all $n \in [|\pi| - 1]$, it holds $(\pi_n, \pi_{n+1}) \in R$. Given a finite path π and a path ρ such that $\text{last}(\pi) = \rho_0$, the *join* of π and ρ, denoted $\pi : \rho$ is defined as the concatenation $\pi_{<|\pi|-1} \cdot \rho$. Notice that since each state in a Kripke structure must have at least one successor, any finite path can be enlarged.

Given a subset $I \subseteq Q$, we write $I Q^*_{\mathcal{S}}$ (resp. $I Q^\omega_{\mathcal{S}}$) for the set of finite (resp. infinite) paths in $\mathcal{S}$ from some $q \in I$; we write $q Q^*_{\mathcal{S}}$ (resp. $q Q^\omega_{\mathcal{S}}$) in case $I = \{q\}$, and $Q^+_{\mathcal{S}}$ (resp. $Q^\omega_{\mathcal{S}}$) in case $I = Q$. With a path π, we associate a *trace* $\ell \circ \pi: [|\pi|] \to \Sigma$, which is a word over Σ.

The *computation tree of $\mathcal{S}$* from q is the Σ-labeled Q-tree $\langle T, l \rangle$ where T is the Q-tree $\{w \in Q^* \mid q \cdot w \in q Q^*_{\mathcal{S}}\}$ and $l(w) = \ell(\text{last}(q \cdot w))$ for all $w \in T$. Notice that from our assumption that each state in $\mathcal{S}$ has at least one outgoing transition, any node in the computation tree has at least one successor. A *branch* in T is an infinite word $w \in \mathcal{Q}^\omega$ such that $q \cdot w \in q Q^\omega_{\mathcal{S}}$. Any finite prefix of a branch is a node of T.

2.3. *Quantified* CTL

The temporal logics CTL* and CTL were defined in the 1980s [32,12,16]. Let AP be a set of atomic propositions. The syntax of CTL* is as follows:

$$\text{CTL}^* \ni \varphi_{\mathsf{s}}, \psi_{\mathsf{s}} ::= p \mid \neg \varphi_{\mathsf{s}} \mid \varphi_{\mathsf{s}} \vee \psi_{\mathsf{s}} \mid \mathbf{E}\varphi_{\mathsf{p}} \mid \mathbf{A}\varphi_{\mathsf{p}}$$

$$\varphi_{\mathsf{p}}, \psi_{\mathsf{p}} ::= \varphi_{\mathsf{s}} \mid \neg \varphi_{\mathsf{p}} \mid \varphi_{\mathsf{p}} \vee \psi_{\mathsf{p}} \mid \mathbf{X}\varphi_{\mathsf{p}} \mid \varphi_{\mathsf{p}} \,\mathbf{U}\, \psi_{\mathsf{p}}$$

where p ranges over AP. Formulas of CTL* are interpreted in the computation tree (hence the name) of a given Kripke structure. Let $\mathcal{S} = \langle Q, R, \ell \rangle$ be such a structure, let ρ be an infinite path in $\mathcal{S}$, and $n \in \mathbb{N}$. That formula φ holds true at position n along ρ in $\mathcal{S}$ is defined inductively as follows:

$$\begin{aligned}
\mathcal{S}, \rho, n \models p \quad &\text{iff} \quad p \in \ell(\rho_n) \\
\mathcal{S}, \rho, n \models \neg \varphi \quad &\text{iff} \quad \mathcal{S}, \rho, n \not\models \varphi \\
\mathcal{S}, \rho, n \models \varphi \vee \psi \quad &\text{iff} \quad \mathcal{S}, \rho, n \models \varphi \text{ or } \mathcal{S}, \rho, n \models \psi
\end{aligned}$$

[1] Personal communication with the authors of [9].

$$\begin{aligned}
\mathcal{S},\rho,n \models \mathbf{E}\varphi_{\mathrm{p}} &\quad \text{iff} \quad \exists \rho' \in \rho_n Q_{\mathcal{S}}^{\omega}.\ \mathcal{S},\rho',0 \models \varphi_{\mathrm{p}}\\
\mathcal{S},\rho,n \models \mathbf{A}\varphi_{\mathrm{p}} &\quad \text{iff} \quad \forall \rho' \in \rho_n Q_{\mathcal{S}}^{\omega}.\ \mathcal{S},\rho',0 \models \varphi_{\mathrm{p}}\\
\mathcal{S},\rho,n \models \mathbf{X}\varphi_{\mathrm{p}} &\quad \text{iff} \quad \mathcal{S},\rho,n+1 \models \varphi_{\mathrm{p}}\\
\mathcal{S},\rho,n \models \varphi_{\mathrm{p}}\,\mathbf{U}\,\psi_{\mathrm{p}} &\quad \text{iff} \quad \exists l \geq 0.\ (\mathcal{S},\rho,n+l \models \psi_{\mathrm{p}} \text{ and}\\
&\qquad\qquad\qquad \forall 0 \leq m < l.\ \mathcal{S},\rho,n+m \models \varphi_{\mathrm{p}})
\end{aligned}$$

Several useful abbreviations can be defined: besides the classical $\top = p \vee \neg p$ (which always evaluates to true), $\bot = \neg\top$ and $\varphi \wedge \psi = \neg(\neg\varphi \vee \neg\psi)$, the following modalities will be used throughout this paper:

$$\mathbf{F}\varphi_{\mathrm{p}} = \top\,\mathbf{U}\,\varphi_{\mathrm{p}} \qquad \mathbf{G}\varphi_{\mathrm{p}} = \neg\,\mathbf{F}\,\neg\,\varphi_{\mathrm{p}}.$$

The former states that φ_{p} will eventually hold true along the current path, while the latter states that it holds true at any position along that path.

An important remark about this semantics is that the evaluation of state formulas (of the form φ_{s} in the grammar defining CTL*) at position n along ρ only depends on ρ_n. In other terms, for any two paths ρ and ρ' and any two positions n and n' such that $\rho_n = \rho'_{n'}$, and for any state formula φ_{s}, it holds

$$\mathcal{S},\rho,n \models \varphi_{\mathrm{s}} \quad \text{iff} \quad \mathcal{S},\rho',n' \models \varphi_{\mathrm{s}}.$$

As a consequence, for a state formula φ_{s}, we often replace $\mathcal{S},\rho,0 \models \varphi_{\mathrm{s}}$ with $\mathcal{S},\rho_0 \models \varphi_{\mathrm{s}}$.

Another remark is that CTL* is invariant under bisimulation [30]: two structures that are bisimilar satisfy the same CTL* formulas. In particular, evaluating a CTL* formula over a Kripke structure and over its computation tree are equivalent. Formally, let $q \in Q$, and w be a branch in the computation tree $\mathcal{T}$ of $\mathcal{S}$ from q (so that $q \cdot w$ is an infinite path in $\mathcal{S}$; we abusively see w as a path in $\mathcal{T}$ starting from the root, when $\mathcal{T}$ is seen as an infinite-state Kripke structure). Let $n \in \mathbb{N}$. Then for any $\varphi \in$ CTL*, it holds

$$\mathcal{S}, q\cdot w, n \models \varphi \quad \text{iff} \quad \mathcal{T}_{\mathcal{S}}, w, n \models \varphi.$$

The fragment CTL of CTL* is obtained by restricting the form of path formulas to the following grammar:

$$\varphi_{\mathrm{p}},\psi_{\mathrm{p}} ::= \neg\,\varphi_{\mathrm{p}} \mid \mathbf{X}\varphi_{\mathrm{s}} \mid \varphi_{\mathrm{s}}\,\mathbf{U}\,\psi_{\mathrm{s}}.$$

In other terms, the modalities **X** and **U** (and negations thereof) can only appear in the immediate scope of a path quantifier **E** or **A**.

We now present an extension of CTL* with quantification over atomic propositions, which will be our main technical tool in the sequel.

For $P \subseteq$ AP, two Kripke structures $\mathcal{S} = \langle Q, R, \ell\rangle$ and $\mathcal{S}' = \langle Q', R', \ell'\rangle$ are *P-equivalent* (denoted $\mathcal{S} \equiv_P \mathcal{S}'$) whenever $Q = Q'$, $R = R'$, and $\ell \equiv_P \ell'$ (*i.e.*, $\ell(q) \cap P = \ell'(q) \cap P$ for any $q \in Q$). In other terms, $\mathcal{S} \equiv_P \mathcal{S}'$ if $\mathcal{S}'$ can be obtained from $\mathcal{S}$ by modifying the labeling function of $\mathcal{S}$ for propositions not in P.

Definition 2. The syntax of QCTL* is defined by the following grammar:

$$\begin{aligned}
\text{QCTL}^* \ni \varphi_{\mathrm{s}},\psi_{\mathrm{s}} &::= p \mid \neg\,\varphi_{\mathrm{s}} \mid \varphi_{\mathrm{s}} \vee \psi_{\mathrm{s}} \mid \mathbf{E}\varphi_{\mathrm{p}} \mid \mathbf{A}\varphi_{\mathrm{p}} \mid \exists p.\ \varphi_{\mathrm{s}}\\
\varphi_{\mathrm{p}},\psi_{\mathrm{p}} &::= \varphi_{\mathrm{s}} \mid \neg\,\varphi_{\mathrm{p}} \mid \varphi_{\mathrm{p}} \vee \psi_{\mathrm{p}} \mid \mathbf{X}\varphi_{\mathrm{p}} \mid \varphi_{\mathrm{p}}\,\mathbf{U}\,\psi_{\mathrm{p}}.
\end{aligned}$$

The fragment QCTL of is the analogous of CTL.

The *structure semantics* of QCTL* is derived from the semantics of CTL* by adding the following rule:

$$\mathcal{S},\rho,n \models \exists p.\ \varphi_{\mathrm{s}} \quad \text{iff} \quad \exists \mathcal{S}'.\ \mathcal{S}' \equiv_{\mathrm{AP}\smallsetminus\{p\}} \mathcal{S} \text{ and } \mathcal{S}',\rho,n \models \varphi_{\mathrm{s}}.$$

In other terms, $\exists p.\ \varphi_{\mathrm{s}}$ means that it is possible to (re)label the Kripke structure with p in order to make φ_{s} hold.

While CTL* is invariant under bisimulation, this is not the case for QCTL*: evaluating QCTL* on a Kripke structure and on its computation tree (seen as an infinite-state Kripke structure) are not equivalent. As a consequence, we consider another semantics for QCTL*, called *tree semantics*, where $\exists p.\ \varphi_{\mathrm{s}}$ holds true if it is possible to label the *computation tree* of the original Kripke structure (instead of the Kripke structure itself) in order to make φ hold. This is the semantics we consider in the sequel.

We refer to [24] for a detailed study of QCTL* and QCTL. Here we just recall the following important properties of these logics. First note that QCTL is actually as expressive as QCTL* (with an effective translation) [17,14]. Secondly model checking and satisfiability are decidable but non-elementary More precisely the complexity depends on the number of alternations of propositional quantifications in the formulas. In the following we will refer to the fragments EQ^{k}CTL and Q^{k}CTL of QCTL and to EQ^{k}CTL* and Q^{k}CTL* the corresponding fragments of QCTL*: EQ^{k}CTL contains the QCTL formulas in prenex normal form (where quantifications are external to the CTL formula), starting with an existential quantification $\exists$ and where the number of alternations is k. In $\mathsf{Q}^k\mathsf{CTL}$ formulas are not supposed to be in prenex normal form but the alternation of quantifier is bounded by k: formally, $\mathsf{Q}^1\mathsf{CTL}$ is $\mathsf{CTL}[\mathsf{EQ}^1\mathsf{CTL}]$, and $\mathsf{Q}^{k+1}\mathsf{CTL}$ is $\mathsf{Q}^1\mathsf{CTL}[\mathsf{Q}^k\mathsf{CTL}]$. The decision procedures for these logics are based on automata construction: given a QCTL formula φ and a (finite) set $D \subset \mathbb{N}$, one can build a tree automaton $\mathcal{A}_{\varphi,D}$ recognizing the D-trees satisfying φ. This provides a decision procedure for model checking as the Kripke structure $\mathcal{S}$ fixes the set D, and it remains to check whether the computation tree of $\mathcal{S}$ is accepted by $\mathcal{A}_{\varphi,\mathcal{D}}$. For satisfiability the decision procedure is obtained by building a formula φ_2 from φ such that φ_2 is satisfied by some [2]-tree if, and only if, φ is satisfied by some finitely-branching tree. Finally, it remains to notice that a QCTL formula is satisfiable if, and only if, it is satisfiable in a finitely-branching tree (since QCTL is as expressive as MSO [24]) to get the decision procedure for QCTL satisfiability. As a consequence, a QCTL formula is satisfiable if, and only if, it is satisfied by a regular tree (corresponding to the computation tree of some finite Kripke structure).

2.4. Concurrent game structures

Game structures extend Kripke structures with several agents acting on the evolution of the system.

Definition 3. (See [3].) A *concurrent game structure* (*CGS*) is a 7-tuple $\mathcal{C} = \langle Q, R, \ell, \mathsf{Agt}, \mathcal{M}, \mathsf{Mov}, \mathsf{Edge}\rangle$ where: $\langle Q, R, \ell\rangle$ is a Kripke structure, $\mathsf{Agt} = \{a_1, \ldots, a_p\}$ is a finite set of *agents*, $\mathcal{M}$ is a non-empty finite or countably infinite set of moves, $\mathsf{Mov}\colon Q \times \mathsf{Agt} \to \mathcal{P}(\mathcal{M}) \smallsetminus \{\varnothing\}$ defines the set of available moves of each agent in each state, and $\mathsf{Edge}\colon Q \times \mathcal{M}^{\mathsf{Agt}} \to R$ is a transition table associating, with each state q and each set of moves of the agents, the resulting transition departing from q.

Let $\mathcal{C} = \langle Q, R, \ell, \mathsf{Agt}, \mathcal{M}, \mathsf{Mov}, \mathsf{Edge}\rangle$ be a CGS. The notions of paths and computation trees of $\mathcal{C}$ are inherited from the underlying Kripke structure $\langle Q, R, \ell\rangle$. The size of $\mathcal{C}$, denoted $|\mathcal{C}|$, is $|Q| + |\mathsf{Edge}|$. Notice that for all $q \in Q$, $\mathsf{Edge}(q)$ is a $|\mathsf{Agt}|$-dimensional table, whose size is then exponential in $|\mathsf{Agt}|$ (provided that $|\mathcal{M}| > 1$). A *move vector* in $\mathcal{C}$ is a mapping $m\colon \mathsf{Agt} \to \mathcal{M}$. For a state $q \in Q$, we define

$$\begin{aligned}\mathsf{Next}(q) = \{q' \in Q \mid \exists m \in \mathcal{M}^{\mathsf{Agt}}.\ \forall a_i \in \mathsf{Agt}.\ m(a_i) \in \mathsf{Mov}(q, a_i)\\ \text{and } \mathsf{Edge}(q, m) = (q, q')\}\end{aligned}$$

and, for a coalition $C \subseteq \mathsf{Agt}$ and a partial move vector m with $\mathsf{dom}(m) = C$,

$$\mathsf{Next}(q, C, m) = \{q' \in Q \mid \exists m' \in \mathcal{M}^{\mathsf{Agt}}.\ m' \simeq_C m \text{ and } \mathsf{Edge}(q, m') = (q, q')\}.$$

A *turn-based* game structure (TBGS for short) is a CGS where each state q is controlled by a single agent, called the *owner* of q (and denoted $\mathsf{Own}(q)$). Formally, for every $q \in Q$, for any two move vectors m and m' with $m =_{\mathsf{Own}(q)} m'$, it holds $\mathsf{Edge}(q, m) = \mathsf{Edge}(q, m')$ (it is sometimes required that for all q, all players but $\mathsf{Own}(q)$ have a single possible move in q; this would make no difference in our setting).

A *strategy* for a player $a_i \in \mathsf{Agt}$ in a CGS $\mathcal{C}$ is a function $f\colon Q^+_{\mathcal{S}} \to \mathcal{M}$ that maps any finite path to a possible move for a_i, *i.e.*, satisfying $f(\pi) \in \mathsf{Mov}(\mathsf{last}(\pi), a_i)$. A strategy f is *memoryless* if $f(\pi) = f(\pi')$ whenever $\mathsf{last}(\pi) = \mathsf{last}(\pi')$. A strategy for a coalition A is a mapping assigning a strategy to each agent in A. The set of strategies for A is denoted $\mathsf{Strat}(A)$ (and $\mathsf{Strat}_0(A)$ is the subset of memoryless strategies). In the sequel, when no ambiguity arises, we subscript the strategies with their domains, writing f_A for a strategy of $A \subseteq \mathsf{Agt}$ and f_i for a strategy of player a_i (hence $f_i = f_A(a_i)$ when $a_i \in A$). Given a strategy $f_A \in \mathsf{Strat}(A)$ and a coalition B, the strategy $(f_A)_{|B}$ (resp. $(f_A)_{\smallsetminus B}$) denotes the restriction of f_A to the coalition $A \cap B$ (resp. $A \smallsetminus B$). Given two strategies $f_A \in \mathsf{Strat}(A)$ and $g_B \in \mathsf{Strat}(B)$, we define $f_A \circ g_B \in \mathsf{Strat}(A \cup B)$ as $(f_A \circ g_B)_j = f_j$ (resp. g_j) if $a_j \in A$ (resp. $a_j \in B \smallsetminus A$).

Let ρ be a finite path in $\mathcal{C}$, and $f_A \in \mathsf{Strat}(A)$ for some coalition A. A path π is *compatible with f_A after ρ* if it is obtained by playing strategy f_A after prefix ρ. Formally, this requires that π contains ρ as a prefix and that, for all $n \in [|\rho|; |\pi| - 1]$, letting $m_n\colon a_i \in A \mapsto f_i(\pi_{<n})$ be the move vector returned by the strategy f_A, it holds $\pi_{n+1} \in \mathsf{Next}(\pi_n, A, m_n)$ (*i.e.*, π_{n+1} is a successor of π_n when coalition A plays according to its strategy f_A). The set of *outcomes of f_A after ρ*, denoted $\mathsf{Out}(\rho, f_A)$, is the set of infinite paths that are compatible with f_A after ρ.

Example 4. Fig. 1 represents two three-state two-player CGSs. The transitions are decorated with their corresponding move vectors. In $\mathcal{C}$, each player has only two allowed moves, while in $\mathcal{C}'$, they have three. One can easily check *e.g.* that $\mathsf{Next}(q_0, a_1, 1)$ is $\{q_0, q_1\}$ in $\mathcal{C}$, and $\mathsf{Next}(q'_0, a_2, 3)$ is $\{q'_0, q'_1, q'_2\}$ in $\mathcal{C}'$.

2.5. ATL *with strategy contexts*

We now introduce our logic, which extends the alternating-time temporal logic of [3] with strategy contexts. We assume a fixed set of atomic propositions AP and a fixed set of agents Agt.

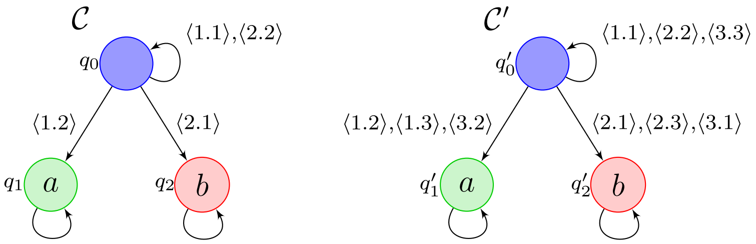
$\mathcal{C}$
q_0
$\langle 1.1\rangle,\langle 2.2\rangle$
$\langle 1.2\rangle$
$\langle 2.1\rangle$
q_1 a
q_2 b
$\mathcal{C}'$
q'_0
$\langle 1.1\rangle,\langle 2.2\rangle,\langle 3.3\rangle$
$\langle 1.2\rangle,\langle 1.3\rangle,\langle 3.2\rangle$
$\langle 2.1\rangle,\langle 2.3\rangle,\langle 3.1\rangle$
q'_1 a
q'_2 b

Fig. 1. Two concurrent game structures.

Definition 5. The formulas of ATL^*_{sc} are defined by the following grammar:

$$\mathsf{ATL}^*_{sc} \ni \varphi_{\mathsf{s}}, \psi_{\mathsf{s}} ::= p \mid \neg\,\varphi_{\mathsf{s}} \mid \varphi_{\mathsf{s}} \vee \psi_{\mathsf{s}} \mid \langle\!\cdot A\cdot\!\rangle\varphi_{\mathsf{p}} \mid \langle\!\cdot \bar{A}\cdot\!\rangle\varphi_{\mathsf{p}} \mid (\!| A |\!)\varphi_{\mathsf{s}} \mid (\!| \bar{A} |\!)\varphi_{\mathsf{s}}$$

$$\varphi_{\mathsf{p}}, \psi_{\mathsf{p}} ::= \varphi_{\mathsf{s}} \mid \neg\,\varphi_{\mathsf{p}} \mid \varphi_{\mathsf{p}} \vee \psi_{\mathsf{p}} \mid \mathbf{X}\varphi_{\mathsf{p}} \mid \varphi_{\mathsf{p}}\,\mathbf{U}\,\psi_{\mathsf{p}}$$

where p ranges over AP and A ranges over 2^{Agt}.

ATL^*_{sc} formulas are interpreted over CGSs, within a context (*i.e.*, a preselected strategy): *state formulas* of the form φ_{s} in the grammar above are evaluated over states, while *path formulas* of the form φ_{p} are evaluated along paths. In order to have a uniform definition, we evaluate all formulas at a given position along a path.

The semantics is quite similar to that of CTL*, but ATL^*_{sc} now has *strategy quantifiers* in place of path quantifiers. When strategy quantifiers assign new strategies to some players, the other players keep on playing their previously assigned strategies. This is what "strategy contexts" refers to. Informally, formula $\langle\!\cdot A\cdot\!\rangle\varphi_{\mathsf{p}}$ holds at position n along ρ under the context F if it is possible to extend F with a strategy for the coalition A such that the outcomes of the resulting strategy after $\rho_{\leq n}$ all satisfy φ_{p}. Formula $\langle\!\cdot \bar{A}\cdot\!\rangle\varphi_{\mathsf{p}}$ is similar, but for the complement of coalition A. This will be useful *e.g.* for quantifying over the strategies of all players independently of the set of players. Strategies can be dropped from the context using operators $(\!| {-} |\!)$ and $(\!| \bar{-} |\!)$. See Remark 6 below for our explanations why we introduced complement coalitions in the syntax.

We now define the semantics formally. Let $\mathcal{C}$ be a CGS, ρ be an infinite path of $\mathcal{C}$, and $n \in \mathbb{N}$ point to a position along ρ. Let $B \subseteq \mathsf{Agt}$ be a coalition, and $f_B \in \mathsf{Strat}(B)$. That a (state or path) formula φ holds at a position n along ρ in $\mathcal{C}$ under strategy context f_B, denoted $\mathcal{C}, \rho, n \models_{f_B} \varphi$, is defined inductively as follows (omitting atomic propositions and Boolean operators):

$$\begin{aligned}
\mathcal{C}, \rho, n \models_{f_B} (\!| A |\!)\varphi_{\mathsf{s}} \quad &\text{iff} \quad \mathcal{C}, \rho, n \models_{(f_B)_{\smallsetminus A}} \varphi_{\mathsf{s}} \\
\mathcal{C}, \rho, n \models_{f_B} (\!| \bar{A} |\!)\varphi_{\mathsf{s}} \quad &\text{iff} \quad \mathcal{C}, \rho, n \models_{(f_B)_{|A}} \varphi_{\mathsf{s}} \\
\mathcal{C}, \rho, n \models_{f_B} \langle\!\cdot A\cdot\!\rangle\varphi_{\mathsf{p}} \quad &\text{iff} \quad \exists g_A \in \mathsf{Strat}(A).\ \forall \rho' \in \mathsf{Out}(\rho_{\leq n}, g_A \circ f_B). \\
&\qquad\qquad \mathcal{C}, \rho', n \models_{g_A \circ f_B} \varphi_{\mathsf{p}} \\
\mathcal{C}, \rho, n \models_{f_B} \langle\!\cdot \bar{A}\cdot\!\rangle\varphi_{\mathsf{p}} \quad &\text{iff} \quad \exists g_{\bar{A}} \in \mathsf{Strat}(\mathsf{Agt} \smallsetminus A).\ \forall \rho' \in \mathsf{Out}(\rho_{\leq n}, g_{\bar{A}} \circ f_B). \\
&\qquad\qquad \mathcal{C}, \rho', n \models_{g_{\bar{A}} \circ f_B} \varphi_{\mathsf{p}} \\
\mathcal{C}, \rho, n \models_{f_B} \mathbf{X}\varphi_{\mathsf{p}} \quad &\text{iff} \quad \mathcal{C}, \rho, n+1 \models_{f_B} \varphi_{\mathsf{p}} \\
\mathcal{C}, \rho, n \models_{f_B} \varphi_{\mathsf{p}}\,\mathbf{U}\,\psi_{\mathsf{p}} \quad &\text{iff} \quad \exists l \geq 0.\ \mathcal{C}, \rho, n+l \models_{f_B} \psi_{\mathsf{p}} \text{ and } \forall 0 \leq m < l. \\
&\qquad\qquad \mathcal{C}, \rho, n+m \models_{f_B} \varphi_{\mathsf{p}}
\end{aligned}$$

Notice how the existential strategy quantifier embeds an implicit universal quantification over the set of outcomes of the selected strategy. Also notice that, given a state formula φ_{s}, two paths ρ and ρ', a position n such that $\rho_{\leq n} = \rho'_{\leq n}$, and a context f_B, it holds

$$\mathcal{C}, \rho, n \models_{f_B} \varphi_{\mathsf{s}} \quad \text{iff} \quad \mathcal{C}, \rho', n \models_{f_B} \varphi_{\mathsf{s}}$$

In particular, when $n = 0$, that $\mathcal{C}, \rho, 0 \models_{f_B} \varphi_{\mathsf{s}}$ does not depend on the whole path ρ but only on its first state ρ_0. In the sequel we equivalently write $\mathcal{C}, \rho_0 \models_{f_B} \varphi_{\mathsf{s}}$ in place of $\mathcal{C}, \rho, 0 \models_{f_B} \varphi_{\mathsf{s}}$. Finally, we write $\mathcal{C}, q_0 \models \varphi_{\mathsf{s}}$ when $\mathcal{C}, q_0 \models_{f_\varnothing} \varphi_{\mathsf{s}}$ (with empty context).

The usual shorthands such as $\mathbf{F}$ and $\mathbf{G}$ are defined as for CTL*. It will also be convenient to use the constructs $[\cdot A\cdot]\varphi_{\mathsf{p}}$ as a shorthand for $\neg\langle\!\cdot A\cdot\!\rangle\neg\,\varphi_{\mathsf{p}}$, and $\langle\!\cdot A\cdot\!\rangle\varphi_{\mathsf{s}}$ as a shorthand for $\langle\!\cdot A\cdot\!\rangle\bot\,\mathbf{U}\,\varphi_{\mathsf{s}}$.

The fragment ATL_{sc} of ATL^*_{sc} is defined as usual, by restricting the set of path formulas to

$$\varphi_{\mathsf{p}}, \psi_{\mathsf{p}} ::= \neg\,\varphi_{\mathsf{p}} \mid \mathbf{X}\varphi_{\mathsf{s}} \mid \varphi_{\mathsf{s}}\,\mathbf{U}\,\psi_{\mathsf{s}}.$$

Remark 6. Previous definitions of ATL^*_{sc} (see [6,13]) did not allow complement coalitions in the syntax for $\langle\cdot\text{-}\cdot\rangle$ and $(\!|\text{-}|\!)$. We add them here for two reasons. The first one is convenience: as already argued, it is sometimes useful to consider coalitions formed by all but a few players; complement coalitions provides an efficient way of naming such coalitions, independently of the set of all players of the game under study. The second reason is theoretical: our expressiveness result of Theorem 9, in the way we state it in this paper, requires talking about complement coalitions: indeed, in order to have generic translations (independent of the set of agents of the game), we sometimes need to talk about complement coalitions. The corresponding statements in our previous papers were weaker, as they did required the set of agents to be fixed. On a similar note, when we deal with satisfiability (see Section 5), the set of agents is not fixed a priori, and complement coalitions could not be expressed explicitly.

Example 7. We consider again the CGSs of Fig. 1. One can check that in both CGSs, player a_1 has a strategy to avoid visiting q_2 (resp. q'_2), but he has no strategy for leaving state q_0 (resp. q_0). But what distinguishes these two CGSs is the following: in $\mathcal{C}$, for any move m_1 of player a_1, it holds $|\mathsf{Next}(q_0, a_1, m_1)| = 2$. On the other hand, for move 3 of a_1 in $\mathcal{C}'$, we have $|\mathsf{Next}(q'_0, a_1, 3)| = 3$. In particular, under move 3 of a_1 in q'_0, player a_2 still has a move to reach q'_1 and another move to reach q'_2, so that formula $\langle\cdot a_1\cdot\rangle(\langle\cdot a_2\cdot\rangle\,\mathbf{X}\,a \wedge \langle\cdot a_2\cdot\rangle\,\mathbf{X}\,b)$ holds true in q'_0. No such move exists in $\mathcal{C}$, and the same formula is false in q_0.

The classical semantics of ATL^* and ATL, as defined in [3], did not involve a strategy context. Syntactically, the logic was defined as

$$\mathsf{ATL}^* \ni \varphi_\mathsf{s}, \psi_\mathsf{s} ::= p \mid \neg\,\varphi_\mathsf{s} \mid \varphi_\mathsf{s} \vee \psi_\mathsf{s} \mid \langle\!\langle A\rangle\!\rangle\varphi_\mathsf{p}$$

$$\varphi_\mathsf{p}, \psi_\mathsf{p} ::= \varphi_\mathsf{s} \mid \neg\,\varphi_\mathsf{p} \mid \varphi_\mathsf{p} \vee \psi_\mathsf{p} \mid \mathbf{X}\varphi_\mathsf{p} \mid \varphi_\mathsf{p}\,\mathbf{U}\,\psi_\mathsf{p}.$$

The semantics was similar to that of ATL^*_{sc}, but without the strategy context. The semantics of the strategy quantifier $\langle\!\langle A\rangle\!\rangle\varphi_\mathsf{p}$ is then defined as follows:

$$\mathcal{C}, \rho, n \models \langle\!\langle A\rangle\!\rangle\varphi_\mathsf{p} \quad \text{iff} \quad \exists g_A \in \mathsf{Strat}(A).\ \forall\rho' \in \mathsf{Out}(\rho_{\le n}, g_A).\ \mathcal{C}, \rho', n \models \varphi_\mathsf{p}.$$

The fragment ATL is obtained by restricting set of path formulas to

$$\varphi_\mathsf{p}, \psi_\mathsf{p} ::= \neg\,\varphi_\mathsf{p} \mid \mathbf{X}\varphi_\mathsf{s} \mid \varphi_\mathsf{s}\,\mathbf{U}\,\psi_\mathsf{s}.$$

Example 8. Consider the formula $\langle\!\langle a_1\rangle\!\rangle(\langle\!\langle a_2\rangle\!\rangle\,\mathbf{X}\,a \wedge \langle\!\langle a_2\rangle\!\rangle\,\mathbf{X}\,b)$ of ATL^*. The first quantification $\langle\!\langle a_1\rangle\!\rangle$ in this formula is useless, since the selected strategy is lost when quantifying over strategies of a_2. As a consequence, this formula is equivalent to $\langle\!\langle a_2\rangle\!\rangle\,\mathbf{X}\,a \wedge \langle\!\langle a_2\rangle\!\rangle\,\mathbf{X}\,b$, which is false both in q_0 and in q'_0 (in the CGSs of Fig. 1). It can be proved that q_0 and q'_0 are *alternating bisimilar* [4], so that they cannot be distinguished by ATL^*.

3. Expressiveness of ATL$_{sc}$ and ATL$^*_{sc}$

We devote this section to expressiveness issues. We begin with some examples of ATL_{sc} formulas, witnessing how useful our new formalism can be. We then give some theoretical results about the expressiveness of ATL_{sc}, showing for instance that ATL_{sc} and ATL^*_{sc} have the same expressive power. We finish the section with a comparison (*w.r.t.* expressiveness) to plain ATL. Comparisons with other formalisms are deferred to Section 6, where we review several logics related to ATL_{sc}.

3.1. Examples of formulas

For the reader to get acquainted with ATL_{sc}, we give in this section several examples of ATL_{sc} formulas. This will also witness the expressive power and usefulness of our logic.

Client-server interactions Consider a situation where a server S controls the access to a resource shared among several clients a_1 to a_n. The server has a double role: it has to ensure that at most one client uses the resource at any point in time (*mutual exclusion*), but also to provide a way for each client to be able to access the resource; the latter mixes collaboration between the server and individual agents, and antagonism between agents. In ATL^*_{sc}, we can express this requirement as follows:

$$\langle\cdot S\cdot\rangle\mathbf{G}\Big[\Big(\bigwedge_{\substack{i,j\in[1,n]\\ i\neq j}} \neg(\mathtt{access}_i \wedge \mathtt{access}_j)\Big) \wedge \Big(\bigwedge_{i\in[1,n]} \langle\cdot a_i\cdot\rangle\,\mathbf{F}\,\mathtt{access}_i\Big)\Big]$$

Nash equilibria In n-player games (with $n > 2$), players will usually have non-zero-sum objectives, and the notion of *winning strategy* is not relevant anymore. Instead, equilibria positions are sought, in which all players has *optimal* outcome. Here *optimal* may have many different meanings. One of them, corresponding to *Nash equilibria* [31], is related to the other players strategies: a Nash equilibrium is a strategy profile in which each individual strategy is the best response to the others' strategies.

ATL_{sc} can express the existence of a Nash equilibrium (here in a setting where the individual objectives of the players are Boolean, and only considering pure strategies): this would be written as

$$\langle\cdot\mathsf{Agt}\cdot\rangle\Big[\bigwedge_{a\in\mathsf{Agt}}(\neg\varphi_a\Rightarrow\neg\langle\cdot a\cdot\rangle\varphi_a)\Big]$$

In this formula, we write that there is a strategy profile (the one witnessing the outermost quantifier) such that no player can improve their payoff (*i.e.*, if they are not winning in the equilibrium, they don't have a way of achieving their goal in this situation).

In ATL_{sc}, we can additionally impose extra requirements to Nash equilibria. Indeed, several Nash equilibria might coexist, and some might be better than others (for instance, Nash equilibria where all the players fail to achieve their objectives might coexist with Nash equilibria where all the objectives are met). In ATL_{sc} we can additionally impose constraints on the equilibria strategies.

Winning secure equilibria A *winning secure equilibrium* [10] is a winning (for all players) Nash equilibria with the additional requirement that if a player deviates and worsens the payoff of some player, then she also worsens her own payoff. In other terms, no player can harm another player without harming herself. The existence of a winning secure equilibrium can be written as

$$\langle\cdot\mathsf{Agt}\cdot\rangle\Big[\bigwedge_{a\in\mathsf{Agt}}\varphi_a\wedge\bigwedge_{a,b\in\mathsf{Agt}}[\cdot a\cdot](\neg\varphi_b\Rightarrow\neg\varphi_a)\Big]$$

Dominant strategy A strategy is said *dominant* if it is a best response to any strategies of the other players. The existence of a dominant strategy for player a_i can be expressed as

$$\langle\cdot a_i\cdot\rangle[\cdot\overline{a_i}\cdot](\langle\cdot a_i\cdot\rangle\varphi_i\Rightarrow\varphi_i).$$

3.2. ATL_{sc} *vs* ATL^*_{sc}

From ATL^*_{sc} *to* ATL_{sc} Surprisingly, strategy contexts bring ATL_{sc} to the same expressiveness as ATL^*_{sc}: any ATL^*_{sc} formula can be translated into an equivalent ATL_{sc} formula. The main idea is to replace the (implicit) universal quantification over outcomes with explicit universal quantification over strategies. This way, all players are assigned a strategy in the context. In that case, there is only one outcome (because our CGSs are deterministic), so that we can insert empty strategy quantifier $\langle\cdot\varnothing\cdot\rangle$ in front of any temporal modality.

Notice that the transformation has to depend on the original context. Actually, for any coalition A, our construction transforms a formula $\Phi\in\mathsf{ATL}^*_{sc}$ into an ATL_{sc} formula $\widehat{\Phi}^A$ that is equivalent to Φ under any context f of domain A, *i.e.*, such that for any CGS $\mathcal{C}$, any state q, and any context f with $\mathsf{dom}(f)=A$, we have $\mathcal{C},q\models_f\Phi$ if, and only if, $\mathcal{C},q\models_f\widehat{\Phi}^A$.

Let Φ be an ATL^*_{sc} formula, and $(B,B')\in 2^{\mathsf{Agt}(\Phi)}\times(2^{\mathsf{Agt}(\Phi)}\cup\mathsf{Agt}_{\mathcal{C}})$ be two coalitions. These coalitions will be used to represent the set $B\cup(\mathsf{Agt}_{\mathcal{C}}\setminus B')$ of players that are *artificially* assigned a strategy in our translation; this will precisely correspond to the complement of the domain of the context in which the formula is evaluated. We allow B' to take the special value $\mathsf{Agt}_{\mathcal{C}}$, in order to keep our translation independent of the underlying CGS. We now define $\widehat{\Phi}^{[B,B']}$ inductively as follows:

$$\begin{aligned}
\widehat{P}^{[B,B']} &= P\\
\widehat{\neg\varphi}^{[B,B']} &= \neg\widehat{\varphi}^{[B,B']} & \widehat{\varphi\wedge\psi}^{[B,B']} &= \widehat{\varphi}^{[B,B']}\wedge\widehat{\psi}^{[B,B']}\\
\widehat{\mathbf{X}\varphi}^{[B,B']} &= \langle\cdot\varnothing\cdot\rangle\,\mathbf{X}\,\widehat{\varphi}^{[B,B']} & \widehat{\varphi\,\mathbf{U}\,\psi}^{[B,B']} &= \langle\cdot\varnothing\cdot\rangle(\widehat{\varphi}^{[B,B']}\,\mathbf{U}\,\widehat{\psi}^{[B,B']})\\
\widehat{\llparenthesis A\rrparenthesis\varphi}^{[B,B']} &= \widehat{\varphi}^{[B\cup A,B']} & \widehat{\llparenthesis\overline{A}\rrparenthesis\varphi}^{[B,B']} &= \widehat{\varphi}^{[B,B'\cap A]}\\
\widehat{\langle\cdot A\cdot\rangle\varphi}^{[B,B']} &= \langle\cdot A\cdot\rangle[\cdot B\setminus A\cdot][\cdot\overline{B'\cup A}\cdot]\widehat{\varphi}^{[B\setminus A,B'\cup A]}\\
\widehat{\langle\cdot\overline{A}\cdot\rangle\varphi}^{[B,B']} &= \langle\cdot\overline{A}\cdot\rangle[\cdot(B\cap A)\cup(A\setminus B')\cdot]\widehat{\varphi}^{[(B\cap A)\cup(A\setminus B'),\mathsf{Agt}_{\mathcal{C}}]}
\end{aligned}$$

Before stating and proving correctness of this transformation, let us first give some more intuition. Assume the context contains strategies for some coalition B, and consider a formula of the form $\langle\cdot A\cdot\rangle\varphi$. This formula is equivalent to $\langle\cdot A\cdot\rangle[\cdot\overline{A\cup B}\cdot]\varphi'$, where we make the quantification on the strategy of the "free" players explicit. Notice that this explicit quantification modifies the context, which is the reason why φ is updated into φ': this is precisely why we have to keep track of the extra strategies that have been made explicit. Now, consider formula $\langle\cdot\overline{A}\cdot\rangle\varphi$ in a context where coalition $B \cup (\mathsf{Agt}_{\mathcal{C}} \setminus B')$ has been artificially assigned a strategy. Then the "free" players, whose strategies must be quantified universally, are precisely those in $B \cup (\mathsf{Agt}_{\mathcal{C}} \setminus B') \setminus (\mathsf{Agt}_{\mathcal{C}} \setminus A)$. Using the fact that $X \setminus \bar{Y} = X \cap Y$, this coalition is easily proved to be $(B \cap A) \cup (A \setminus B')$.

Clearly, $\widehat{\Phi}^{[B,B']}$ is an ATL_{sc} formula, thanks to the $\langle\cdot\varnothing\cdot\rangle$ inserted in front of the temporal modalities. Notice that the resulting formula only involves coalitions that appear in the original formula, and does not depend on $\mathsf{Agt}_{\mathcal{C}}$ (because $\mathsf{Agt}_{\mathcal{C}} \cup A = \mathsf{Agt}_{\mathcal{C}}$, $A \setminus \mathsf{Agt}_{\mathcal{C}} = \varnothing$, $\mathsf{Agt}_{\mathcal{C}} \cap A = A$, and $[\cdot\overline{\mathsf{Agt}_{\mathcal{C}}}\cdot]\psi$ is equivalent to $[\cdot\varnothing\cdot]\psi$). This transformation achieves the following result (the full proof of which is rather technical, so we moved it to Appendix A).

Theorem 9. *Given a formula $\varphi \in \mathsf{ATL}^*_{sc}$ and a coalition B', there exists an ATL_{sc} formula $\widehat{\varphi}^{[\varnothing,B']}$, involving only players in $\mathsf{Agt}_\varphi \cup B'$, such that for any strategy context f with $\mathsf{dom}(f) = B'$, φ and $\widehat{\varphi}^{[\varnothing,B']}$ are equivalent under context f.*

Remark 10. One can notice that the resulting formula does not make use of $\llparenthesis\text{-}\rrparenthesis$, even when applied to an ATL_{sc} formula. Hence, as a side result, we obtain that the $\llparenthesis\text{-}\rrparenthesis$ operator does not increase the expressive power of ATL_{sc}.

3.3. *Comparison with* ATL

Clearly enough, ATL^* properties can be expressed in ATL^*_{sc}: indeed, the ATL strategy quantifier $\langle\!\langle A\rangle\!\rangle$ is equivalent to $\llparenthesis\overline{\varnothing}\rrparenthesis\langle\cdot A\cdot\rangle$. Notice that following Examples 7 and 8, ATL_{sc} is actually strictly more expressive than ATL^*.

Similarly, CTL^* is translated in ATL^*_{sc} by rewriting $\mathbf{E}\varphi$ as $\langle\cdot\overline{\varnothing}\cdot\rangle\varphi$ and $\mathbf{A}\varphi$ as $\llparenthesis\overline{\varnothing}\rrparenthesis\langle\cdot\varnothing\cdot\rangle\varphi$. Notice that these transformations do not preserve the strategy context, and that they are different from the path quantifiers used in Game Logic (GL) [3]. There, the path quantifiers range over the set of outcomes of the strategies already in use. More precisely, in GL, we have

$$\mathcal{C}, \rho, i \models_f \mathbf{E}_{\mathsf{GL}}\varphi_{\mathsf{p}} \quad \text{iff} \quad \exists \rho' \in \mathsf{Out}(\rho_{\le i}, f).\, \mathcal{C}, \rho', i \models_f \varphi_{\mathsf{p}}$$

The universal path quantifier is dual. Both path quantifiers can be expressed in ATL^*_{sc} as follows:

$$\mathbf{E}_{\mathsf{GL}}\varphi_{\mathsf{p}} \equiv \neg\langle\cdot\varnothing\cdot\rangle\neg\varphi_{\mathsf{p}} \qquad \mathbf{A}_{\mathsf{GL}}\varphi_{\mathsf{p}} \equiv \langle\cdot\varnothing\cdot\rangle\varphi_{\mathsf{p}}$$

4. Model checking

Model checking is the problem of deciding whether $\mathcal{C}, q_0 \models \varphi$, for a given CGS $\mathcal{C}$, a state q_0 and a formula φ. In this section, we present an algorithm for model checking ATL_{sc} and ATL^*_{sc}, and study its complexity. Our algorithm is based on a translation of the model-checking problem from ATL^*_{sc} into the model-checking problem for QCTL^*. Using a translation in the other direction, we prove that ATL^*_{sc} model checking is complete for k-EXPTIME (where k is the quantifier height of the formula).

4.1. *From* ATL^*_{sc} *to* QCTL^*

Let $\mathcal{C} = \langle Q, R, \ell, \mathsf{Agt}, \mathcal{M}, \mathsf{Mov}, \mathsf{Edge}\rangle$ be a finite-state CGS, with a finite set of moves $\mathcal{M} = \{m_1, \ldots, m_k\}$ and $\mathsf{Agt} = \{a_1, \ldots, a_n\}$. We consider the following sets of fresh atomic propositions: $\mathsf{P}_Q = \{\mathsf{p}_q \mid q \in Q\}$, $\mathsf{P}^j_{\mathcal{M}} = \{\mathsf{m}^j_1, \ldots, \mathsf{m}^j_k\}$ for every $a_j \in \mathsf{Agt}$, and write $\mathsf{P}_{\mathcal{M}} = \bigcup_{a_j \in \mathsf{Agt}} \mathsf{P}^j_{\mathcal{M}}$. Let $\mathcal{S}_{\mathcal{C}}$ be the Kripke structure $\langle Q, R, \ell_+\rangle$ where for any state q, we have: $\ell_+(q) = \ell(q) \cup \{\mathsf{p}_q\}$. A strategy for an agent a_j from q can be seen as a function $f_j\colon qQ^*_{\mathcal{S}} \to \mathsf{P}^j_{\mathcal{M}}$ labeling the computation tree of $\mathcal{S}_{\mathcal{C}}$ with propositions in $\mathsf{P}^j_{\mathcal{M}}$.

Let C be a coalition in Agt, $f_C \in \mathsf{Strat}(C)$ be a strategy context, and $\Phi \in \mathsf{ATL}^*_{sc}$. We transform the question whether $\mathcal{C}, q \models_{f_C} \Phi$ into an instance of QCTL^* model checking over $\mathcal{S}_{\mathcal{C}}$ (assuming the tree semantics). For this, we define a QCTL^* formula $\overline{\Phi}^C$ inductively. Except for strategy quantifiers, the translation is straightforward:

$$\begin{aligned}
\overline{\varphi \wedge \psi}^C &= \overline{\varphi}^C \wedge \overline{\psi}^C & \qquad \overline{\neg\varphi}^C &= \neg\,\overline{\varphi}^C & \qquad \overline{p}^C &= p \\
\overline{\llparenthesis A\rrparenthesis\varphi_{\mathsf{s}}}^C &= \overline{\varphi_{\mathsf{s}}}^{C\smallsetminus A} & \qquad \overline{\varphi_{\mathsf{p}}\,\mathbf{U}\,\psi_{\mathsf{p}}}^C &= \overline{\varphi_{\mathsf{p}}}^C\,\mathbf{U}\,\overline{\psi_{\mathsf{p}}}^C & \qquad \overline{\mathbf{X}\varphi_{\mathsf{p}}}^C &= \mathbf{X}\overline{\varphi_{\mathsf{p}}}^C
\end{aligned}$$

For a formula $\langle\cdot A\cdot\rangle\varphi_{\mathsf{p}}$ with $A = \{a_{j_1}, \ldots, a_{j_l}\}$ and coalition C s.t. $A \cup C \neq \varnothing$, we let:

$$\begin{aligned}
\overline{\langle\cdot A\cdot\rangle\varphi_{\mathsf{p}}}^C = {}& \exists \mathsf{m}^{a_{j_1}}_1 \ldots \mathsf{m}^{a_{j_1}}_k \ldots \mathsf{m}^{a_{j_l}}_1 \ldots \mathsf{m}^{a_{j_l}}_k .\mathsf{p}_{\mathsf{out}}. \\
& \Big(\Phi_{\mathsf{strat}}(A) \wedge \Phi_{\mathsf{out}}(A \cup C) \wedge \mathbf{A}\Big(\mathbf{G}\mathsf{p}_{\mathsf{out}} \Rightarrow \overline{\varphi_{\mathsf{p}}}^{C\cup A}\Big)\Big)
\end{aligned}$$

with

$$\Phi_{\mathsf{strat}}(B) = \bigwedge_{a\in B}\bigwedge_{q\in Q} \mathbf{AG}\Big(\mathsf{p}_q \Rightarrow \bigvee_{m_i\in\mathsf{Mov}(q,a)} (\mathsf{m}_i^a \wedge \bigwedge_{j\neq i} \neg\, \mathsf{m}_j^a)\Big)$$

$$\Phi_{\mathsf{out}}(B) = \mathsf{p}_{\mathsf{out}} \wedge \mathbf{AG}\Big[\neg\, \mathsf{p}_{\mathsf{out}} \Rightarrow \mathbf{AX}\neg\, \mathsf{p}_{\mathsf{out}}\Big] \wedge \mathbf{AG}\Big[\mathsf{p}_{\mathsf{out}} \Rightarrow \bigvee_{q\in Q}\Big(\mathsf{p}_q \wedge \bigvee_{\overline{m}\in\mathsf{Mov}(q,B)} \big(\mathsf{p}_{\overline{m}} \wedge \mathbf{AX}\big((\bigvee_{q'\in\mathsf{Next}(q,B,\overline{m})} \mathsf{p}_{q'}) \Leftrightarrow \mathsf{p}_{\mathsf{out}}\big)\big)\Big)\Big]$$

for any non-empty coalition $B \subseteq \mathsf{Agt}$. In $\Phi_{\mathsf{out}}(B)$, for $\overline{m} = (m_{i_a})_{a\in B} \in \mathsf{Mov}(q, B)$, we write $\mathsf{p}_{\overline{m}}$ for the propositional formula $\bigwedge_{a\in B} \mathsf{m}_{i_a}^a$ characterizing $\overline{m}$. Formula $\Phi_{\mathsf{strat}}(B)$ ensures that the labeling of propositions $\mathsf{m}_i^{a_j}$ describes a feasible strategy for B; formula $\Phi_{\mathsf{out}}(B)$ ensures that the outcomes of the strategy assigned to coalition B (and only those outcomes) are labeled by the atomic proposition $\mathsf{p}_{\mathsf{out}}$. Note that $\Phi_{\mathsf{out}}(B)$ is based on the transition table Edge of $\mathcal{C}$ (via $\mathsf{Next}(q, B, \overline{m})$) and its size is in $O(|Q|^2 \cdot |\mathcal{M}|^{|B|})$ (*i.e.*, in $O(|Q| \cdot |\mathsf{Edge}|)$). When $A \cup C = \varnothing$, we define $\overline{\langle\cdot\varnothing\cdot\rangle\varphi_{\mathsf{p}}}^{\varnothing} = \mathbf{A}\overline{\varphi_{\mathsf{p}}}^{\varnothing}$. For $\overline{\llparenthesis \overline{A} \rrparenthesis\varphi}^C$ and $\overline{\langle\cdot\overline{A}\cdot\rangle\varphi_{\mathsf{p}}}^C$, we simply replace A with $\mathsf{Agt}\setminus A$ and apply the same definitions as above.

In this translation, each strategy quantifier in the original ATL^*_{sc} formula induces a strategy quantifier in the QCTL^* formula (except $\langle\cdot\varnothing\cdot\rangle$ interpreted in an empty context, as there is no need to mark the outcomes in this case). There is another exception to this rule, in which the translation can be adapted to not involve quantification: in some cases, it is not necessary to *recompute* the labeling with $\mathsf{p}_{\mathsf{out}}$. This occurs with formulas of the form $\overline{\langle\cdot\varnothing\cdot\rangle\varphi_{\mathsf{p}}}^C$ in the direct scope of another strategy quantifier, with no $\llparenthesis\cdot\rrparenthesis$ operator in-between; in such cases, $\overline{\langle\cdot\varnothing\cdot\rangle\varphi_{\mathsf{p}}}^C$ can be defined as $\mathbf{A}(\mathbf{G}\mathsf{p}_{\mathsf{out}} \Rightarrow \overline{\varphi_{\mathsf{p}}}^C)$, using propositions $\mathsf{p}_{\mathsf{out}}$ resulting from the previous quantification. In the following, such occurrences of modality $\langle\cdot\varnothing\cdot\rangle$ (in an empty context or in the scope of another strategy quantifier with no $\llparenthesis\cdot\rrparenthesis$ in-between) are said to be *trivial*.

Let $\mathcal{T}_{\mathcal{C}} = \langle T, \ell\rangle$ be the computation tree of the Kripke structure $\mathcal{S}_{\mathcal{C}}$, ρ be a path in $\mathcal{S}_{\mathcal{C}}$, n be a position along ρ, and f_A be a strategy for some coalition A from $\rho_{\leq n}$. Let ℓ' be a labeling extending ℓ with propositions $(\mathsf{m}_i^a)_{a\in A, 1\leq i\leq k}$ and $\mathsf{p}_{\mathsf{out}}$. We say that ℓ' is an f_A-labeling after $\rho_{\leq n}$ if, for every finite path π containing $\rho_{\leq n}$ as a prefix, it holds $\mathsf{m}_i^a \in \ell'(\pi)$ if, and only if, $f_A(a)(\pi) = m_i$, and $\mathsf{p}_{\mathsf{out}} \in \ell'(\pi)$ if, and only if, π is compatible with f_A after $\rho_{\leq n}$.

For such an f_A-labeling ℓ', we clearly have $\langle T, \ell'\rangle, \rho, n \models \Phi_{\mathsf{strat}}(A) \wedge \Phi_{\mathsf{out}}(A)$. The converse is also true: if $\langle T, \ell'\rangle, \rho, n \models \Phi_{\mathsf{strat}}(A) \wedge \Phi_{\mathsf{out}}(A)$, then propositions $(\mathsf{m}_i^a)_{a\in A, 1\leq i\leq k}$ encode a strategy f_A in the subtree rooted at node $\rho_{\leq n}$, and ℓ' is an f_A-labeling after $\rho_{\leq n}$. Indeed, consider a path π from $\rho_{\leq n}$ and a position $i \geq n$. Then formula $\Phi_{\mathsf{strat}}(A)$ enforces that for all $a \in A$, node $\pi_{\leq i}$ is labeled with some m_i^a corresponding to a move $m_i \in \mathsf{Mov}(q, a)$. Now, by induction, one easily shows that the node corresponding to finite paths that are compatible with f_A after $\rho_{\leq n}$ are labeled with $\mathsf{p}_{\mathsf{out}}$, while the other nodes are not labeled with $\mathsf{p}_{\mathsf{out}}$. Notice that this is easily extended to include a strategy context g_C.

The following result is a direct consequence of the above:

Theorem 11. *Let ρ be an infinite path in a CGS $\mathcal{C}$, and n be a position along ρ. Let Φ be an ATL^*_{sc} formula, and f_C be a strategy context for some coalition C. Let $\mathcal{T}_{\mathcal{C}}(\rho(0)) = \langle T, \ell\rangle$ be the computation tree $\mathcal{S}_{\mathcal{C}}$ from $\rho(0)$, and ℓ_{f_C} be an f_C-labeling extending ℓ. Then $\mathcal{C}, \rho, n \models_{f_C} \Phi$ if, and only if, $\langle T, \ell_{f_C}\rangle, \rho, n \models \overline{\Phi}^C$.*

Combined with the (non-elementary) decision procedure for QCTL^* model checking, we get a model-checking algorithm for model checking ATL^*_{sc}.

We now consider complexity issues more precisely. We have

$$|\overline{\Phi}^{\varnothing}| = O(|\Phi| \cdot |Q|(|\mathsf{Agt}| \cdot |\mathcal{M}|^2 + |\mathsf{Edge}|)),$$

which is polynomial in $|\Phi|$ and $|\mathcal{C}|$. Moreover, $\overline{\Phi}^{\varnothing}$ belongs to $\mathsf{Q}^k\mathsf{CTL}^*$, where k is the *depth of* $\langle\cdot\text{-}\cdot\rangle$ in Φ (which we define as the maximal number of nested *non-trivial* $\langle\cdot\text{-}\cdot\rangle$ quantifiers). Given an ATL^*_{sc} formula Φ of depth k and a CGS $\mathcal{C}$, the reduction yields a model checking algorithm running in $(k + 1)$-$\mathsf{EXPTIME}$ [24].

Finally note that when starting from an ATL_{sc} formula, the QCTL^* formula we obtain can easily be translated into QCTL: it contains the $\mathsf{CTL}+$ formula $\mathbf{A}(\mathbf{G}\mathsf{p}_{\mathsf{out}} \Rightarrow \varphi_{\mathsf{p}})$, which can be succinctly written in CTL (for instance, when $\varphi_{\mathsf{p}} = \varphi_{\mathsf{s}}\ \mathbf{U}\ \psi_{\mathsf{s}}$, $\mathbf{A}(\mathbf{G}\mathsf{p}_{\mathsf{out}} \Rightarrow \varphi_{\mathsf{p}})$ is equivalent to $\mathbf{A}(\varphi_{\mathsf{s}} \wedge \mathsf{p}_{\mathsf{out}})\ \mathbf{U}\ (\psi_{\mathsf{s}} \vee \neg\, \mathsf{p}_{\mathsf{out}})$). Thus it provides a $\mathsf{Q}^k\mathsf{CTL}$ formula whose size is in $O(|\Phi| \cdot |Q|(|\mathsf{Agt}| \cdot |\mathcal{M}|^2 + |\mathsf{Edge}|))$ if Φ is an ATL_{sc} formula of $\langle\cdot\text{-}\cdot\rangle$-depth k. This yields a model-checking algorithm in k-$\mathsf{EXPTIME}$.

4.2. From QCTL^* *back to* ATL^*_{sc}

We now propose a reduction in the converse direction, from an instance of the QCTL^* model-checking problem into an instance of the ATL^*_{sc} model-checking problem. Intuitively, strategies in the resulting game will correspond to labeling with atomic propositions in the original Kripke structure.

Let Φ be a QCTL formula and $\mathcal{S} = \langle Q, R, \ell\rangle$ be a Kripke structure. W.l.o.g., we assume that every atomic proposition in Φ is quantified at most once. We write $\mathsf{AP}_f(\Phi)$ for the set of free atomic propositions in Φ (which are intended to already label the Kripke structure), and $\mathsf{AP}_{\mathcal{Q}}(\Phi) = \{P_1, \ldots, P_k\}$ for the set of atomic propositions that are quantified in Φ. We build a TBGS $\mathcal{C}_{\mathcal{S}}$ and an ATL_{sc} formula $\widetilde{\Phi}$ such that $\mathcal{S}, q \models_t \Phi$ if, and only if, $\mathcal{C}_{\mathcal{S}}, q \models \widetilde{\Phi}$.

The game $\mathcal{C}_{\mathcal{S}} = \langle Q', R', \ell', \mathsf{Agt}, \mathcal{M}, \mathsf{Mov}, \mathsf{Edge}\rangle$ is defined as follows. The set of agents is $\mathsf{Agt} = \{a_0, \ldots, a_k\}$: Player a_0 is in charge of selecting the transitions in $\mathcal{S}$, while Player a_i (for $i \geq 1$) has to decide the truth value of P_i. The set of states Q' is $Q \cup \{c_{q,i} \mid i = 1, \ldots, k\} \cup \{p_i \mid i = 0, \ldots, k\}$: states $q \in Q$ are controlled by a_0, while states $c_{q,i}$ are controlled by a_i. States p_i only carry a self-loop, and are not explicitly controlled. The transition set R' contains every transition $(q, q') \in R$, and also transitions $(q, c_{q,i})$, $(c_{q,i}, p_i)$, and $(c_{q,i}, p_0)$ for $i = 1, \ldots, k$ and $q \in Q$. The labeling ℓ' is as follows: $\ell'(q) = \ell(q) \cup \{P_Q\}$ if $q \in Q$ (P_Q is a fresh atomic proposition), $\ell'(c_{q,i}) = \ell'(p_0) = \varnothing$, and $\ell'(p_i) = P_i$ for $i \geq 1$.

In a state $q \in Q$, a_0 can choose either a successor state q' (*i.e.*, an $\mathcal{S}$-transition (q, q')) or some $c_{q,i}$, the latter choice being used to check whether P_i holds true in q. Indeed in $c_{q,i}$, a_i has two available moves: move m_1 goes to P_i, and while mode m_0 goes to P_0. Thus as soon as a_i has selected a strategy, $c_{q,i}$ has a unique successor; this encodes the labeling for atomic proposition $P_i \in \mathsf{AP}_{\mathcal{Q}}(\Phi)$. Note also that for any path in $\mathcal{C}_{\mathcal{S}}$ of the form $\rho \cdot c_{q,i}$, ρ is a path in $\mathcal{S}$ ending in q. Finally note that as $\mathcal{C}_{\mathcal{S}}$ is a TBGS, its size is in $O(|Q| \cdot |\Phi| + |R|)$, *i.e.* in $O(|\mathcal{S}| \cdot |\Phi|)$.

Following the ideas above, we define $\widetilde{\Phi}$ inductively:

$$\widetilde{\neg\psi} = \neg\widetilde{\varphi} \qquad \widetilde{\varphi \wedge \psi} = \widetilde{\varphi} \wedge \widetilde{\psi} \qquad \widetilde{\exists P_i.\varphi} = \langle\cdot a_i\cdot\rangle\widetilde{\varphi}$$

$$\widetilde{\mathbf{X}\varphi} = \mathbf{X}\widetilde{\varphi} \qquad \widetilde{\varphi\,\mathbf{U}\,\psi} = \widetilde{\varphi}\,\mathbf{U}\,\widetilde{\psi} \qquad \widetilde{\mathsf{E}\varphi} = \langle\cdot a_0\cdot\rangle(\mathbf{G}P_Q \wedge \widetilde{\varphi})$$

$$\widetilde{P_i} = \begin{cases} \langle\cdot a_0\cdot\rangle\,\mathbf{X}\,\langle\cdot a_0\cdot\rangle\,\mathbf{X}\,P & \text{if } P \in \mathsf{AP}_{\mathcal{Q}}(\Phi) \\ P & \text{otherwise} \end{cases}$$

The size of $\widetilde{\Phi}$ is in $O(|\Phi|)$. We state the correctness of the reduction as follows:

Proposition 12. *Let Φ be a QCTL^* formula with $\mathsf{AP}_{\mathcal{Q}}(\Phi) = \{P_1, \ldots, P_k\}$ and ψ be a Φ-subformula. Let $\mathcal{I}$ be the indices of propositions in $\mathsf{AP}_f(\psi) \cap \mathsf{AP}_{\mathcal{Q}}(\Phi)$. Let $\mathcal{S} = \langle Q, R, \ell\rangle$ be a KS, ρ be a path in $\mathcal{S}$ and n be a position along ρ. Let $\mathcal{T}_{\mathcal{S}}(\rho(0)) = \langle T, \ell_{\mathcal{I}}\rangle$ be the computation tree from $\rho(0)$ with a labeling function $\ell_{\mathcal{I}}$ that extends ℓ for $\{P_i \mid i \in \mathcal{I}\}$. Let f be the strategy context for $\{a_i\}_{i\in\mathcal{I}}$ such that: $f(a_i)(\rho' \cdot c_{q,i}) = \mathsf{m}_1$ if, and only if, $\ell_{\mathcal{T}}(\rho') \ni P_i$ for every $\mathcal{T}$-node ρ'. Then we have:*

$$\langle T, \ell_{\mathcal{I}}\rangle, \rho, n \models_s \psi \quad \textit{iff} \quad \mathcal{C}_{\mathcal{S}}, \rho, n \models_f \widetilde{\psi}$$

Proof. The proof is based on the fact that any strategy for agent a_i in $\mathcal{C}_{\mathcal{S}}$ corresponds to a (unique) P_i labeling of T. Indeed such a strategy is a mapping from paths of the form $\rho' \cdot c_{q,i}$ with $\rho' \in Q^+$ and as noticed above, we have: for any such a $\mathcal{C}_{\mathcal{S}}$ path $\rho' \cdot c_{q,i}$, ρ' is a path in $\mathcal{S}$ ending in q which implies that ρ is a state of T. Using this observation, the proof is straightforward. □

A complexity lower bound for ATL_{sc} model checking Now consider a model checking instance $\mathcal{C} \models \Phi$ with $\Phi \in \mathsf{EQ}^k\mathsf{CTL}$. This problem is k-$\mathsf{EXPTIME}$-complete [24]. Now we can adapt the previous reduction from QCTL to ATL_{sc} in order to obtain a formula with $\langle\cdot\text{-}\cdot\rangle$-depth equals to k:

- First we can replace all occurrences of the quantifier $\langle\cdot a_0\cdot\rangle$ in $\widetilde{\Phi}$ with $\neg\langle\cdot\varnothing\cdot\rangle\neg$, because every such subformula is interpreted in a context where every agent in $\{a_1, \ldots, a_k\}$ has a fixed strategy, so that quantifying on the ability of the last agent (a_0) to select a path in the structure is equivalent to looking for a path in the outcomes of the context (and $\mathbf{E}_c \equiv \neg\langle\cdot\varnothing\cdot\rangle\neg$). Notice that such $\langle\cdot\varnothing\cdot\rangle$ correspond to *trivial* quantification, because no $\llparenthesis\text{-}\rrparenthesis$ is used in the formula.
- Second, formula $\widetilde{\mathbf{E}\varphi}$ is in ATL^*_{sc}, as it involves a conjunction of path formulas in the scope of a strategy quantifier. However, this is easily overcome in pretty much the same way as we did at the end of Section 4.1.

From this reduction, we get a k-$\mathsf{EXPTIME}$-hardness lower bound for ATL_{sc} model checking (notice that this already holds for TBGSs). The same approach can be followed for $\mathsf{EQ}^k\mathsf{CTL}^*$ and ATL^*_{sc}, yielding a $(k+1)$-$\mathsf{EXPTIME}$-hardness bound. As a result:

Theorem 13. *Model checking the fragment of ATL_{sc} (resp. ATL^*_{sc}) with at most k non-trivial nested strategy quantifiers is k-$\mathsf{EXPTIME}$-hard (resp. $(k+1)$-$\mathsf{EXPTIME}$-hard), even for TBGS.*

To sum up our results about model checking, we have:

Corollary 14. *Model checking ATL_{sc} and ATL^*_{sc} is Tower-complete. More precisely, model checking the fragment of ATL_{sc} (resp. ATL^*_{sc}) with k non-trivial nested strategy quantifiers is k-$\mathsf{EXPTIME}$-complete (resp. $(k+1)$-$\mathsf{EXPTIME}$-complete).*

5. Satisfiability

Satisfiability checking is the problem of deciding whether there exists a CGS $\mathcal{C}$ and a state q_0 such that $\mathcal{C}, q_0 \models \varphi$, for a given formula φ. The above translation to QCTL* works for model checking, but does not extend to satisfiability: the QCTL* formula we build depends both on the formula and on the structure. Actually, as was proved recently in [35], satisfiability is undecidable for ATL_{sc}, both when looking for infinite or finite CGS. For the sake of completeness, and because it has interesting consequences, we sketch a (modified) proof of this result in this section.

We then establish decidability of satisfiability in two different settings: first when restricting to turn-based games, and then in the case where the set of actions allowed to the players is fixed. A consequence of our decidability proofs (based on automata constructions) is that in both settings, ATL_{sc} does have the finite-model property (thanks to Rabin's regularity theorem).

Before we proceed to the algorithms for satisfiability, we prove a generic result about the number of players needed in a game in order to satisfy a formula involving a given set of agents. This result has already been proved for ATL (*e.g.* in [37]). Given a formula $\Phi \in \text{ATL}^*_{sc}$, we use $\text{Agt}(\Phi)$ to denote the set of all agents involved in the strategy quantifiers of Φ.

Proposition 15. *An ATL^*_{sc} formula Φ is satisfiable if, and only if, it is satisfiable in a CGS with $|\text{Agt}(\Phi)| + 1$ agents.*

Proof. Assume that Φ is satisfied in a CGS $\mathcal{C} = \langle Q, R, \ell, \text{Agt}, \mathcal{M}, \text{Mov}, \text{Edge}\rangle$. If $|\text{Agt}| \leq |\text{Agt}(\Phi)|$, it suffices to add extra idle players in $\mathcal{C}$. Otherwise, if $|\text{Agt}| > |\text{Agt}(\Phi)| + 1$, we can replace the agents $\{b_1, \ldots, b_k\}$ in Agt that do not belong to $\text{Agt}(\Phi)$ by a unique agent a mimicking the actions of the removed players. Notice that this requires extending the set of moves for Player a to $\mathcal{M}^k$. □

Note also that this result still holds when considering turn-based CGS.

5.1. General case

In [35], Troquard and Walther show that satisfiability of ATL_{sc} is undecidable. The proof consists in reducing the satisfiability problem for the modal logic $S5^n$ to the satisfiability problem for ATL_{sc}. The construction is elegant and induces several important results, which is the reason why we include it here (with a few changes).

The multi-modal logic $S5^n$ The logic $S5^n$ is a multidimensional modal logic [23], whose formulas are built from Boolean operators, atomic propositions $P \in \text{AP}$ and modalities $\Diamond_i$. These formulas are interpreted over models $\mathcal{M} = \langle \mathcal{F}, \mathcal{V}\rangle$ where $\mathcal{F}$ is a product frame $W_1 \times \ldots \times W_n$, and $\mathcal{V}$ is a valuation for atomic propositions over $\mathcal{F}$. The (implicit) transition relation over W_i is universal: for any world $w = (w_1, \ldots, w_n)$ and any $w'_i \in W_i$, there is an i-transition to $(w_1, \ldots, w_{i-1}, w'_i, w_{i+1}, \ldots, w_n)$. This provides the semantics of $\Diamond_i\varphi$: $\mathcal{M}, w \models \Diamond_i \varphi$ if, and only if, there exists $w'_i \in W_i$ such that $\mathcal{M}, w[w_i \to w'_i] \models \varphi$. When $n > 2$ (which we assume from now on), satisfiability (both over finite and infinite models) is undecidable for $S5^n$ [26], and $S5^n$ does not have the finite-model property [22].

Let Φ be an $S5^n$ formula. From Φ, we build an ATL_{sc} formula $\widehat{\Phi}$ inductively as follows:

$$\widehat{\varphi \wedge \psi} = \widehat{\varphi} \wedge \widehat{\psi} \qquad \widehat{\neg\psi} = \neg\,\widehat{\psi} \qquad \widehat{P} = \langle\cdot\varnothing\cdot\rangle\, \mathbf{X}\, P \qquad \widehat{\Diamond_i \psi} = \langle\cdot a_i \cdot\rangle\, \widehat{\psi}$$

The following result connects both satisfiability problems:

Proposition 16. *(See [35].) Let Φ be an $S5^n$ formula and $\widehat{\Phi}$ be the resulting ATL_{sc} formula, obtained as above. Then Φ is satisfiable in a finite (resp. infinite) $S5^n$ model if, and only if, $\langle\cdot\varnothing\cdot\rangle\,\widehat{\Phi}$ is satisfiable in a finite (resp. infinite) CGS.*

Proof. First assume that there exists a model $\mathcal{M} = (\mathcal{F}, \mathcal{V})$ for Φ, with $\mathcal{F} = W_1 \times \ldots \times W_n$. Take w such that $\mathcal{M}, w \models \Phi$. For every i, the states in W_i are denoted $w^i_1, w^i_2, \ldots$. We define a CGS $\mathcal{C}_{\mathcal{M}} = \langle Q, R, \ell, \text{Agt}, \mathcal{M}, \text{Mov}, \text{Edge}\rangle$ with $\mathcal{M}$ as its underlying transition system: $Q = \mathcal{F}$, $R = Q \times Q$, and $\ell(w) = \mathcal{V}(w)$. We let $\text{Agt} = \{a_1, \ldots, a_n\}$. The action alphabet $\mathcal{M}$ is $\{1, \ldots, \max_{1 \leq i \leq n} |W_i|\}$ if $\mathcal{M}$ is finite, and $\mathbb{N}_{>0}$ otherwise. In every world $w \in Q$, Player a_i can choose the next position in W_i: in other terms, $\text{Mov}(w, a_i) = \{1, \ldots, |W_i|\}$ if W_i is finite, or $\mathbb{N}_{>0}$ otherwise, and $\text{Edge}(w, \overline{m}) = w_{\overline{m}}$ with $w_{\overline{m}} = \langle w^1_{m_1}, w^2_{m_2}, \ldots, w^n_{m_n}\rangle$ when $\overline{m}$ is $\langle m_1, \ldots, m_n\rangle$. Note that as $\mathcal{M}$ is universal, the transition table does not depend on the current state w. As a world w in $\mathcal{M}$ is also a state in $\mathcal{C}_{\mathcal{M}}$ and also corresponds to a move in the game structure, we might abusively write $\text{Edge}(w, \overline{m}) = \overline{m}$ or $\text{Edge}(w, w') = w'$ in the sequel.

In our reduction, formula $\widehat{\Phi}$ only involves non-nested occurrences of the $\mathbf{X}$ modality. Hence we are only interested in the first move proposed by strategies. Given a world w in $\mathcal{F}$, we write F_w for the class of all strategies for Agt such that $F_w(\varepsilon) = w$. In other terms, any strategy in F_w enforces the first transition to go to w. Now we can easily see that, for every

$S5^n$ formula Φ and for any two worlds w and w':

$$\mathcal{M}, w \models \Phi \quad \Leftrightarrow \quad \mathcal{C}_{\mathcal{M}}, w' \models_{F_w} \widehat{\Phi}$$

The proof is by structural induction over Φ. The cases of Boolean operators are direct. We only consider $\Diamond_i$ and propositions P:

- When $\Phi = P$: $\mathcal{M}, w \models P$ is equivalent to $P \in \mathcal{V}(w)$, which in turn is equivalent to $\mathcal{C}_{\mathcal{M}}, w' \models_{F_w} \langle\cdot\varnothing\cdot\rangle\, \mathbf{X}\, P$ because the strategy F_w ensures a first transition to w, where P holds true.
- When $\Phi = \Diamond_i \psi$: if $\mathcal{M}, w \models \Diamond_i \psi$, then there exists w'' such that $w''_j = w_j$ for $j \neq i$ and $\mathcal{M}, w'' \models \psi$. By induction hypothesis, we get $\mathcal{C}_{\mathcal{M}}, w' \models_{F_{w''}} \widehat{\psi}$, which implies $\mathcal{C}_{\mathcal{M}}, w' \models_{F_w} \langle\cdot a_i \cdot\rangle\, \widehat{\psi}$ because $F_{w''}$ is clearly equivalent (when considering only the first transition) to F_w modified by a new move for Player a_i.

Now consider any state w' in Q. Since $\mathcal{M}, w \models \Phi$, we have $\mathcal{C}_{\mathcal{M}}, w' \models \langle\cdot\overline{\varnothing}\cdot\rangle\, \widehat{\Phi}$, since the strategy quantifier $\langle\cdot\overline{\varnothing}\cdot\rangle$ (*i.e.* $\langle\cdot\mathsf{Agt}\cdot\rangle$) allows us to select a strategy in F_w to ensure $\mathcal{C}_{\mathcal{M}}, w' \models_{F_w} \widehat{\Phi}$. Thus $\langle\cdot\overline{\varnothing}\cdot\rangle\, \widehat{\Phi}$ is satisfiable.

Now assume that there exists $\mathcal{C} = \langle Q, R, \ell, \mathsf{Agt}, \mathcal{M}, \mathsf{Mov}, \mathsf{Edge}\rangle$ and $q \in Q$ such that $\mathcal{C}, q \models \langle\cdot\overline{\varnothing}\cdot\rangle\, \widehat{\Phi}$. We first show that there is such a structure involving n agents. We then deduce that the $S5^n$ formula Φ is satisfiable.

Assume that $\mathcal{C}$ involves $n+1$ players $\{a_0, a_1, \ldots, a_n\}$ (following Proposition 15). As $\mathcal{C}, q \models \langle\cdot\overline{\varnothing}\cdot\rangle\, \widehat{\Phi}$, there exists a strategy F for $\{a_0, a_1, \ldots, a_n\}$ such that $\mathcal{C}, q \models_F \widehat{\Phi}$. Pick such an F. In $\widehat{\Phi}$, the strategy quantifiers only deal with Players a_1 to a_n. The strategy (and in particular the first move) for a_0 is fixed by F, and is not updated by $\widehat{\Phi}$. Now, consider the structure $\mathcal{C}' = \langle Q, R', \ell, \{a_1, \ldots, a_n\}, \mathcal{M}, \mathsf{Mov}', \mathsf{Edge}'\rangle$, in which $\mathsf{Edge}'(q, \langle m_1, \ldots, m_n\rangle)$ is defined as $\mathsf{Edge}(q, \langle m_0, m_1, \ldots, m_n\rangle)$, where m_0 is the first move proposed by F_{a_0}. Then $\mathcal{C}', q \models \langle\cdot\overline{\varnothing}\cdot\rangle\, \widehat{\Phi}$.

We now pick an n-player CGS $\mathcal{C}$ such that $\mathcal{C}, q \models \langle\cdot\overline{\varnothing}\cdot\rangle\, \widehat{\Phi}$. We define $\mathcal{M}_{\mathcal{C}} = (\mathcal{F}, \mathcal{V})$ as follows: $\mathcal{F} = W_1 \times \ldots \times W_n$ with $W_i = \mathsf{Mov}(q, a_i)$, and $\mathcal{V}(\overline{m}) = \ell(\mathsf{Edge}(q, \overline{m}))$. In other terms, the states of $\mathcal{M}_{\mathcal{C}}$ are the move vectors of $\mathcal{C}$, and they are labeled with the atomic propositions of the state they lead to. Associated with a universal relation, this defined an $S5^n$ model. We now show that it is a model for Φ.

Given a strategy F in $\mathcal{C}$ for every player, $F(q)$ defines a unique move vector $\overline{m_{F(q)}}$ from q; it also corresponds to some world $w_{F(q)}$ in $\mathcal{M}_{\mathcal{C}}$. We clearly have:

$$\mathcal{C}, q \models_F \widehat{\Phi} \quad \Leftrightarrow \quad \mathcal{M}_{\mathcal{C}}, w_{F(q)} \models \Phi$$

The proof works exactly as in the previous case. Finally we have $\mathcal{C}, q \models \langle\cdot\overline{\varnothing}\cdot\rangle\, \widehat{\Phi}$, and then there exists a complete strategy F such that $\mathcal{C}, q \models_F \widehat{\Phi}$. From the previous result, we get $\mathcal{M}_{\mathcal{C}}, w_{F(q)} \models \Phi$, and then Φ is satisfiable.

Finally note that $\mathcal{M}$ is infinite if, and only if, $\mathcal{C}_{\mathcal{M}}$ has an infinite action alphabet. Conversely, $\mathcal{C}$ has an infinite action alphabet if, and only if, $\mathcal{M}_{\mathcal{C}}$ is infinite. Therefore Φ is finitely (resp. infinitely) satisfiable if, and only if, $\langle\cdot\overline{\varnothing}\cdot\rangle\, \widehat{\Phi}$ is satisfiable in a finite (resp. infinite) CGS.[2] □

Proposition 16 and its proof entail the following results:

Corollary 17.

- ATL_{sc} *does not have the finite-model property.*
- ATL_{sc} *does not have the finite-branching property.*
- *Satisfiability of* ATL_{sc} *is undecidable for finite or infinite CGS [35].*

5.2. *Turn-based games*

In this section, we consider the restriction of satisfiability to turn-based games: given an ATL_{sc} formula, we look for a *turn-based* game structure satisfying Φ. As we now explain, this problem turns out to be decidable.

Let Φ be an ATL_{sc} formula. Write $\mathsf{Agt}(\Phi) = \{a_1, \ldots, a_n\}$ for the set of players involved in Φ. Following Proposition 15, let Agt be the set $\mathsf{Agt}(\Phi) \cup \{a_0\}$, where a_0 is an additional player. Pick a TBGS $\mathcal{C}$, and consider its computation tree

[2] Notice that finiteness refers here to the total size of the CGS, not only its number of states.

$\mathcal{T}_{\mathcal{C}} = \langle T, \ell\rangle$. Since $\mathcal{C}$ is turn-based, we may assume that ℓ labels each node of the tree with the owner of the corresponding state. Formally, for each node $\pi \in T$, we have $\ell(\pi) \ni \mathsf{turn}_j$ if, and only if, $a_j = \mathsf{Own}(\mathsf{last}(\pi))$. A strategy for an agent a_j is then encoded using an atomic proposition mov_j: indeed, a strategy for a_j precisely corresponds to a selection of one *successor* of every turn_j-state (notice that this is a crucial difference with CGSs). The outcomes of a strategy of Player a_j are the runs in which every turn_j-state is followed by a mov_j-state. The decidability proof now consists in using this encoding of strategies together with the translation from ATL^*_{sc} into QCTL^*.

We again consider the computation tree $\mathcal{T}_{\mathcal{C}} = \langle T, \ell\rangle$ of $\mathcal{C}$. Given a coalition $B \subseteq \mathsf{Agt}$ and a strategy f_B for that coalition, a labeling function ℓ' extending ℓ is an *f_B-tb-labeling* if ℓ' labels T with propositions mov_j for all $a_j \in B$ as dictated by f_B, and with proposition $\mathsf{p_{out}}$ in order to mark outcomes. More precisely,

- for any node π labeled with $\mathsf{turn}_j \in B$, we have $\ell'(\pi \cdot q) = \mathsf{mov}_j$ if, and only if, $f_B(j)(\pi) = q$,
- when a state is labeled with $\mathsf{p_{out}}$ and turn_j for some $a_j \in B$, then only its mov_j-successor is labeled with $\mathsf{p_{out}}$; states labeled with $\mathsf{p_{out}}$ and turn_k with $a_k \notin B$, have all their successors labeled with $\mathsf{p_{out}}$; finally, states not labeled with $\mathsf{p_{out}}$ have none of their successors labeled with $\mathsf{p_{out}}$.

Given a coalition C (which we intend to represent the agents that have a strategy in the current context), we translate an ATL^*_{sc} formula Φ into a QCTL^* formula $\widehat{\Phi}^C$ inductively as follows:

$$\widehat{(\!|A|\!)\varphi}^{C} = \widehat{\varphi}^{C\smallsetminus A} \qquad \widehat{\varphi\wedge\psi}^{C} = \widehat{\varphi}^{C} \wedge \widehat{\psi}^{C} \qquad \widehat{\neg\psi}^{C} = \neg\widehat{\varphi}^{C}$$
$$\widehat{\varphi\,\mathbf{U}\,\psi}^{C} = \widehat{\varphi}^{C}\,\mathbf{U}\,\widehat{\psi}^{C} \qquad \widehat{\mathbf{X}\varphi}^{C} = \mathbf{X}\widehat{\varphi}^{C} \qquad \widehat{P}^{C} = P$$

For formulas of the form $\langle\!\cdot A\cdot\!\rangle\varphi$ with $A = \{a_{j_1}, \ldots, a_{j_l}\}$ that correspond to non-trivial $\langle\!\cdot\cdot\!\rangle$ quantifiers, we let:

$$\widehat{\langle\!\cdot A\cdot\!\rangle\varphi}^{C} = \exists \mathsf{mov}_{j_1} \ldots \mathsf{mov}_{j_l}.\mathsf{p_{out}}. \\ \Big(\Phi^{\mathsf{tb}}_{\mathsf{strat}}(A) \wedge \Phi^{\mathsf{tb}}_{\mathsf{out}}(A \cup C) \wedge \mathbf{A}\Big(\mathbf{G}\mathsf{p_{out}} \Rightarrow \widehat{\varphi_{\mathsf{p}}}^{C\cup A}\Big)\Big)$$

with

$$\Phi^{\mathsf{tb}}_{\mathsf{strat}}(B) = \mathbf{AG} \bigwedge_{a_j\in B} (\mathsf{turn}_j \Rightarrow \mathbf{EX}_1 \mathsf{mov}_j)$$

$$\Phi^{\mathsf{tb}}_{\mathsf{out}}(B) = \mathsf{p_{out}} \wedge \mathbf{AG}\big[\neg\,\mathsf{p_{out}} \Rightarrow \mathbf{AX}\neg\,\mathsf{p_{out}}\big] \wedge \mathbf{AG}\Big[\mathsf{p_{out}} \Rightarrow \\ \Big(\bigwedge_{a_j\in B}\big(\mathsf{turn}_j \Rightarrow \mathbf{AX}(\mathsf{mov}_j \Leftrightarrow \mathsf{p_{out}})\big) \wedge \bigwedge_{a_j\notin B}\big(\mathsf{turn}_j \Rightarrow \mathbf{AX}\mathsf{p_{out}}\big)\Big)\Big]$$

where $B \subseteq \mathsf{Agt}$, and $\mathbf{EX}_1\alpha = \mathbf{EX}\alpha\ \wedge\ \forall p.\Big(\mathbf{EX}(\alpha \wedge p) \Rightarrow \mathbf{AX}(\alpha \Rightarrow p)\Big)$ expresses the existence of a unique successor satisfying α. For trivial occurrences of $\langle\!\cdot\varnothing\cdot\!\rangle$, we let:

$$\widehat{\langle\!\cdot\varnothing\cdot\!\rangle\varphi_{\mathsf{p}}}^{\varnothing} = \mathbf{A}\widehat{\varphi_{\mathsf{p}}}^{\varnothing} \qquad \widehat{\langle\!\cdot\varnothing\cdot\!\rangle\varphi_{\mathsf{p}}}^{C} = \mathbf{A}(\mathbf{G}\mathsf{p_{out}} \Rightarrow \widehat{\varphi_{\mathsf{p}}}^{C}) \quad \text{if } C \neq \varnothing$$

Now we have the following proposition, the proof of which is done by structural induction over the formula:

Proposition 18. *Let $\Phi \in \mathsf{ATL}^*_{sc}$, and $\mathsf{Agt} = \mathsf{Agt}(\Phi) \cup \{a_0\}$ as above. Let $\mathcal{C}$ be a TBGS, ρ be a path of $\mathcal{C}$, n be a position along ρ, and f_B be a strategy context whose domain is $B \subseteq \mathsf{Agt}$. Let $\mathcal{T}_{\mathcal{C}}(\rho(0)) = \langle T, \ell_{f_B}\rangle$ be the computation tree of the Kripke structure underlying $\mathcal{C}$ from $\rho(0)$, labeled with an f_B-tb-labeling ℓ_{f_B}. Then we have:*

$$\mathcal{C}, \rho, n \models_{f_B} \Phi \quad \textit{iff} \quad \langle T, \ell_{f_B}\rangle, \rho, n \models \widehat{\Phi}^{B}$$

Proof. The proof is by structural induction over Φ. The cases of atomic propositions, Boolean operators and temporal modalities are straightforward.

- If $\Phi = \langle\!\cdot A\cdot\!\rangle\varphi_{\mathsf{p}}$: assume $\mathcal{C}, \rho, n \models_{f_B} \Phi$. Then there exists $g_A \in \mathsf{Strat}(A)$ such that for any $\rho' \in \mathsf{Out}(\rho_{\leq n}, g_A \circ f_B)$, we have $\mathcal{C}, \rho', n \models_{(g_A\circ f_B)} \varphi$. Let $\ell_{g_A\circ f_B}$ be some $(g_A \circ f_B)$-tb-labeling built from ℓ_{f_B} by updating the labeling for propositions $(\mathsf{mov}_j)_{a_j\in A}$ and $\mathsf{p_{out}}$. By induction hypothesis, for any $\rho' \in \mathsf{Out}(\rho_{\leq n}, g_A \circ f_B)$, we have $\langle T, \ell_{(g_A\circ f_B)}\rangle, \rho', n \models \widehat{\varphi_{\mathsf{p}}}^{A\cup B}$. Then, by definition of $\ell_{g_A\circ f_B}$, the outcomes generated by $g_A \circ f_B$ are exactly the runs satisfying $\mathbf{G}\mathsf{p_{out}}$, and then we clearly have $\langle T, \ell_{(g_A\circ f_B)}\rangle, \rho, n \models \mathbf{A}(\mathbf{G}\mathsf{p_{out}} \Rightarrow \widehat{\varphi_{\mathsf{p}}}^{A\cup B})$. Moreover $\Phi^{\mathsf{tb}}_{\mathsf{strat}}(A)$ and $\Phi^{\mathsf{tb}}_{\mathsf{out}}(A \cup B)$ also hold true for $\langle T, \ell_{(g_A\circ f_B)}\rangle, \rho, n$. Therefore we have $\langle T, \ell_{f_B}\rangle, \rho, n \models \widehat{\langle\!\cdot A\cdot\!\rangle\varphi_{\mathsf{p}}}^{B}$, since it suffices to extend ℓ_{f_B} in order to encode the strategy g_A and update the truth value of $\mathsf{p_{out}}$ accordingly.

Conversely, assume $\langle T, \ell_{f_B}\rangle, \rho, n \models \widehat{\langle\cdot A\cdot\rangle\varphi_{\mathsf{p}}}^{B}$. Then there exists a labeling ℓ' for $(\mathsf{mov}_j)_{a_j\in A}$ and for $\mathsf{p}_{\mathsf{out}}$ so that (1) $\Phi^{\mathsf{tb}}_{\mathsf{strat}}(A)$ holds true, which ensures that the labeling with $(\mathsf{mov}_j)_{a_j\in A}$ corresponds to a strategy g_A for A from $\rho_{\leq n}$, and (2) $\Phi^{\mathsf{tb}}_{\mathsf{out}}(A\cup B)$ also holds true, which ensures that $\mathsf{p}_{\mathsf{out}}$ marks the outcomes from $\rho_{\leq n}$ induced by $g_A \circ f_B$. This implies that ℓ' is a $(g_A\circ f_B)$-tb-labeling. Finally we also know that $\mathbf{A}(\mathbf{G}\mathsf{p}_{\mathsf{out}} \Rightarrow \widehat{\varphi_{\mathsf{p}}}^{A\cup B})$ holds for $\langle T, \ell_{f_B}\rangle, \rho, n$, which entails that every outcome of $g_A\circ f_B$ satisfies $\widehat{\varphi_{\mathsf{p}}}^{A\cup B}$). The induction hypothesis entails the expected result.

- If $\Phi = (\!|A|\!)\varphi_{\mathsf{s}}$: assume $\mathcal{C}, \rho, n \models_{f_B} \Phi$. Then $\mathcal{C}, \rho, n \models_{f_{B\setminus A}} \varphi_{\mathsf{s}}$. Applying the induction hypothesis, we get $\langle T, \ell_{f_{B\setminus A}}\rangle, \rho, n \models \widehat{\varphi_{\mathsf{s}}}^{B\setminus A}$. It follows that $\langle T, \ell_{f_B}\rangle, \rho, n \models \widehat{\varphi_{\mathsf{s}}}^{B\setminus A}$, because the labeling of strategies for coalition A in f_B is not used for evaluating $\widehat{\varphi_{\mathsf{s}}}^{B\setminus A}$, and the labeling with proposition $\mathsf{p}_{\mathsf{out}}$ will be updated at the next occurrence of a $\langle\cdot\text{-}\cdot\rangle$ quantifier.
  Conversely, assume that $\langle T, \ell_{f_B}\rangle, \rho, n \models \widehat{\varphi_{\mathsf{s}}}^{B\setminus A}$. For the same reason as above, we have $\langle T, \ell_{f_{B\setminus A}}\rangle, \rho, n \models \widehat{\varphi_{\mathsf{s}}}^{B\setminus A}$. Applying the induction hypothesis, we get $\mathcal{C}, \rho, \models_{f_{B\setminus A}} \varphi_{\mathsf{s}}$, and then $\mathcal{C}, \rho, n \models_{f_B} \Phi$. □

Finally, it remains to enforce that the Kripke structure satisfying $\widehat{\Phi}^{\varnothing}$ corresponds to a turn-based game structure. This is achieved by also requiring

$$\Phi_{\mathsf{tb}} = \mathbf{AG}\Big[\bigvee_{a_j\in\mathsf{Agt}}\Big(\mathsf{turn}_j \wedge \bigwedge_{a_l\neq a_j} \neg\,\mathsf{turn}_l\Big)\Big].$$

Finally, we let $\widehat{\Phi}$ be the formula $\Phi_{\mathsf{tb}} \wedge \widehat{\Phi}^{\varnothing}$.

Proposition 19. *Let Φ be an* ATL$^*_{sc}$ *formula and $\widehat{\Phi}$ be the* QCTL* *formula defined as above. Then Φ is satisfiable in a turn-based CGS if, and only if, $\widehat{\Phi}$ is satisfiable (in the tree semantics).*

Proof. If Φ is satisfiable in a TBGS, then there exists such a structure $\mathcal{C}$ with $|\mathsf{Agt}(\Phi)|+1$ agents. Pick such a structure $\mathcal{C}$, and a path ρ such that $\mathcal{C}, \rho, 0 \models \Phi$. Now consider the computation tree $\mathcal{T}_{\mathcal{C}}(\rho(0)) = \langle T, \ell\rangle$. From Proposition 18, we have $\langle T, \ell\rangle, \rho, 0 \models \widehat{\Phi}^{\varnothing}$. Thus clearly $\langle T, \ell\rangle, \rho, 0 \models \widehat{\Phi}$.

Conversely assume $\mathcal{T} \models \widehat{\Phi}$. Thus $\mathcal{T} \models \Phi_{\mathsf{tb}} \wedge \widehat{\Phi}^{\varnothing}$: the first part of the formula ensures that every state of the Kripke structure can be assigned to a unique agent, hence defining a TBGS. The second part ensures that Φ holds in the corresponding game, thanks to Proposition 18. □

The above translation from ATL$^*_{sc}$ into QCTL* transforms a formula with k strategy quantifiers into a formula with at most $k+1$ nested blocks of quantifiers. By slightly modifying the definition of $\widehat{\langle\cdot A\cdot\rangle\varphi_{\mathsf{p}}}^{C}$, we can obtain a translation from ATL$_{sc}$ into QCTL with the same property. Satisfiability of a QCTL* (resp. QCTL) formula with $k+1$ blocks of quantifiers is in $(k+3)$-EXPTIME (resp. $(k+2)$-EXPTIME) [24]. Hence the algorithm is in Tower.

We now prove that this high complexity cannot be avoided:

Proposition 20. *Satisfiability of* ATL$_{sc}$ *and* ATL$^*_{sc}$ *formulas over turn-based CGSs is* Tower*-hard (i.e., it is k-*EXPTIME*-hard, for all k).*

Proof (sketch). Model checking ATL$_{sc}$ over turn-based games is Tower-hard (Theorem 13), and it can easily be encoded as a satisfiability problem. Indeed, let $\mathcal{C} = \langle Q, R, \ell, \mathsf{Agt}, \mathcal{M}, \mathsf{Mov}, \mathsf{Edge}\rangle$ be a TBGS, and Φ be an ATL$_{sc}$ formula. Let $(\mathsf{p}_q)_{q\in Q}$ be fresh atomic propositions. We define an ATL$_{sc}$ formula $\Psi_{\mathcal{C}}$ to describe the game $\mathcal{C}$ as follows:

$$\Psi_{\mathcal{C}} = \mathbf{AG}\Big(\bigvee_{q\in Q}(\mathsf{p}_q \wedge \bigwedge_{q'\neq q}\neg\,\mathsf{p}_{q'} \wedge \bigwedge_{P\in\ell(q)} P \wedge \bigwedge_{P'\notin\ell(q)} \neg\, P')\Big) \wedge$$
$$\mathbf{AG}\Big[\bigwedge_{q\in Q}\Big(\mathsf{p}_q \Rightarrow (\bigwedge_{q\to q'} \langle\!\langle \mathsf{Own}(q)\rangle\!\rangle\,\mathbf{X}\,\mathsf{p}_{q'} \wedge \bigwedge_{q'.\ q\not\to q'} \neg\langle\!\langle \mathsf{Own}(q)\rangle\!\rangle\,\mathbf{X}\,\mathsf{p}_{q'})\Big)\Big]$$

where $q \to q'$ denotes the existence of a transition from q to q' in $\mathcal{C}$. Any TBGS $\mathcal{C}'$ satisfying $\Psi_{\mathcal{C}}$ corresponds to a kind of unfolding of $\mathcal{C}$ with possibly duplications of transitions (and of the corresponding moves). First note that duplicating transitions does not change the truth value of ATL$_{sc}$ formula: in a turn-based structure, duplicating a transition consists in adding a new move for the owner of the source state and this move is completely equivalent to the previous move. Thus we can assume that $\mathcal{C}'$ corresponds to some unfolding of $\mathcal{C}$ (with extra labeling for propositions p_q), thus it yields the same computation tree as $\mathcal{C}$; this ensures that both structures satisfy the same ATL$_{sc}$ formulas when the extra propositions are not used. In particular, $\mathcal{C}'$ satisfies Φ if, and only if, $\mathcal{C}$ does.

Finally we clearly have that $\mathcal{C}, q \models \Phi$ if, and only if, $\Psi_{\mathcal{C}} \wedge \mathsf{p}_q \wedge \Phi$ is satisfiable in a turn-based structure. □

Theorem 21. *Satisfiability for* ATL$_{sc}$ *and* ATL$^*_{sc}$ *over turn-based CGSs is* Tower*-complete.*

We conclude this section with proving that in TBGS, ATL^*_{sc} has the finite-model property:

Proposition 22. *If an ATL^*_{sc} formula Φ is satisfiable in a turn-based CGS, then there exists a finite turn-based CGS satisfying Φ.*

Proof. Assume Φ is satisfiable, then the QCTL* formula $\widehat{\Phi}$ is satisfiable and there exists a tree satisfying $\widehat{\Phi}$. Such a tree $\mathcal{T}$ can be chosen to be regular (QCTL* models can be characterized by alternating parity tree automata [24]); we can consider the underlying finite Kripke structure $K_{\mathcal{T}}$, and apply the same construction as we did for proving Proposition 19, obtaining a finite TBGS satisfying Φ. □

5.3. Games with a bounded action alphabet

We now consider another setting where the reduction to QCTL* can be used to solve the satisfiability for ATL^*_{sc}: we look for CGSs involving a given set of moves, and a given set of players.[3] Formally, the problem is defined as follows:

Problem: $(\mathsf{Agt}, \mathcal{M})$**-satisfiability**
Input: a finite set of moves $\mathcal{M}$, a set of agents Agt, and an ATL^*_{sc} formula Φ involving the agents in Agt;
Question: does there exist a CGS $\mathcal{C} = \langle Q, R, \ell, \mathsf{Agt}, \mathcal{M}, \mathsf{Mov}, \mathsf{Edge}\rangle$ and a state $q \in Q$ such that $\mathcal{C}, q \models \Phi$.

We fix $\mathcal{M} = \{m_1, \ldots, m_k\}$ and $\mathsf{Agt} = \{a_1, \ldots, a_n\}$. Moreover we assume w.l.o.g. that every agent may choose all k moves from every state (*i.e.*, $\mathsf{Mov}(q, a) = \mathcal{M}$ for any $q \in Q$ and $a \in \mathsf{Agt}$). Therefore we know that we are looking for a CGS whose computation tree is a $[k^n]$-tree. In the following, we use a specific alphabet $\mathcal{M}^a$ to represent the moves chosen by Player a, and we use atomic propositions of the form m_i^a to label strategies in the computation tree (of the Kripke structure): a node q is labeled with m_i^a when the strategy for a requires to play m_i from q.

We use $\mathcal{DC}_\Phi$ to denote the set of non-empty coalitions for which outcomes are considered while evaluating Φ. Informally, $\mathcal{DC}_\Phi$ is obtained by traversing the tree representing Φ and computing at each node the "accumulated coalition" since the root of the tree. For example, considering $\Phi = \langle\cdot a_1\cdot\rangle \Big[\langle\cdot a_2\cdot\rangle\, \mathbf{X}\, P_1 \wedge \langle\cdot a_3\cdot\rangle (\!| a_1 |\!)\, \mathbf{X}\, P_2\Big] \vee \langle\cdot a_2, a_4\cdot\rangle\, \mathbf{X}\, P_3$, the set $\mathcal{DC}_\Phi$ is $\{\{a_1\}, \{a_1, a_2\}, \{a_1, a_3\}, \{a_3\}, \{a_2, a_4\}\}$. Note that $|\mathcal{DC}_\Phi|$ is bounded by $|\Phi|$.

We encode the transition table of the CGS in its computation tree: we have to specify the successor state when each player has chosen his move. We also have to deal with *partial* moves: what are the possible successor states when some coalition A plays some move $\overline{m}$ (where $\overline{m}$ is of the form $(m^a_{i_a})_{a\in A}$)? For this, we use atomic propositions to represent the moves that have been chosen by *coalitions*: for every coalition A in $\mathcal{DC}_\Phi \cup \{\mathsf{Agt}\}$ and every move $\overline{m}$ for A (again, with $\overline{m}$ of the form $(m^a_{i_a})_{a\in A}$), we define the atomic proposition $\mathsf{after}(\overline{m})$ to specify that the coalition A has played the move $\overline{m}$ in the *parent* node. Now we have to ensure that a complete move $\overline{m} \in \mathcal{M}^{\mathsf{Agt}}$ fixes a unique successor and that every state (except the root) corresponds to at least one move from its parent:

$$\Phi_{\mathsf{Edge}} = \mathbf{AG}\Big(\bigwedge_{\overline{m}\in\mathcal{M}^{\mathsf{Agt}}} \mathbf{EX}_1 \mathsf{after}(\overline{m})\Big) \wedge \mathbf{AX}\,\mathbf{AG}\Big(\bigvee_{\overline{m}\in\mathcal{M}^{\mathsf{Agt}}} \mathsf{after}(\overline{m})\Big)$$

Moreover the labeling of $\mathsf{after}(\overline{m})$ for a partial move for coalition A has to be consistent *w.r.t.* the labeling of complete moves, thus we have:

$$\begin{aligned}\Phi'_{\mathsf{Edge}} = \mathbf{AG}\Big(&\bigwedge_{\overline{m}\in\mathcal{M}^{\mathsf{Agt}}} \mathsf{after}(\overline{m}) \Rightarrow \bigwedge_{\mathsf{A}\in\mathcal{DC}_\Phi} \mathsf{after}(\overline{m}_{|A})\Big) \wedge \\ \mathbf{AG}\Big(&\bigwedge_{A\in\mathcal{DC}_\Phi}\ \bigwedge_{\overline{m}\in\mathcal{M}^A} \mathsf{after}(\overline{m}) \Rightarrow \bigvee_{\substack{\overline{m'}\in\mathcal{M}^{\mathsf{Agt}}\text{ s.t.}\\ \overline{m'}_{|A}=\overline{m}}} \mathsf{after}(\overline{m'})\Big)\end{aligned}$$

where $\mathsf{after}(\overline{m}_{|A})$ is the atomic proposition corresponding to the restriction of $\overline{m}$ to players in A. Thus we clearly have that for any coalition $A \in \mathcal{DC}_\Phi$ and any move $\overline{m}$ for A, a state q is labeled by $\mathsf{after}(\overline{m})$ if, and only if, $q \in \mathsf{Next}(q', A, \overline{m})$ where q' is the parent of q. The number of such atomic propositions $\mathsf{after}(\overline{m})$ is bounded by $|\Phi| \cdot |\mathcal{M}|^n$, and the size of $\Phi_{\mathsf{Edge}} \wedge \Phi'_{\mathsf{Edge}}$ is in $O(|\Phi| \cdot |\mathcal{M}|^{2n+1})$.

From an ATL^*_{sc} formula Φ, we now define a QCTL* formula $\widehat{\overline{\Phi}}^C$ in a similar way as we defined formula $\overline{\Phi}^C$ in Section 4 (when reducing the ATL^*_{sc} model-checking problem to the QCTL model-checking problem). Here, the sub-formulas Φ_{strat} and Φ_{out} are defined using propositions after(-) instead of the transition table (which is not known at that point). For

[3] One could be tempted to only bound the number of moves, and apply Proposition 15 in order to bound the number of players. However, Proposition 15 does not apply when the set of moves is fixed.

formulas of the form $\langle\cdot A\cdot\rangle\varphi$ with $A=\{a_{j_1},\dots,a_{j_l}\}$ that correspond to non-trivial $\langle\cdot\varnothing\cdot\rangle$ quantifiers, we let:

$$\widehat{\langle\cdot A\cdot\rangle\varphi_{\mathsf{p}}}^{C}=\exists \mathsf{m}_1^{a_{j_1}}\dots\mathsf{m}_k^{a_{j_1}}\dots\mathsf{m}_1^{a_{j_l}}\dots\mathsf{m}_k^{a_{j_l}}.\mathsf{p}_{\mathsf{out}}.$$
$$\Big(\Phi^{\mathsf{ba}}_{\mathsf{strat}}(A)\wedge\Phi^{\mathsf{ba}}_{\mathsf{out}}(A\cup C)\wedge\mathbf{A}\Big(\mathbf{G}\mathsf{p}_{\mathsf{out}}\Rightarrow\widehat{\varphi_{\mathsf{p}}}^{C\cup A}\Big)\Big)$$

with:

$$\Phi^{\mathsf{ba}}_{\mathsf{strat}}(B)=\bigwedge_{a\in B}\mathbf{AG}\Big(\bigvee_{m_i\in\mathcal{M}}(\mathsf{m}_i^a\wedge\bigwedge_{j\neq i}\neg\,\mathsf{m}_j^a)\Big)$$
$$\Phi^{\mathsf{ba}}_{\mathsf{out}}(B)=\mathsf{p}_{\mathsf{out}}\wedge\mathbf{AG}\big(\neg\,\mathsf{p}_{\mathsf{out}}\Rightarrow\mathbf{AX}\neg\,\mathsf{p}_{\mathsf{out}}\big)\wedge$$
$$\mathbf{AG}\Big[\mathsf{p}_{\mathsf{out}}\Rightarrow\bigvee_{\overline{m}\in\mathcal{M}^B}\Big(\mathsf{p}_{\overline{m}}\wedge\mathbf{AX}(\mathsf{after}(\overline{m})\Leftrightarrow\mathsf{p}_{\mathsf{out}})\Big)\Big]$$

where, given $\overline{m}=(m_j^a)_{a\in B}$, $\mathsf{p}_{\overline{m}}$ stands for $\bigwedge_{a\in B}\mathsf{m}_j^a$. As for trivial occurrences of the $\langle\cdot\varnothing\cdot\rangle$ quantifier, we still have:

$$\widehat{\langle\cdot\varnothing\cdot\rangle\varphi_{\mathsf{p}}}^{\varnothing}=\mathbf{A}\widehat{\varphi_{\mathsf{p}}}^{\varnothing}\qquad\widehat{\langle\cdot\varnothing\cdot\rangle\varphi_{\mathsf{p}}}^{C}=\mathbf{A}(\mathbf{G}\mathsf{p}_{\mathsf{out}}\Rightarrow\widehat{\varphi_{\mathsf{p}}}^{C})\quad\text{if } C\neq\varnothing$$

This yields a formula whose size is in $O(|\Phi|\cdot|\mathsf{Agt}|^2\cdot|\mathcal{M}|^{|\mathsf{Agt}|+1})$.

As for the turn-based case, given a strategy f_B, we say that a labeling function ℓ is an f_B-ba-labeling when propositions $(\mathsf{m}^a_{i_a})_{a\in B}$ describe the moves proposed by f_B, and when $\mathsf{p}_{\mathsf{out}}$ labels the outcomes of f_B. We also assume that the labeling of proposition $\mathsf{after}(\text{-})$ in f_B satisfies the formulas Φ_{Edge} and Φ'_{Edge}. Our translation has the expected property (we omit the proof as it follows exactly the same lines as for Proposition 18):

Proposition 23. *Let $\Phi\in\mathsf{ATL}^*_{sc}$. Fix a finite set $\mathcal{M}=\{m_1,\dots,m_k\}$ of actions, and a finite set $\mathsf{Agt}=\{a_1,\dots,a_n\}$ of agents (containing $\mathsf{Agt}(\Phi)$). Let $\mathcal{C}$ be a CGS based on $\mathcal{M}$ and Agt. Let ρ be a path of $\mathcal{C}$, n be a position along ρ, and f_B be a strategy context whose domain is $B\subseteq\mathsf{Agt}$. Let $\mathcal{T}_{\mathcal{C}}(\rho(0))=\langle T,\ell_{f_B}\rangle$ be an computation tree of the Kripke structure underlying $\mathcal{C}$ from $\rho(0)$, labeled with an f_B-ba-labeling ℓ_{f_B}. Then we have:*

$$\mathcal{C},\rho,n\models_{f_B}\Phi\quad\textit{iff}\quad\langle T,\ell_{f_B}\rangle,\rho,n\models\widehat{\widehat{\Phi}}^{B}$$

We can then relate the truth values of Φ and of $\widehat{\widehat{\Phi}}=\Phi_{\mathsf{Edge}}\wedge\Phi'_{\mathsf{Edge}}\wedge\widehat{\widehat{\Phi}}^{\varnothing}$:

Proposition 24. *Let Φ be an ATL^*_{sc} formula, $\mathsf{Agt}=\{a_1,\dots,a_n\}$ be a finite set of agents, $\mathcal{M}=\{m_1,\dots,m_k\}$ be a finite set of moves, and $\widehat{\widehat{\Phi}}$ be the formula defined above. Then Φ is $(\mathsf{Agt},\mathcal{M})$-satisfiable in a CGS if, and only if, the QCTL^* formula $\widehat{\widehat{\Phi}}$ is satisfiable (in the tree semantics).*

Sketch of proof. If Φ is $(\mathsf{Agt},\mathcal{M})$-satisfiable, then we can derive a tree satisfying $\widehat{\widehat{\Phi}}$ (thanks to Proposition 23). Conversely if $\widehat{\widehat{\Phi}}$ is satisfied in some tree $\mathcal{T}$, then this tree corresponds to some CGS based on Agt and $\mathcal{M}$: indeed, formula $\Phi_{\mathsf{Edge}}\wedge\Phi'_{\mathsf{Edge}}$ ensures that for every node, the labeling of the successor nodes with $\mathsf{after}(\overline{m})$ for every $\overline{m}\in\mathcal{M}^n$ defines a transition table. The end of the proof is similar to proof for the turn-based case. □

Notice that if Φ has $k\geq 1$ nested quantifiers, then so does $\widehat{\widehat{\Phi}}$. However, the size of $\widehat{\widehat{\Phi}}$ is exponential in $|\mathsf{Agt}|$. Satisfiability of $\mathsf{Q}^k\mathsf{CTL}^*$ (resp. $\mathsf{Q}^k\mathsf{CTL}$) formula being $(k+2)$-EXPTIME-complete (resp. $(k+1)$-EXPTIME-complete), we end up with an algorithm in $(k+3)$-EXPTIME (resp. $(k+2)$-EXPTIME) for ATL^*_{sc} (resp. ATL_{sc}) formulas involving at most $k\geq 1$ nested quantifiers.

Proposition 25. *The $(\mathsf{Agt},\mathcal{M})$-satisfiability problem for ATL_{sc} and ATL^*_{sc} is* Tower*-hard.*

Proof. The proof uses similar ideas as for the case of TBGSs (see Proposition 20). Since the reduction is from the model-checking problem, we already know the set of agents and actions. The main difficulty is to require that the satisfying CGS be turn-based. This can be achieved using the following formula:

$$\mathbf{AG}\left[\bigwedge_{q\in\mathsf{Q}}\bigvee_{a\in\mathsf{Agt}}\left(p_q\Rightarrow\Big[(\bigwedge_{q\to q'}\langle\langle a\rangle\rangle\,\mathbf{X}\,p_{q'})\wedge([\![a]\!]\bigvee_{q\to q'}\langle\cdot\varnothing\cdot\rangle\,\mathbf{X}\,p_{q'})\Big]\right)\right]$$

The rest of the proof is similar to the proof of Proposition 20. □

Theorem 26. $(\mathsf{Agt},\mathcal{M})$*-satisfiability for ATL_{sc} and ATL^*_{sc} is* Tower*-complete.*

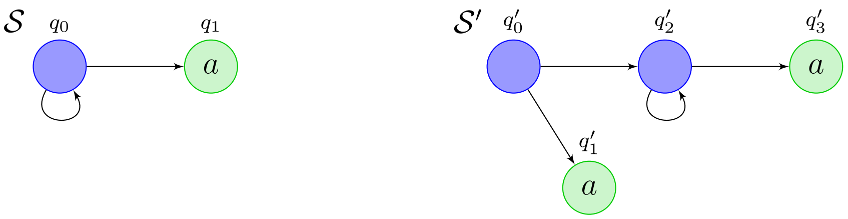

Fig. 2. Two CGS that can be distinguished by $\mathsf{ATL}_{sc,0}$, but not by ATL_{sc}.

6. Extensions of ATL_{sc}

In this section, we explain how our technique of using QCTL^* applies in two other settings: first, for the variant of ATL^*_{sc} where strategy quantifiers are restricted to range over *memoryless* strategies; second, for strategy logic (SL), a different formalism for expressing properties of multi-agent systems.

6.1. ATL_{sc} with memoryless strategy quantifiers

In this section, we consider the logic obtained from ATL_{sc} by restricting strategy quantifiers to range over *memoryless* strategies. Notice that the memoryless requirement only applies to explicitly quantified strategies: for instance, $\langle\cdot A\cdot\rangle_0\varphi$ states that coalition A has a memoryless strategy to enforce φ, whatever the other players do, even if they have memory:

$$\begin{aligned}&\mathcal{C},\pi,n\models_f \langle\cdot A\cdot\rangle_0\varphi \qquad \Leftrightarrow\\ &\qquad \exists f_A\in\mathsf{Strat}_0(A).\ \forall\pi'\in\mathsf{Out}(\pi_{\le n},f_A\circ f).\ \mathcal{C},\pi',n\models_{f_A\circ f}\varphi.\end{aligned}$$

Enforcing memoryless strategies for the opponent coalition can be achieved by making the strategy quantification explicit: formula $\langle\cdot A\cdot\rangle_0[\cdot B\cdot]_0\langle\cdot\varnothing\cdot\rangle_0\varphi$ only considers memoryless strategies of players in B. Notice that because of this difference between implicit and explicit strategy quantification, our translation from ATL^*_{sc} to ATL_{sc} does not apply for memoryless strategies.

It is well known that the strategies witnessing an ATL property can be chosen memoryless: in other terms, $\langle\langle A\rangle\rangle\varphi$ and $\langle\langle A\rangle\rangle_0\varphi$ are equivalent (when $\langle\langle A\rangle\rangle\varphi$ is interpreted with the classical semantics of ATL). Moreover, $\langle\cdot a_1\cdot\rangle_0(\langle\cdot a_2\cdot\rangle_0\mathbf{X}a\wedge\langle\cdot a_2\cdot\rangle_0\mathbf{X}b)$ is true on only one of the two CGSs of Fig. 1. It follows that $\mathsf{ATL}_{sc,0}$ is strictly more expressive than ATL. Actually, $\mathsf{ATL}_{sc,0}$ can even distinguish CGSs that ATL_{sc} cannot: consider the two one-player CGSs $\mathcal{S}$ and $\mathcal{S}'$ of Fig. 2; they involve only one player, and can be seen as Kripke structures; as Kripke structures, they are bisimilar, so that they satisfy the same CTL^* formulas, and consequently also the same ATL_{sc} formulas (any ATL_{sc} formula is easily translated into an equivalent CTL^* formula for these one-player models). But the $\mathsf{ATL}_{sc,0}$ formula $\langle\cdot a\cdot\rangle_0(\mathbf{X}\neg a\wedge\mathbf{X}\mathbf{X}a)$ holds true in state q'_0 of $\mathcal{S}'$, while it fails to hold in q_0 in $\mathcal{S}$.

Model checking The number of memoryless strategies for one player being bounded (with $|\mathcal{M}|^{|Q|}$), we can easily enumerate all of them, and store each strategy within polynomial space. Hence:

Theorem 27. *Model checking* $\mathsf{ATL}_{sc,0}$ *and* $\mathsf{ATL}^*_{sc,0}$ *is* PSPACE*-complete.*

Proof. We again rely on our translation to QCTL^*, but this time in the *structure semantics* [24]: instead of ranging over labellings of the computation tree, propositional quantification then ranges over labellings of the Kripke structure. Indeed, a memoryless strategy is simply a function mapping each state of the game to an available move.

However, we cannot directly reuse the translation of Section 4.1: indeed, in this translation, we quantify over atomic proposition $\mathsf{p}_{\mathsf{out}}$ to mark the outcomes of the selected strategies. This is not correct in the structure semantics of QCTL^*, since this would only involve ultimately-periodic outcomes.

To overcome this problem, we propose a slightly different translation: instead of quantifying over $\mathsf{p}_{\mathsf{out}}$, we use a CTL^* formula to characterize the outcomes: the translation of $\langle\cdot A\cdot\rangle_0\varphi$ in a context with domain C now reads as follows (reusing the notations of Section 4.1):

$$\begin{aligned}\overline{\langle\cdot A\cdot\rangle_0\varphi_{\mathsf p}}^{C} = {}&\exists\mathsf{m}_1^{a_{j_1}}\dots\mathsf{m}_k^{a_{j_1}}\dots\mathsf{m}_1^{a_{j_l}}\dots\mathsf{m}_k^{a_{j_l}}.\\ &\Big(\Phi_{\mathsf{strat}}(A)\wedge\mathbf{A}\Big[(\Phi'_{\mathsf{out}}(A\cup C))\Rightarrow\overline{\varphi_{\mathsf p}}^{C\cup A}\Big)\Big]\end{aligned}$$

where

$$\Phi'_{\mathsf{out}}(B)=\mathbf{G}\left(\bigwedge_{q\in Q}\ \bigwedge_{\overline{m}\in\mathsf{Mov}(q,B)}\left[(\mathsf{p}_q\wedge\mathsf{p}_{\overline{m}})\Rightarrow\bigvee_{q'\in\mathsf{Next}(q,B,\overline{m})}\mathbf{X}\mathsf{p}_{q'}\right]\right).$$

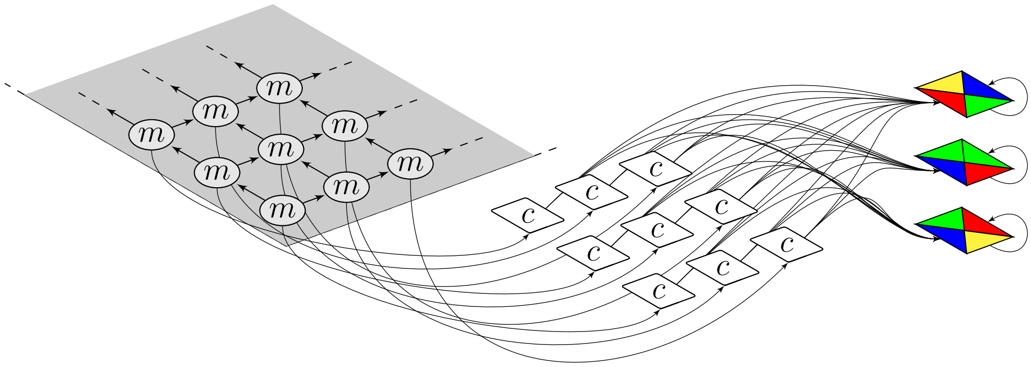

Fig. 3. The turn-based game encoding the tiling problem.

Formula $\Phi'_{\text{out}}(B)$ characterizes the outcomes of the strategies in use for some coalition B. In the end, the QCTL* formula $\overline{\Phi}^{\varnothing}$ has size $O(|\Phi| \cdot |Q| \cdot (|\mathsf{Agt}| \cdot |\mathcal{M}|^2 + |Q| \cdot |\mathsf{Edge}|))$. Using the PSPACE algorithm for model checking QCTL* in the structure semantics, we obtain a PSPACE algorithm for model checking $\mathsf{ATL}^*_{sc,0}$.

Finally, hardness in PSPACE is obtained for $\mathsf{ATL}_{sc,0}$ by a straightforward encoding of QBF. □

Satisfiability While restricting to memoryless strategies makes model checking easier, it actually makes satisfiability undecidable:

Theorem 28. *Satisfiability of* $\mathsf{ATL}_{sc,0}$ *(with memoryless-strategy quantification) is undecidable, even when restricting to turn-based games or when the set of agents and actions is fixed.*

While this result may look surprising given our previous results, it is the natural counterpart of the fact that QCTL satisfiability is undecidable over finite graphs. The proof of Theorem 28 uses the same ideas as for the undecidability of QCTL satisfiability over graphs [24].

Proof. The proof of this result is long and technical; we postpone the full proof to Appendix B, and only give the main ideas here. We encode the following tiling problem: given a finite set of square tiles with a color on each of their four edges, is it possible to tile the quadrant $\mathbb{N} \times \mathbb{N}$, with neighboring tiles sharing the same color on their common edge? This problem was proved undecidable in [18]. Our reduction consists in building a formula that is satisfiable if, and only if, there exists a valid tiling. The formula is the conjunct of two subformulas:

- One subformula characterizes those CGS that have the shape of a grid: each state has two successors (right and up, say), and their successors share a common successor. This is the most technical part of the formula.
- The second subformula states that the grid can be tiled: each state of the grid-shaped CGS has transitions, controlled by a one of the players, to "tile states"; a tiling can then be encoded by a memoryless strategy of that player, and its correctness is expressed by requiring that under that strategy, the colors of the selected tiles match between a state and its successors.

This reduction is depicted on Fig. 3, where we also have intermediary c-states, mainly for having a turn-based game. □

6.2. Strategy logic

Strategy Logic (SL) [11,29] extends LTL with explicit quantification and use of strategies. SL allows first-order quantification over strategies, and those strategies are then assigned to players.

Formula $\langle\langle x\rangle\rangle\varphi$ expresses the existence of a strategy enforcing φ; the strategy is stored in variable x for later use in φ: the *agent binding* operator (a, x) can be used to assign strategy x to agent a. An assignment χ is a partial function from $\mathsf{Agt} \cup \mathsf{Var}$ to Strat. An SL formula φ is interpreted over pairs (χ, q) where q is a state of some CGS and χ is an assignment such that any free[4] strategy variable or agent occurring in φ belongs to $\mathsf{dom}(\chi)$. Note that we must have $\mathsf{Agt} \subseteq \mathsf{dom}(\chi)$ when temporal modalities $\mathbf{X}$ and $\mathbf{U}$ are interpreted: this implies that the set of outcomes is restricted to a single execution generated by all the strategies assigned to players in Agt, and the temporal modalities are therefore interpreted along this execution. We refer to [29] for a complete definition of SL.

[4] We use the standard notion of freeness for strategy variables, with the hypothesis that $\langle\langle x\rangle\rangle$ binds x, and for the agents with the hypothesis that (a, x) binds a and that every agent in Agt is free in temporal subformulas (*i.e.*, with $\mathbf{U}$ or $\mathbf{X}$ as root).

In the following we assume w.l.o.g. that every quantifier $\langle\!\langle x\rangle\!\rangle$ introduces a fresh strategy variable x: this allows us to permanently use variable x to denote the selected strategy for a. Moreover, we require that every player may play any move in any state ($\mathsf{Mov}(q,a)=\mathcal{M}$): this rules out the problem whether a selected strategy can be assigned to a player when evaluating a formula. We omit the formal proofs of the results stated in this part, as they closely follow the same arguments as for ATL^*_{sc}.

Model checking Following the ideas developed for ATL^*_{sc} model checking, we reduce the model-checking problem for SL to the model-checking problem for QCTL^*. Consider a CGS $\mathcal{C}=\langle Q,R,\ell,\mathsf{Agt},\mathcal{M},\mathsf{Mov},\mathsf{Edge}\rangle$ with $\mathcal{M}=\{m_1,\ldots,m_k\}$ and $\mathsf{Agt}=\{a_1,\ldots,a_n\}$, and where Mov constantly returns $\mathcal{M}$. Let $\Phi\in\mathsf{SL}$ and V a partial function $V:\mathsf{Agt}\rightharpoonup\mathsf{Var}$ assigning strategy variables to some of the agents. We build a QCTL^* formula $\overline{\Phi}_V$ as follows:

$$\begin{aligned}
\overline{\varphi\wedge\psi}^V &= \overline{\varphi}^V\wedge\overline{\psi}^V & \overline{\neg\varphi}^V &= \neg\,\overline{\varphi}^V & \overline{\varphi\,\mathbf{U}\,\psi}^V &= \mathbf{A}\Big(\Phi_{\mathsf{out}}(V)\Rightarrow\overline{\varphi_{\mathsf{p}}}^V\,\mathbf{U}\,\overline{\psi}^V\Big)\\
\overline{(a,x)\varphi}^V &= \overline{\varphi}^{V[a\mapsto x]} & \overline{p}^V &= p & \overline{\mathbf{X}\varphi_{\mathsf{p}}}^V &= \mathbf{A}\Big(\Phi_{\mathsf{out}}(V)\Rightarrow\mathbf{X}\overline{\varphi_{\mathsf{p}}}^V\Big)
\end{aligned}$$

Strategy quantification is handled as follows:

$$\overline{\langle\!\langle x\rangle\!\rangle\varphi}^V=\exists\mathsf{m}^x_1\ldots\mathsf{m}^x_k.\Big(\Phi_{\mathsf{strat}}(x)\wedge\overline{\varphi}\Big)$$

with:

$$\begin{aligned}
\Phi_{\mathsf{strat}}(x) &= \mathbf{AG}\Big(\bigvee_{m_i\in\mathcal{M}}(\mathsf{m}^x_i\wedge\bigwedge_{j\neq i}\neg\,\mathsf{m}^x_j)\Big)\\
\Phi_{\mathsf{out}}(V) &= \mathbf{G}\Big[\bigvee_{q\in Q}\Big(\mathsf{p}_q\wedge\bigvee_{\overline{m}\in\mathcal{M}^{\mathsf{Agt}}}\big(\mathsf{p}^V_{\overline{m}}\wedge\mathbf{X}\mathsf{p}_{\mathsf{Edge}(q,\overline{m})}\big)\Big)\Big]
\end{aligned}$$

where $\mathsf{p}^V_{\overline{m}}$ stands for $\bigwedge_{a\in\mathsf{Agt}}\mathsf{m}^{V(a)}_{\alpha_i}$ when $\overline{m}=(m_{\alpha_1},\ldots,m_{\alpha_n})$.

We then have $\mathcal{C},q\models\Phi$ if, and only if, $\langle T,\ell\rangle,q\models\overline{\Phi}^{V\varnothing}$, where $\langle T,\ell\rangle$ is the computation tree of $\mathcal{C}$ with each state q being labeled with a corresponding atomic proposition p_q.

Theorem 29. *Let $\mathcal{C}$ be a CGS $=\langle Q,R,\ell,\mathsf{Agt},\mathcal{M},\mathsf{Edge}\rangle$ and let $\langle T,\ell\rangle$ be the computation tree of $\mathcal{C}$ such that ℓ includes a labeling for propositions p_q. Let Φ be an* SL *formula and $\overline{\Phi}^{V\varnothing}$ be the* QCTL* *formula defined as above. Then $\mathcal{C},q\models\Phi$ if, and only if, $\langle T,\ell\rangle q\models\overline{\Phi}^{V\varnothing}$.*

Satisfiability for turn-based case One easily sees that ATL^*_{sc} can be expressed in SL. It follows that satisfiability is undecidable for SL. We thus restrict to our two decidable cases (turn-based games and bounded set of moves and players), and prove decidability of satisfiability for SL in both cases.

Given an SL formula Φ and a partial function $V:\mathsf{Agt}\rightharpoonup\mathsf{Var}$, we define a QCTL^* formula $\widehat{\Phi}^V$ inductively as follows (Boolean cases omitted):

$$\widehat{\langle\!\langle x\rangle\!\rangle\varphi}^V=\exists\mathsf{mov}_x.\Big[\mathbf{AG}\Big(\mathbf{EX}_1\,\mathsf{mov}_x\Big)\wedge\widehat{\varphi}^V\Big]\qquad\widehat{(a,x)\varphi}^V=\widehat{\varphi}^{V[a\mapsto x]}$$

Note that in this case we require that *every* reachable state has a (unique) successor labeled with mov_x: indeed when one quantifies over a strategy x, the agent(s) who will use this strategy are not known a priori. However, in the turn-based case, a given strategy should be dedicated to a single agent: there is no natural way to share a strategy between two different agents (or the other way around, any two strategies for two different agents can be seen as a single strategy), as they are not playing in the same states. When the strategy x is assigned to some agent a, only the choices made in the a-states are considered.

The temporal modalities are treated as follows:

$$\begin{aligned}
\widehat{\varphi\,\mathbf{U}\,\psi}^V &= \mathbf{A}\Big[\mathbf{G}\Big(\bigwedge_{a_j\in\mathsf{Agt}}(\mathsf{turn}_j\Rightarrow\mathbf{X}\mathsf{mov}_{V(a_j)})\Big)\Rightarrow\widehat{\varphi}^V\,\mathbf{U}\,\widehat{\psi}^V\Big]\\
\widehat{\mathbf{X}\varphi}^V &= \mathbf{A}\Big[\mathbf{G}\Big(\bigwedge_{a_j\in\mathsf{Agt}}(\mathsf{turn}_j\Rightarrow\mathbf{X}\mathsf{mov}_{V(a_j)})\Big)\Rightarrow\mathbf{X}\widehat{\varphi}^V\Big]
\end{aligned}$$

Finally, we let $\widehat{\Phi}$ be the formula $\Phi_{\mathsf{tb}}\wedge\widehat{\Phi}^{V\varnothing}$. Then we have the following theorem:

Theorem 30. *Let Φ be an* SL *formula and $\widetilde{\Phi}$ be the* QCTL* *formula defined as above. Then Φ is satisfiable in a TBGS if, and only if, $\widetilde{\Phi}$ is satisfiable (in the tree semantics).*

Satisfiability for bounded action alphabet Let $\mathcal{M}$ be $\{m_1, \ldots, m_k\}$ and $\mathsf{Agt} = \{a_1, \ldots, a_n\}$. The reduction carried out for ATL_{sc} can also be adapted for SL in this case. First note that it is not necessary to consider partial moves in the construction: in SL, temporal modalities are also interpreted under a *complete* context (*i.e.* where every agent has a strategy). This makes the construction a bit simpler, in particular we will only consider atomic propositions $\mathsf{after}(\overline{m})$ for moves in $\mathcal{M}^n$, thus we will not use the formula Φ'_{Edge} any more. Given an SL formula Φ and a partial function $V\colon \mathsf{Agt} \rightharpoonup \mathsf{Var}$, we define the QCTL* formula $\widehat{\Phi}^V$ inductively as follows:

$$\widehat{\langle\langle x\rangle\rangle\varphi}^V = \exists \mathsf{m}_1^x \ldots \exists \mathsf{m}_k^x . \Phi_{\mathsf{strat}}(x) \wedge \widehat{\varphi}^V \qquad \widehat{(a,x)\varphi}^V = \widehat{\varphi}^{V[a\mapsto x]}$$

The temporal modalities are handled as follows:

$$\widehat{\varphi\,\mathbf{U}\,\psi}^V = \mathbf{A}\Big[\Big(\mathbf{G} \bigwedge_{\overline{m}\in\mathcal{M}^{\mathsf{Agt}}} \big(\mathsf{p}_{\overline{m}}^V \Rightarrow \mathbf{X}\,\mathsf{after}(\overline{m})\big)\Big) \Rightarrow \Big(\widehat{\varphi}^V\,\mathbf{U}\,\widehat{\psi}^V\Big)\Big]$$

$$\widehat{\mathbf{X}\varphi}^C = \mathbf{A}\Big[\Big(\mathbf{G} \bigwedge_{\overline{m}\in\mathcal{M}^{\mathsf{Agt}}} \big(\mathsf{p}_{\overline{m}}^V \Rightarrow \mathbf{X}\,\mathsf{after}(\overline{m})\big)\Big) \Rightarrow \Big(\mathbf{X}\widehat{\varphi}^V\Big)\Big]$$

where $\mathsf{p}_{\overline{m}}^V$ stands for $\bigwedge_{a\in\mathsf{Agt}} \mathsf{m}_{\alpha_i}^{V(a)}$ when $\overline{m} = (m_{\alpha_1}, \ldots, m_{\alpha_n})$.

Finally, let $\widehat{\Phi}$ be the formula $\Phi_{\mathsf{Edge}} \wedge \widehat{\Phi}_{\varnothing}^V$. We have:

Theorem 31. *Let Φ be an SL formula based on the set $\mathsf{Agt} = \{a_1, \ldots, a_n\}$, let $\mathcal{M} = \{m_1, \ldots, m_k\}$ be a finite set of moves, and $\widehat{\Phi}$ be the QCTL* formula defined as above. Then Φ is $(\mathsf{Agt}, \mathcal{M})$-satisfiable if, and only if, $\widehat{\Phi}$ is satisfiable (in the tree semantics).*

7. Conclusion

We developed a tight link between the extension of ATL with strategy contexts and the logic QCTL. We believe that our logic ATL_{sc} (and similar formalisms such as SL) is very well suited for reasoning about complex, multi-agent systems: it can express useful properties in non-zero-sum games, and provide much better granularity than Nash equilibria and similar solution concepts. But the technical formalism of games blurs the setting, and we believe that QCTL is the formalism of choice for fully understanding ATL_{sc} and related logics.

Our translation to QCTL provides us with a uniform presentation of verification algorithms for ATL_{sc}—when such algorithms exist. In view of this, we will keep on developing our knowledge and understanding of QCTL, for instance in terms of the behavioral equivalence it characterizes.

We believe that ATL_{sc} forms a powerful formalism for designing multi-agent systems. Our projects regarding the possible continuations of this work include identifying fragments of ATL_{sc} that preserve a good expressive power while enjoying reasonably efficient model-checking algorithms. The two-alternation fragment of ATL_{sc} (where alternation is understood as the alternation depth of the corresponding QCTL or QCTL* formula) could be a good candidate: all the interesting formulas listed in Section 3 have alternation two. Implementing a model-checker for QCTL (with bounded alternation) might be a good option, given our rather simple translation from ATL_{sc}.

Other possible directions of research include the use of randomized strategies in place of deterministic ones. Since a stochastic version of ATL_{sc} has already been shown undecidable [5], our aim here would be to stick to qualitative formulas expressing sure, almost-sure or limit-sure winning, in the same way as [15] do for *randomized* ATL.

Acknowledgments

The research reported in this article has been partly sponsored by ERC Stg Grant EQualIS (308087) and FP7 FET project Cassting (601148).

Appendix A. Proof of Theorem 9

Theorem 9. *Given a formula $\varphi \in \mathsf{ATL}^*_{sc}$ and a coalition B', there exists an ATL_{sc} formula $\widehat{\varphi}^{[\varnothing,B']}$, involving only players in $\mathsf{Agt}_\varphi \cup B'$, such that for any strategy context f with $\mathsf{dom}(f) = B'$, φ and $\widehat{\varphi}^{[\varnothing,B']}$ are equivalent under context f.*

This result is proved through the following two lemmas: Theorem 9 is a special case of the second result of Lemma 33 for $B = \mathsf{dom}(g) = \varnothing$.

Lemma 32. *Let $\mathcal{C}$ be a CGS with set of agents $\mathsf{Agt}_{\mathcal{C}}$, ρ be a path and n be a position along ρ. For any state-formula $\varphi \in \mathsf{ATL}^*_{sc}$, for any strategy contexts f, g and g' such that $\mathsf{dom}(g) \subseteq \mathsf{dom}(g')$, $g'_{|\mathsf{dom}(g)} = g$ (in other terms, g' extends g), and $\mathsf{dom}(f) \cap \mathsf{dom}(g') = \varnothing$,* *for any coalitions B and B' s.t.* $\mathrm{dom}(f) = (\mathrm{Agt}_{\mathcal{C}} \setminus B) \cap B'$, *and for any outcome* $\pi \in \mathrm{Out}(\rho_{\le n}, g' \circ f)$, *we have:*

$$\mathcal{C}, \pi, n \models_{g \circ f} \widehat{\varphi}^{[B,B']} \;\Leftrightarrow\; \mathcal{C}, \pi, n \models_{g' \circ f} \widehat{\varphi}^{[B,B']}.$$

Proof. We prove this result by structural induction, omitting the easy cases of atomic propositions and Boolean operators.

For the case where $\varphi = \langle\cdot A\cdot\rangle\psi$, assume $\mathcal{C}, \pi, n \models_{g \circ f} \widehat{\langle\cdot A\cdot\rangle\psi}^{[B,B']}$. Then by definition, $\mathcal{C}, \pi, n \models_{g \circ f} \langle\cdot A\cdot\rangle[\cdot B \setminus A\cdot][\cdot \overline{B' \cup A}\cdot] \times \widehat{\psi}^{[B\setminus A, B'\cup A]}$. Thus there exists $f_A \in \mathrm{Strat}(A)$ such that for any $f' \in \mathrm{Strat}((B \setminus A) \cup (\overline{B' \cup A}))$, it holds $\mathcal{C}, \pi', n \models_{f' \circ f_A \circ g \circ f} \widehat{\psi}^{[B\setminus A, B'\cup A]}$, where π' is the unique path in $\mathrm{Out}(\rho_{\le n}, f' \circ f_A \circ g \circ f)$. Equivalently, with the above f_A and for any $f' \in \mathrm{Strat}((B \setminus A) \cup (\overline{B' \cup A}))$, $\mathcal{C}, \pi', n \models_{f' \circ f_A \circ g' \circ f} \widehat{\psi}^{[B\setminus A, B'\cup A]}$ because $\mathrm{dom}(g) \subseteq \mathrm{dom}(g') \subseteq \mathrm{dom}(f) \subseteq \mathrm{dom}(f') \cup A$. In the end, we get $\mathcal{C}, \pi, n \models_{g' \circ f} \widehat{\varphi}^{[B,B']}$, as expected. The converse implication is proven similarly, as well as the case where $\varphi = \langle\cdot \overline{A}\cdot\rangle\psi$.

Now assume $\varphi = (\!|A|\!)\psi$. If $\mathcal{C}, \pi, n \models_{g \circ f} \widehat{\varphi}^{[B,B']}$, then $\mathcal{C}, \pi, n \models_{g \circ f} \widehat{\psi}^{[B\cup A, B']}$, so that $\mathcal{C}, \pi, n \models_{g' \circ f} \widehat{\psi}^{[B\cup A, B']}$ by induction hypothesis (as ψ is a state-formula). Thus $\mathcal{C}, \pi, n \models_{g' \circ f} \widehat{\varphi}^{[B,B']}$. The converse direction follows the same lines. The proof for $\varphi = (\!|\overline{A}|\!)\psi$ is similar. □

Lemma 33. *Let $\mathcal{C}$ be a CGS with set of agents* $\mathrm{Agt}_{\mathcal{C}}$, *$\rho$ be a path and n be a position along ρ, and f be a strategy context with* $\mathrm{dom}(f) \subseteq \mathrm{Agt}_{\Phi}$. *Then for any* ATL^*_{sc} *formula φ, for any strategy context g s.t.* $\mathrm{dom}(g) = \mathrm{Agt}_{C} \smallsetminus \mathrm{dom}(f)$, *for any outcome* $\pi \in \mathrm{Out}(\rho_{\le n}, g \circ f)$, *and for any coalitions B and B' s.t.* $\mathrm{dom}(f) = (\mathrm{Agt}_{\mathcal{C}} \setminus B) \cap B'$, *it holds: $\mathcal{C}, \pi, n \models_f \varphi \Leftrightarrow \mathcal{C}, \pi, n \models_{g \circ f} \widehat{\varphi}^{[B,B']}$. Moreover, if φ is a state-formula, this result extends to any strategy context g s.t.* $\mathrm{dom}(g) \cap \mathrm{dom}(f) = \varnothing$.

Proof. We prove the result by induction on the structure of φ. The cases of atomic propositions and Boolean connectives are straightforward.

- If $\varphi = \mathbf{X}\psi$, then $\mathcal{C}, \pi, n \models_f \varphi$ is equivalent to $\mathcal{C}, \pi, n+1 \models_f \psi$. Applying the induction hypothesis, this is equivalent to $\mathcal{C}, \pi, n+1 \models_{g \circ f} \widehat{\psi}^{[B,B']}$. This means that $\mathcal{C}, \pi, n \models_{g \circ f} \mathbf{X}\widehat{\psi}^{[B,B']}$, which in turn is equivalent to $\mathcal{C}, \pi, n \models_{g \circ f} \langle\cdot\varnothing\cdot\rangle\, \mathbf{X}\, \widehat{\psi}^{[B,B']}$ because $\mathrm{Out}(\pi_{\le n}, g \circ f) = \{\pi\}$.
- If $\varphi = \psi_1 \,\mathbf{U}\, \psi_2$: this case can be handled in a similar way as for the previous case, and we omit it.
- If $\varphi = \langle\cdot A\cdot\rangle\psi$: as this is a state formula, we prove the second, more general statement. Let g be a strategy context with $\mathrm{dom}(g) \cap \mathrm{dom}(f) = \varnothing$, and π be an outcome of $g \circ f$ from $\rho_{\le n}$. Finally, fix B and B' such that $\mathrm{dom}(f) = (\mathrm{Agt}_{\mathcal{C}} \setminus B) \cap B'$.

$$\begin{aligned}
&\mathcal{C}, \pi, n \models_f \langle\cdot A\cdot\rangle\psi \\
&\Leftrightarrow \quad \exists f_A \in \mathrm{Strat}(A).\ \forall \pi' \in \mathrm{Out}(\pi_{\le n}, f_A \circ f).\ \mathcal{C}, \pi', n \models_{f_A \circ f} \psi \\
&\Leftrightarrow \quad \exists f_A \in \mathrm{Strat}(A).\ \forall f' \in \mathrm{Strat}(\mathrm{Agt}_{\mathcal{C}} \smallsetminus \mathrm{dom}(f_A \circ f)). \\
&\qquad\qquad \forall \pi' \in \mathrm{Out}(\pi_{\le n}, f' \circ f_A \circ f).\ \mathcal{C}, \pi', n \models_{f_A \circ f} \psi \\
&\Leftrightarrow \quad \exists f_A \in \mathrm{Strat}(A).\ \forall f' \in \mathrm{Strat}(\mathrm{Agt}_{\mathcal{C}} \smallsetminus \mathrm{dom}(f_A \circ f)). \\
&\qquad\qquad \forall \pi' \in \mathrm{Out}(\pi_{\le n}, f' \circ f_A \circ f).\ \mathcal{C}, \pi', n \models_{f' \circ f_A \circ f} \widehat{\psi}^{[B\setminus A, B'\cup A]} \\
&\qquad\qquad \text{(by i.h., because } \mathrm{dom}(f_A \circ f) = (\mathrm{Agt}_{\mathcal{C}} \setminus (B \setminus A)) \cap (B' \cup A)) \\
&\Leftrightarrow \quad \exists f_A \in \mathrm{Strat}(A).\ \forall f' \in \mathrm{Strat}(\mathrm{Agt}_{\mathcal{C}} \smallsetminus \mathrm{dom}(f_A \circ f)). \\
&\qquad\qquad \forall \pi' \in \mathrm{Out}(\pi_{\le n}, f' \circ f_A \circ f).\ \mathcal{C}, \pi', n \models_{f' \circ f_A \circ g \circ f} \widehat{\psi}^{[B\setminus A, B'\cup A]} \\
&\qquad\qquad \text{(because } f' \circ f_A \circ g \circ f = f' \circ f_A \circ f) \\
&\Leftrightarrow \quad \exists f_A \in \mathrm{Strat}(A).\ \forall f' \in \mathrm{Strat}(\mathrm{Agt}_{\mathcal{C}} \smallsetminus \mathrm{dom}(f_A \circ f)). \\
&\qquad\qquad \exists \pi' \in \mathrm{Out}(\pi_{\le n}, f' \circ f_A \circ f).\ \mathcal{C}, \pi', n \models_{f' \circ f_A \circ g \circ f} \widehat{\psi}^{[B\setminus A, B'\cup A]} \\
&\qquad\qquad \text{(because } |\mathrm{Out}(\pi_{\le n}, f' \circ f_A \circ g \circ f)| = 1) \\
&\Leftrightarrow \quad \exists f_A \in \mathrm{Strat}(A).\ \mathcal{C}, \pi, n \models_{f_A \circ g \circ f} [\cdot B \setminus A\cdot][\cdot \overline{B' \cup A}\cdot]\widehat{\psi}^{[B\setminus A, B'\cup A]} \\
&\qquad\qquad \text{(because } \mathrm{dom}(f') = (B \setminus A) \cup (\mathrm{Agt}_{\mathcal{C}} \setminus (B' \cup A))) \\
&\Leftrightarrow \quad \mathcal{C}, \pi, n \models_{g \circ f} \langle\cdot A\cdot\rangle[\cdot B \setminus A\cdot][\cdot \overline{B' \cup A}\cdot]\widehat{\psi}^{[B\setminus A, B'\cup A]} \\
&\Leftrightarrow \quad \mathcal{C}, \pi, n \models_{g \circ f} \widehat{\varphi}^{[B,B']}
\end{aligned}$$

- If $\varphi = \langle\cdot \overline{A}\cdot\rangle\psi$: a similar sequence of equivalences applies, but now $\mathrm{dom}(f_{\overline{A}} \circ f) = \mathrm{Agt}_{\mathcal{C}} \setminus [(B \cap A) \cup (A \setminus B')]$.

- If $\varphi = (\!|A|\!)\psi$: again, we prove the second statement. Let g be a strategy context with $\mathsf{dom}(g) \cap \mathsf{dom}(f) = \varnothing$, and π be an outcome of $g \circ f$ after $\rho_{\leq n}$. Let B and B' such that $\mathsf{dom}(f) = (\mathsf{Agt}_{\mathcal{C}} \setminus B) \cap B'$. We have:

$$\begin{aligned}
\mathcal{C}, \pi, n \models_f (\!|A|\!)\psi \quad &\Leftrightarrow \quad \mathcal{C}, \pi, n \models_{f_{\backslash A}} \psi \\
&\Leftrightarrow \quad \mathcal{C}, \pi, n \models_{g \circ f_{\backslash A}} \widehat{\psi}^{[B \cup A, B']} \\
&\qquad\text{(by i.h., because } \mathsf{dom}(f_{\backslash A}) = (\mathsf{Agt}_{\mathcal{C}} \setminus (B \cup A)) \cap B'\text{)} \\
&\Leftrightarrow \quad \mathcal{C}, \pi, n \models_{g \circ f_{|A} \circ f_{\backslash A}} \widehat{\psi}^{[B \cup A, B']} \quad \text{(by Lemma 32)} \\
&\Leftrightarrow \quad \mathcal{C}, \pi, n \models_{g \circ f} \widehat{\psi}^{[B \cup A, B']}
\end{aligned}$$

- If $\varphi = (\!|\overline{A}|\!)\psi$, a similar sequence of equivalences applies. □

Appendix B. Proof of Theorem 28

Theorem 28. *Satisfiability of* $\mathsf{ATL}_{sc,0}$ *(with memoryless-strategy quantification) is undecidable, even when restricting to turn-based games or when the set of agents and actions is fixed.*

Proof. We prove the result for infinite-state turn-based games, by adapting the corresponding proof for QCTL under the structure semantics [17], which consists in encoding the problem of tiling a quadrant. The result for finite-state turn-based games can be obtained using similar (but more involved) ideas, by encoding the problem of tiling all finite grids (see [24] for the corresponding proof for QCTL).

We consider a finite set T of tiles, and two binary relations H and V indicating which tile(s) may appear on the right and above (respectively) a given tile. Our proof consists in writing a formula that is satisfiable only on a grid-shaped (turn-based) game structure representing a tiling of the quadrant (*i.e.*, of $\mathbb{N} \times \mathbb{N}$). The reduction involves two players: Player 1 controls square states (which are labeled with □), while Player 2 controls circle states (labeled with ◯). Each state of the grid is intended to represent one cell of the quadrant to be tiled. For technical reasons, the reduction is not that simple, and our game structure will have three kinds of states (see Fig. 3):

- the "main" states (controlled by Player 2), which form the grid. Each state in this main part has a *right* neighbor and a *top* neighbor, which we assume we can identify: more precisely, we make use of two atomic propositions v_1 and v_2 which alternate along the horizontal lines of the grid. The *right* successor of a v_1-state is labeled with v_2, while its *top* successor is labeled with v_1;
- the "tile" states, labeled with one item of T (seen as atomic propositions). Each tile state only has outgoing transition(s) to a tile state labeled with the same tile;
- the "choice" states, which appear between "main" states and "tile" states: there is one choice state associated with each main state, and each choice state has a transition to each tile state. Choice states are controlled by Player 1.

Assuming that we have such a structure, a tiling of the grid corresponds to a *memoryless* strategy of Player 1 (who only plays in the "choice" states). Once such a memoryless strategy for Player 1 has been selected, that it corresponds to a valid tiling can be expressed easily: for instance, in any cell of the grid (assumed to be labeled with v_1), there must exist a pair of tiles $(t_1, t_2) \in H$ such that $v_1 \wedge \langle\cdot 2\cdot\rangle_0 \mathbf{X}\mathbf{X} t_1 \wedge \langle\cdot 2\cdot\rangle_0 \mathbf{X}(v_2 \wedge \mathbf{X}\mathbf{X} t_2)$. This would be written as follows:

$$\langle\cdot 1\cdot\rangle_0 \mathbf{G} \left[\begin{array}{l} v_1 \Rightarrow \displaystyle\bigvee_{(t_1,t_2) \in H} \langle\cdot 2\cdot\rangle_0 \mathbf{X}\mathbf{X} t_1 \wedge \langle\cdot 2\cdot\rangle_0 \mathbf{X}(v_2 \wedge \mathbf{X}\mathbf{X} t_2) \\ \wedge \\ v_2 \Rightarrow \displaystyle\bigvee_{(t_1,t_2) \in H} \langle\cdot 2\cdot\rangle_0 \mathbf{X}\mathbf{X} t_1 \wedge \langle\cdot 2\cdot\rangle_0 \mathbf{X}(v_1 \wedge \mathbf{X}\mathbf{X} t_2) \end{array} \right].$$

The same can be imposed for vertical constraints, and for imposing a fairness constraint on the base line (under the same memoryless strategy for Player 1).

It remains to build a formula characterizing an infinite grid. This requires a slight departure from the above description of the grid: each main state will in fact be a gadget composed of four states, as depicted on Fig. B.4. The first state of each gadget will give the opportunity to Player 1 to *color* the state with either α or β. This will be used to enforce "confluence" of several transitions to the same state (which we need to express that the two successors of any cell of the grid share a common successor).

We now start writing our formula, which we present as a conjunction of several subformulas. We require that the main states be labeled with m, the choice states be labeled with c, and the tile states be labeled with the names of the tiles. We let $\mathsf{AP}' = \{m, c\} \cup T$ and $\mathsf{AP} = \mathsf{AP}' \cup \{v_1, v_2, \alpha, \beta, \square, \bigcirc\}$. The first part of the formula reads as follows (where universal

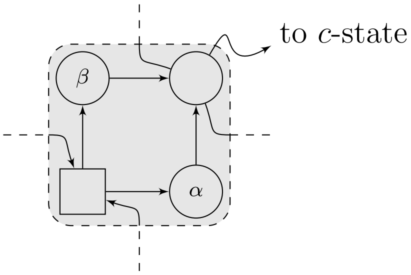

Fig. B.4. The cell gadget.

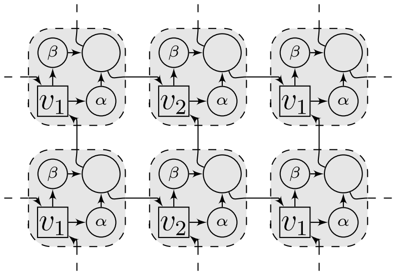

Fig. B.5. Several cells forming (part of) a grid.

path quantification can be encoded, as long as the context is empty, using $\langle\cdot\varnothing\cdot\rangle_0$):

$$\mathbf{A}(m\,\mathbf{W}\,c) \wedge \mathbf{AG}\left[\bigvee_{p\in\mathsf{AP}'} p \wedge \bigwedge_{p'\in\mathsf{AP}'\setminus\{p\}} \neg\, p'\right] \wedge \mathbf{AG}(\Box \Leftrightarrow \neg \bigcirc) \wedge$$

$$\mathbf{AG}\left[\begin{array}{l} c \Rightarrow \left(\Box \wedge \bigwedge_{t\in T}\langle\cdot 1\cdot\rangle_0\,\mathbf{X}\,t \wedge \mathbf{A}\,\mathbf{X}\left(\bigvee_{t\in T}\mathbf{AG}t\right)\right)\\ \wedge \\ \Box\Rightarrow\left(\bigwedge_{p\in\mathsf{AP}}(\mathbf{E}\,\mathbf{X}\,p\Leftrightarrow\langle\cdot 1\cdot\rangle_0\,\mathbf{X}\,p)\right)\\ \wedge\\ \bigcirc\Rightarrow\left(\bigwedge_{p\in\mathsf{AP}}(\mathbf{E}\,\mathbf{X}\,p\Leftrightarrow\langle\cdot 2\cdot\rangle_0\,\mathbf{X}\,p)\right)\end{array}\right] \tag{B.1}$$

This formula enforces that each state is labeled with exactly one proposition from AP′. It also enforces that any path will wander through the main part until it possibly goes to a choice state (this is expressed as $\mathbf{A}(m\,\mathbf{W}\,c)$, where $m\,\mathbf{W}\,c$ means $\mathbf{G}m \vee m\,\mathbf{U}\,c$, and can be expressed as a negated-until formula). Finally, the second part of the formula enforces the witnessing structures to be turn-based.

Now we have to impose that the m-part has the shape of a grid (Fig. B.5): intuitively, each cell has three successors: one "to the right" and one "to the top" in the main part of the grid, and one c-state which we will use for associating a tile with this cell. For technical reasons, the situation is not that simple, and each cell is actually represented by the gadget depicted on Fig. B.4. Each state of the gadget is labeled with m. We constrain the form of the cells as follows:

$$\mathbf{AG}\left[\begin{array}{l} m\Rightarrow((\Box\wedge\neg\alpha\wedge\neg\beta)\vee(\bigcirc\wedge\neg(\alpha\wedge\beta)))\\ \wedge\\ \big((m\wedge\Box)\Rightarrow(v_1\Leftrightarrow\neg v_2)\big)\wedge\big((v_1\vee v_2)\Rightarrow(m\wedge\Box)\big)\\ \wedge\\ (m\wedge\Box)\Rightarrow\big[\mathbf{A}\,\mathbf{X}\,\big(m\wedge\bigcirc\wedge(\alpha\vee\beta)\wedge\mathbf{A}\,\mathbf{X}\,(m\wedge\bigcirc\wedge\neg\alpha\wedge\neg\beta)\big)\wedge\\ \qquad\qquad\langle\cdot 1\cdot\rangle_0\,\mathbf{X}\,\alpha\wedge\langle\cdot 1\cdot\rangle_0\,\mathbf{X}\,\beta\big]\end{array}\right] \tag{B.2}$$

This says that there are four types of states in each cell, and specifies the possible transitions within such cells. We now express constraints on the transitions leaving a cell:

$$\mathbf{AG}\Big[(\mathbf{E}\,\mathbf{X}\,c\vee\mathbf{E}\,\mathbf{X}\,v_1\vee\mathbf{E}\,\mathbf{X}\,v_2)\Rightarrow(m\wedge\bigcirc\wedge\neg\alpha\wedge\neg\beta)\Big]\wedge$$

$$\mathbf{AG}\Big[(m\wedge\bigcirc\wedge\neg\alpha\wedge\neg\beta)\Rightarrow(\mathbf{E}\,\mathbf{X}\,c\wedge\mathbf{E}\,\mathbf{X}\,v_1\wedge\mathbf{E}\,\mathbf{X}\,v_2\wedge\mathbf{A}\,\mathbf{X}\,(c\vee v_1\vee v_2)\Big] \tag{B.3}$$

It remains to enforce that the successor of the α and β states are the same. This is obtained by the following formula:

$$\mathbf{AG}\Big[(m \wedge \Box) \Rightarrow [\cdot 2 \cdot]_0 \big(\langle\cdot\varnothing\cdot\rangle_0\, \mathbf{X}^3\ (c \vee v_1) \vee \langle\cdot\varnothing\cdot\rangle_0\, \mathbf{X}^3\ (c \vee v_2)\big)\Big] \tag{B.4}$$

Indeed, assume that some cell has two different "final" states; then there would exist a strategy for Player 2 (consisting in playing differently in those two final states) that would violate Formula (B.4). Hence each cell as a single final state.

We now impose that each cell in the main part has exactly two m-successors, and these two m-successors have an m-successor in common. For the former property, Formula (B.3) already imposes that each cell has at least two m-successors (one labeled with v_1 and one with v_2). We enforce that there cannot be more than two:

$$\mathbf{AG}\Big[(m \wedge \Box) \Rightarrow [\cdot 1 \cdot]_0 \Big[\big(\langle\cdot 2\cdot\rangle_0\, \mathbf{X}^3\ (v_1 \wedge \mathbf{X}\alpha) \wedge \langle\cdot 2\cdot\rangle_0\, \mathbf{X}^3\ (v_2 \wedge \mathbf{X}\alpha)\big) \Rightarrow$$
$$[\cdot 2 \cdot]_0 \langle\cdot\varnothing\cdot\rangle_0\, \mathbf{X}^3\, \mathbf{X}\alpha\Big]\Big]. \tag{B.5}$$

Notice that $[\cdot 2 \cdot]_0 \langle\cdot\varnothing\cdot\rangle_0 \varphi$ means that φ has to hold along any outcome of any *memoryless* strategy of Player 2. Assume that a cell has three (or more) successor cells. Then at least one is labeled with v_1 and at least one is labeled with v_2. There is a strategy for Player 1 to color one v_1-successor cell and one v_2-successor cell with α, and a third successor cell with β, thus violating Formula (B.5) (as Player 2 has a strategy to reach a successor cell colored with β).

For the latter property (the two successors have a common successor), we add the following formula (as well as its v_2-counterpart):

$$[\cdot 1 \cdot]_0 \langle\cdot\varnothing\cdot\rangle_0 \mathbf{G}\Big[(m \wedge \Box \wedge v_1) \Rightarrow \Big(\big[\langle\cdot 2\cdot\rangle_0\, \mathbf{X}^3\ (v_1 \wedge [\cdot 2 \cdot]_0\, \mathbf{X}^3\, \mathbf{X}\alpha)\big] \Rightarrow$$
$$\big[\langle\cdot 2\cdot\rangle_0\, \mathbf{X}^3\ (\neg\, v_1 \wedge \mathbf{X}^3(\neg\, v_1 \wedge \mathbf{X}\alpha))\big]\Big)\Big] \tag{B.6}$$

In this formula, the initial (universal) quantification over strategies of Player 1 fixes a color for each cell. The formula claims that whatever this choice, if we are in some v_1-cell and can move to another v_1-cell whose two successors have color α, then also we can move to a v_2-cell having one α successor (which we require to be a v_2-cell). As this must hold for any coloring, both successors of the original v_1-cell share a common successor. Notice that this does not prevent the grid to be collapsed: this would just indicate that there is a *regular* infinite tiling.

We conclude by requiring that the initial state be in a square state of a cell in the main part. □

References

[1] T. Ågotnes, V. Goranko, W. Jamroga, Alternating-time temporal logics with irrevocable strategies, in: Proceedings of the 11th Conference on Theoretical Aspects of Rationality and Knowledge, TARK'07, 2007, pp. 15–24.

[2] T. Ågotnes, V. Goranko, W. Jamroga, Strategic commitment and release in logics for multi-agent systems (extended abstract), Technical Report 08-01, Clausthal University of Technology, Germany, 2008.

[3] R. Alur, T.A. Henzinger, O. Kupferman, Alternating-time temporal logic, J. ACM 49 (5) (2002) 672–713.

[4] R. Alur, T.A. Henzinger, O. Kupferman, M.Y. Vardi, Alternating refinement relations, in: Proceedings of the 9th International Conference on Concurrency Theory, CONCUR'98, in: LNCS, vol. 1466, Springer, 1998, pp. 163–178.

[5] Ch. Baier, T. Brázdil, M. Größer, A. Kučera, Stochastic game logic, in: Proceedings of the 4th International Conference on Quantitative Evaluation of Systems, QEST'07, IEEE Comp. Soc. Press, 2007, pp. 227–236.

[6] Th. Brihaye, A. Da Costa, F. Laroussinie, N. Markey, ATL with strategy contexts and bounded memory, in: Proceedings of the International Symposium Logical Foundations of Computer Science, LFCS'09, in: LNCS, vol. 5407, Springer, 2009, pp. 92–106.

[7] P. Čermák, A model checker for strategy logic, Master's thesis, Dept. of Computing, Imperial College, London, UK, 2014.

[8] P. Čermák, A. Lomuscio, F. Mogavero, A. Murano, MCMAS-SLK: a model checker for the verification of strategy logic specifications, in: Proceedings of the 26th International Conference on Computer Aided Verification, CAV'14, in: LNCS, vol. 8559, Springer, 2014, pp. 525–532.

[9] C. Chareton, J. Brunel, D. Chemouil, Towards an updatable strategy logic, in: Proceedings of the 1st International Workshop on Strategic Reasoning, SR'13, in: Electronic Proceedings in Theoretical Computer Science, vol. 112, 2013, pp. 91–98.

[10] K. Chatterjee, T.A. Henzinger, M. Jurdziński, Games with secure equilibria, Theor. Comput. Sci. 365 (1–2) (2006) 67–82.

[11] K. Chatterjee, T.A. Henzinger, N. Piterman, Strategy logic, in: Proceedings of the 18th International Conference on Concurrency Theory, CONCUR'07, in: LNCS, vol. 4703, Springer, 2007, pp. 59–73.

[12] E.M. Clarke, E.A. Emerson, Design and synthesis of synchronization skeletons using branching-time temporal logic, in: Proceedings of the 3rd Workshop on Logics of Programs, LOP'81, in: LNCS, vol. 131, Springer, 1982, pp. 52–71.

[13] A. Da Costa, F. Laroussinie, N. Markey, ATL with strategy contexts: expressiveness and model checking, in: Proceedings of the 30th Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS'10, in: LIPIcs, vol. 8, Leibniz-Zentrum für Informatik, 2010, pp. 120–132.

[14] A. Da Costa, F. Laroussinie, N. Markey, Quantified CTL: expressiveness and model checking, in: Proceedings of the 23rd International Conference on Concurrency Theory, CONCUR'12, in: LNCS, vol. 7454, Springer, 2012, pp. 177–192.

[15] L. de Alfaro, T.A. Henzinger, O. Kupferman, Concurrent reachability games, in: Proceedings of the 39th Annual Symposium on Foundations of Computer Science, FOCS'98, IEEE Comp. Soc. Press, 1998, pp. 564–575.

[16] E.A. Emerson, J.Y. Halpern, "Sometimes" and "not never" revisited: on branching versus linear time, in: Conference Record of the 10th ACM Symposium on Principles of Programming Languages, POPL'83, ACM Press, 1983, pp. 127–140.

[17] T. French, Decidability of quantified propositional branching time logics, in: Proceedings of the 14th Australian Joint Conference on Artificial Intelligence, AJCAI'01, in: LNCS, vol. 2256, Springer, 2001, pp. 165–176.

[18] D. Harel, Recurring dominoes: making the highly undecidable highly understandable, Ann. Discrete Math. 24 (1985) 51–72.

[19] A. Herzig, E. Lorini, D. Walther, Reasoning about actions meets strategic logics, in: Proceedings of the 4th Workshop on Logic, Rationality, and Interaction, LORI'13, in: LNCS, vol. 8196, Springer, 2013, pp. 162–175.

[20] C.-H. Huang, S. Schewe, F. Wang, Model checking iterated games, in: Proceedings of the 19th International Conference on Tools and Algorithms for Construction and Analysis of Systems, TACAS'13, in: LNCS, vol. 7795, Springer, 2013, pp. 154–168.

[21] O. Kupferman, Augmenting branching temporal logics with existential quantification over atomic propositions, in: Proceedings of the 7th International Conference on Computer Aided Verification, CAV'95, in: LNCS, vol. 939, Springer, 1995, pp. 325–338.

[22] A. Kurucz, $S5 \times S5 \times S5$ lacks the finite model property, in: Proceedings of the 3rd Workshop on Advances in Modal Logic, AIML'00, World Scientific, 2002, pp. 321–327.

[23] A. Kurucz, F. Wolter, M. Zakharyaschev, D.M. Gabbay, Many-Dimensional Modal Logics: Theory and Applications, Elsevier Science, 2003.

[24] F. Laroussinie, N. Markey, Quantified CTL: expressiveness and complexity, Research Report LSV-13-07, Lab. Spécification & Vérification, ENS Cachan, France, 2013.

[25] F. Laroussinie, N. Markey, Satisfiability of ATL with strategy contexts, in: Proceedings of the 4th International Symposium on Games, Automata, Logics and Formal Verification, GandALF'13, in: Electronic Proceedings in Theoretical Computer Science, vol. 119, 2013, pp. 208–223.

[26] R. Maddux, The equational theory of ca_3 is undecidable, J. Symb. Log. 45 (2) (1980) 311–316.

[27] F. Mogavero, A. Murano, G. Perelli, M.Y. Vardi, Reasoning about strategies: on the model-checking problem, Research report arXiv:1112.6275 [cs.LO], 2012.

[28] F. Mogavero, A. Murano, G. Perelli, M.Y. Vardi, What makes ATL* decidable? a decidable fragment of strategy logic, in: Proceedings of the 23rd International Conference on Concurrency Theory, CONCUR'12, in: LNCS, vol. 7454, Springer, 2012, pp. 193–208.

[29] F. Mogavero, A. Murano, M.Y. Vardi, Reasoning about strategies, in: Proceedings of the 30th Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS'10, in: LIPIcs, vol. 8, Leibniz-Zentrum für Informatik, 2010, pp. 133–144.

[30] F. Moller, A. Rabinovich, On the expressive power of CTL*, in: Proceedings of the 14th Annual Symposium on Logic in Computer Science, LICS'99, IEEE Comp. Soc. Press, 1999, pp. 360–368.

[31] J.F. Nash Jr., Equilibrium points in n-person games, Proc. Natl. Acad. Sci. USA 36 (1) (1950) 48–49.

[32] J.-P. Queille, J. Sifakis, Specification and verification of concurrent systems in CESAR, in: Proceedings of the 5th International Symposium on Programming, SOP'82, in: LNCS, vol. 137, Springer, 1982, pp. 337–351.

[33] S. Schmitz, Complexity hierarchies beyond elementary, Research report arXiv:1312.5686 [cs.CC], 2013.

[34] A.P. Sistla, Theoretical issues in the design and verification of distributed systems, PhD thesis, Harvard University, Cambridge, Massachussets, USA, 1983.

[35] N. Troquard, D. Walther, On satisfiability in ATL with strategy contexts, in: Proceedings of the 13th European Conference in Logics in Artificial Intelligence, JELIA'12, in: LNCS, vol. 7519, Springer, 2012, pp. 398–410.

[36] W. van der Hoek, W. Jamroga, M. Wooldridge, A logic for strategic reasoning, in: Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS'05, ACM Press, 2005, pp. 157–164.

[37] D. Walther, C. Lutz, F. Wolter, M. Wooldridge, ATL satisfiability is indeed EXPTIME-complete, J. Log. Comput. 16 (6) (2006) 765–787.

[38] D. Walther, W. van der Hoek, M. Wooldridge, Alternating-time temporal logic with explicit strategies, in: Proceedings of the 11th Conference on Theoretical Aspects of Rationality and Knowledge, TARK'07, 2007, pp. 269–278.

[39] F. Wang, C.-H. Huang, F. Yu, A temporal logic for the interaction of strategies, in: Proceedings of the 22nd International Conference on Concurrency Theory, CONCUR'11, in: LNCS, vol. 6901, Springer, 2011, pp. 466–481.