Satisfiability in Strategy Logic can be Easier than Model Checking

Erman Acar[1], **Massimo Benerecetti**[2], **Fabio Mogavero**[2]
[1] Vrije Universiteit Amsterdam, The Netherlands
[2] Università di Napoli "Federico II", Italy
[1]erman.acar@vu.nl, [2]{massimo.benerecetti, fabio.mogavero}@unina.it

Abstract

In the design of complex systems, *model-checking* and *satisfiability* arise as two prominent decision problems. While model-checking requires the designed system to be provided in advance, satisfiability allows to check if such a system even exists. With very few exceptions, the second problem turns out to be harder than the first one from a complexity-theoretic standpoint. In this paper, we investigate the connection between the two problems for a non-trivial fragment of *Strategy Logic* (SL, for short). SL extends LTL with first-order quantifications over strategies, thus allowing to explicitly reason about the strategic abilities of agents in a multi-agent system. Satisfiability for the full logic is known to be highly undecidable, while model-checking is non-elementary.

The SL fragment we consider is obtained by preventing strategic quantifications within the scope of temporal operators. The resulting logic is quite powerful, still allowing to express important game-theoretic properties of multi-agent systems, such as existence of *Nash and immune equilibria*, as well as to formalize the *rational synthesis* problem. We show that satisfiability for such a fragment is PSPACE-COMPLETE, while its model-checking complexity is 2EXPTIME-HARD. The result is obtained by means of an elegant encoding of the problem into the satisfiability of *conjunctive-binding first-order logic*, a recently discovered decidable fragment of first-order logic.

Introduction

Model checking and *satisfiability* are two of the most prominent decision problems in logic. Many questions in various branches of *Computer Science*, from *Formal Verification*, to *Database Theory* and *Artificial Intelligence*, can be solved by encoding them as instances of these problems for a suitable logic. Some examples are verification of hardware and software, planning and scheduling, query containment, reactive and controller synthesis.

In general, given a logic $\mathcal{L}$, a class of structures $\mathcal{C}$ over which $\mathcal{L}$ is interpreted, and a formula $\phi \in \mathcal{L}$, the first problem asks whether a given structure $\mathcal{M} \in \mathcal{C}$ makes the formula ϕ true, while the second one requires deciding whether such a structure exists. The model-checking problem is, then, a typical verification problem, where a proposed solution, namely the structure $\mathcal{M}$, is checked for correctness *w.r.t.* the formula ϕ. On the other hand, satisfiability corresponds to a solution problem, where a correct solution, some witness structure $\mathcal{M}$ satisfying ϕ, has to be produced.

From a complexity-theoretical standpoint, satisfiability is almost always at least as complex as model checking. For instance, SAT, the satisfiability problem for *propositional logic*, is NPTIME-COMPLETE, while the corresponding model-checking problem is in PTIME. Similarly, satisfiability is PSPACE for plain *modal logic*, while model checking can be solved in linear time. In the context of temporal logics, the CTL and CTL* satisfiability problems are EXPTIME-COMPLETE (Kupferman, Vardi, and Wolper 2000) and 2EXPTIME-COMPLETE (Vardi and Stockmeyer 1985), respectively, while their model-checking problems are PTIME-COMPLETE and PSPACE-COMPLETE, respectively (Emerson and Clarke 1982) and (Clarke, Emerson, and Sistla 1983). Both model checking and satisfiability for LTL are PSPACE-COMPLETE (Pnueli 1977; Gabbay et al. 1980; Sistla and Clarke 1985; Vardi and Wolper 1986). Similarly, the two problems have the same complexity, namely 2EXPTIME-COMPLETE, also for more expressive logics suitable for strategic reasoning in multi-agent settings, such as ATL* (Alur, Henzinger, and Kupferman 2002; Schewe 2008) and the *One-Binding fragment* of *Strategy Logic* (Mogavero et al. 2012; 2014; 2017), while both problems are non-elementary for STL (Benerecetti, Mogavero, and Murano 2013; 2015). These results seem to support the common sense intuition that solving a problem (finding a satisfying model) is not easier than verifying the correctness of a proposed solution (checking that a given model is indeed satisfying).

The picture is, however, not as simple, since examples of fragments of known logics whose satisfiability is simpler than model checking do exist. In (Markey 2002), two fragments are identified for which model checking is PSPACE-COMPLETE, while satisfiability is NPTIME: the fragment only allowing for the temporal operators $\mathtt{F}$ (eventually) and $\mathtt{X}$ (next); and the one allowing only the operators $\mathtt{G}$ (globally) and $\mathtt{S}$ (since). In both fragments no temporal operators are allowed in the scope of a negation, hence they are not closed under negation. Therefore, the result holds for very weak fragments of LTL and the complexity gap rests on the assumption that NPTIME $\neq$ PSPACE. For branching-time logics, the only known results are those in (Goranko and

Vester 2014), where the authors show that satisfiability for non-trivial fragments of ATL* can be solved in PSPACE. In the identified fragments no nesting of strategic quantifiers in the scope of temporal operators is allowed, while full LTL is permitted in the scope of the strategic quantifications. The result is obtained by means of model-theoretic arguments very specific to the particular semantics of ATL*. In particular, the technique used there heavily relies on the fact that ATL* only allows for a single alternation of the implicit strategy quantifiers. Even though the authors do not mention the corresponding model-checking problems, the result entails that the model checking for that fragment is strictly more complex, being still 2EXPTIME-COMPLETE.

In this paper we investigate the connection between the two problems for a non-trivial fragment of Strategy Logic (SL, for short) (Chatterjee, Henzinger, and Piterman 2007; Mogavero, Murano, and Vardi 2010). SL extends LTL by means of two strategy quantifiers, the existential $\exists x$ and the universal $\forall x$, as well as agent bindings (a, x), where a is an agent and x a strategy variable. Intuitively, these elements can be respectively read as "there exists a strategy x", "for all strategies x", and "bind agent a to the strategy associated with x". The main technical differences between the two logics is that SL considers strategies as first class citizens and can express properties requiring an arbitrary alternation of the strategic quantifiers, while ATL* only allows for at most one such alternation. From a semantic viewpoint, this entails that ATL* cannot encode arbitrary functional dependencies among strategies. The ability to encode such dependencies is crucial to express relevant multi-agent systems.

The fragment we consider, called *Flat Conjunctive-Goal Strategy Logic* (FSL[CG], for short), strictly contains all the ATL* fragments studied in (Goranko and Vester 2014). As in those fragments, we prevent strategic quantification within the scope of temporal operators. Essentially, the allowed formulas are Boolean combinations of sentences in a specific prenex normal forms, where each sentence $\phi \triangleq \wp\eta$ is formed by a quantification prefix $\wp$ and a conjunction η of temporal goals, each one of the form $\flat\psi$, with $\flat$ a binding prefix and ψ a LTL formula. In other words, the considered logic is the conjunctive goal fragment SL[CG] of SL, as introduced in (Mogavero, Murano, and Sauro 2013), where quantifications are not allowed within temporal operators. The main result of the paper is that satisfiability in this fragment can be solved in PSPACE, a strictly lower complexity than the corresponding model-checking problem, which remains 2EXPTIME-COMPLETE. Clearly, since FSL[CG], like SL, allows for arbitrary alternation of quantifiers, the technique used in (Goranko and Vester 2014) cannot be applied. The result is obtained, instead, by exploiting a characterization of the semantics of FSL[CG] in a fragment of FOL, called FOL[CB], whose satisfiability was recently proved to be in Σ_3^{P} (Bova and Mogavero 2017). Even though the naive translation is exponential in the size of the original formula, a non-deterministic SMT-like procedure, which uses LTL as the background theory, can be defined that only requires polynomial space, hence delivering a PSPACE upper bound.

The significance of the result is twofold. On the one hand, it provides the first satisfiability procedure for a fragment of SL[CB]. To date, the only known satisfiability results for SL are the undecidability for the full logic (Mogavero, Murano, and Vardi 2010; Mogavero et al. 2017) and the completeness for 2EXPTIME for its one-binding fragment SL[1G] (Mogavero et al. 2012; 2017). On the other hand, it identifies an expressive fragment of SL, rich enough to express important game-theoretic properties, such as the existence of Nash equilibria, or arbitrary alternation of strategic quantification, whose satisfiability problem can be solved in PSPACE, the same complexity of classic LTL, and whose model checking is still 2EXPTIME-COMPLETE. The technique used to prove the result is also quite general and can, we believe, be extended even to richer fragments.

Preliminaries

We introduce the notion of multi-agent concurrent game (Alur, Henzinger, and Kupferman 2002), *i.e.*, the mathematical structure that describes the interaction between the agents and the associated properties. Here we rely on definitions similar to those reported in (Mogavero, Murano, and Sauro 2013; 2014; Mogavero et al. 2017).

A *concurrent game structure* (CGS, for short) is a tuple $\mathcal{G} \triangleq \langle \mathrm{AP}, \mathrm{Ag}, \mathrm{Ac}, \lambda, \mathrm{St}, \tau, s_0 \rangle$, where AP and Ag are finite non-empty sets of *atomic propositions* and *agents*, Ac and St are enumerable non-empty sets of *actions* and *states*, $s_0 \in \mathrm{St}$ is a designated *initial state*, and $\lambda : \mathrm{St} \to 2^{\mathrm{AP}}$ is a *labeling function* that maps each state to the set of atomic propositions true in that state. Let $\mathrm{Dc} \triangleq \mathrm{Ac}^{\mathrm{Ag}}$ be the set of *decisions*, *i.e.*, functions from Ag to Ac representing the choices of an action for each agent. Then, $\tau \colon \mathrm{St} \times \mathrm{Dc} \to \mathrm{St}$ is a *transition function* mapping states and decisions to states. A game structure $\mathcal{G}$ naturally induces a graph $\langle \mathrm{St}, \mathrm{Mv} \rangle$ with $\mathrm{Mv} \triangleq \{(s_1, s_2) : \exists d \in \mathrm{Dc}. \tau(s_1, d) = s_2\}$, where the finite (*resp.*, infinite) paths starting at the initial state s_0 represent all possible *histories* (*resp.*, *paths*), whose set is denoted by Hst (*resp.*, Pth). A *strategy* is a function $\sigma \in \mathrm{Str} \triangleq \mathrm{Hst} \to \mathrm{Ac}$ prescribing which action has to be performed at a certain history. Given a set of variables Vr, a *strategy assignment* in a CGS $\mathcal{G}$ is a partial function $\chi \in \mathrm{Asg} \triangleq \mathrm{Vr} \cup \mathrm{Ag} \to \mathrm{Str}$ mapping variables and agents to a strategy. An assignment χ is *complete* if it is defined on all agents, *i.e.*, $\mathrm{Ag} \subseteq \mathsf{dom}(\chi)$. A path $\pi \in \mathrm{Pth}$ is a *play w.r.t.* a complete assignment $\chi \in \mathrm{Asg}$ (χ-play, for short) if, for all $i \in \mathbb{N}$, it holds that $(\pi)_{i+1} = \tau((\pi)_i, d)$, where the decision $d \in \mathrm{Dc}$ is uniquely identified by the property $d(a) = \chi(a)((\pi)_{\leq i})$, for each agent $a \in \mathrm{Ag}$, where $(\pi)_{\leq i}$ is the history prefix of π up to index i. The partial function $\mathsf{play} \colon \mathrm{Asg} \to \mathrm{Pth}$, with $\chi \in \mathsf{dom}(\mathsf{play})$ *iff* χ is complete, returns the χ-play $\mathsf{play}(\chi)$.

Flat Strategy Logic Fragment

Strategy Logic (Chatterjee, Henzinger, and Piterman 2007; Mogavero, Murano, and Vardi 2010) is extension of LTL with first-order strategy quantifications $\exists x$ and $\forall x$ and agent bindings (a, x) that connect agent a with the adopted strategies x. The logic is geared towards the specification of game-theoretic properties of multi-agent systems and motivated by the lack of expressiveness of ATL*. In its full version its satisfiability problem is undecidable (Mogavero, Murano, and

Vardi 2010; Mogavero et al. 2017), while its model-checking is non-elementary hard. These negative results have triggered the study of fragments with more favorable computational properties (see, *e.g.*, (Mogavero, Murano, and Sauro 2013; 2014; P. Gardy and P. Bouyer and N. Markey 2018)). However, satisfiability has been proved decidable in 2EXPTIME only for its One-Goal fragment and still undecidable for its Boolean-Goal fragment (Mogavero et al. 2017). Here we consider a weaker version of this last fragment, where strategy quantifications and agent binding in the scope of temporal operators and disjunctions of goals are not allowed. We shall denote by $\wp = \mathtt{Qn}_1 x_1, \ldots \mathtt{Qn}_n x_2 \in \mathrm{Qnt}$, with $\mathtt{Qn}_i \in \{\forall, \exists\}$, and $\flat = (a_1, x_1), \ldots, (a_n, x_n) \in \mathrm{Bnd}$, with $a_i \in \mathrm{Ag}$, respectively, a generic quantification and binding prefix.

Definition 1 (FSL[CG] Syntax)**.** *Flat Conjunctive-Goal SL (*FSL[CG]*) formulas are built inductively from the sets of quantifier and binding prefixes* Qnt *and* Bnd *and atomic propositions* AP*, by using the following grammar:*

$$\varphi := \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \wp\eta \textit{ with } \wp \in \mathrm{Qnt},$$
$$\eta := \eta \wedge \eta \mid \flat\psi \textit{ with } \flat \in \mathrm{Bnd},$$
$$\psi := p \in \mathrm{AP} \mid \neg\psi \mid \psi \wedge \psi \mid \psi \vee \psi \mid \mathtt{X}\psi \mid \psi\mathtt{U}\psi \mid \psi\mathtt{R}\psi.$$

As it is usual for predicative logics, FSL[CG] requires the notion of *free variables and agents* in order to formalize its semantics. Intuitively, by $\mathsf{free}(\phi)$ we denote the set of variables that are not bound by any quantifier and agents for which there is no binding mentioning it in the scope of a temporal operator, for any FSL[CG] formula ϕ produced by one of the above rules. For the sake of space, we refer to (Mogavero et al. 2014; 2017) for the formal definition.

Similarly to ATL*, the semantics of FSL[CG] is defined *w.r.t.* CGS. For a FSL[CG] formula ϕ, a CGS $\mathcal{G}$, and an assignment χ with $\mathsf{free}(\phi) \subseteq \mathsf{dom}(\chi)$, we write $\mathcal{G}, \chi, s_0 \models \phi$ to indicate that ϕ holds at the initial state of $\mathcal{G}$ under χ. The semantics of FSL[CG] formulas involving atomic propositions, the Boolean connectives $\neg$, $\wedge$, and $\vee$, as well as the temporal operators X, U, and R is defined as usual. Although equivalent to the one reported in (Mogavero et al. 2014; 2017), the formalization for strategy quantifications, agent bindings, and play evaluation is reported for completeness.

Definition 2 (FSL[CG] Semantics)**.**

1. *For a variable $x \in \mathrm{Vr}$ and a formula ϕ, we set that:*
   - $\mathcal{G}, \chi \models \exists x.\phi$ *if there is a strategy $\sigma \in \mathrm{Str}$ such that* $\mathcal{G}, \chi[x \mapsto \sigma] \models \phi$*;*
   - $\mathcal{G}, \chi \models \forall x.\phi$ *if, for all strategies $\sigma \in \mathrm{Str}$, it holds that* $\mathcal{G}, \chi[x \mapsto \sigma] \models \phi$*.*
2. *For an agent $a \in \mathrm{Asg}$, a variable $x \in \mathrm{Vr}$, and a formula ϕ, we set that $\mathcal{G}, \chi \models (a, x)\phi$ if $\mathcal{G}, \chi[a \mapsto \chi(x)] \models \phi$.*
3. *Finally, for a complete assignment χ and a* LTL *formula ψ, we set that $\mathcal{G}, \chi \models \psi$ if $\mathsf{play}(\chi) \models_{\mathrm{LTL}} \psi$.*

In (Alur, Henzinger, and Kupferman 2002) it has been proved that the model checking of a single state formula of the form $\langle\langle \mathrm{A} \rangle\rangle\psi$, with ψ pure LTL, is 2EXPTIME-COMPLETE. Obviously, this formula belongs to FSL[CG], by using the standard embedding of ATL* into SL[1G], which is a fragment of SL[CG] (Mogavero et al. 2014). Indeed, $\langle\langle \mathrm{A} \rangle\rangle\psi$ is equivalent to the SL[1G] sentence $\exists x_1 \cdots x_i \forall x_{i+1} \cdots x_n (a_{j_1}, x_1) \cdots (a_{j_n}, x_n)\psi$, where $\mathrm{A} = \{a_{j_1}, \ldots, a_{j_n}\}$. Thus, the next theorem follows immediately.

Theorem 1 (FSL[CG] Model Checking)**.** FSL[CG] *model-checking problem is 2*EXPTIME-HARD.

Beside enjoying a decidable and even PSPACE satisfiability problem, as we shall show shortly, FSL[CG] is an interesting fragment as it allows to express non-trivial game-theoretic properties of multi-agent systems.

Consider the n agents in $\mathrm{Ag} = \{a_1, \ldots, a_n\}$, each one having a temporal goal described by one of the LTL formulas $\psi_1, \ldots, \psi_n$. Then, we can express the existence of a *Nash equilibrium* (Nash 1950; Osborne and Rubinstein 1994) via the FSL[CG] sentence

$$\varphi_{NE} \triangleq \bigvee_{\mathrm{A} \subseteq \mathrm{Ag}} \exists x_1 \cdots \exists x_n \forall y \,.\, \eta_{NE}, \text{ where}$$
$$\eta_{NE} \triangleq \left(\bigwedge_{a_i \in \mathrm{A}} \flat_i \neg\psi_{a_i} \wedge \bigwedge_{a \in \mathrm{Ag} \setminus \mathrm{A}} \flat\psi_a \right),$$

with $\flat \triangleq \prod_{i=1}^{n} (a_i, x_i)$ and $\flat_i \triangleq (a_i, y) \cdot \prod_{j=1, j \neq i}^{n} (a_j, x_j)$. Intuitively, the n agents enjoy a strategy profiles identified by the variables $x_1, \ldots, x_n$ having the property to be a Nash equilibrium if there is a subset A of them that are not able to satisfy the desired goal independently of the strategy, while all the others achieve their goal by sticking to the strategy specified in the profile. In this way, we are sure that there is no agent that can improve its payoff by deviating from what is prescribed in the profile.

An *immune equilibrium* (Halpern 2011), instead, ensures the existence of a strategy profile for which a deviation of one agent from its strategy cannot induce a decrease in the payoff of a different agent. Also this property can be formalized via a FSL[CG] sentence as follows:

$$\varphi_{IE} \triangleq \bigvee_{\mathrm{A} \subseteq \mathrm{Ag}} \exists x_1 \cdots \exists x_n \forall y \, \eta_{IE}, \text{ where}$$
$$\eta_{IE} \triangleq \left(\bigwedge_{a \in \mathrm{A}} \flat\neg\psi_a \wedge \bigwedge_{a \in \mathrm{Ag} \setminus \mathrm{A}} \bigwedge_{a_i \in \mathrm{Ag} \setminus a} \flat_i \psi_a \right).$$

Intuitively, there is a set of agents A that cannot satisfy their goals, while all the others can do it, independently of the deviations of some different agent.

We can also specify the existence of a Nash equilibrium that is also immune as follows:

$$\varphi_{INE} \triangleq \bigvee_{\mathrm{A} \subseteq \mathrm{Ag}} \exists x_1 \cdots \exists x_n \forall y \, (\eta_{IE} \wedge \eta_{NE}).$$

Finally, we can express the notion of *rational synthesis* (Fisman, Kupferman, and Lustig 2010; Kupferman, Perelli, and Vardi 2016), a recent improvement of the classical reactive synthesis in the context of system design, where the adversarial environment is not a monolithic block, but a subset $\mathrm{E} \subset \mathrm{Ag}$ of all the agents, each of them having their own goal. The aim of the environment is thus to oppose the system agents $\mathrm{Ag} \setminus \mathrm{E}$, while still finding an equilibrium for its own goals. In case of an *immune-Nash equilibrium*, we can formalize this problem via the FSL[CG] sentence

$$\varphi_{RS} \triangleq \bigvee_{\mathrm{A} \subseteq \mathrm{E}} \exists x_1 \cdots \exists x_n \forall y\, (\eta_{IE} \wedge \eta_{NE}).$$

The rational synthesis problem *w.r.t.* immune-Nash equilibria is solvable if there is a strategy profile for all the agents in $\mathrm{Ag} \setminus \mathrm{E}$ that, once extended with an equilibrium for those in E, identifies a play that satisfies their corresponding LTL goals. Note that none of the above properties is expressible in ATL* (Chatterjee, Henzinger, and Piterman 2007).

Satisfiability Procedure for FSL[CG]

The satisfiability decision procedure for the conjunctive-goal fragment of flat SL, namely FSL[CG], is based on a first-order characterization of the associated semantics. More specifically, we show the existence of a computable first-order sentence $\mathsf{fol}(\varphi)$, for each FSL[CG] sentence φ, that is satisfiable *iff* φ is satisfiable as well (see Theorem 2). This sentence belongs to a recently discovered decidable fragment of FOL, namely the conjunctive-binding fragment (FOL[CB], for short), proved to enjoy a Σ_3^{P} satisfiability problem (Bova and Mogavero 2017). Since its length is exponential in the size of φ, a direct reduction between the decision problems of the two logics would merely provide an EXPSPACE procedure (see Corollary 1). To obtain a lower complexity, we present a satisfiability criterion (see Theorem 4), whose verification can be carried out in PSPACE. The pseudo code of the entire procedure is reported in Algorithm 1. Notice that PSPACE-hardness easily follows from the fact that FSL[CG] naturally embeds LTL, whose decision problem is known to be PSPACE-COMPLETE (Sistla and Clarke 1982).

The idea behind the derivation of $\mathsf{fol}(\varphi)$ from φ is rather simple: first we replace every LTL formula ψ in φ with a corresponding fresh first-order relation r_ψ and, then, we axiomatize the LTL semantics, by determining which LTL formulas can be satisfied on the same path for a given labeling of the initial state. The intuition is that strategies in a CGS model of φ play the *rôle* of the domain elements in the first-order model of $\mathsf{fol}(\varphi)$ and a relation r_ψ is satisfied on an assignment for the first-order variables *iff* ψ is satisfied on the play induced by the corresponding strategies. Formally, $\mathsf{fol}\colon$ FSL[CG] $\rightarrow$ FOL[CB] is a function mapping every FSL[CG] sentence φ to a FOL[CB] sentence $\mathsf{fol}(\varphi) \triangleq \mathsf{trn}(\varphi) \wedge \mathsf{axm}(\varphi)$, where the two functions $\mathsf{trn}, \mathsf{axm}\colon$ FSL[CG] $\rightarrow$ FOL[CB] take care of encoding the two steps described above. Notice that, *w.l.o.g.*, we assume the input sentences φ to be in positive normal form, *i.e.*, negation operators only occur inside the LTL part of φ. The function trn is defined inductively as follows:

- $\mathsf{trn}(\varphi_1 \,\mathsf{Op}\, \varphi_2) \triangleq \mathsf{trn}(\varphi_1) \,\mathsf{Op}\, \mathsf{trn}(\varphi_2)$, for $\mathsf{Op} \in \{\wedge, \vee\}$;
- $\mathsf{trn}(\wp\, \eta) \triangleq \wp\, \mathsf{trn}(\eta)$;
- $\mathsf{trn}(\eta_1 \wedge \eta_2) \triangleq \mathsf{trn}(\eta_1) \wedge \mathsf{trn}(\eta_2)$;
- $\mathsf{trn}(\flat\psi) \triangleq r_\psi(\vec{\mathsf{t}_\flat})$, where the tuple of variables $\vec{\mathsf{t}_\flat} \in \mathrm{Vr}^{|\mathrm{Ag}|}$ is such that its i-th variable $(\vec{\mathsf{t}_\flat})_i$ is the one bound in $\flat$ to a_i, the i-th element in the enumeration of the agents in φ.

The translation function trn simply replaces each goal $\flat\psi$, *i.e.*, an application of a binding prefix $\flat$ to a LTL formula ψ, with an atom $r_\psi(\vec{\mathsf{t}_\flat})$, where the variables in $\vec{\mathsf{t}_\flat}$ are put in place of the positions the agents are associated with in the binding $\flat$. Notice that this translation abstracts away the LTL semantics of the formula ψ. As a consequence, two relations r_{ψ_1} and r_{ψ_2} can be satisfied on the same strategy assignment, even though the two corresponding LTL formulas ψ_1 and ψ_2 cannot hold on the same play, namely the conjunction $\psi_1 \wedge \psi_2$ is unsatisfiable. To recover the semantic relation between the LTL formulas ψ_1 and ψ_2 in the first-order encoding we need to enforce that the conjunction of r_{ψ_1} and r_{ψ_2} cannot be true on the same assignments, by adding the constraint $\forall \vec{x}.\neg(r_{\psi_1}(\vec{x}) \wedge r_{\psi_2}(\vec{x}))$ to the translation. The axm function generalizes this argument to arbitrary unsatisfiable set of LTL formulas. For a given set $\aleph$ of atomic propositions true at the initial state of a CGS, let

$$\mathrm{V}_\aleph \triangleq \left\{ \mathrm{L} \subseteq 2^{\mathrm{LTL}(\varphi)} : \not\models_{\mathrm{LTL}} \bigwedge(\mathrm{L} \cup \aleph \cup \neg(\mathrm{AP} \setminus \aleph)) \right\}$$

contain all those sets of LTL formulas in φ that are unsatisfiable along the same path. In the following, we abbreviate $\bigwedge(\mathrm{L} \cup \aleph \cup \neg(\mathrm{AP} \setminus \aleph))$ with $\bigwedge(\mathrm{L} \cup \widehat{\aleph})$, by setting $\widehat{\aleph} \triangleq \aleph \cup \neg(\mathrm{AP} \setminus \aleph)$. The axiomatization function produces a sentence that first guesses the labeling $\aleph$ of the initial state of the candidate CGS model of φ and then, for each set L in $\mathrm{V}_\aleph$, adds the negation of the conjunction of the corresponding relations, namely $\forall \vec{x}.\neg \bigwedge_{\psi \in \mathrm{L}} r_\psi(\vec{x})$. This enforces those relations to not overlap on the same assignments. Formally, $\mathsf{axm}(\varphi) \triangleq \bigvee_{\aleph \subseteq \mathrm{AP}} \forall \vec{x}. \bigwedge_{\mathrm{L} \in \mathrm{V}_\aleph} \neg \bigwedge_{\psi \in \mathrm{L}} r_\psi(\vec{x})$.

Let us consider the FSL[CG] formula φ_{NE} for the existence of a Nash equilibrium of the previous section. Applying the translation above, we obtain that $\mathsf{trn}(\varphi_{NE})$ is the disjunction, for each subset of agents $\mathrm{A} \subseteq \mathrm{Ag}$, of the sentences $\exists x_1 \cdots \exists x_n \forall y\, (\eta^1_{\mathrm{A}} \wedge \eta^2_{\mathrm{A}})$, where

- $\eta^1_{\mathrm{A}} \triangleq \bigwedge_{a_i \in \mathrm{A}} r_{\neg\psi_{a_i}}(x_1, \ldots, x_{i-1}, y, x_{i+1}, \ldots, x_n)$ and
- $\eta^2_{\mathrm{A}} \triangleq \bigwedge_{a \in \mathrm{Ag} \setminus \mathrm{A}} r_{\psi_a}(x_1, \ldots, x_n)$.

Moreover, if AP is the set of atomic propositions occurring in the LTL subformulas ψ_a, we have that $\mathsf{axm}(\varphi_{NE}) = \bigvee_{\aleph \subseteq \mathrm{AP}} \forall x_1 \cdots \forall x_n \bigwedge_{\mathrm{L} \in \mathrm{V}_\aleph} \neg \bigwedge_{\psi \in \mathrm{L}} r_\psi(x_1, \ldots, x_n)$, where the set $\mathrm{V}_\aleph$ is defined in such a way that $\mathrm{L} \in \mathrm{V}_\aleph$ *iff* the LTL formula $\bigwedge_{\psi \in \mathrm{L}} \psi \wedge \bigwedge_{p \in \aleph} p \wedge \bigwedge_{q \in \mathrm{AP} \setminus \aleph} \neg q$ is unsatisfiable, with $\mathrm{L} \subseteq \{\psi_a, \neg\psi_a : a \in \mathrm{Ag}\}$.

We can now prove the first result, establishing the equivalence between the satisfiability of a FSL[CG] sentence φ and of the corresponding FOL[CB] sentence $\mathsf{fol}(\varphi)$. Intuitively, given a CGS $\mathcal{G}$ satisfying φ, we can construct a first-order model $\mathcal{F}_\mathcal{G}$ for $\mathsf{fol}(\varphi)$, where the strategies in $\mathcal{G}$ play the *rôle* of the domain elements in $\mathcal{F}_\mathcal{G}$ and where a relation r_ψ is satisfied on a tuple of these strategies *iff* ψ is satisfied on the associated play. Conversely, given a first-order structure $\mathcal{F}$ satisfying φ, we can build a CGS model $\mathcal{G}_\mathcal{F}$ for φ, where the domain elements in $\mathcal{F}$ play the *rôle* of the actions in $\mathcal{G}_\mathcal{F}$ and the plays are uniquely identified by the decision chosen at the initial state (see Figure 1).

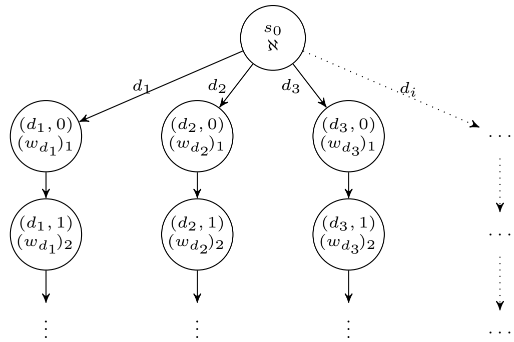

Figure 1: The CGS $\mathcal{G}_{\mathcal{F}}$.

Theorem 2 (FSL[CG] First-Order Characterization)**.** *For every* FSL[CG] *sentence φ, it holds that φ is satisfiable* iff $\mathsf{fol}(\varphi)$ *is satisfiable.*

Proof. ***(Only if).*** Suppose that the FSL[CG] sentence φ over the sets of atomic propositions AP and agent Ag is satisfiable. Then, there exists a CGS $\mathcal{G} = \langle \mathrm{AP}, \mathrm{Ag}, \mathrm{Ac}, \lambda, \mathrm{St}, \tau, s_0 \rangle$ such that $\mathcal{G} \models \varphi$. Now, consider the first-order model $\mathcal{F}_{\mathcal{G}}$, whose domain $\mathrm{F} \triangleq \mathrm{Str}_{\mathcal{G}}$ contains the possible strategies of the CGS and where the interpretation of the relations r_ψ is defined as follows: $\mathtt{t} \in r_\psi{}^{\mathcal{F}_{\mathcal{G}}}$ *iff* $\mathcal{G}, \chi, s_0 \models \psi$, where $\chi(a_i) = \mathtt{t}_i$, for all tuples $\mathtt{t} \in \mathrm{F}^{|\mathrm{Ag}|}$, strategy assignments χ, and LTL formulas $\psi \in \mathrm{LTL}(\varphi)$. Intuitively, a tuple of strategies $\mathtt{t}$ is part of the interpretation of a relation r_ψ *iff* the associated play starting from the initial state s_0 satisfies the LTL formula ψ. It is not hard to prove that $\mathcal{F}_{\mathcal{G}} \models \mathsf{fol}(\varphi)$, where we recall that $\mathsf{fol}(\varphi) = \mathsf{trn}(\varphi) \wedge \bigvee_{\aleph \subseteq \mathrm{AP}} \vartheta_\aleph$, with $\vartheta_\aleph \triangleq \forall \vec{x}. \bigwedge_{\mathrm{L} \in \mathrm{V}_\aleph} \neg \bigwedge_{\psi \in \mathrm{L}} r_\psi(\vec{x})$. Let us consider the two conjuncts separately.

Claim 1. *$\mathcal{F}_{\mathcal{G}} \models \mathsf{trn}(\varphi)$ and $\mathcal{F}_{\mathcal{G}} \models \vartheta_{\lambda(s_0)}$.*

The first part can be proved by a simple but tedious induction on the syntactic structure of the formula $\mathsf{trn}(\varphi)$. The base case, namely $\mathcal{G}, \chi, s_0 \models \flat\psi$ *iff* $\mathcal{F}_{\mathcal{G}}, \chi \models r_\psi(\vec{\mathtt{t}_\flat})$, for any strategy assignment χ, directly derives from the definition of the first-order model $\mathcal{F}_{\mathcal{G}}$, *i.e.*, from the fact that $\mathtt{t} \in r_\psi{}^{\mathcal{F}_{\mathcal{G}}}$ *iff* $\mathcal{G}, \chi, s_0 \models \psi$, as stated above. For the second part, suppose, by contradiction, that $\mathcal{F}_{\mathcal{G}} \not\models \vartheta_{\lambda(s_0)}$. Then, $\mathcal{F}_{\mathcal{G}} \models \exists \vec{x}. \bigvee_{\mathrm{L} \in \mathrm{V}_{\lambda(s_0)}} \bigwedge_{\psi \in \mathrm{L}} r_\psi(\vec{x})$. This means that there exist a tuple of strategies $\mathtt{t} \in \mathrm{F}^{|\mathrm{Ag}|}$ and a set of LTL formulas $\mathrm{L} \in \mathrm{V}_{\lambda(s_0)}$ such that $\mathtt{t} \in r_\psi{}^{\mathcal{F}_{\mathcal{G}}}$, for all $\psi \in \mathrm{L}$. However, by definition of the interpretation of the relations in $\mathcal{F}_{\mathcal{G}}$, we have that $\mathcal{G}, \chi, s_0 \models \psi$, for all $\psi \in \mathrm{L}$, where $\chi(a_i) = \mathtt{t}_i$. Therefore, there exists a LTL model for the formula $\bigwedge(\mathrm{L} \cup \lambda(s_0) \cup \neg(\mathrm{AP} \setminus \lambda(s_0)))$, contradicting the fact that $\mathrm{L} \in \mathrm{V}_{\lambda(s_0)}$.

(If). Suppose that the FOL[CB] sentence $\mathsf{fol}(\varphi)$ is satisfiable and let $\mathcal{F}$ be one of its models. Hence, $\mathcal{F} \models \mathsf{trn}(\varphi)$ and there exists a set of atomic propositions $\aleph \subseteq \mathrm{AP}$ such that $\mathcal{F} \models \vartheta_\aleph$. We can construct a CGS $\mathcal{G}_{\mathcal{F}} = \langle \mathrm{AP}, \mathrm{Ag}, \mathrm{Ac}, \lambda, \mathrm{St}, \tau, s_0 \rangle$ as follows. The actions are the elements of the domain of $\mathcal{F}$, *i.e.*, we set $\mathrm{Ac} \triangleq \mathrm{F}$. As a consequence, each decision function $d \in \mathrm{Dc}$ is represented by a tuple $\mathtt{t} \in \mathrm{F}^{|\mathrm{Ag}|}$, in the sense that $d(a_i) = (\mathtt{t})_i$, for each agent a_i. Moreover the set of states is $\mathrm{St} \triangleq \{s_0\} \cup \mathrm{Dc} \times \mathbb{N}$. Intuitively, the CGS is formed by an initial state s_0 and infinite sequences of states, one for each decision in Dc (see Figure 1). From s_0, following a decision $d \in \mathrm{Dc}$, we reach the state $(d, 0)$ that is connect to $(d, 1)$, independently of the decision of the agents, which is in turn connect to $(d, 2)$, and so on. Formally, we have $\tau(s_0, d) = (d, 0)$ and $\tau((d, i), d') = (d, i + 1)$, for all $d, d' \in \mathrm{Dc}$ and $i \in \mathbb{N}$. It remains to define the labeling function λ. We first need to prove the following claim. Let $\mathtt{t}_d$ be the tuple representing the decision funcion d and $\mathrm{L}_d \triangleq \left\{\psi \in \mathrm{LTL}(\varphi) : \mathtt{t}_d \in r_\psi{}^{\mathcal{F}}\right\}$ the set of LTL formulas ψ associated with a relation r_ψ, whose interpretation contains $\mathtt{t}_d$.

Claim 2. *$\bigwedge(\mathrm{L}_d \cup \widehat{\aleph})$ is satisfiable, for all $d \in \mathrm{Dc}$.*

Assume, by contradiction, that $\bigwedge(\mathrm{L}_d \cup \widehat{\aleph})$ is unsatisfiable, for some decision $d \in \mathrm{Dc}$. Then, by definition of the set $\mathrm{V}_\aleph$, it holds that $\mathrm{L}_d \in \mathrm{V}_\aleph$. This implies that $\mathcal{F} \models \neg \bigwedge_{\psi \in \mathrm{L}_d} r_\psi(\mathtt{t}_d)$, since, by assumption, we have already observed that $\mathcal{F} \models \vartheta_\aleph$. Thus, $\mathtt{t}_d \notin r_\psi{}^{\mathcal{F}}$, for some $\psi \in \mathrm{L}_d$, contradicting the definition of the set L_d itself.

To conclude the construction of the CGS $\mathcal{G}_{\mathcal{F}}$, consider, for each $d \in \mathrm{Dc}$, a LTL model w_d of the formula $\bigwedge(\mathrm{L}_d \cup \widehat{\aleph})$, *i.e.*, an infinite word $w_d \in (2^{\mathrm{AP}})^\omega$ such that $w_d \models \bigwedge(\mathrm{L}_d \cup \widehat{\aleph})$. Notice that, due to the $\widehat{\aleph} = \aleph \cup \neg(\mathrm{AP} \setminus \aleph)$ part of the formula, we necessarily have $(w_d)_0 = \aleph$. Hence, the labeling function can be defined as follows: $\lambda(s_0) = \aleph = (w_d)_0$ and $\lambda((d, i)) = (w_d)_{i+1}$, for all $d \in \mathrm{Dc}$ and $i \in \mathbb{N}$.

Claim 3. *$\mathcal{G}_{\mathcal{F}} \models \varphi$.*

The claim can be proved by induction on the syntactic structure of the formula φ, where, as base case, we show that $\mathcal{F}, \zeta \models r_\psi(\vec{\mathtt{t}_\flat})$ *iff* $\mathcal{G}_{\mathcal{F}}, \chi, s_0 \models \flat\psi$, for all assignments ζ and χ such that the action chosen in χ for a variable x at the initial state s_0 equals the one chosen in ζ for the same variable, *i.e.*, $\chi(x)(s_0) = \zeta(x)$. This easily follows from the definitions of the transition and labeling functions of $\mathcal{G}_{\mathcal{F}}$. $\square$

The above proof also shows that, if a FSL[CG] sentence is satisfiable, it is so on a bounded-width model, *i.e.*, a model with a finite number of actions. This follows from the finite model property of FOL[CB] (Bova and Mogavero 2017).

Since the length of $\mathsf{fol}(\varphi)$ is exponentially bounded by the size of φ and the satisfiability problem for FOL[CB] can be decided, as mentioned above, in Σ_3^{P}, the next result follows.

Corollary 1 (FSL[CG] Complexity Upper Bound)**.** *The satisfiability problem for* FSL[CG] *can be solved in* EXPSPACE.

In order to obtain an exponential improvement on the above result, we now introduce a satisfiability criterion for FSL[CG] that can be automatically verified in PSPACE. First of all, let us consider the Skolem normal form of the FOL[CB] sentence $\mathsf{fol}(\varphi)$. This can be obtained via the function $\mathsf{fol}^\forall \colon \mathrm{FSL[CG]} \to \mathrm{FOL}[\forall\mathrm{CB}]$, with $\mathsf{fol}^\forall(\varphi) =$

$\mathsf{trn}^{\forall}(\varphi) \wedge \mathsf{axm}(\varphi)$, where $\mathsf{trn}^{\forall}(\varphi)$ computes the Skolemization of the FOL[CB] sentence $\mathsf{trn}(\varphi)$. Observe that, being $\mathsf{axm}(\varphi)$ universal, the Skolemization procedure applied to $\mathsf{fol}(\varphi)$ only affects the $\mathsf{trn}(\varphi)$ component. Formally:

- $\mathsf{trn}^{\forall}(\varphi_1 \,\mathsf{Op}\, \varphi_2) \triangleq \mathsf{trn}^{\forall}(\varphi_1) \,\mathsf{Op}\, \mathsf{trn}^{\forall}(\varphi_2)$, for $\mathsf{Op} \in \{\wedge, \vee\}$;
- $\mathsf{trn}^{\forall}(\wp\, \eta) \triangleq \wp' \, \mathsf{trn}^{\forall}_{\wp}(\eta)$, where $\wp'$ is obtained by removing all existential quantifiers from $\wp$;
- $\mathsf{trn}^{\forall}_{\wp}(\eta_1 \wedge \eta_2) \triangleq \mathsf{trn}^{\forall}_{\wp}(\eta_1) \wedge \mathsf{trn}^{\forall}_{\wp}(\eta_2)$;
- $\mathsf{trn}^{\forall}_{\wp}(\flat\psi) \triangleq r_{\psi}(\vec{\mathsf{t}}_{\wp,\flat})$, where the term $\vec{\mathsf{t}}_{\wp,\flat}$ is obtained by replacing every existential variable in the term $\vec{\mathsf{t}}_{\flat}$ with the associated Skolem function *w.r.t.* the quantifier prefix $\wp$.

Considering again the Nash equilibrium specification φ_{NE}, we have that $\mathsf{trn}^{\forall}(\varphi_{NE})$ is the disjunction, for each subset of agents $\mathrm{A} \subseteq \mathrm{Ag}$, of the sentences $\forall y(\eta^{\forall 1}_{\mathrm{A}} \wedge \eta^{\forall 2}_{\mathrm{A}})$ built on the following quantifier-free formulas:

- $\eta^{\forall 1}_{\mathrm{A}} \triangleq \bigwedge_{a_i \in \mathrm{A}} r_{\neg\psi_{a_i}}(c_1, \ldots, c_{i-1}, y, c_{i+1}, \ldots, c_n)$;
- $\eta^{\forall 2}_{\mathrm{A}} \triangleq \bigwedge_{a \in \mathrm{Ag} \setminus \mathrm{A}} r_{\psi_a}(c_1, \ldots, c_n)$,

where c_i is the Skolem constant of the existential variable x_i under the scope of the disjunct identified by A.

The Skolem normal form is used in (Bova and Mogavero 2017) (see Theorems 4 and 5) to obtain a canonical form for FOL[CB], namely FOL[∀1B], for which it is possible to define a Σ^{P}_3-verifiable satisfiability criterion. For completeness, a simple variation of this criterion is reported in Theorem 3.

In order to formalize the criterion, we need to introduce some necessary notions first. An implicant for a positive Boolean formula β is a subset $\mathrm{I} \in \mathrm{Im}(\beta)$ of the propositions occurring in β such that $\mathrm{I} \models \beta$. This notion can be easily lifted to any FSL[CG] or FOL[CB] sentence φ, by considering it as positive Boolean formula over the set of sentences in prenex form $\wp\eta$ occurring in φ. For instance, given the sentence $\varphi \triangleq \wp_1\eta_1 \vee (\wp_2\eta_2 \wedge (\wp_3\eta_3 \vee \wp_4\eta_4))$, we have that $\mathrm{Im}(\varphi) = \{\{\wp_1\eta_1\}, \{\wp_2\eta_2, \wp_3\eta_3\}, \{\wp_2\eta_2, \wp_4\eta_4\}\}$. For a set of FOL[CB] prenex sentences $\mathrm{U} = \{\wp_1\eta_1, \ldots, \wp_n\eta_n\}$, with $\mathrm{Tr}(\mathrm{U})$ we denote the set of terms occurring in all quantifier-free formulas $\mathrm{Mt}(\mathrm{U}) = \{\eta_1, \ldots, \eta_n\}$. For instance, $\mathrm{Tr}(\mathrm{U}) = \{\mathsf{t}_1, \mathsf{t}_2, \mathsf{t}_3\}$ where $\mathrm{U} = \{\wp_1\eta_1, \wp_2\eta_2\}$ and $\mathrm{Mt}(\mathrm{U}) = \{\eta_1, \eta_2\}$ with $\eta_1 = r_1(\mathsf{t}_1) \wedge r_2(\mathsf{t}_1), \eta_2 = r_1(\mathsf{t}_2) \vee (r_2(\mathsf{t}_2) \wedge r_3(\mathsf{t}_3))$. Finally, for an arbitrary Boolean formula γ over atoms of the form $r(\mathsf{t})$, we indicate by $\mathsf{bool}(\gamma)$ the Boolean formula over the syntactic relations obtained from γ by erasing all the occurrences of the terms, *e.g.*, $\mathsf{bool}(\eta_2) = r_1 \vee (r_2 \wedge r_3)$.

In the following, we will diffusely use the classic notion of unifiability of terms (Baader and Snyder 2001), where we intuitively say that a set of terms $\mathrm{T} = \{\mathsf{t}_1, \ldots, \mathsf{t}_n\}$ unifies if there exists a substitution of variables μ that, once applied to each t_i, returns the same term $\mathsf{t} = \mathsf{t}_i^{\mu}$ (see (Bova and Mogavero 2017) for more details). This notion can be lifted to sets of quantifier-binding prefix pairs (*qb-pairs*, for short) $(\wp, \flat)$ occurring in a given FSL[CG] formula φ as follows. We say that a set $\mathrm{S} = \{(\wp_1, \flat_1), \ldots (\wp_n, \flat_n)\}$ unifies if the associated set of terms $\mathrm{T} = \{\vec{\mathsf{t}}_{\wp_1,\flat_1}, \ldots, \vec{\mathsf{t}}_{\wp_n,\flat_n}\}$ obtained from the application of $\mathsf{fol}^{\forall}$ to φ unifies as well.

Theorem 3 ((Bova and Mogavero 2017, Theorem 5))**.** *For every* FOL[∀CB] *sentence φ, the following are equivalent: (i) φ is satisfiable; (ii) there exists an implicant* $\mathrm{J} \in \mathrm{Im}(\varphi)$ *such that, for all subsets of sentences* $\mathrm{U} \subseteq \{\forall\vec{x}.\eta_j : \forall\vec{x}. \bigwedge_{i=1}^{n} \eta_i \in \mathrm{J}\}$ *whose set of terms* $\mathrm{Tr}(\mathrm{U})$ *is unifiable, the Boolean formula* $\bigwedge_{\gamma \in \mathrm{Mt}(\mathrm{U})} \mathsf{bool}(\gamma)$ *is satisfiable.*

The intuitive idea behind the criterion stated in the next theorem is the following. First notice that for a FSL[CG] sentence φ to be satisfiable one of its implicants $\mathrm{I} = \{\wp_1\eta_1, \ldots, \wp_n\eta_n\} \in \mathrm{Im}(\varphi)$ must necessarily exist such that the conjunction $\bigwedge_{i=1}^{n} \wp_n\eta_n$ is satisfiable. Now, it can be shown that if a subset of its qb-pairs $\mathrm{S} = \{(\wp_{i_1}, \flat_{i_1}), \ldots, (\wp_{i_k}, \flat_{i_k})\}$ is unifiable, all the associated LTL formulas $\mathrm{L} = \{\psi_{i_1}, \ldots, \psi_{i_k}\}$ are verified on at least one common path. This means that the conjunction $\bigwedge_{j=1}^{k} \psi_{i_j}$ must be satisfiable. On the contrary, if S is not unifiable, it is possible to construct a CGS in such a way that all LTL formulas in L are verified on different paths. In the following, given an implicant I and qb-pairs set S, let $\Delta^{\mathrm{S}}_{\mathrm{I}}$ contain all the formulas ψ_i such that $(\wp, \flat_i)$ occurs in S and $\flat_i\psi_i$ occurs in some sentence $\wp \bigwedge_{j=1}^{n} \flat_j\psi_j$ in I.

Theorem 4 (FSL[CG] Satisfiability Characterization)**.** *For every* FSL[CG] *sentence φ, the following are equivalent: (i) φ is satisfiable; (ii) there exist a subset of atomic propositions* $\aleph \subseteq \mathrm{AP}$ *and an implicant* $\mathrm{I} \in \mathrm{Im}(\varphi)$ *such that, for all unifiable sets of qb-pairs* $\mathrm{S} \subseteq \left\{(\wp, \flat_i) : \wp \bigwedge_{j=1}^{n} \flat_j\psi_j \in \mathrm{I}\right\}$*, the* LTL *formula* $\bigwedge(\Delta^{\mathrm{S}}_{\mathrm{I}} \cup \widehat{\aleph})$ *is satisfiable.*

Proof. By Theorem 2, φ is satisfiable *iff* $\mathsf{fol}(\varphi)$ is satisfiable. Thus, due to the standard first-order Skolem Theorem, we have that the original FSL[CG] sentence is satisfiable *iff* the FOL[∀CB] sentence $\mathsf{fol}^{\forall}(\varphi)$ is. Thanks to the satisfiability characterization reported in Theorem 3, we have that φ is satisfiable *iff (∗)* there exists an implicant $\mathrm{J} \in \mathrm{Im}(\mathsf{fol}^{\forall}(\varphi))$ such that, for all subsets of sentences $\mathrm{U} \subseteq \left\{\forall\vec{x}.r_{\psi_i}(\vec{\mathsf{t}}_{\wp,\flat_i}) : \forall\vec{x}. \bigwedge_{j=1}^{n} r_{\psi_j}(\vec{\mathsf{t}}_{\wp,\flat_j}) \in \mathrm{J}\right\}$ whose set of terms $\mathrm{Tr}(\mathrm{U})$ is unifiable, the Boolean formula $\bigwedge_{\gamma \in \mathrm{Mt}(\mathrm{U})} \mathsf{bool}(\gamma)$ is also satisfiable. Hence, to conclude the proof, we just need to show that the two statements *(∗)* and *(ii)* are equivalent. To do this, recall that $\mathsf{fol}^{\forall}(\varphi) = \mathsf{trn}^{\forall}(\varphi) \wedge \bigvee_{\aleph \subseteq \mathrm{AP}} \vartheta_{\aleph}$, where $\vartheta_{\aleph} \triangleq \forall\vec{x}. \bigwedge_{\mathrm{L} \in \mathrm{V}_{\aleph}} \neg \bigwedge_{\psi \in \mathrm{L}} r_{\psi}(\vec{x})$.

(Only if). For the forward direction, we show how to extract from the implicant J of $\mathsf{fol}^{\forall}(\varphi)$ satisfying *(∗)*, whose existence is assumed by hypothesis, two sets $\aleph$ and I satisfying the statement *(ii)*. Since $\mathrm{J} \in \mathrm{Im}(\mathsf{fol}^{\forall}(\varphi))$, there must exist a subset of atomic propositions $\aleph \subseteq \mathrm{AP}$ such that $\vartheta_{\aleph} \in \mathrm{J}$. Moreover, $\mathrm{J} \setminus \{\vartheta_{\aleph}\} \in \mathrm{Im}(\mathsf{trn}^{\forall}(\varphi))$. Therefore, due to the definition of the Skolemized translation function $\mathsf{trn}^{\forall}$, the set of FSL[CG] sentences $\mathrm{I} \triangleq \left\{\wp \bigwedge_{i=1}^{n} \flat_i\psi_i \in \mathrm{FSL[CG]} : \forall\vec{x}. \bigwedge_{i=1}^{n} r_{\psi_i}(\vec{\mathsf{t}}_{\wp,\flat_i}) \in \mathrm{J} \setminus \{\vartheta_{\aleph}\}\right\}$ is an implicant of φ, *i.e.*, $\mathrm{I} \in \mathrm{Im}(\varphi)$. Now, consider an arbitrary unifiable set of qb-pairs $\mathrm{S} \subseteq \left\{(\wp, \flat_i) : \wp \bigwedge_{i=1}^{n} \flat_i\psi_i \in \mathrm{I}\right\}$ and the associated set of sentences $\mathrm{U} \subseteq \mathrm{J}$ defined as follows: $\mathrm{U} \triangleq \left\{\forall\vec{x}. \bigwedge_{i=1}^{n} r_{\psi_i}(\vec{\mathsf{t}}_{\wp,\flat_i}) \in \mathrm{J} : (\wp, \flat_i) \in \mathrm{S}\right\} \cup \{\vartheta_{\aleph}\}$.

By definition of unifiability for S, we have that the set of terms $\mathrm{Tr}(\mathrm{U})$ is necessarily unifiable, since the terms in $\vartheta_{\aleph}$ contains only variables and cannot obstruct the unification. As a consequence, by $(*)$, the Boolean formula $\beta = \bigwedge_{\gamma \in \mathrm{Mt}(\mathrm{U})} \mathsf{bool}(\gamma)$ is satisfiable, where one can observe that $\beta = \bigwedge_{\forall \vec{x}. r_{\psi_i}(\vec{\mathsf{t}}_{\wp, \flat_i}) \in \mathrm{U} \setminus \{\vartheta_{\aleph}\}} r_{\psi_i} \wedge \mathsf{bool}(\vartheta_{\aleph}) = \bigwedge_{\psi \in \Delta^{\mathrm{S}}_{\mathrm{I}}} r_{\psi} \wedge \mathsf{bool}(\vartheta_{\aleph})$, with $\mathsf{bool}(\vartheta_{\aleph}) = \bigwedge_{\mathrm{L} \in \mathrm{V}_{\aleph}} \neg \bigwedge_{\psi \in \mathrm{L}} r_{\psi}$. At this point, it is easy to see that the LTL formula $\bigwedge(\Delta^{\mathrm{S}}_{\mathrm{I}} \cup \widehat{\aleph})$ is also satisfiable. Indeed, if this were not the case, we would have, by definition of the set $\mathrm{V}_{\aleph}$, that $\Delta^{\mathrm{S}}_{\mathrm{I}} \in \mathrm{V}_{\aleph}$. But this in turn would imply that β is unsatisfiable, contradicting the hypothesis on the truth of *$(*)$*.

(If). For the opposite direction, we derive a set J satisfying the statement *$(*)$* from the existence of a set of atomic propositions $\aleph$ and the implicant I of φ satisfying *(ii)*. As first step, consider the set of $\mathrm{FOL}[\forall \textsc{cb}]$ sentences $\mathrm{J} \triangleq \left\{ \forall \vec{x}. \bigwedge_{j=1}^{n} r_{\psi_j}(\vec{\mathsf{t}}_{\wp, \flat_j}) \in \mathrm{FOL}[\forall \textsc{cb}] : \wp \bigwedge_{j=1}^{n} \flat_j \psi_j \in \mathrm{I} \right\} \cup \{\vartheta_{\aleph}\}$. By definition of the Skolemized translation function $\mathsf{trn}^{\forall}$, we have that $\mathrm{J} \in \mathrm{Im}(\mathsf{fol}^{\forall}(\varphi))$, since $\mathrm{I} \in \mathrm{Im}(\varphi)$ and $\{\vartheta_{\aleph}\} \in \mathrm{Im}(\bigvee_{\aleph \subseteq \mathrm{AP}} \vartheta_{\aleph})$. Now, let $\mathrm{U} \subseteq \left\{ \forall \vec{x}. r_{\psi_i}(\vec{\mathsf{t}}_{\wp, \flat_i}) : \forall \vec{x}. \bigwedge_{j=1}^{n} r_{\psi_j}(\vec{\mathsf{t}}_{\wp, \flat_j}) \in \mathrm{J} \right\}$ be an arbitrary set of sentences whose set of terms $\mathrm{Tr}(\mathrm{U})$ is unifiable. If $\vartheta_{\aleph} \notin \mathrm{U}$, the Boolean formula $\beta = \bigwedge_{\gamma \in \mathrm{Mt}(\mathrm{U})} \mathsf{bool}(\gamma)$ does not contain negated relations, thus, it is trivially satisfiable. If, on the other hand, $\vartheta_{\aleph} \in \mathrm{U}$, consider the set of qb-pairs $\mathrm{S} \triangleq \left\{ (\wp, \flat_i) : \wp \bigwedge_{j=1}^{n} \flat_j \psi_j \in \mathrm{I} \wedge \forall \vec{x}. r_{\psi_i}(\vec{\mathsf{t}}_{\wp, \flat_i}) \in \mathrm{U} \right\}$ associated with U. By construction, it holds that S is unifiable. Therefore, by assumption, we have that the LTL formula $\bigwedge(\Delta^{\mathrm{S}}_{\mathrm{I}} \cup \widehat{\aleph})$ is satisfiable. Hence, by definition of $\mathrm{V}_{\aleph}$, it is immediate to observe that, for every $\mathrm{L} \subseteq \Delta^{\mathrm{S}}_{\mathrm{I}}$ it holds that $\mathrm{L} \notin \mathrm{V}_{\aleph}$. Moreover, $\beta = \bigwedge_{\gamma \in \mathrm{Mt}(\mathrm{U})} \mathsf{bool}(\gamma) = \bigwedge_{\psi \in \Delta^{\mathrm{S}}_{\mathrm{I}}} r_{\psi} \wedge \bigwedge_{\mathrm{L} \in \mathrm{V}_{\aleph}} \neg \bigwedge_{\psi \in \mathrm{L}} r_{\psi}$. Consequently, the Boolean formula β is satisfiable, as required by the statement *$(*)$*. □

Algorithm 1: $\mathrm{FSL}[\textsc{cg}]$ Satisfiability Procedure.

signature $\mathsf{sat} \colon \mathrm{FSL}[\textsc{cg}] \to \mathbb{B}$
function $\mathsf{sat}(\varphi)$

1 $sat \leftarrow \mathtt{ff}$
2 **foreach** $\aleph \subseteq \mathrm{AP}$ **and** $\mathrm{I} \in \mathrm{Im}(\varphi)$ **do**
3 $\quad sat \leftarrow \mathtt{tt}$
4 $\quad$ **foreach** $\mathrm{S} \subseteq \left\{ (\wp, \flat_i) : \wp \bigwedge_{j=1}^{n} \flat_j \psi_j \in \mathrm{I} \right\}$ **do**
5 $\quad\quad$ **if** S *unifiable* **then**
6 $\quad\quad\quad$ **if not** $\mathsf{sat}_{\mathrm{LTL}}(\bigwedge(\Delta^{\mathrm{S}}_{\mathrm{I}} \cup \bigwedge \widehat{\aleph}))$ **then**
7 $\quad\quad\quad\quad sat \leftarrow \mathtt{ff}$
8 $\quad\quad\quad\quad$ **break**
9 $\quad$ **if** sat **then break**
10 **return** sat

Algorithm 1 reports a procedure that, for any $\mathrm{FSL}[\textsc{cg}]$ sentence φ, verifies the corresponding criterion stated in Theorem 4. More specifically, the existential search for both a set of atomic propositions $\aleph$ and an implicant I of φ is done at Line 2, while the universal search for a unifiable set of qb-pairs S is performed at Lines 4 and 5. Finally, the LTL satisfiability test is performed at Line 6.

The next corollary immediately follows from the fact that the three set variables $\aleph$, I, and S used in the algorithm only require a number of bits that is polynomially bounded by the length of φ. The external call to the LTL solver only needs polynomial space in the input formula $\bigwedge(\Delta^{\mathrm{S}}_{\mathrm{I}} \cup \bigwedge \widehat{\aleph})$ (Sistla and Clarke 1982), whose length is polynomial in the size of ϕ.

Corollary 2 ($\mathrm{FSL}[\textsc{cg}]$ Satisfiability Complexity)**.** *The satisfiability problem for* $\mathrm{FSL}[\textsc{cg}]$ *is* PSPACE-COMPLETE*.*

As a final remark, observe that, although the $\mathrm{FSL}[\textsc{cg}]$ sentences used to formalize Nash and immune equilibria, as well as rational synthesis, are exponential in the number of agents, their satisfiability can still be checked in PSPACE. Indeed, every disjunct has polynomial size in the number of agents, and, given the set of agents A, can be computed in PSPACE independently of the other ones. Thus, we can apply Algorithm 1 to each of them separately and determine if it returns true for at least one. The same holds for any $\mathrm{FSL}[\textsc{cg}]$ sentence φ in disjunctive normal form, *i.e.*, where φ can be viewed as a DNF Boolean formula over the atoms $\wp\eta$.

Discussion

We have studied the relationship between model-checking and satisfiability in Strategy Logic. In particular, we identified a fragment of SL that allows for conjunction of goals but prevents strategic quantifications and agent bindings within temporal operators. From a semantic viewpoint, this restriction inhibits agents from changing their strategies during an execution. Despite this limitations, the resulting logic is still expressive enough to encode relevant game-theoretic properties of multi-agent systems, like existence of Nash equilibria. As a main contribution, we have shown that the logic enjoys a PSPACE-COMPLETE satisfiability problem, while its model-checking problem is strictly harder, being 2EXPTIME-HARD. This also provides the first decidability result for satisfiability in SL, other than the One-Goal fragment $\mathrm{SL}[1\mathrm{G}]$.

Our result is, in a sense, a non-trivial generalization of the one reported in (Goranko and Vester 2014), where the authors prove the same bounds for a fragment of $\mathrm{ATL}^{\star}$ that is much weaker than $\mathrm{FSL}[\textsc{cg}]$. It also requires a more powerful proof technique, since the one used in there rests on model-theoretic properties of $\mathrm{ATL}^{\star}$ that are not shared by $\mathrm{FSL}[\textsc{cg}]$. In particular, the fact that $\mathrm{FSL}[\textsc{cg}]$, unlike $\mathrm{ATL}^{\star}$, allows for arbitrary alternation of quantifiers and conjunctions of different goals makes the argument of (Goranko and Vester 2014) inapplicable to our case. The approach followed here, instead, is to translate formulas of $\mathrm{FSL}[\textsc{cg}]$ into equisatisfiable formulas in $\mathrm{FOL}[\textsc{cb}]$, the conjunctive-binding fragment of FOL, whose decidability in Σ^{P}_{3} was recently proved. The reduction approach used for $\mathrm{FSL}[\textsc{cg}]$ does not depend on the specific syntactic features of the logics and can be applied to more expressive fragments of $\mathrm{SL}[\textsc{bg}]$. In particular, we conjecture that we can allow the nesting of strategy quantifications inside the first argument of a release operator or in the second argument of an until operator while still preserving a PSPACE satisfiability procedure.

Theorem 1 and Corollary 2, showing that satisfiability can be easier than model-checking, witness an already rare phenomenon. Even more so, considering that the SL fragment involved is still quite powerful. In general, however, a clear understanding of the actual relationship between model-checking and satisfiability is not only interesting from a theoretical perspective, but it may also carry important practical consequences. Indeed, the usual approach to system design is to manually produce the system first and, then, check that it satisfies some required properties. The checking phase can in some cases be automated, by using, for instance, a model-checking procedure. A second, more appealing, approach is to directly synthesize the system starting from the desired property alone. This could be done by solving, when decidable, a satisfiability problem for that property and taking the witness model produced by the decision procedure as the desired system. Given the often much higher complexity of solving satisfiability, the first approach is almost always the most convenient, if not the only feasible one, despite the cost of the manual design. An immediate consequence of our result is that automatically synthesizing a multi-agent system satisfying properties expressible in FSL[CG] is not only a viable option, but actually preferable to the first design approach also from a complexity-theoretical viewpoint.

References

Alur, R.; Henzinger, T.; and Kupferman, O. 2002. Alternating-Time Temporal Logic. *JACM* 49(5):672–713.

Baader, F., and Snyder, W. 2001. Unification Theory. In *Handbook of Automated Reasoning (vol. 1)*. Elsevier & MIT Press. 445–532.

Benerecetti, M.; Mogavero, F.; and Murano, A. 2013. Substructure Temporal Logic. In *LICS'13*, 368–377. IEEECS.

Benerecetti, M.; Mogavero, F.; and Murano, A. 2015. Reasoning About Substructures and Games. *TOCL* 16(3):25:1–46.

Bova, S., and Mogavero, F. 2017. Herbrand Property, Finite Quasi-Herbrand Models, and a Chandra-Merlin Theorem for Quantified Conjunctive Queries. In *LICS'17*, 1–12. ACM.

Chatterjee, K.; Henzinger, T.; and Piterman, N. 2007. Strategy Logic. In *CONCUR'07*, LNCS 4703, 59–73. Springer.

Clarke, E.; Emerson, E.; and Sistla, A. 1983. Automatic Verification of Finite-State Concurrent Systems Using Temporal Logic Specifications: A Practical Approach. In *POPL'83*, 117–126. ACM.

Emerson, E., and Clarke, E. 1982. Design and Synthesis of Synchronization Skeletons Using Branching-Time Temporal Logic. *SCP* 2(3):241–266.

Fisman, D.; Kupferman, O.; and Lustig, Y. 2010. Rational Synthesis. In *TACAS'10*, LNCS 6015, 190–204. Springer.

Gabbay, D.; Pnueli, A.; Shelah, S.; and Stavi, J. 1980. On the Temporal Basis of Fairness. In *POPL'80*, 163–173. ACM.

Goranko, V., and Vester, S. 2014. Optimal Decision Procedures for Satisfiability in Fragments of Alternating-time Temporal Logics. In *AIML'14*, 234–253. College Pub.

Halpern, J. 2011. Beyond Nash Equilibrium: Solution Concepts for the 21st Century. In *GAMESEC'11*, LNCS 7037, 1–3. Springer.

Kupferman, O.; Perelli, G.; and Vardi, M. 2016. Synthesis with Rational Environments. *AMAI* 78(1):3–20.

Kupferman, O.; Vardi, M.; and Wolper, P. 2000. An Automata Theoretic Approach to Branching-Time Model Checking. *JACM* 47(2):312–360.

Markey, N. 2002. Past is for Free: on the Complexity of Verifying Linear Temporal Properties with Past. *ENTCS* 68(2):87–104.

Mogavero, F.; Murano, A.; Perelli, G.; and Vardi, M. 2012. What Makes ATL* Decidable? A Decidable Fragment of Strategy Logic. In *CONCUR'12*, LNCS 7454, 193–208. Springer.

Mogavero, F.; Murano, A.; Perelli, G.; and Vardi, M. 2014. Reasoning About Strategies: On the Model-Checking Problem. *TOCL* 15(4):34:1–42.

Mogavero, F.; Murano, A.; Perelli, G.; and Vardi, M. 2017. Reasoning About Strategies: On the Satisfiability Problem. *LMCS* 13(1:9):1–37.

Mogavero, F.; Murano, A.; and Sauro, L. 2013. On the Boundary of Behavioral Strategies. In *LICS'13*, 263–272. IEEECS.

Mogavero, F.; Murano, A.; and Sauro, L. 2014. Strategy Games: A Renewed Framework. In *AAMAS'14*, 869–876. IFAAMAS.

Mogavero, F.; Murano, A.; and Vardi, M. 2010. Relentful Strategic Reasoning in Alternating-Time Temporal Logic. In *LPAR'10*, LNAI 6355, 371–387. Springer.

Nash, J. 1950. Equilibrium Points in n-Person Games. *PNAS* 36(1):48–49.

Osborne, M., and Rubinstein, A. 1994. *A Course in Game Theory*. MIT Press.

P. Gardy and P. Bouyer and N. Markey. 2018. Dependences in Strategy Logic. In *STACS'18*, LIPIcs 96, 34:1–15. Leibniz-Zentrum fuer Informatik.

Pnueli, A. 1977. The Temporal Logic of Programs. In *FOCS'77*, 46–57. IEEECS.

Schewe, S. 2008. ATL* Satisfiability is 2ExpTime-Complete. In *ICALP'08*, LNCS 5126, 373–385. Springer.

Sistla, A., and Clarke, E. 1982. The Complexity of Propositional Linear Temporal Logics. In *STOC'82*, 159–168. ACM.

Sistla, A., and Clarke, E. 1985. The Complexity of Propositional Linear Temporal Logics. *JACM* 32(3):733–749.

Vardi, M., and Stockmeyer, L. 1985. Improved Upper and Lower Bounds for Modal Logics of Programs: Preliminary Report. In *STOC'85*, 240–251. ACM.

Vardi, M., and Wolper, P. 1986. Automata-Theoretic Techniques for Modal Logics of Programs. *JCSS* 32(2):183–221.