

# Dependences in Strategy Logic\*

Patrick Gardy<sup>1</sup>, Patricia Bouyer<sup>1</sup>, and Nicolas Markey<sup>1,2</sup>

1 LSV, CNRS & ENS Paris-Saclay, Univ. Paris-Saclay, France

2 Univ. Rennes, CNRS, Inria, France

---

## Abstract

Strategy Logic (SL) is a very expressive temporal logic for specifying and verifying properties of multi-agent systems: in SL, one can quantify over strategies, assign them to agents, and express LTL properties of the resulting plays. Such a powerful framework has two drawbacks: first, model checking SL has non-elementary complexity; second, the exact semantics of SL is rather intricate, and may not correspond to what is expected. In this paper, we focus on *strategy dependences* in SL, by tracking how existentially-quantified strategies in a formula may (or may not) depend on other strategies selected in the formula, revisiting the approach of [Mogavero *et al.*, Reasoning about strategies: On the model-checking problem, 2014]. We explain why *elementary* dependences, as defined by Mogavero *et al.*, do not exactly capture the intended concept of behavioral strategies. We address this discrepancy by introducing *timeline* dependences, and exhibit a large fragment of SL for which model checking can be performed in 2-EXPTIME under this new semantics.

**1998 ACM Subject Classification** F.3.1 Specifying and Verifying and Reasoning about Programs, F.4.1 Mathematical Logic

**Keywords and phrases** Game theory; strategy logic; behavioural strategies

**Digital Object Identifier** 10.4230/LIPIcs.STACS.2018.34

## 1 Introduction

**Temporal logics.** Since Pnueli's seminal paper [24] in 1977, temporal logics have been widely used in theoretical computer science, especially by the formal-verification community. Temporal logics provide powerful languages for expressing properties of reactive systems, and enjoy efficient algorithms for satisfiability and model checking [9]. Since the early 2000s, new temporal logics have appeared to address *open* and *multi-agent systems*. While classical temporal logics (e.g. CTL [8, 25] and LTL [24]) could only deal with one or all the behaviours of the whole system, ATL [2] expresses properties of (executions generated by) behaviours of individual components of the system. ATL has been extensively studied since then, both about its expressiveness and about its verification algorithms [2, 13, 16].

**Strategic interactions in ATL.** Strategies in ATL are handled in a very limited way, and there are no real *strategic interactions* in that logic (which, in return, enjoys a polynomial-time model-checking algorithm). Over the last 10 years, various extensions have been defined and studied in order to allow for more interactions [1, 7, 6, 18, 26]. *Strategy Logic* (SL for short) [7, 18] is such a powerful approach, in which strategies are first-class objects; formulas can quantify (universally and existentially) over strategies, store those strategies in variables, assign them to players, and express properties of the resulting plays. As a simple example, the existence of a winning strategy for Player *A* (with objective  $\varphi_A$ ) against any strategy of

---

\* Supported by ERC project EQualIS



© Patrick Gardy, Patricia Bouyer, Nicolas Markey;  
licensed under Creative Commons License CC-BY

35th Symposium on Theoretical Aspects of Computer Science (STACS 2018).

Editors: Rolf Niedermeier and Brigitte Vallée; Article No. 34; pp. 34:1–34:31



Leibniz International Proceedings in Informatics

LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Player  $B$  would be written as  $\exists\sigma_A. \forall\sigma_B. \text{assign}(A \mapsto \sigma_A; B \mapsto \sigma_B). \varphi_A$ . This makes the logic both expressive and easy to use (at first sight), at the expense of a very high complexity: SL model checking has non-elementary complexity, and satisfiability is undecidable [18, 15].

**Understanding SL.** Since it enjoys decidable model checking and high expressiveness, SL is the logic of choice for showing that some game problems are decidable (e.g. rational synthesis [12, 14, 10] or assume-admissible synthesis [5]). For instance, the existence of an admissible strategy for player  $A$  (i.e., a strategy that is strictly dominated by no other strategies [5]) is expressed as

$$\exists\sigma_A. \forall\sigma'_A. \left[ \begin{array}{l} \exists\sigma_B. \text{assign}(A \mapsto \sigma_A, B \mapsto \sigma_B). \varphi \wedge \text{assign}(A \mapsto \sigma'_A, B \mapsto \sigma_B). \neg\varphi \\ \forall\sigma'_B. \text{assign}(A \mapsto \sigma_A, B \mapsto \sigma'_B). \varphi \vee \text{assign}(A \mapsto \sigma'_A, B \mapsto \sigma'_B). \neg\varphi \end{array} \right].$$

However, it has been noticed in recent works that the nice expressiveness of SL comes with unexpected phenomena. One such phenomenon is induced by the separation of strategy quantification and strategy assignment: are the events between strategy quantifications and strategy assignments part of the *memory* of the strategy? While both options may make sense depending on the applications, only one of them makes model checking decidable [4].

A second phenomenon—which is the main focus of the present paper—concerns *strategy dependences* [18]: in a formula such as  $\forall\sigma_A. \exists\sigma_B. \xi$ , the existentially-quantified strategy  $\sigma_B$  may depend on *the whole* strategy  $\sigma_A$ ; in other terms, the action returned by strategy  $\sigma_B$  after some finite history  $\rho$  may depend on what strategy  $\sigma_A$  would play on any other history  $\rho'$ . Again, in some contexts, it may be desirable that the value of strategy  $\sigma_B$  after history  $\rho$  can be *computed* based solely on what has been observed along  $\rho$  (see Fig. 2 for an illustration). This approach was initiated in [18, 21], conjecturing that large fragments of SL (subsuming ATL\*) would have 2-EXPTIME model-checking algorithms with such limited dependences.

**Our contributions.** We follow this line of work by performing a more thorough exploration of strategy dependences in (a fragment of) SL. We mainly follow the framework of [21], based on a kind of Skolemization of the formula: for instance, a formula of the form  $(\forall x_i \exists y_i)_i. \xi$  is satisfied if there exists a *dependence map*  $\theta$  defining each existentially-quantified strategy  $y_j$  based on the universally-quantified strategies  $(x_i)_i$ . In order to recover the classical semantics of SL, it is only required that the strategy  $\theta((x_i)_i)(y_j)$  (i.e. the strategy assigned to the existentially-quantified variable  $y_j$  by  $\theta((x_i)_i)$ ) only depends on  $(x_i)_{i < j}$ .

Based on this definition, other constraints can be imposed on dependence maps, in order to refine the dependences of existentially-quantified strategies on universally-quantified ones. *Elementary dependences* [21] only allows existentially-quantified strategy  $y_j$  to depend on the values of  $(x_i)_{i < j}$  along the current history. This gives rise to two different semantics in general, but fragments of SL have been defined (SL[1G] in [17], SL[CG] and SL[DG] in [20]) on which the classic and elementary semantics would coincide.

The coincidence actually only holds for SL[1G]. As we explain in this paper, elementary dependences as defined and used in [17, 20] do not exactly capture the intuition that strategies should depend on the “behavior [of universal strategies] on the history of interest only” [20]: indeed, they only allow dependences on universally-quantified strategies *that appear earlier in the formula*, while we claim that the behaviour of all universally-quantified strategies should be considered. We address this discrepancy by introducing another kind of dependences, which we call *timeline dependences*, and which extend elementary dependences by allowing existentially-quantified strategies to additionally depend on *all* universally-quantified strategies along *strict prefixes* of the current history (as illustrated on Fig. 4).

We study and compare those three dependences (classic, elementary and timeline), showing that they correspond to three distinct semantics. Because the semantics based on dependence maps is defined in terms of the *existence* of a witness map, we show that the syntactic negation of a formula may not correspond to its semantic negation: there are cases where both a formula  $\varphi$  and its syntactic negation  $\neg\varphi$  fail to hold (i.e., none of them has a witness map). This phenomenon is already present, but had not been formally identified, in [18, 21]. The main contribution of the present paper is the definition of a large (and, in a sense, maximal) fragment of SL for which syntactic and semantic negations coincide under the timeline semantics. As an (important) side result, we show that model checking this fragment under the timeline semantics is  $\mathcal{L}$ -EXPTIME-complete.

## 2 Definitions

### 2.1 Concurrent game structures

Let AP be a set of atomic propositions,  $\mathcal{V}$  be a set of variables, and  $\text{Agt}$  be a set of agents. A *concurrent game structure* is a tuple  $\mathcal{G} = (\text{Act}, \mathcal{Q}, \Delta, \text{lab})$  where  $\text{Act}$  is a finite set of actions,  $\mathcal{Q}$  is a finite set of states,  $\Delta: \mathcal{Q} \times \text{Act}^{\text{Agt}} \rightarrow \mathcal{Q}$  is the transition function, and  $\text{lab}: \mathcal{Q} \rightarrow 2^{\text{AP}}$  is a labelling function. An element of  $\text{Act}^{\text{Agt}}$  will be called a *move vector*. For any  $q \in \mathcal{Q}$ , we let  $\text{succ}(q)$  be the set  $\{q' \in \mathcal{Q} \mid \exists m \in \text{Act}^{\text{Agt}}. q' = \Delta(q, m)\}$ . For the sake of simplicity, we assume in the sequel that  $\text{succ}(q) \neq \emptyset$  for any  $q \in \mathcal{Q}$ . A game  $\mathcal{G}$  is said *turn-based* whenever for every state  $q \in \mathcal{Q}$ , there is a player  $\text{own}(q) \in \text{Agt}$  (named the *owner* of  $q$ ) such that for any two move vectors  $m_1$  and  $m_2$  with  $m_1(\text{own}(q)) = m_2(\text{own}(q))$ , it holds  $\Delta(q, m_1) = \Delta(q, m_2)$ . Figure 1 displays an example of a (turn-based) game.

Fix a state  $q \in \mathcal{Q}$ . A *play* in  $\mathcal{G}$  from  $q$  is an infinite sequence  $\pi = (q_i)_{i \in \mathbb{N}}$  of states in  $\mathcal{Q}$  such that  $q_0 = q$  and  $q_i \in \text{succ}(q_{i-1})$  for all  $i > 0$ . We write  $\text{Play}_{\mathcal{G}}(q)$  for the set of plays in  $\mathcal{G}$  from  $q$ . In this and all similar notations, we might omit to mention  $\mathcal{G}$  when it is clear from the context, and  $q$  when we consider the union over all  $q \in \mathcal{Q}$ . A (strict) prefix of a play  $\pi$  is a finite sequence  $\rho = (q_i)_{0 \leq i \leq L}$ , for some  $L \in \mathbb{N}$ . We write  $\text{Pref}(\pi)$  for the set of strict prefixes of play  $\pi$ . Such finite prefixes are called *histories*, and we let  $\text{Hist}_{\mathcal{G}}(q) = \text{Pref}(\text{Play}_{\mathcal{G}}(q))$ . We extend the notion of strict prefixes and the notation  $\text{Pref}$  to histories in the natural way, requiring in particular that  $\rho \notin \text{Pref}(\rho)$ . A (finite) extension of a history  $\rho$  is any history  $\rho'$  such that  $\rho \in \text{Pref}(\rho')$ . Let  $\rho = (q_i)_{i \leq L}$  be a history. We define  $\text{first}(\rho) = q_0$  and  $\text{last}(\rho) = q_L$ . Let  $\rho' = (q'_j)_{j \leq L'}$  be a history from  $\text{last}(\rho)$ . The *concatenation* of  $\rho$  and  $\rho'$  is then defined as the path  $\rho \cdot \rho' = (q''_k)_{k \leq L+L'}$  such that  $q''_k = q_k$  when  $k \leq L$  and  $q''_k = q'_{k-L}$  when  $L \leq k \leq L+L'$  (notice that we required  $q'_0 = q_L$ ).

A *strategy* from  $q$  is a mapping  $\delta: \text{Hist}_{\mathcal{G}}(q) \rightarrow \text{Act}$ . We write  $\text{Strat}_{\mathcal{G}}(q)$  for the set of strategies in  $\mathcal{G}$  from  $q$ . Given a strategy  $\delta \in \text{Strat}(q)$  and a history  $\rho$  from  $q$ , the *translation*  $\delta_{\vec{\rho}}$  of  $\delta$  by  $\rho$  is the strategy  $\delta_{\vec{\rho}}$  from  $\text{last}(\rho)$  defined by  $\delta_{\vec{\rho}}(\rho') = \delta(\rho \cdot \rho')$  for any  $\rho' \in \text{Hist}(\text{last}(\rho))$ . A *valuation* from  $q$  is a partial function  $\chi: \mathcal{V} \cup \text{Agt} \rightarrow \text{Strat}(q)$ . As usual, for any partial function  $f$ , we write  $\text{dom}(f)$  for the domain of  $f$ .

Let  $q \in \mathcal{Q}$  and  $\chi$  be a valuation from  $q$ . If  $\text{Agt} \subseteq \text{dom}(\chi)$ , then  $\chi$  induces a unique play from  $q$ , called its *outcome*, and defined as  $\text{out}(q, \chi) = (q_i)_{i \in \mathbb{N}}$  such that  $q_0 = q$  and for every  $i \in \mathbb{N}$ , we have  $q_{i+1} = \Delta(q_i, m_i)$  with  $m_i(A) = \chi(A)((q_j)_{j \leq i})$  for every  $A \in \text{Agt}$ .

### 2.2 Strategy Logic with boolean goals

Strategy Logic (SL for short) was introduced in [7], and further extended and studied in [22, 18], as a rich logical formalism for expressing properties of games. SL manipulates

strategies as first-order elements, assigns them to players, and expresses LTL properties on the outcomes of the resulting strategic interactions. This results in a very expressive temporal logic, for which satisfiability is undecidable [22, 19] and model checking is TOWER-complete [18, 3]. In this paper, we focus on a restricted fragment of SL, called  $\text{SL}[\text{BG}]^b$  (where BG stands for *boolean goals* [18], and the symbol  $b$  indicates that we do not allow nesting of (closed) subformulas; we discuss this latter restriction below).

**Syntax.** Formulas in  $\text{SL}[\text{BG}]^b$  are built along the following grammar

$$\begin{aligned} \text{SL}[\text{BG}]^b \ni \varphi &::= \exists x. \varphi \mid \forall x. \varphi \mid \xi & \xi &::= \neg \xi \mid \xi \wedge \xi \mid \xi \vee \xi \mid \beta \\ \beta &::= \text{assign}(\sigma). \psi & \psi &::= \neg \psi \mid \psi \vee \psi \mid \psi \wedge \psi \mid \mathbf{X} \psi \mid \psi \mathbf{U} \psi \mid p \end{aligned}$$

where  $x$  ranges over  $\mathcal{V}$ ,  $\sigma$  ranges over the set  $\mathcal{V}^{\text{Agt}}$  of *full assignments*, and  $p$  ranges over AP. A *goal* is a formula of the form  $\beta$  in the grammar above; it expresses an LTL property  $\psi$  on the outcome of the mapping  $\sigma$ . Formulas in  $\text{SL}[\text{BG}]^b$  are thus made of an initial block of first-order quantifiers (selecting strategies for variables in  $\mathcal{V}$ ), followed by a boolean combination of such goals.

**Free variables.** With any subformula  $\zeta$  of some formula  $\varphi \in \text{SL}[\text{BG}]^b$ , we associate its set of *free agents and variables*, which we write  $\text{free}(\zeta)$ . It contains the agents and variables that have to be associated with a strategy in order to unequivocally evaluate  $\zeta$  (as will be seen from the definition of the semantics of  $\text{SL}[\text{BG}]^b$  below). The set  $\text{free}(\zeta)$  is defined inductively:

$$\begin{aligned} \text{free}(p) &= \emptyset \quad \text{for all } p \in \text{AP} & \text{free}(\mathbf{X} \psi) &= \text{Agt} \cup \text{free}(\psi) \\ \text{free}(\neg \alpha) &= \text{free}(\alpha) & \text{free}(\psi_1 \mathbf{U} \psi_2) &= \text{Agt} \cup \text{free}(\psi_1) \cup \text{free}(\psi_2) \\ \text{free}(\alpha_1 \vee \alpha_2) &= \text{free}(\alpha_1) \cup \text{free}(\alpha_2) & \text{free}(\exists x. \varphi) &= \text{free}(\varphi) \setminus \{x\} \\ \text{free}(\alpha_1 \wedge \alpha_2) &= \text{free}(\alpha_1) \cup \text{free}(\alpha_2) & \text{free}(\forall x. \varphi) &= \text{free}(\varphi) \setminus \{x\} \\ \text{free}(\text{assign}(\sigma). \varphi) &= (\text{free}(\varphi) \cup \sigma(\text{Agt} \cap \text{free}(\varphi))) \setminus \text{Agt} \end{aligned}$$

Subformula  $\zeta$  is said to be *closed* whenever  $\text{free}(\zeta) = \emptyset$ . We can now comment on our choice of considering the flat fragment of  $\text{SL}[\text{BG}]$ : the full fragment, as defined in [18], allows for nesting *closed*  $\text{SL}[\text{BG}]$  formulas in place of atomic propositions. The meaning of such nesting in our setting is ambiguous, because our semantics (in Sections 3 to 5) are defined in terms of the *existence of a witness*, which does not easily propagate in formulas. In particular, as we explain later in the paper, the semantics of the negation of a formula (there is a witness for  $\neg \varphi$ ) does not coincide with the negation of the semantics (there is no witness for  $\varphi$ ); thus substituting a subformula and substituting its negation may return different results.

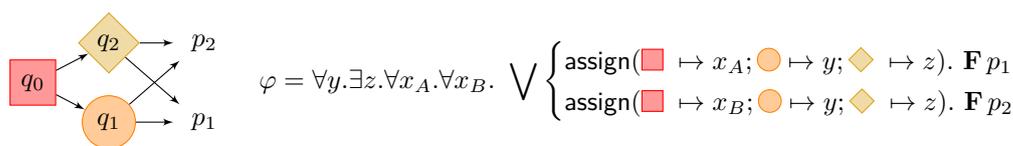
**Semantics.** Fix a state  $q \in Q$ , and a valuation  $\chi: \mathcal{V} \cup \text{Agt} \rightarrow \text{Strat}(q)$ . We inductively define the semantics of a subformula  $\alpha$  of a formula of  $\text{SL}[\text{BG}]^b$  at  $q$  under valuation  $\chi$ , requiring  $\text{free}(\alpha) \subseteq \text{dom}(\chi)$ . We omit the easy cases of boolean combinations and atomic propositions.

Given a mapping  $\sigma: \text{Agt} \rightarrow \mathcal{V}$ , the semantics of strategy assignments is defined as follows:

$$\mathcal{G}, q \models_{\chi} \text{assign}(\sigma). \psi \quad \Leftrightarrow \quad \mathcal{G}, q \models_{\chi[A \in \text{Agt} \mapsto \chi(\sigma(A))]} \psi.$$

Notice that, writing  $\chi' = \chi[A \in \text{Agt} \mapsto \chi(\sigma(A))]$ , we have  $\text{free}(\psi) \subseteq \text{dom}(\chi')$  if  $\text{free}(\alpha) \subseteq \text{dom}(\chi)$ , so that our inductive definition is sound.

We now consider path formulas  $\psi = \mathbf{X} \psi_1$  and  $\psi = \psi_1 \mathbf{U} \psi_2$ . Since  $\text{Agt} \subseteq \text{free}(\psi) \subseteq \text{dom}(\chi)$ , the valuation  $\chi$  induces a unique outcome  $\text{out}(q, \chi) = (q_i)_{i \in \mathbb{N}}$  from  $q$ . For  $n \in \mathbb{N}$ , we write  $\text{out}_n(q, \chi) = (q_i)_{i \leq n}$ , and define  $\chi_{\vec{n}}$  as the valuation obtained by shifting all



■ **Figure 1** A game and a SL[BG] formula.

the strategies in the image of  $\chi$  by  $\text{out}_n(q, \chi)$ . Under the same conditions, we also define  $q_{\vec{n}} = \text{last}(\text{out}_n(q, \chi))$ . We then set

$$\begin{aligned} \mathcal{G}, q \models_{\chi} \mathbf{X} \psi_1 &\Leftrightarrow \mathcal{G}, q_{\vec{1}} \models_{\chi_{\vec{1}}} \psi_1 \\ \mathcal{G}, q \models_{\chi} \psi_1 \mathbf{U} \psi_2 &\Leftrightarrow \exists k \in \mathbb{N}. \mathcal{G}, q_{\vec{k}} \models_{\chi_{\vec{k}}} \psi_2 \quad \text{and} \quad \forall 0 \leq j < k. \mathcal{G}, q_{\vec{j}} \models_{\chi_{\vec{j}}} \psi_1. \end{aligned}$$

In the sequel, we use classical shorthands, such as  $\top$  for  $p \vee \neg p$  (for any  $p \in \text{AP}$ ),  $\mathbf{F} \psi$  for  $\top \mathbf{U} \psi$  (*eventually*  $\psi$ ), and  $\mathbf{G} \psi$  for  $\neg \mathbf{F} \neg \psi$  (*always*  $\psi$ ). It remains to define the semantics of the strategy quantifiers. This is actually what this paper is all about. We provide here the original semantics, and discuss alternatives in the following sections:

$$\mathcal{G}, q \models_{\chi} \exists x. \varphi \Leftrightarrow \exists \delta \in \text{Strat}(q). \mathcal{G}, q \models_{\chi[x \mapsto \delta]} \varphi.$$

► **Example 1.** We consider the (turn-based) game  $\mathcal{G}$  is depicted on Fig. 1. We name the players after the shape of the state they control. The SL[BG] formula  $\varphi$  to the right of Fig. 1 has four quantified variables and two goals. We show that this formula evaluates to true at  $q_0$ : fix a strategy  $\delta_y$  (to be played by player  $\circ$ ); because  $\mathcal{G}$  is turn-based, we identify the actions of the owner of a state with the resulting target state, so that  $\delta_y(q_0 q_1)$  will be either  $p_1$  or  $p_2$ . We then define strategy  $\delta_z$  (to be played by  $\diamond$ ) as  $\delta_z(q_0 q_2) = \delta_y(q_0 q_1)$ . Then clearly, for any strategy assigned to player  $\square$ , one of the goals of formula  $\varphi$  holds true, so that  $\varphi$  itself evaluates to true.

**Subclasses of SL[BG].** Because of the high complexity and subtlety of reasoning with SL and SL[BG], several restrictions of SL[BG] have been considered in the literature [17, 20, 21], by adding further restrictions to boolean combinations in the grammar defining the syntax:

- SL[1G] restricts SL[BG] to a unique goal.  $\text{SL}[1G]^b$  is then defined from the grammar of  $\text{SL}[BG]^b$  by setting  $\xi ::= \beta$  in the grammar;
- the larger fragment SL[CG] allows for *conjunctions* of goals.  $\text{SL}[CG]^b$  corresponds to formulas defined with  $\xi ::= \xi \wedge \xi \mid \beta$ ;
- similarly, SL[DG] only allows *disjunctions* of goals, i.e.  $\xi ::= \xi \vee \xi \mid \beta$ ;
- finally, SL[AG] mixes conjunctions and disjunctions in a restricted way. Goals in  $\text{SL}[AG]^b$  can be combined using the following grammar:  $\xi ::= \beta \wedge \xi \mid \beta \vee \xi \mid \beta$ .

In the sequel, we write a generic  $\text{SL}[BG]^b$  formula  $\varphi$  as  $(Q_i x_i)_{1 \leq i \leq l}. \xi(\beta_j. \psi_j)_{j \leq n}$  where:

- $(Q_i x_i)_{1 \leq i \leq l}$  is a block of quantifications, with  $\{x_i \mid 1 \leq i \leq l\} \subseteq \mathcal{V}$  and  $Q_i \in \{\exists, \forall\}$ , for every  $1 \leq i \leq l$ ;
- $\xi(g_1, \dots, g_n)$  is a boolean combination of its arguments;
- for all  $1 \leq j \leq n$ ,  $\beta_j. \psi_j$  is a goal:  $\beta_j$  is a full assignment and  $\psi_j$  is an LTL formula.

### 3 Strategy dependences

We now follow the framework of [18, 21] and define the semantics of  $\text{SL}[BG]^b$  in terms of *dependence maps*. This approach provides a fine way of controlling how *existentially-quantified*

strategies depend on other strategies (in a quantifier block). Using dependence maps, we can limit such dependences.

**Dependence maps.** Consider an  $\text{SL}[\text{BG}]^b$  formula  $\varphi = (Q_i x_i)_{1 \leq i \leq l} \cdot \xi(\beta_j \cdot \varphi_j)_{j \leq n}$ , assuming w.l.o.g. that  $\{x_i \mid 1 \leq i \leq l\} = \mathcal{V}$ . We let  $\mathcal{V}^\forall = \{x_i \mid Q_i = \forall\} \subseteq \mathcal{V}$  be the set of universally-quantified variables of  $\varphi$ . A function  $\theta: \text{Strat}^{\mathcal{V}^\forall} \rightarrow \text{Strat}^{\mathcal{V}}$  is a  $\varphi$ -map (or *map* when  $\varphi$  is clear from the context) if  $\theta(w)(x_i)(\rho) = w(x_i)(\rho)$  for any  $w \in \text{Strat}^{\mathcal{V}^\forall}$ , any  $x_i \in \mathcal{V}^\forall$ , and any history  $\rho$ . In other words,  $\theta(w)$  extends  $w$  to  $\mathcal{V}$ . This general notion allows any existentially-quantified variable to depend on *all* universally-quantified ones (dependence on existentially-quantified variables is implicit: all existentially-quantified variables are assigned through a single map, hence they all depend on the others); we add further restrictions later on. Using maps, we may then define new semantics for  $\text{SL}[\text{BG}]^b$ : generally speaking, formula  $\varphi = (Q_i x_i)_{1 \leq i \leq l} \cdot \xi(\beta_j \cdot \varphi_j)_{j \leq n}$  holds true if there exists a  $\varphi$ -map  $\theta$  such that, for any  $w: \mathcal{V}^\forall \rightarrow \text{Strat}$ , the valuation  $\theta(w)$  makes  $\xi(\beta_j \cdot \varphi_j)_{j \leq n}$  hold true.

*Classic maps* are dependence maps in which the order of quantification is respected:

$$\forall w_1, w_2 \in \text{Strat}^{\mathcal{V}^\forall}. \forall x_i \in \mathcal{V} \setminus \mathcal{V}^\forall. \\ (\forall x_j \in \mathcal{V}^\forall \cap \{x_j \mid j < i\}. w_1(x_j) = w_2(x_j)) \Rightarrow (\theta(w_1)(x_i) = \theta(w_2)(x_i)). \quad (\text{C})$$

In words, if  $w_1$  and  $w_2$  coincide on  $\mathcal{V}^\forall \cap \{x_j \mid j < i\}$ , then  $\theta(w_1)$  and  $\theta(w_2)$  coincide on  $x_i$ .

*Elementary maps* [18, 17] have to satisfy a more restrictive condition: for those maps, the value of an existentially-quantified strategy at any history  $\rho$  may only depend on the value of earlier universally-quantified strategies *along*  $\rho$ . This may be written as:

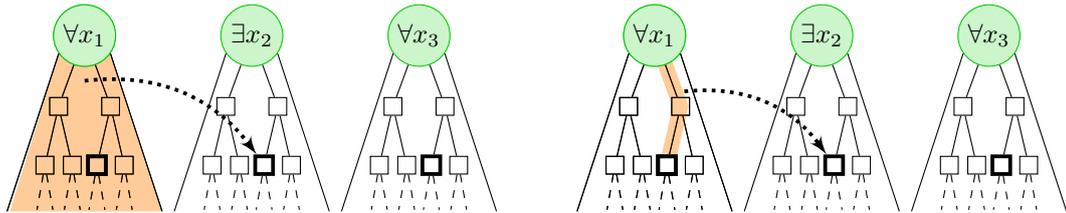
$$\forall w_1, w_2 \in \text{Strat}^{\mathcal{V}^\forall}. \forall x_i \in \mathcal{V} \setminus \mathcal{V}^\forall. \forall \rho \in \text{Hist}. \\ (\forall x_j \in \mathcal{V}^\forall \cap \{x_k \mid k < i\}. \forall \rho' \in \text{Pref}(\rho) \cup \{\rho\}. w_1(x_j)(\rho') = w_2(x_j)(\rho')) \Rightarrow \\ (\theta(w_1)(x_i)(\rho) = \theta(w_2)(x_i)(\rho)). \quad (\text{E})$$

In this case, for any history  $\rho$ , if two valuations  $w_1$  and  $w_2$  of the universally-quantified variables coincide on the variables quantified before  $x_i$  all along  $\rho$ , then  $\theta(w_1)(x_i)$  and  $\theta(w_2)(x_i)$  have to coincide at  $\rho$ .

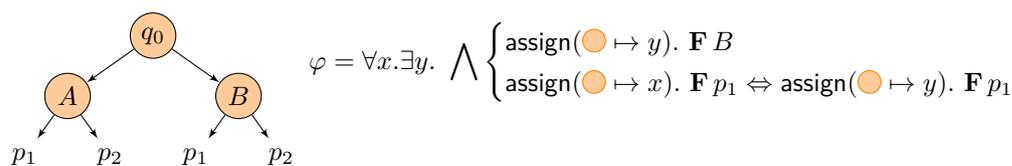
The difference between both kinds of dependences is illustrated on Fig. 2: for classic maps, the existentially-quantified strategy  $x_2$  may depend on the whole strategy  $x_1$ , while it may only depend on the value of  $x_1$  along the current history for elementary maps. Notice, as we prove in Appendix A, that a map satisfying (E) also satisfies (C).

**Satisfaction relations.** Pick a formula  $\varphi = (Q_i x_i)_{1 \leq i \leq l} \cdot \xi(\beta_j \cdot \varphi_j)_{j \leq n}$  in  $\text{SL}[\text{BG}]^b$ . We define:

$$\mathcal{G}, q \models^C \varphi \quad \text{iff} \quad \exists \theta \text{ satisfying (C)}. \forall w \in \text{Strat}^{\mathcal{V}^\forall}. \mathcal{G}, q \models_{\theta(w)} \xi(\beta_j \varphi_j)_{j \leq n}$$



■ **Figure 2** Classical (left) vs elementary (right) dependences for a formula  $\forall x_1. \exists x_2. \forall x_3. \xi$



■ **Figure 3** A game  $\mathcal{G}$  and an  $\text{SL}[\text{BG}]^b$  formula  $\varphi$  such that  $\mathcal{G}, q_0 \models^E \varphi$  and  $\mathcal{G}, q_0 \not\models^E \neg\varphi$ .

As explained above, this actually corresponds to the usual semantics of  $\text{SL}[\text{BG}]^b$  as given in Section 2 [18, Theorem 4.6]. When  $\mathcal{G}, q \models^C \varphi$ , a map  $\theta$  satisfying the conditions above is called a *C-witness* of  $\varphi$  for  $\mathcal{G}$  and  $q$ . Similarly, we define the *elementary semantics* [18] as:

$$\mathcal{G}, q \models^E \varphi \quad \text{iff} \quad \exists \theta \text{ satisfying (E)}. \forall w \in \text{Strat}^{\forall}. \mathcal{G}, q \models_{\theta(w)} \xi(\beta_j \varphi_j)_{j \leq n}$$

Again, when such a map exists, it is called an *E-witness*. Notice that since Property (E) implies Property (C), we have  $\mathcal{G}, q \models^E \varphi \Rightarrow \mathcal{G}, q \models^C \varphi$  for any  $\varphi \in \text{SL}[\text{BG}]^b$ . This corresponds to the intuition that it is harder to satisfy a  $\text{SL}[\text{BG}]^b$  formula when dependences are more restricted. The contrapositive statement then raises questions about the negation of formulas.

**The syntactic vs. semantic negations.** If  $\varphi = (Q_i x_i)_{1 \leq i \leq l} \xi(\beta_j \varphi_j)_{j \leq n}$  is an  $\text{SL}[\text{BG}]^b$  formula, its syntactic negation  $\neg\varphi$  is the formula  $(\overline{Q}_i x_i)_{i \leq l} (\neg\xi)(\beta_j \varphi_j)_{j \leq n}$ , where  $\overline{Q}_i = \exists$  if  $Q_i = \forall$  and  $\overline{Q}_i = \forall$  if  $Q_i = \exists$ . Looking at the definitions of  $\models^C$  and  $\models^E$ , it could be the case that e.g.  $\mathcal{G}, q \models^C \varphi$  and  $\mathcal{G}, q \models^C \neg\varphi$ : this only requires the existence of two adequate maps. However, since  $\models^C$  and  $\models$  coincide, and since  $\mathcal{G}, q \models \varphi \Leftrightarrow \mathcal{G}, q \not\models \neg\varphi$  in the usual semantics, we get  $\mathcal{G}, q \models^C \varphi \Leftrightarrow \mathcal{G}, q \not\models^C \neg\varphi$ . Also, since  $\mathcal{G}, q \models^E \varphi \Rightarrow \mathcal{G}, q \models^C \varphi$ , we also get  $\mathcal{G}, q \models^E \varphi \Rightarrow \mathcal{G}, q \not\models^E \neg\varphi$ . As we now show, the converse implication holds for  $\text{SL}[\text{1G}]^b$ , but may fail to hold for  $\text{SL}[\text{BG}]^b$ .

► **Proposition 1.** *There exist a (one-player) game  $\mathcal{G}$  with initial state  $q_0$  and a formula  $\varphi \in \text{SL}[\text{BG}]^b$  such that  $\mathcal{G}, q_0 \not\models^E \varphi$  and  $\mathcal{G}, q_0 \not\models^E \neg\varphi$ .*

**Proof.** Consider the formula and the one-player game of Fig. 3. We start by proving that  $\mathcal{G}, q_0 \not\models^E \varphi$ . For a contradiction, assume that a witness map  $\theta$  satisfying (E) exists, and pick any valuation  $w$  for the universal variable  $x$ . First, for the first goal in the conjunction to be fulfilled, the strategy assigned to  $y$  must play to  $B$  from  $q_0$ . We abbreviate this as  $\theta(w)(y)(q_0) = B$  in the sequel. Now, consider two valuations  $w_1$  and  $w_2$  such that  $w_1(x)(q_0) = w_2(x)(q_0) = A$  and  $w_1(x)(q_0 \cdot B) = w_2(x)(q_0 \cdot B)$ , but such that  $w_1(x)(q_0 \cdot A) = p_1$  and  $w_2(x)(q_0 \cdot A) = p_2$ . In order to fulfill the second goal under both valuations  $w_1$  and  $w_2$ , we must have  $\theta(w_1)(y)(q_0 \cdot B) = p_1$  and  $\theta(w_2)(y)(q_0 \cdot B) = p_2$ . But this violates Property (E): since  $w_1(x)$  and  $w_2(x)$  coincide on  $q_0$  and on  $q_0 \cdot B$ , we must have  $\theta(w_1)(y)(q_0 \cdot B) = \theta(w_2)(y)(q_0 \cdot B)$ .

We now prove that  $\mathcal{G}, q_0 \not\models^E \neg\varphi$ . Indeed, following the previous discussion, we easily get that  $\mathcal{G}, q_0 \models^C \varphi$ , by letting  $\theta(w)(y)(q_0) = B$  and  $\theta(w)(y)(q_0 \cdot B) = w(x)(q_0 \cdot A)$  if  $w(x)(q_0) = A$ , and  $\theta(w)(y)(q_0 \cdot B) = w(x)(q_0 \cdot B)$  if  $w(x)(q_0) = B$ . As explained above, this entails  $\mathcal{G}, q_0 \not\models^C \neg\varphi$ , and  $\mathcal{G}, q_0 \not\models^E \neg\varphi$ . ◀

► **Proposition 2.** *For any game  $\mathcal{G}$  with initial state  $q_0$ , and any formula  $\varphi \in \text{SL}[\text{1G}]^b$ , it holds  $\mathcal{G}, q_0 \models^E \varphi \Leftrightarrow \mathcal{G}, q_0 \not\models^E \neg\varphi$ .*

**Sketch of proof.** This result follows from [18, Corollary 4.21], which states that  $\models^C$  and  $\models^E$  coincide on  $\text{SL}[\text{1G}]$ . Because it is central in our approach, we sketch a direct proof here (with

a full proof in Appendix A.1), using similar ingredients: it consists in encoding the problem whether  $\mathcal{G}, q_0 \models^E \varphi$  into a two-player turn-based game with a parity-winning objective.

The construction is as follows: the interaction between existential and universal quantifications of the formula is integrated into the game structure, replacing each state of  $\mathcal{G}$  with a tree-shaped subgame where Player  $P_{\exists}$  selects existentially-quantified actions and Player  $P_{\forall}$  selects universally-quantified ones. The unique goal of the formula is then incorporated into the game via a deterministic parity automaton, yielding a two-player turn-based parity game. We then show that  $\mathcal{G}, q_0 \models^E \varphi$  if, and only if, Player  $P_{\exists}$  has a winning strategy in the resulting turn-based parity game, while  $\mathcal{G}, q_0 \models^E \neg\varphi$  if, and only if, Player  $P_{\forall}$  has a winning strategy. Those equivalences hold for the elementary semantics because memoryless strategies are sufficient in parity games. Proposition 2 then follows by determinacy of those games [11, 23]. ◀

Note that the construction of the parity game gives an effective algorithm for the model-checking problem of  $\text{SL}[1\text{G}]^b$ , which runs in time doubly-exponential in the size of the formula, and polynomial in the size of the game structure; we recover the result of [18] for that problem.

**Comparison of  $\models^C$  and  $\models^E$ .** A consequence of Prop. 2 is that  $\models^C$  and  $\models^E$  coincide on  $\text{SL}[1\text{G}]^b$  (Corollary 4.21 of [18]). However, when considering larger fragments, the satisfaction relations are distinct (see the proof of Prop. 1 for a candidate formula in  $\text{SL}[\text{CG}]^b$ ):

► **Proposition 3.** *The relations  $\models^C$  and  $\models^E$  differ on  $\text{SL}[\text{CG}]^b$ , as well as on  $\text{SL}[\text{DG}]^b$ .*

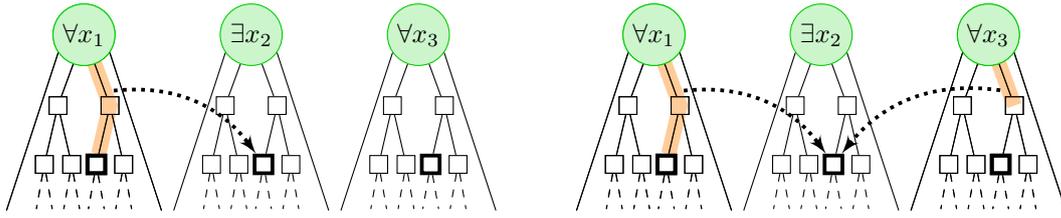
► **Remark.** Proposition 3 contradicts the claim in [20] that  $\models^E$  and  $\models^C$  coincide on  $\text{SL}[\text{CG}]$  (and  $\text{SL}[\text{DG}]$ ). Indeed, in [20], the satisfaction relation for  $\text{SL}[\text{DG}]$  and  $\text{SL}[\text{CG}]$  is encoded into a two-player game in pretty much the same way as we did in the proof of Prop. 2. While this indeed rules out dependences outside the current history, it also gives information to Player  $P_{\exists}$  about the values (over prefixes of the current history) of strategies that are universally-quantified later in the quantification block. This proof technique works with  $\text{SL}[1\text{G}]^b$  because the single goal can be encoded as a parity objective, for which memoryless strategies exist, so that the extra information is not crucial. In the next section, we investigate the role of this extra information for larger fragments of  $\text{SL}[\text{BG}]^b$ .

## 4 Timeline dependences

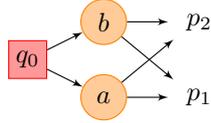
Following the discussion above, we introduce a new type of dependences between strategies (which we call *timeline dependences*). They allow strategies to also observe (and depend on) *all* other universally-quantified strategies on the strict prefix of the current history. For instance, for a block of quantifiers  $\forall x_1. \exists x_2. \forall x_3$ , the value of  $x_2$  after history  $\rho$  may depend on the value of  $x_1$  on  $\rho$  and its prefixes (as for elementary maps), but also on the value of  $x_3$  on the (strict) prefixes of  $\rho$ . Such dependences are depicted on Fig. 4. We believe that such dependences are relevant in many situations, especially for reactive synthesis, since in this framework strategies really base their decisions on what they could observe along the current history.

Formally, a map  $\theta$  is a *timeline map* if it satisfies the following condition:

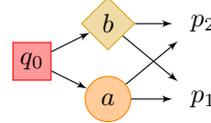
$$\forall w_1, w_2 \in \text{Strat}^{\mathcal{V}^{\forall}}. \forall x_i \in \mathcal{V} \setminus \mathcal{V}^{\forall}. \forall \rho \in \text{Hist}. \\ \left( \begin{array}{l} \forall x_j \in \mathcal{V}^{\forall} \cap \{x_k \mid k < i\}. \forall \rho' \in \text{Pref}(\rho) \cup \{\rho\}. w_1(x_j)(\rho) = w_2(x_j)(\rho) \\ \wedge \forall x_j \in \mathcal{V}^{\forall}. \forall \rho' \in \text{Pref}(\rho). w_1(x_j)(\rho) = w_2(x_j)(\rho) \end{array} \right) \Rightarrow \\ (\theta(w_1)(x_i)(\rho) = \theta(w_2)(x_i)(\rho)). \quad (\text{T})$$



■ **Figure 4** Elementary (left) vs timeline (right) dependences for a formula  $\forall x_1. \exists x_2. \forall x_3. \xi$



■ **Figure 5**  $\models^E$  and  $\models^T$  differ on  $SL[CG]^b$



■ **Figure 6**  $\models^E$  and  $\models^T$  differ on  $SL[DG]^b$

Using those maps, we introduce the *timeline semantics* of  $SL[BG]^b$ :

$$\mathcal{G}, q \models^T \varphi \text{ iff } \exists \theta \text{ satisfying (T). } \forall w \in \text{Strat}^{\forall}. \mathcal{G}, q \models_{\theta(w)} \xi(\beta_j \varphi_j)_{j \leq n}$$

Such a map, if any, is called a T-witness of  $\varphi$  for  $\mathcal{G}$  and  $q$ . As in the previous section, it is easily seen that Property (E) implies Property (T), so that an E-witness is also a T-witness, and  $\mathcal{G}, q \models^E \varphi \Rightarrow \mathcal{G}, q \models^T \varphi$  for any formula  $\varphi \in SL[BG]^b$ .

► **Example 2.** Consider again the game of Fig 1 in Section 2. We have seen that  $\mathcal{G}, q_0 \models^C \varphi$  in Section 2, and that  $\mathcal{G}, q_0 \not\models^E \varphi$  in the proof of Prop. 3. With timeline dependences, we have  $\mathcal{G}, q_0 \models^T \varphi$ . Indeed, now  $\theta(w)(z)(q_0 \cdot q_2)$  may depend on  $w(x_A)(q_0)$  and  $w(x_B)(q_0)$ : we could then have e.g.  $\theta(w)(z)(q_0 \cdot q_2) = p_1$  when  $w(x_A)(q_0) = q_2$ , and  $\theta(w)(z)(q_0 \cdot q_2) = p_2$  when  $w(x_A)(q_0) = q_1$ . It is easily checked that this map is a T-witness of  $\varphi$  for  $q_0$ .

**Comparison of  $\models^E$  and  $\models^T$ .** As explained at the end of Section 3, the proof of Prop. 2 actually shows the following result:

► **Proposition 4.** For any game  $\mathcal{G}$  with initial state  $q_0$ , and any formula  $\varphi \in SL[1G]^b$ , it holds  $\mathcal{G}, q_0 \models^E \varphi \Leftrightarrow \mathcal{G}, q_0 \models^T \varphi$ .

We now prove that this does not extend to  $SL[CG]^b$  and  $SL[DG]^b$ :

► **Proposition 5.** The relations  $\models^E$  and  $\models^T$  differ on  $SL[CG]^b$ , as well as on  $SL[DG]^b$ .

**Proof.** For  $SL[CG]^b$ , we consider the game structure of Fig. 5, and formula

$$\varphi_C = \exists y. \forall x_A. \exists x_B. \bigwedge \begin{cases} \text{assign}(\text{orange} \mapsto y; \text{red} \mapsto x_A). \mathbf{F} p_1 \\ \text{assign}(\text{orange} \mapsto y; \text{red} \mapsto x_B). \mathbf{F} p_2 \end{cases}$$

We first notice that  $\mathcal{G}, q_0 \not\models^E \varphi_C$ : indeed, in order to satisfy the first goal under any choice of  $x_A$ , the strategy for  $y$  has to point to  $p_1$  from both  $a$  and  $b$ . But then no choice of  $x_B$  will make the second goal true.

On the other hand, considering the timeline semantics, strategy  $y$  after  $q_0 \cdot a$  and  $q_0 \cdot b$  may depend on the choice of  $x_A$  in  $q_0$ . When  $w(x_A)(q_0) = a$ , we let  $\theta(w)(y)(q_0 \cdot a) = p_1$  and  $\theta(w)(y)(q_0 \cdot b) = p_2$  and  $\theta(w)(x_B)(q_0) = b$ , which makes both goals hold true. Conversely, if  $w(x_A)(q_0) = b$ , then we let  $\theta(w)(y)(q_0 \cdot b) = p_1$  and  $\theta(w)(y)(q_0 \cdot a) = p_2$  and  $\theta(w)(x_B)(q_0) = a$ .

For  $\text{SL}[\text{DG}]^b$ , we consider the game of Fig. 6, and easily prove that formula  $\varphi_D$  below has a T-witness but no E-witness:

$$\varphi_D = \exists y. \forall x_A. \forall x_B. \forall z. \bigvee \left\{ \begin{array}{l} \text{assign}(\color{orange}{\circ} \mapsto y; \color{red}{\square} \mapsto x_A; \color{yellow}{\diamond} \mapsto z). \mathbf{F} p_1 \\ \text{assign}(\color{orange}{\circ} \mapsto y; \color{red}{\square} \mapsto x_B; \color{yellow}{\diamond} \mapsto z). \mathbf{F} p_2 \end{array} \right. \blacktriangleleft$$

**The syntactic vs. semantic negations.** While both semantics differ, we now prove that the situation w.r.t. the syntactic vs. semantic negations is similar. First, following Prop. 4 and 2, the two negations coincide on  $\text{SL}[\text{1G}]^b$  under the timeline semantics. Moreover:

► **Proposition 6.** *For any formula  $\varphi$  in  $\text{SL}[\text{BG}]^b$ , for any game  $\mathcal{G}$  and any state  $q_0$ , we have  $\mathcal{G}, q_0 \models^T \varphi \Rightarrow \mathcal{G}, q_0 \not\models^T \neg\varphi$ .*

**Sketch of proof.** Write  $\varphi = (Q_i x_i)_{1 \leq i \leq l} \xi(\beta_j \varphi_j)_{j \leq n}$ . For a contradiction, assume that there exist two maps  $\theta$  and  $\bar{\theta}$  witnessing  $\mathcal{G}, q_0 \models^T \varphi$  and  $\mathcal{G}, q_0 \models^T \neg\varphi$ , respectively. Then for any strategy valuations  $w$  and  $\bar{w}$  for  $\mathcal{V}^\forall$  and  $\mathcal{V}^\exists$ , we have that  $\mathcal{G}, q_0 \models_{\theta(w)} \xi(\beta_j \varphi_j)_j$  and  $\mathcal{G}, q_0 \models_{\bar{\theta}(\bar{w})} \neg\xi(\beta_j \varphi_j)_j$ . We can then inductively (on histories and on the sequence of quantified variables) build a strategy valuation  $\chi$  on  $\mathcal{V}$  such that  $\theta(\chi|_{\mathcal{V}^\forall}) = \bar{\theta}(\chi|_{\mathcal{V}^\exists}) = \chi$ . Then under valuation  $\chi$ , both  $\xi(\beta_j \varphi_j)_j$  and  $\neg\xi(\beta_j \varphi_j)_j$  hold in  $q_0$ , which is impossible. ◀

► **Proposition 7.** *There exists a formula  $\varphi \in \text{SL}[\text{BG}]^b$ , a (turn-based) game  $\mathcal{G}$  and a state  $q_0$  such that  $\mathcal{G}, q_0 \not\models^T \varphi$  and  $\mathcal{G}, q_0 \not\models^T \neg\varphi$ .*

## 5 The fragment $\text{SL}[\text{EG}]^b$

In this section, we focus on the timeline semantics  $\models^T$ . We exhibit a fragment<sup>1</sup>  $\text{SL}[\text{EG}]^b$  of  $\text{SL}[\text{BG}]^b$ , containing  $\text{SL}[\text{CG}]^b$  and  $\text{SL}[\text{DG}]^b$ , for which the syntactic and semantic negations coincide, and for which we prove model-checking is in 2-EXPTIME:

► **Theorem 8.** *For any  $\varphi \in \text{SL}[\text{EG}]^b$  and any state  $q_0$ , it holds:  $\mathcal{G}, q_0 \models^T \varphi \Leftrightarrow \mathcal{G}, q_0 \not\models^T \neg\varphi$ . Moreover, model checking  $\text{SL}[\text{EG}]^b$  for the timeline semantics is 2-EXPTIME-complete.*

### 5.1 Semi-stable sets.

For  $n \in \mathbb{N}$ , we let  $\{0, 1\}^n$  be the set of mappings from  $[1, n]$  to  $\{0, 1\}$ . We write  $\mathbf{0}^n$  (or  $\mathbf{0}$  if the size  $n$  is clear) for the function that maps all integers in  $[1, n]$  to 0, and  $\mathbf{1}^n$  (or  $\mathbf{1}$ ) for the function that maps  $[1, n]$  to 1. For  $f, g \in \{0, 1\}^n$ , we define:

$$\bar{f}: i \mapsto 1 - f(i) \quad f \wedge g: i \mapsto \min\{f(i), g(i)\} \quad f \vee g: i \mapsto \max\{f(i), g(i)\}.$$

We then introduce the notion of semi-stable sets, on which the definition of  $\text{SL}[\text{EG}]^b$  relies: a set  $F^n \subseteq \{0, 1\}^n$  is *semi-stable* if for any  $f$  and  $g$  in  $F^n$ , it holds that

$$\forall s \in \{0, 1\}^n. \quad (f \wedge s) \vee (g \wedge \bar{s}) \in F^n \text{ or } (g \wedge s) \vee (f \wedge \bar{s}) \in F^n.$$

► **Example 3.** *Obviously, the set  $\{0, 1\}^n$  is semi-stable, as well as the empty set. For  $n = 2$ , the set  $\{(0, 1), (1, 0)\}$  is easily seen not to be semi-stable: taking  $f = (0, 1)$  and  $g = (1, 0)$  with  $s = (1, 0)$ , we get  $(f \wedge s) \vee (g \wedge \bar{s}) = (0, 0)$  and  $(g \wedge s) \vee (f \wedge \bar{s}) = (1, 1)$ . Similarly,  $\{(0, 0), (1, 1)\}$  is not semi-stable. Any other subset of  $\{0, 1\}^2$  is semi-stable.*

<sup>1</sup> We name our fragment  $\text{SL}[\text{EG}]^b$  as it comes as a natural continuation after fragments  $\text{SL}[\text{AG}]^b$  [21],  $\text{SL}[\text{BG}]^b$  [18], and  $\text{SL}[\text{CG}]^b$  and  $\text{SL}[\text{DG}]^b$  [20].

We then define  $\text{SL}[\text{EG}]^b$  as follows:

$$\begin{aligned} \text{SL}[\text{EG}]^b \ni \varphi &::= \forall x. \varphi \mid \exists x. \varphi \mid \xi & \xi &::= F^n((\beta_i)_{1 \leq i \leq n}) \\ \beta &::= \text{assign}(\sigma). \psi & \psi &::= \neg \psi \mid \psi \vee \psi \mid \mathbf{X} \psi \mid \psi \mathbf{U} \psi \mid p \end{aligned}$$

where  $F^n$  ranges over semi-stable subsets of  $\{0, 1\}^n$ , for all  $n \in \mathbb{N}$ . The semantics of the operator  $F^n$  is defined as

$$\mathcal{G}, q \models_{\chi} F^n((\beta_i)_{i \leq n}) \iff \exists f \in F^n. \forall 1 \leq i \leq n. (f(i) = 1 \text{ iff } \mathcal{G}, q \models_{\chi} \beta_i).$$

Notice that if  $F^n$  would range over all subsets of  $\{0, 1\}^n$ , then this definition would exactly correspond to  $\text{SL}[\text{BG}]^b$ . Similarly, the case where  $F^n = \{\mathbf{1}^n\}$  corresponds to  $\text{SL}[\text{CG}]^b$ , while  $F^n = \{0, 1\}^n \setminus \{\mathbf{0}^n\}$  gives rise to  $\text{SL}[\text{DG}]^b$ .

► **Example 4.** Consider the following formula, expressing the existence of a Nash equilibrium for two players with respective LTL objectives  $\psi_1$  and  $\psi_2$ :

$$\exists x_1. \exists x_2. \forall y_1. \forall y_2. \bigwedge \left\{ \begin{array}{l} (\text{assign}(A_1 \mapsto y_1; A_2 \mapsto x_2). \psi_1) \Rightarrow (\text{assign}(A_1 \mapsto x_1; A_2 \mapsto x_2). \psi_1) \\ (\text{assign}(A_1 \mapsto x_1; A_2 \mapsto y_2). \psi_2) \Rightarrow (\text{assign}(A_1 \mapsto x_1; A_2 \mapsto x_2). \psi_2) \end{array} \right.$$

This formula has four goals, and it corresponds to the set

$$F^4 = \{(a, b, c, d) \in \{0, 1\}^4 \mid a \leq b \text{ and } c \leq d\}$$

Taking  $f = (1, 1, 0, 0)$  and  $g = (0, 0, 1, 1)$ , with  $s = (1, 0, 1, 0)$  we have  $(f \wedge s) \vee (g \wedge \bar{s}) = (1, 0, 0, 1)$  and  $(g \wedge s) \vee (f \wedge \bar{s}) = (0, 1, 1, 0)$ , none of which is in  $F^4$ . Hence our formula is not (syntactically) in  $\text{SL}[\text{EG}]^b$ .

The definition of  $\text{SL}[\text{EG}]$  may look artificial. The main reason why we work with  $\text{SL}[\text{EG}]$  is that it is maximal for the first claim of Theorem 8 (see Prop. 11). But as the next result shows, it is actually a large fragment encompassing  $\text{SL}[\text{AG}]$  (hence also  $\text{SL}[\text{CG}]$  and  $\text{SL}[\text{DG}]$ ):

► **Proposition 9.**  $\text{SL}[\text{EG}]^b$  contains  $\text{SL}[\text{AG}]^b$ . The inclusion is strict (syntactically).

## 5.2 Defining quasi-orders from semi-stable sets.

For  $F^n \subseteq \{0, 1\}^n$ , we write  $\overline{F^n}$  for the complement of  $F^n$ . Fix such a set  $F^n$ , and pick  $s \in \{0, 1\}^n$ . For any  $h \in \{0, 1\}^n$ , we define

$$\begin{aligned} \mathbb{F}^n(h, s) &= \{h' \in \{0, 1\}^n \mid (h \wedge s) \vee (h' \wedge \bar{s}) \in F^n\} \\ \overline{\mathbb{F}^n}(h, s) &= \{h' \in \{0, 1\}^n \mid (h \wedge s) \vee (h' \wedge \bar{s}) \in \overline{F^n}\} \end{aligned}$$

Trivially  $\mathbb{F}^n(h, s) \cap \overline{\mathbb{F}^n}(h, s) = \emptyset$  and  $\mathbb{F}^n(h, s) \cup \overline{\mathbb{F}^n}(h, s) = \{0, 1\}^n$ . If we assume  $F^n$  to be semi-stable, then the family  $(\mathbb{F}^n(h, s))_{h \in \{0, 1\}^n}$  enjoys the following property:

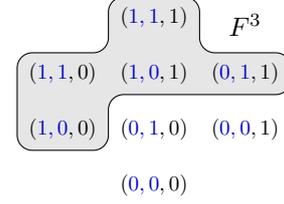
► **Lemma 10.** Fix a semi-stable set  $F^n$  and  $s \in \{0, 1\}^n$ . For any  $h_1, h_2 \in \{0, 1\}^n$ , either  $\mathbb{F}^n(h_1, s) \subseteq \mathbb{F}^n(h_2, s)$  or  $\mathbb{F}^n(h_2, s) \subseteq \mathbb{F}^n(h_1, s)$ .

Given a semi-stable set  $F^n$  and  $s \in \{0, 1\}^n$ , we can use the inclusion relation of Lemma 10 to define a relation  $\preceq_s^{F^n}$  (written  $\preceq_s$  when  $F^n$  is clear) over the elements of  $\{0, 1\}^n$ . It is defined as follows:  $h_1 \preceq_s h_2$  if, and only if,  $\mathbb{F}^n(h_1, s) \subseteq \mathbb{F}^n(h_2, s)$ .

This relation is a quasi-order: its reflexivity and transitivity both follow from the reflexivity and transitivity of the inclusion relation  $\subseteq$ . By Lemma 10, this quasi-order is total. Intuitively,  $\preceq_s$  orders the elements of  $\{0, 1\}^n$  based on how “easy” it is to complete

their restriction to  $s$  so that the completion belongs to  $F^n$ . In particular, only the indices on which  $s$  take value 1 are used to check whether  $h_1 \preceq_s h_2$ : given  $h_1, h_2 \in \{0, 1\}^n$  such that  $(h_1 \wedge s) = (h_2 \wedge s)$ , we have  $\mathbb{F}(h_1, s) = \mathbb{F}(h_2, s)$ , and  $h_1 \equiv_s h_2$ .

► **Example 5.** Consider the set  $F^3 = \{(1, 0, 0), (1, 1, 0), (1, 0, 1), (0, 1, 1), (1, 1, 1)\}$  represented on the figure opposite, and which can be shown to be semi-stable. Fix  $s = (1, 1, 0)$ . Then  $\mathbb{F}^3((0, 1, \star), s) = \{0, 1\}^2 \times \{1\}$ , while  $\mathbb{F}^3((1, 1, \star), s) = \mathbb{F}^3((1, 0, \star), s) = \{0, 1\}^3$  and  $\mathbb{F}^3((0, 0, \star), s) = \emptyset$ . It follows that  $(0, 0, \star) \preceq_s (0, 1, \star) \preceq_s (1, 0, \star) \equiv_s (1, 1, \star)$ .



### 5.3 Sketch of proof of Theorem 8

The approach we used in Prop 2 does not extend in general to formulas with several goals. Consider for instance formula  $(Q_i x_i)_{i \leq l} (\beta_1 \cdot \psi_1 \Leftrightarrow \beta_2 \cdot \psi_2)$ : if at some points the two goals give rise to two different outcomes (hence to two different subgames), the winning objectives in one subgame depends on what is achieved in the other subgame.

$\text{SL}[\text{EG}]^b$  has been designed to prevent such situations: when two (or more) outcomes are available at a given position, each *subgame* can be assigned an independent winning objective. This objective can be obtained from the quasi-orders  $\preceq_s$  associated with the  $\text{SL}[\text{EG}]^b$  formula being considered. Consider again Example 5: associating the set  $F^3$  with three goals  $\beta_1$ ,  $\beta_2$  and  $\beta_3$ , we get a formula in  $\text{SL}[\text{EG}]^b$ . Assume that the moves selected by the players give rise to the same transition for  $\beta_1$  and  $\beta_2$ , and to a different transition for  $\beta_3$ ; then in the subgame reached when following the transition of  $\beta_1$  and  $\beta_2$  (hence with  $s = (1, 1, 0)$ ), the optimal way of fulfilling goals  $\beta_1$  and  $\beta_2$  is given by  $(0, 0, \star) \preceq_s (0, 1, \star) \preceq_s (1, 0, \star) \equiv_s (1, 1, \star)$ , independently of what may happen in the subgame reached by following the transition given by  $\beta_3$ .

We exploit this idea in our proof: first, in order to keep track of the truth values of the LTL formulas  $\psi_i$  of each goal, we define a family of parity automata, one for each subset of goals of the formula under scrutiny. A subgame, as considered above, is characterized by a state  $q$  of the original concurrent game, a state  $d_p$  of each of the parity automata, and a vector  $s \in \{0, 1\}^n$  defining which goals are still active. For each subgame, we can compute, by induction on  $s$ , the optimal set of goals that can be fulfilled from that configuration. The optimal strategies of both players in each subgame can be used to define (partial) optimal timeline dependence maps. We can then combine these partial maps together to get optimal dependence maps  $\theta$  and  $\bar{\theta}$ ; using similar arguments as for the proof of Prop. 6, we get a valuation  $\chi$  such that  $\theta(\chi|_{\forall}) = \chi = \bar{\theta}(\chi|_{\exists})$ , from which we deduce that exactly one of  $\varphi$  and  $\neg\varphi$  holds.

### 5.4 Maximality of $\text{SL}[\text{EG}]^b$

Finally, we prove that  $\text{SL}[\text{EG}]^b$  is, in a sense, maximal for the first property of Theorem 8:

► **Proposition 11.** For any non-semi-stable boolean set  $F^n \subseteq \{0, 1\}^n$ , there exists a  $\text{SL}[\text{BG}]^b$  formula  $\varphi$  built on  $F^n$ , a game  $\mathcal{G}$  and a state  $q_0$  such that  $\mathcal{G}, q_0 \not\models^T \neg\varphi$  and  $\mathcal{G}, q_0 \not\models^T \varphi$ .

Whether  $\text{SL}[\text{EG}]^b$  is also maximal for having a 2-EXPTIME model-checking algorithm remains open. Actually, we do not know if  $\text{SL}[\text{BG}]^b$  model checking is decidable under the timeline semantics. These questions will be part of our future works on this topic.

---

**References**

---

- 1 Thomas Ågotnes, Valentin Goranko, and Wojciech Jamroga. Alternating-time temporal logics with irrevocable strategies. In Dov Samet, editor, *Proceedings of the 11th Conference on Theoretical Aspects of Rationality and Knowledge (TARK'07)*, pages 15–24, June 2007.
- 2 Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49(5):672–713, September 2002. doi:10.1145/585265.585270.
- 3 Patricia Bouyer, Patrick Gardy, and Nicolas Markey. Weighted strategy logic with boolean goals over one-counter games. In Prahladh Harsha and G. Ramalingam, editors, *Proceedings of the 35th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'15)*, volume 45 of *Leibniz International Proceedings in Informatics*, pages 69–83. Leibniz-Zentrum für Informatik, December 2015. doi:10.4230/LIPIcs.FSTTCS.2015.69.
- 4 Patricia Bouyer, Patrick Gardy, and Nicolas Markey. On the semantics of strategy logic. *Information Processing Letters*, 116(2):75–79, February 2016. doi:10.1016/j.ipl.2015.10.004.
- 5 Romain Brenguier, Jean-François Raskin, and Ocan Sankur. Assume-admissible synthesis. *Acta Informatica*, 54(1):41–83, February 2017. doi:10.1007/s00236-016-0273-2.
- 6 Thomas Brihaye, Arnaud Da Costa, François Laroussinie, and Nicolas Markey. ATL with strategy contexts and bounded memory. In Sergei N. Artemov and Anil Nerode, editors, *Proceedings of the International Symposium Logical Foundations of Computer Science (LFCS'09)*, volume 5407 of *Lecture Notes in Computer Science*, pages 92–106. Springer-Verlag, January 2009. doi:10.1007/978-3-540-92687-0\_7.
- 7 Krishnendu Chatterjee, Thomas A. Henzinger, and Nir Piterman. Strategy logic. In Luís Caires and Vasco T. Vasconcelos, editors, *Proceedings of the 18th International Conference on Concurrency Theory (CONCUR'07)*, volume 4703 of *Lecture Notes in Computer Science*, pages 59–73. Springer-Verlag, September 2007. doi:10.1007/978-3-540-74407-8\_5.
- 8 Edmund M. Clarke and E. Allen Emerson. Design and synthesis of synchronization skeletons using branching-time temporal logic. In Dexter C. Kozen, editor, *Proceedings of the 3rd Workshop on Logics of Programs (LOP'81)*, volume 131 of *Lecture Notes in Computer Science*, pages 52–71. Springer-Verlag, 1982. doi:10.1007/BFb0025774.
- 9 Edmund M. Clarke, Orna Grumberg, and Doron A. Peled. *Model checking*. MIT Press, 2000.
- 10 Rodica Condurache, Emmanuel Filiot, Raffaella Gentilini, and Jean-François Raskin. The complexity of rational synthesis. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *Proceedings of the 43rd International Colloquium on Automata, Languages and Programming (ICALP'16) – Part II*, volume 55 of *Leibniz International Proceedings in Informatics*, pages 121:1–121:15. Leibniz-Zentrum für Informatik, July 2016. doi:10.4230/LIPIcs.ICALP.2016.121.
- 11 E. Allen Emerson and Charanjit S. Jutla. Tree automata, mu-calculus and determinacy. In *Proceedings of the 32nd Annual Symposium on Foundations of Computer Science (FOCS'91)*, pages 368–377. IEEE Comp. Soc. Press, October 1991. doi:10.1109/SFCS.1991.185392.
- 12 Dana Fisman, Orna Kupferman, and Yoad Lustig. Rational synthesis. In Javier Esparza and Rupak Majumdar, editors, *Proceedings of the 16th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'10)*, volume 6015 of *Lecture Notes in Computer Science*, pages 190–204. Springer-Verlag, March 2010. doi:10.1007/978-3-642-12002-2\_16.
- 13 Valentin Goranko and Govert van Drimmelen. Complete axiomatization and decidability of alternating-time temporal logic. *Theoretical Computer Science*, 353(1-3):93–117, March 2006.

- 14 Orna Kupferman, Giuseppe Perelli, and Moshe Y. Vardi. Synthesis with rational environments. *Annals of Mathematics and Artificial Intelligence*, 78(1):3–20, September 2016. doi:10.1007/s10472-016-9508-8.
- 15 François Laroussinie and Nicolas Markey. Augmenting ATL with strategy contexts. *Information and Computation*, 245:98–123, December 2015. doi:10.1016/j.ic.2014.12.020.
- 16 François Laroussinie, Nicolas Markey, and Ghassan Oreiby. On the expressiveness and complexity of ATL. *Logical Methods in Computer Science*, 4(2), May 2008. doi:10.2168/LMCS-4(2:7)2008.
- 17 Fabio Mogavero, Aniello Murano, Giuseppe Perelli, and Moshe Y. Vardi. What makes ATL\* decidable? A decidable fragment of strategy logic. In Maciej Koutny and Irek Ulidowski, editors, *Proceedings of the 23rd International Conference on Concurrency Theory (CONCUR'12)*, volume 7454 of *Lecture Notes in Computer Science*, pages 193–208. Springer-Verlag, September 2012.
- 18 Fabio Mogavero, Aniello Murano, Giuseppe Perelli, and Moshe Y. Vardi. Reasoning about strategies: On the model-checking problem. *ACM Transactions on Computational Logic*, 15(4):34:1–34:47, August 2014. doi:10.1145/2631917.
- 19 Fabio Mogavero, Aniello Murano, Giuseppe Perelli, and Moshe Y. Vardi. Reasoning about strategies: On the satisfiability problem. *Logical Methods in Computer Science*, 13(1), March 2017. doi:10.23638/LMCS-13(1:9)2017.
- 20 Fabio Mogavero, Aniello Murano, and Luigi Sauro. On the boundary of behavioral strategies. In *Proceedings of the 28th Annual Symposium on Logic in Computer Science (LICS'13)*, pages 263–272. IEEE Comp. Soc. Press, June 2013.
- 21 Fabio Mogavero, Aniello Murano, and Luigi Sauro. A behavioral hierarchy of strategy logic. In Nils Bulling, Leendert W. N. van der Torre, Serena Villata, Wojciech Jamroga, and Wamberto Weber Vasconcelos, editors, *Proceedings of the 15th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA'14)*, volume 8624 of *Lecture Notes in Artificial Intelligence*, pages 148–165. Springer-Verlag, August 2014. doi:10.1007/978-3-319-09764-0\_10.
- 22 Fabio Mogavero, Aniello Murano, and Moshe Y. Vardi. Reasoning about strategies. In Kamal Lodaya and Meena Mahajan, editors, *Proceedings of the 30th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'10)*, volume 8 of *Leibniz International Proceedings in Informatics*, pages 133–144. Leibniz-Zentrum für Informatik, December 2010. doi:10.4230/LIPIcs.FSTTCS.2010.133.
- 23 Andrzej Mostowski. Games with forbidden positions. Research Report 78, University of Danzig, 1991.
- 24 Amir Pnueli. The temporal logic of programs. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science (FOCS'77)*, pages 46–57. IEEE Comp. Soc. Press, October–November 1977. doi:10.1109/SFCS.1977.32.
- 25 Jean-Pierre Queille and Joseph Sifakis. Specification and verification of concurrent systems in CESAR. In Mariangiola Dezani-Ciancaglini and Ugo Montanari, editors, *Proceedings of the 5th International Symposium on Programming (SOP'82)*, volume 137 of *Lecture Notes in Computer Science*, pages 337–351. Springer-Verlag, April 1982. doi:10.1007/3-540-11494-7\_22.
- 26 Farn Wang, Chung-Hao Huang, and Fang Yu. A temporal logic for the interaction of strategies. In Joost-Pieter Katoen and Barbara König, editors, *Proceedings of the 22nd International Conference on Concurrency Theory (CONCUR'11)*, volume 6901 of *Lecture Notes in Computer Science*, pages 466–481. Springer-Verlag, September 2011.

# Appendix

## A Proofs of Section 3

Before proving Propositions 2 and 3, we show that any map satisfying (E) also satisfies (C). Indeed, consider a map  $\theta$  satisfying (E), and pick two strategy valuations  $w_1$  and  $w_2$  and an existential variable  $x_i$  such that

$$\forall x_j \in \mathcal{V} \cap \{x_j \mid j < i\}. w_1(x_j) = w_2(x_j).$$

In particular, for those  $x_j$ , we have  $w_1(x_j)(\rho) = w_2(x_j)(\rho)$  for any history  $\rho$  (hence also for any of its prefixes). By (E), it follows  $\theta(w_1)(x_i)(\rho) = \theta(w_2)(x_i)(\rho)$ . Since this holds for any history, we have shown  $\theta(w_1)(x_i) = \theta(w_2)(x_i)$ .

### A.1 Proof of Proposition 2

► **Proposition 2.** *For any game  $\mathcal{G}$  with initial state  $q_0$ , and any formula  $\varphi \in \text{SL}[1G]^b$ , it holds  $\mathcal{G}, q_0 \models^E \varphi \Leftrightarrow \mathcal{G}, q_0 \not\models^E \neg\varphi$ .*

**Proof.** We begin with intuitive explanations before going into full details. We encode the satisfaction relation  $\mathcal{G}, q_0 \models^E \varphi$  into a two-player turn-based parity game: the first player of the parity game will be in charge of selecting the existentially-quantified strategies, and her opponent will select the universally-quantified ones. This will be encoded by replacing each state of  $\mathcal{G}$  with a tree-shaped module as depicted on Fig. 7. Following the strategy assignment of the  $\text{SL}[1G]$  formula  $\varphi$ , the strategies selected by those players will define a unique play, along which the LTL objective has to be fulfilled; this verification is encoded into a (doubly-exponential) parity automaton.

We prove that  $\mathcal{G}, q_0 \models^E \varphi$  if, and only if, the first player wins; conversely,  $\mathcal{G}, q_0 \not\models^E \varphi$  if the second player wins. Both claims crucially rely on the existence of memoryless optimal strategies for two-player parity games. Finally, by determinacy of those games, we get the expected result.

**Building a turn-based parity game  $\mathcal{H}$  from  $\mathcal{G}$  and  $\varphi$ .** For the rest of the proof, we fix a game  $\mathcal{G}$  and a  $\text{SL}[1G]$  formula  $\varphi = (Q_i x_i)_{i \leq l} \beta \varphi$ . Each state of  $\mathcal{G}$  is replaced with a copy of the *quantification game* depicted on Fig. 7. A quantification game  $\mathcal{Q}_\varphi$  is formally defined as follows:

- they involve two players  $P_\exists$  and  $P_\forall$ ;
- the set of states is  $S_\varphi = \{\mathbf{m} \in \text{Act}^* \mid 0 \leq |\mathbf{m}| \leq l\}$ , thereby defining a tree of depth  $l + 1$  with directions  $\text{Act}$ . A state  $\mathbf{m}$  in  $S_\varphi$  with  $0 \leq |\mathbf{m}| < l$  belongs to Player  $P_\exists$  if, and only if,  $Q_{|\mathbf{m}|+1} = \exists$ . States with  $|\mathbf{m}| = l$  will have only one outgoing transition.
- from each  $\mathbf{m}$  with  $0 \leq |\mathbf{m}| < l$ , for all  $a \in \text{Act}$ , there is a transition to  $\mathbf{m} \cdot a$ . The empty word  $\varepsilon \in S_\varphi$  is the starting node of the quantification game, and has no incoming transitions; states with  $|\mathbf{m}| = l$  (currently) have no outgoing transitions.

A leaf (i.e., a state  $\mathbf{m}$  with  $|\mathbf{m}| = l$ ) in a quantification game represents a move vector of domain  $\mathcal{V} = \{x_i \mid 1 \leq i \leq l\}$ : we identify each leaf  $\mathbf{m}$  with the move vector  $\mathbf{m}$ , hence writing  $\mathbf{m}(x_i)$  for  $\mathbf{m}(i)$ .

We denote by  $D$  a deterministic parity automaton over  $2^{\text{AP}}$  associated with  $\varphi$ . We write  $d_0$  for the initial state of  $D$ . Using quantification games, we can now define the turn-based parity game  $\mathcal{H}$ :

- it involves both players  $P_{\exists}$  and  $P_{\forall}$ ;
- for each state  $q$  of  $\mathcal{G}$  and each state  $d$  of  $D$ ,  $\mathcal{H}$  contains a copy of the quantification game  $\mathcal{Q}_\varphi$ , which we call the  $(q, d)$ -copy. Hence the set of states of  $\mathcal{H}$  is the product of the state spaces of  $\mathcal{G}$ ,  $D$  and  $\mathcal{Q}_\varphi$ .
- the transitions in  $\mathcal{H}$  are of two types:
  - internal transitions in each copy of the quantification game are preserved;
  - consider a state  $(q, d, \mathbf{m})$  where  $|\mathbf{m}| = l$ ; this is a leaf in the quantification game. If there exists a state  $q'$  such that  $q' = \Delta(q, m_\beta)$  where  $m_\beta: \mathbf{Agt} \rightarrow \mathbf{Act}$  is the move vector over  $\mathbf{Agt}$  defined by  $m_\beta(A) = \mathbf{m}(i-1)$  where  $x_i = \beta(A)$  (i.e., assigning to each player  $A \in \mathbf{Agt}$  the action  $\mathbf{m}(\beta(A))$ ), then we add a transition from  $(q, d, \mathbf{m})$  to  $(q', d', \varepsilon)$  where  $d'$  is the state of  $D$  reached from  $d$  when reading  $\text{lab}(q')$ . Notice that  $(q, d, \mathbf{m})$  then has at most one outgoing transition.
- the priorities are inherited from those in  $D$ : state  $(q, d, \mathbf{m})$  has the same priority as  $d$ .

**Correspondence between  $\mathcal{G}$  and  $\mathcal{H}$ .** We define a correspondence between  $\mathcal{G}$  and  $\mathcal{H}$  through the notion of *lanes*:

► **Definition 12.** A *lane* in  $\mathcal{G}$  is a tuple  $(\rho, u, b, t)$  made of

- a history  $\rho = (q_j)_{0 \leq j \leq a}$  (for some integer  $a$ );
  - a function  $u: \mathcal{V} \times \text{Pref}(\rho) \rightarrow \mathbf{Act}$ ;
  - an integer  $b \in [0; l]$ ;
  - a function  $t: \{x_1, \dots, x_b\} \rightarrow \mathbf{Act}$  ( $t$  is the empty function if  $b = 0$ );
- and such that

$$\forall 0 \leq j < a. \quad \Delta(q_j, (m_j(\beta(A)))_{A \in \mathbf{Agt}}) = q_{j+1} \quad \text{with } m_j: \mathcal{V} \rightarrow \mathbf{Act} \quad (1)$$

$$x \mapsto u(x, \rho_{\leq j})$$

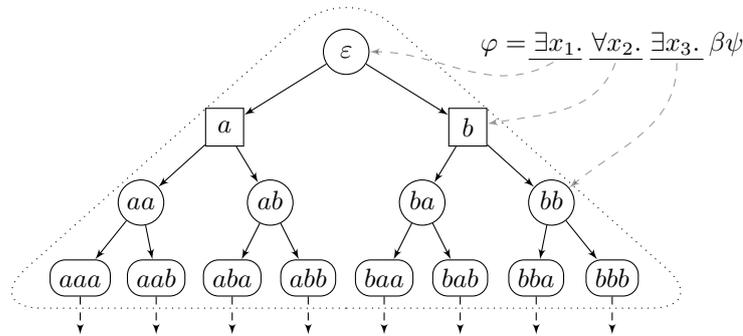
We can then build a one-to-one application  $HtoG_p$  between histories in  $\mathcal{H}$  and lanes in  $\mathcal{G}$ . With a history  $\pi$  in  $\mathcal{H}$ , written

$$\pi = \left( \prod_{0 \leq j < a} \prod_{0 \leq i \leq l} (q_j, d_j, \mathbf{m}_{j,i}) \right) \cdot \prod_{0 \leq i \leq b} (q_a, d_a, \mathbf{m}_{a,i}),$$

having length  $a \cdot (l+1) + b + 1$  with  $0 \leq b < l$ , we associate a lane  $HtoG_p(\pi) = ((q_j)_{j \leq a}, u, b, t)$  with

$$u: \mathcal{V} \times \text{Pref}(\rho) \rightarrow \mathbf{Act} \quad t: \{x_1, \dots, x_b\} \rightarrow \mathbf{Act}$$

$$x_i, (q_j)_{j \leq c} \mapsto \mathbf{m}_{c,i} \quad (\forall c < a) \quad x_i \mapsto \mathbf{m}_{a,i}$$



■ **Figure 7** The quantification game for  $\varphi = \exists x_1. \forall x_2. \exists x_3. \beta\psi$ .

The resulting function  $HtoG_p$  is clearly injective (different histories will correspond to different lanes), but also surjective. To prove the latter statement, we build the inverse function  $GtoH_p$ : for a lane  $((q_j)_{j \leq a}, u, b, t)$ , we set  $GtoH_p((q_j)_{j \leq a}, u, b, t) = \pi$  where  $\pi$  is the history in  $\mathcal{H}$  of length  $a \cdot (l + 1) + b + 1$  defined as

$$\pi = \prod_{0 \leq j < a} \prod_{0 \leq i \leq l} (q_j, d_j, u(x_i, (q_{j'})_{j' \leq j})) \cdot \prod_{0 \leq i \leq b} (q_a, d_a, t(x_i, (q_j)_{j \leq a}))$$

where  $d_j$  is the state of  $D$  reached on input  $(q_k)_{0 \leq k \leq j-1}$ .

Because of the coherence condition (1),  $GtoH_p((q_j)_{j \leq a}, u, i, t)$  is indeed a history in  $\mathcal{H}$ . From the definitions, one can easily check that

$$GtoH_p(HtoG_p(\pi)) = \pi$$

and deduce that  $GtoH_p$  is the inverse function of  $HtoG_p$ ; therefore

► **Lemma 13.** *The application  $HtoG_p$  is a bijection between lanes of  $\mathcal{G}$  and histories in  $\mathcal{H}$ , and  $GtoH_p$  is its inverse function.*

**Extending the correspondence.** We can use  $HtoG_p$  to describe another correspondence  $HtoG$  between (positional) strategies for  $P_{\exists}$  in  $\mathcal{H}$  and (elementary) maps in  $\mathcal{G}$ . Remember that a map is a function  $\theta: (\text{Hist}_{\mathcal{G}} \rightarrow \text{Act})^{\forall} \rightarrow (\text{Hist}_{\mathcal{G}} \rightarrow \text{Act})^{\forall}$ . Remember also that if  $Q_j = \forall$ , then  $\theta(w)(x_i)(\rho) = w(x_i)(\rho)$ , so that we only have to define the map for the existentially-quantified variables.

Formally, the application  $HtoG$  takes as input a strategy  $\delta$  for player  $P_{\exists}$  in  $\mathcal{H}$ , and returns a map in  $\mathcal{G}$ . It will enjoy the following properties:

- for any finite outcome  $\pi$  of  $\delta$  in  $\mathcal{H}$  ending at the root of a quantification game, there exists a function  $w$  such that  $HtoG_p(\pi) = (\rho, u, 0, t_{\emptyset})$  where  $\rho$  is the outcome of  $HtoG(\delta)(w)$  in  $\mathcal{G}$  under the assignment defined by  $\beta$ ;
- conversely, for any path  $\rho$  in  $\mathcal{G}$  that is an outcome of  $HtoG(\delta)(w)$  for some  $w$  and under the assignment defined by  $\beta$ , then letting  $u(x, \rho') = HtoG(\delta)(w)(x)(\rho')$ , we have that  $(\rho, u, 0, t_{\emptyset})$  is a lane in  $\mathcal{G}$  and  $GtoH_p(\rho, u, 0, t_{\emptyset})$  is an outcome of  $\delta$  in  $\mathcal{H}$  ending in the root of a quantification game.

We fix  $\delta$ , and for all  $w, \rho$  and  $x_i$ , we define  $HtoG(\delta)(w)(x_i)(\rho)$  by a double induction, first on the length of the history  $\rho$  in  $\mathcal{G}$ , and second on the sequence of variables  $x_i$ .

- **initial step:** we begin with the case where  $\rho$  is the single state  $q_0$ . We proceed by induction on existentially-quantified variables, merging the initialization step with the induction step as they are similar. Consider an existentially-quantified variable  $x_i$  in  $\mathcal{V}$ . Given  $w: \mathcal{V}^{\forall} \times \text{Pref}(\rho) \cup \{\rho\} \rightarrow \text{Act}$ , we define a function  $t_{i,w}: [x_1; x_{i-1}] \rightarrow \text{Act}$  such that  $t_{i,w}(x) = w(x, q_0)$  for  $x \in \mathcal{V}^{\forall} \cap [x_1; x_{i-1}]$ , and  $t_{i,w}(x) = HtoG(\delta)(w)(x)(q_0)$  for  $x \in \mathcal{V}^{\exists} \cap [x_1; x_{i-1}]$ , assuming that they have been defined in the previous induction steps on variables. We can then create the lane  $lane_{i,w} = (\varepsilon, u_{\emptyset}, i - 1, t)$  and define

$$HtoG(\delta)(w)(x_i)(q_0) = \delta(GtoH_p(lane_{i,w}))$$

Pick an outcome  $\pi$  of  $\delta$  in  $\mathcal{H}$  of length  $l + 2$ , and write  $\mathbf{m}$  for its  $l + 1$ -st state: it defines a valuation for the variables in  $\mathcal{V}$ , hence defining a move vector  $m_{\beta}$  under the assignment  $\beta$ . in  $\text{Act}$ . By construction of  $\mathcal{H}$ , this outcome ends in the state  $(q_1, d_1, \varepsilon)$  where  $q_1 = \Delta(q_0, m_{\beta})$  and  $d_1$  is the successor of the initial state  $d_0$  of  $D$  when reading  $\text{lab}(q_1)$ . We now prove that  $q_0 \cdot q_1$  is the outcome of  $HtoG(\delta)(w)$  for some  $w$ . For this, we let  $w(x_i) = \mathbf{m}_i$  for all  $x_i \in \mathcal{V}^{\forall}$ . By construction,  $HtoG(\delta)(w)(x_j)(q_0)$  precisely corresponds to  $\mathbf{m}(j)$ , for

all  $x_j \in \mathcal{V}^\exists$ . In the end, under assignment  $\beta$ ,  $HtoG(\delta)(w)$  precisely returns the move vector  $m_\beta$ , hence proving our result.

The proof of the converse statement follows similar arguments: consider an outcome  $\rho = q_0 \cdot q_1$  of  $HtoG(\delta)(w)$  for some  $w$ . The lane  $(\rho, u, 0, t_\emptyset)$  defined with  $u(x, q_0) = HtoG(\delta)(w)(x)(q_0)$  then corresponds through  $GtoH_p$  to a play ending in  $(q_1, d_1, \varepsilon)$ , and visiting the leaf  $\mathbf{m}$  defined as  $\mathbf{m}_i = u(x_i, q_0)$ . By construction, this is an outcome of  $\delta$  in  $\mathcal{H}$ .

- **induction step:** we consider a history  $\rho$  in  $\mathcal{G}$ , assuming we have define  $HtoG(\delta)(w)(x_i)(\rho')$  for all prefixes  $\rho'$  of  $\rho$ , for all  $w$  and all variables  $x_i$ . We now define  $HtoG(\delta)(w)(x_i)(\rho)$ , by induction on the list of variables. Again, the initialization step is merged with the induction step as they rely on the same arguments.

Consider an existentially-quantified variable  $x_i$ , and  $w: \mathcal{V}^\forall \times \text{Pref}(\rho) \cup \{\rho\} \rightarrow \text{Act}$ . We define a function  $t_{i,w}: [x_1; x_{i-1}] \rightarrow \text{Act}$  where  $t_{i,w}$  associate with  $x \in \mathcal{V}^\forall \cap [x_1; x_{i-1}]$  the action  $w(x)(\pi)$ , and with  $x \in \mathcal{V}^\exists \cap [x_1; x_{i-1}]$  the action  $HtoG(\delta)(w)(x)(\rho)$ . We also define  $u_w: \mathcal{V} \times \text{Pref}(\rho) \rightarrow \text{Act}$  as  $u_w(x, \rho') = HtoG(\delta)(w)(x)(\rho')$ , for all prefixes  $\rho'$  of  $\rho$ . We can then create the lane  $lane_{i,w} = (\pi, u_w, i-1, t_{i,w})$  and finally define

$$HtoG(\delta)(w)(x_i)(\rho) = \delta(GtoH_p(lane_{i,w})).$$

Using the same arguments as in the initial step, we prove our correspondence between the outcomes of  $\delta$  in  $\mathcal{H}$  and the outcomes of  $HtoG(\delta)$  in  $\mathcal{G}$ .

Notice that in the construction above,  $HtoG(\delta)(w)(x_i)(\rho)$  may depend on the value of  $w(x_j, \rho')$  for  $j > i$  and  $\rho' \in \text{Pref}(\rho)$ : indeed, in the inductive definition, we define  $HtoG(\delta)(w)(x_j)(\rho')$  before defining  $HtoG(\delta)(w)(x_i)(\rho)$ . Hence in general  $HtoG(\delta)$  is *not* an elementary map.

However, in case  $\delta$  is memoryless, we notice that  $HtoG(\delta)(w)(x_i)(\rho)$  only depends on value of  $\delta$  in the last state of the lane  $lane_{i,w}$ , hence in particular not on  $u_w$ . This removes the above dependence, and makes  $HtoG(\delta)$  elementary.

Finally, notice that we can define a dual correspondence  $\overline{HtoG}$  relating strategies of Player  $P_\forall$  and elementary maps in  $\mathcal{G}$  where existential and universal variables are swapped.

**Concluding the proof.** Using  $HtoG$ , we prove our final correspondence between  $\mathcal{H}$  and  $\mathcal{G}$ :

► **Lemma 14.** *Assume that  $P_\exists$  is winning in  $\mathcal{H}$  and let  $\delta$  be a positional winning strategy. Then the elementary map  $HtoG(\delta)$  is a witness that  $\mathcal{G}, q_0 \models^E \varphi$ .*

*Similarly, assume that  $P_\forall$  is winning in  $\mathcal{H}$  and let  $\bar{\delta}$  be a positional winning strategy. Then the elementary map  $\overline{HtoG}(\bar{\delta})$  is a witness that  $\mathcal{G}, q_0 \models^E \neg\varphi$ .*

**Proof.** We prove the first point, the second one following similar arguments. Assume that  $P_\exists$  is winning in  $\mathcal{H}$ , and pick a memoryless winning strategy  $\delta$ . Toward a contradiction, assume further that  $HtoG(\delta)$  is not a witness of  $\mathcal{G}, q_0 \models^E \varphi$ . Then there exists  $w_0: \mathcal{V}^\forall \rightarrow (\text{Hist}_{\mathcal{G}} \rightarrow \text{Act})$  s.t.  $\mathcal{G}, q_0 \not\models_{HtoG(\delta)(w_0)} \beta\varphi$ . We use  $w_0$  to build a strategy  $\bar{\delta}$  for Player  $P_\forall$  in  $\mathcal{H}$ . Given a history

$$\pi = \prod_{0 \leq j < a} \prod_{0 \leq i \leq l} (q_j, d_j, \mathbf{m}_{j,i}) \cdot \prod_{0 \leq i \leq b} (q_a, d_a, \mathbf{m}_{a,i})$$

in  $\mathcal{H}$ , we define  $\rho = \prod_{0 \leq j \leq a} q_j$  and set  $\bar{\delta}(\pi) = HtoG(\delta)(w)(x_b)(\eta)$  where

- $w: \text{Pref}(\rho) \cup \{\rho\} \times (\mathcal{V}^\forall \cap [x_1; x_b]) \rightarrow \text{Act}$  is such that  $w(\rho', x_i)$  is the action to be played for going from  $\pi_{\leq |\rho'| \cdot (l+1) + i - 1}$  to  $\pi_{\leq |\rho'| \cdot (l+1) + i}$  in  $\mathcal{H}$ ;
- $\eta = \prod_{0 \leq j < a} \prod_{0 \leq i \leq l} (q_j, d_j, \mathbf{m}_{j,i})$ .

Write  $\nu = (q_j)_{j \in \mathbb{N}}$  for the outcome of  $\theta(w_0)$  under strategy assignment  $\beta$  in  $\mathcal{G}$ . Then, by construction of  $\bar{\delta}$ , the outcome of  $\delta$  and  $\bar{\delta}$  in  $\mathcal{H}$  will visit the  $(q_j, d_j)_{j \in \mathbb{N}}$ -copies of the quantification game, where  $d_j$  is the state reached by reading  $(q_{j'})_{j' \leq j}$  in the deterministic automaton  $D$ . Now, since  $\mathcal{G}, q_0 \not\models_{HtoG(\delta)} \beta\varphi$ , we get that  $\nu$  does not satisfy  $\varphi$  and therefore the outcome of  $\delta$  and  $\bar{\delta}$  in  $\mathcal{H}$  does not satisfy the parity condition. This is in contradiction with  $\delta$  being the winning strategy of  $P_{\exists}$ , and proves that  $HtoG(\delta)$  must be a witness that  $\mathcal{G}, q_0 \models^E \varphi$ .  $\blacktriangleleft$

Proposition 14, together with the determinacy of parity games [11, 23] immediately imply that at least one of  $\varphi$  and  $\neg\varphi$  must hold in  $\mathcal{G}$  for  $\models^E$ . This concludes our proof.  $\blacktriangleleft$

## A.2 Proof of Proposition 3

► **Proposition 3.** *The relations  $\models^C$  and  $\models^E$  differ on  $SL[CG]^b$ , as well as on  $SL[DG]^b$ .*

**Proof.** The proof of Prop. 1 provides an example of a game and a formula in  $SL[CG]^b$  where  $\models^C$  and  $\models^E$  differ. We prove the result for  $SL[DG]^b$ : consider again the game of Fig. 1 in Section 2. We already proved that  $\mathcal{G}, q_0 \models^C \varphi$ ; we show that  $\mathcal{G}, q_0 \not\models^E \varphi$ . For this, consider the following four valuations for the universally-quantified strategies:

$$\begin{array}{lll} w_1(x_A)(q_0) = q_1 & w_1(x_B)(q_0) = q_2 & w_1(y)(q_0 \cdot q_1) = p_2 \\ w_2(x_A)(q_0) = q_2 & w_2(x_B)(q_0) = q_1 & w_2(y)(q_0 \cdot q_1) = p_1 \end{array}$$

(assuming that they coincide in any other situation). Let  $\theta$  be an elementary  $\varphi$ -map: then it must be such that  $\theta(w_1)(z)(q_0 \cdot q_2) = \theta(w_2)(z)(q_0 \cdot q_2)$ . Then:

- if  $\theta(w_1)(z)(q_0 \cdot q_2) = \theta(w_2)(z)(q_0 \cdot q_2) = p_1$ , then the first goal goes to  $p_2$  via  $q_1$ , and the second goal goes to  $p_1$  via  $q_2$ . None of those goals is fulfilled;
- if  $\theta(w_1)(z)(q_0 \cdot q_2) = \theta(w_2)(z)(q_0 \cdot q_2) = p_2$ , then the first goal goes to  $p_2$  via  $q_2$ , and the second goal goes to  $p_1$  via  $q_1$ . Again, both goals are missed.  $\blacktriangleleft$

## B Proofs of Section 4

### B.1 Proof of Proposition 6

► **Proposition 6.** *For any formula  $\varphi$  in  $SL[BG]^b$ , for any game  $\mathcal{G}$  and any state  $q_0$ , we have  $\mathcal{G}, q_0 \models^T \varphi \Rightarrow \mathcal{G}, q_0 \not\models^T \neg\varphi$ .*

**Proof.** For a contradiction, assume that there exist two maps  $\theta$  and  $\bar{\theta}$  witnessing  $\mathcal{G}, q_0 \models^T \varphi$  and  $\mathcal{G}, q_0 \models^T \neg\varphi$  resp. Then

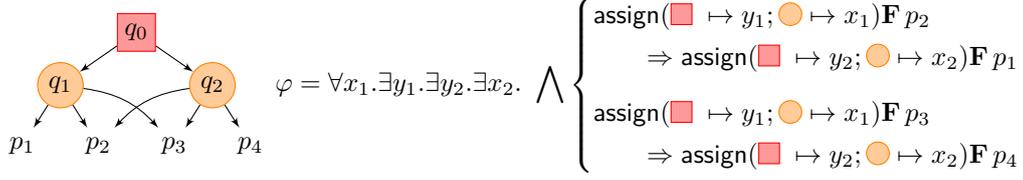
$$\forall w: \mathcal{V}^\forall \rightarrow (\text{Hist} \rightarrow \text{Act}). \quad \mathcal{G}, q_0 \models_{\theta(w)} \xi(\beta_j \varphi_j)_{j \leq n} \tag{2}$$

$$\forall \bar{w}: \mathcal{V}^\exists \rightarrow (\text{Hist} \rightarrow \text{Act}). \quad \mathcal{G}, q_0 \models_{\bar{\theta}(\bar{w})} \neg\xi(\beta_j \varphi_j)_{j \leq n} \tag{3}$$

From  $\theta$  and  $\bar{\theta}$ , we build a strategy valuation  $\chi$  on  $\mathcal{V}$  such that  $\theta(\chi|_{\mathcal{V}^\forall}) = \bar{\theta}(\chi|_{\mathcal{V}^\exists}) = \chi$ . By Equations (2) and (3), we get that  $\mathcal{G}, q_0 \models_\chi \xi(\beta_j \varphi_j)_{j \leq n}$  and  $\mathcal{G}, q_0 \models_\chi \neg\xi(\beta_j \varphi_j)_{j \leq n}$ , which for LTL formulas is impossible.

We define  $\chi(x)(\rho)$  inductively on histories and on the list of quantified variables. When  $\rho$  is the empty history  $q_0$ , we consider two cases:

- if  $x_1 \in \mathcal{V}^\forall$ , then  $\bar{\theta}(\bar{w})(x_1)(q_0)$  does not depend on  $\bar{w}$  at all, since  $\bar{\theta}$  is a timeline-map. Hence we let  $\chi(x_1)(q_0) = \bar{\theta}(\bar{w})(x_1)(q_0)$ , for any  $\bar{w}$ .



■ **Figure 8** A game  $\mathcal{G}$  and a formula  $\varphi$  such that  $\mathcal{G}, q_0 \models^T \varphi$  and  $\mathcal{G}, q_0 \not\models^T \neg\varphi$

■ similarly, if  $x_1 \in \mathcal{V}^\exists$ , we let  $\chi(x_1)(q_0) = \theta(w)(x_1)(q_0)$ , which again does not depend on  $w$ . Similarly, when  $\chi(x)(q_0)$  has been defined for all  $x \in \{x_1, \dots, x_{i-1}\}$ , we again consider two cases:

- if  $x_i \in \mathcal{V}^\forall$ , we define  $\bar{w}(x_j)(q_0) = \chi(x_j)(q_0)$  for all  $x_j \in \mathcal{V}^\exists \cap \{x_1, \dots, x_{i-1}\}$ , and let  $\chi(x_i)(q_0) = \bar{\theta}(\bar{w})(x_i)(q_0)$ , which again does not depend on the value of  $\bar{w}$  besides those defined above;
- symmetrically, if  $x_i \in \mathcal{V}^\exists$ , we define  $w(x_j)(q_0) = \chi(x_j)(q_0)$  for all  $x_j \in \mathcal{V}^\forall \cap \{x_1, \dots, x_{i-1}\}$ , and let  $\chi(x_i)(q_0) = \theta(w)(x_i)(q_0)$ .

Notice that this indeed enforces that  $\theta(\chi|_{\mathcal{V}^\forall})(x_i)(q_0) = \chi(x_i)(q_0)$  when  $x_i \in \mathcal{V}^\exists$ , and  $\bar{\theta}(\chi|_{\mathcal{V}^\exists})(x_i)(q_0) = \chi(x_i)(q_0)$  when  $x_i \in \mathcal{V}^\forall$ .

The induction step is proven similarly: consider a history  $\rho$  and a variable  $x_i$ , assuming that  $\chi$  has been defined for all variables on all prefixes of  $\rho$ , and for variables in  $\{x_1, \dots, x_{i-1}\}$  on  $\rho$  itself. Then:

- if  $x_i \in \mathcal{V}^\forall$ , we define  $\bar{w}(x_j)(\rho') = \chi(x_j)(\rho')$  for all  $x_j \in \mathcal{V}$  and all  $\rho' \in \text{Pref}(\rho)$ , and  $\bar{w}(x_j)(\rho) = \chi(x_j)(\rho)$  for all  $x_j \in \mathcal{V}^\exists \cap \{x_1, \dots, x_{i-1}\}$ . We then let  $\chi(x_i)(\rho) = \bar{\theta}(\bar{w})(x_i)(q_0)$ , which does not depend on the value of  $\bar{w}$  besides those defined above;
- the construction for the case when  $x_i \in \mathcal{V}^\exists$  is similar.

As in the initial step, it is easy to check that this construction enforces  $\theta(\chi|_{\mathcal{V}^\forall}) = \bar{\theta}(\chi|_{\mathcal{V}^\exists}) = \chi$ , as required. ◀

## B.2 Proof of Proposition 7

► **Proposition 7.** *There exists a formula  $\varphi \in \text{SL}[\text{BG}]^b$ , a (turn-based) game  $\mathcal{G}$  and a state  $q_0$  such that  $\mathcal{G}, q_0 \models^T \varphi$  and  $\mathcal{G}, q_0 \not\models^T \neg\varphi$ .*

**Proof.** Consider the turn-based game  $\mathcal{G}$  and the  $\text{SL}[\text{BG}]^b$  formula  $\varphi$  of Fig. 8. First,  $\mathcal{G}, q_0 \models^T \varphi$ , since for any choice of  $x_1$  and  $y_1$ , one of the goals holds vacuously, and the other one can be made true by correctly selecting  $y_2$  and  $x_2$ . We now prove that  $\mathcal{G}, q_0 \not\models^T \neg\varphi$ : since timeline dependences are allowed,  $\theta(w)(x_1)(q_0 \cdot q_1)$  and  $\theta(w)(x_1)(q_0 \cdot q_2)$  may depend on the values of  $w(y_1)(q_0)$  and  $w(y_2)(q_0)$ . We thus consider four cases:

- if  $w(y_1)(q_0) = w(y_2)(q_0) = q_1$ , then we let  $\theta(w)(x_1)(q_0 \cdot q_1) = p_3$ ; then the second goal of  $\varphi$  is not fulfilled, whatever  $w(x_2)$ ;
- if  $w(y_1)(q_0) = w(y_2)(q_0) = q_2$ , then symmetrically, we let  $\theta(w)(x_1)(q_0 \cdot q_2) = p_2$ , so that the first goal of  $\varphi$  fails to hold for any  $w(x_2)$ ;
- if  $w(y_1)(q_0) = q_1$  and  $w(y_2)(q_0) = q_2$ , then we let  $\theta(w)(x_1)(q_0 \cdot q_1) = p_2$ , and again the first goal holds, whatever  $w(x_2)$ ;
- if  $w(y_1)(q_0) = q_2$  and  $w(y_2)(q_0) = q_1$ , then we let  $\theta(w)(x_1)(q_0 \cdot q_2) = p_3$ , and again the second goal fails to hold independently of  $w(x_2)$ . ◀

## C

 Proofs of Section 5

### C.1 Proof of Proposition 9

► **Proposition 9.**  $SL[EG]^b$  contains  $SL[AG]^b$ . The inclusion is strict (syntactically).

**Proof.** Remember that boolean combinations in  $SL[AG]^b$  follow the grammar  $\xi ::= \xi \vee \beta \mid \xi \wedge \beta \mid \beta$ . In terms of subsets of  $\{0, 1\}^n$ , it corresponds to considering sets defined in one of the following two forms:

$$F_\xi^n = \{f \in \{0, 1\}^n \mid f(n) = 1\} \cup \{g \in \{0, 1\}^n \mid g_{[1;n-1]} \in F_{\xi'}^{n-1}\}$$

$$F_\xi^n = \{f \in \{0, 1\}^n \mid f(n) = 1 \text{ and } f_{[1;n-1]} \in F_{\xi'}^{n-1}\}$$

depending whether  $\xi(p_j)_j = \xi'(p_j)_j \vee p_n$  or  $\xi(p_j)_j = \xi'(p_j)_j \wedge p_n$ . Assuming (by induction) that  $F_{\xi'}^{n-1}$  is semi-stable, then we can prove that  $F_\xi^n$  also is. We detail the proof for the second case, the first case being similar.

Consider the case where  $F_\xi^n = \{f \in \{0, 1\}^n \mid f(n) = 1 \text{ and } f_{[1;n-1]} \in F_{\xi'}^{n-1}\}$ . Pick any two elements  $f$  and  $g$  in  $F_\xi^n$ , and  $s \in \{0, 1\}^n$ . Since  $f(n) = g(n) = 1$ , we have  $[(f \wedge s) \vee (g \wedge \bar{s})](n) = [(f \wedge \bar{s}) \vee (g \wedge s)](n) = 1$ . Moreover, the restriction of  $[(f \wedge s) \vee (g \wedge \bar{s})]$  and of  $[(f \wedge \bar{s}) \vee (g \wedge s)]$  to their first  $n-1$  bits is computed from the restriction of  $f$ ,  $g$  and  $s$  to their first  $n-1$  bits. Since  $F_{\xi'}^{n-1}$  is semi-stable, one of  $[(f \wedge s) \vee (g \wedge \bar{s})]_{[1;n-1]}$  and  $[(f \wedge \bar{s}) \vee (g \wedge s)]_{[1;n-1]}$  belongs to  $F_{\xi'}^{n-1}$ , so that one of  $[(f \wedge s) \vee (g \wedge \bar{s})]$  and  $[(f \wedge \bar{s}) \vee (g \wedge s)]$  is in  $F_\xi^n$ .

That the inclusion is strict is proven by considering the semi-stable set  $H^3 = \{(1, 1, 1), (1, 1, 0), (1, 0, 1), (0, 1, 1)\}$ . Assume that it corresponds to a formula in  $SL[AG]^b$ : then the boolean combination  $\xi(x_1, x_2, x_3)$  of that formula must be in one of the following forms:

$$\xi'(x_1, x_2) \wedge x_3 \quad \xi'(x_1, x_2) \vee x_3 \quad \xi'(x_1, x_2) \wedge \neg x_3 \quad \xi'(x_1, x_2) \vee \neg x_3.$$

It remains to prove that none of these cases corresponds to  $H^3$ : the first case does not allow  $(1, 1, 0)$ ; the second case allows  $(0, 0, 1)$ ; the third case does not allow  $(1, 0, 1)$ ; the last case allows  $(0, 0, 0)$ . ◀

### C.2 Proof of Lemma 10

► **Lemma 10.** Fix a semi-stable set  $F^n$  and  $s \in \{0, 1\}^n$ . For any  $h_1, h_2 \in \{0, 1\}^n$ , either  $\mathbb{F}^n(h_1, s) \subseteq \mathbb{F}^n(h_2, s)$  or  $\mathbb{F}^n(h_2, s) \subseteq \mathbb{F}^n(h_1, s)$ .

**Proof.** Assume otherwise, there is  $h'_1 \in \mathbb{F}^n(h_1, s) \setminus \mathbb{F}^n(h_2, s)$  and  $h'_2 \in \mathbb{F}^n(h_2, s) \setminus \mathbb{F}^n(h_1, s)$ . We then have:

$$(h_1 \wedge s) \vee (h'_1 \wedge \bar{s}) \in F^n \quad (h_2 \wedge s) \vee (h'_1 \wedge \bar{s}) \notin F^n$$

$$(h_2 \wedge s) \vee (h'_2 \wedge \bar{s}) \in F^n \quad (h_1 \wedge s) \vee (h_2 \wedge \bar{s}) \notin F^n$$

Now consider  $(h_1 \wedge s) \vee (h'_1 \wedge \bar{s})$ ,  $(h_2 \wedge s) \vee (h'_2 \wedge \bar{s})$  and  $s$ . As  $F^n$  is semi-stable, one of the two following vector is in  $F^n$ :

$$((h_1 \wedge s) \vee (h'_1 \wedge \bar{s}) \wedge s) \vee ((h_2 \wedge s) \vee (h'_2 \wedge \bar{s}) \wedge \bar{s})$$

$$((h_2 \wedge s) \vee (h'_2 \wedge \bar{s}) \wedge s) \vee ((h_1 \wedge s) \vee (h'_1 \wedge \bar{s}) \wedge \bar{s})$$

The first vector is equal to  $(h_1 \wedge s) \vee (h'_2 \wedge \bar{s})$  and the second to  $(h_2 \wedge s) \vee (h'_1 \wedge \bar{s})$  and both are supposed to be in  $F^n$ , we get a contradiction. ◀

While it is not related to the lemma above, we prove here a result that will be useful for the proof of Lemma 24 in Appendix C.4.

► **Lemma 15.** *Given a semi-stable set  $F^n$ ,  $s_1, s_2 \in \{0, 1\}^n$  such that  $s_1 \wedge s_2 = \mathbf{0}$  and  $f, g \in \{0, 1\}^n$  such that  $f \preceq_{s_1} g$  and  $f \preceq_{s_2} g$ . Then  $f \preceq_{s_1 \vee s_2} g$ .*

**Proof.** Because  $f \preceq_{s_1} g$  and  $f \preceq_{s_2} g$ , we have

$$\forall i \in \{1, 2\} \forall h \in \{0, 1\}^n \quad (f \wedge s_i) \vee (h \wedge \bar{s}_i) \in F^n \Rightarrow (g \wedge s_i) \vee (h \wedge \bar{s}_i) \in F^n \quad (4)$$

Consider  $h' \in \{0, 1\}^n$  such that  $\alpha = (f \wedge (s_1 \vee s_2)) \vee (h' \wedge \overline{(s_1 \vee s_2)})$  is in  $F^n$ . Define the element  $h = \alpha \wedge \bar{s}_2$ , then  $(f \wedge s_2) \vee (h \wedge \bar{s}_2) = (f \wedge (s_1 \vee s_2)) \vee (h' \wedge \overline{(s_1 \vee s_2)}) \in F^n$ . Using (4) with  $s_2$  and  $h$ , we get  $\beta = (g \wedge s_2) \vee (h \wedge \bar{s}_2)$ . As  $s_1 \wedge s_2 = \mathbf{0}$ , we can write  $\beta = (f \wedge s_1) \vee (g \wedge s_2) \vee (h' \wedge \overline{(s_1 \vee s_2)}) \in F^n$ .

Now consider  $h = \beta \wedge \bar{s}_1$ , we have  $(f \wedge s_1) \vee (h \wedge \bar{s}_1) = \beta \in F^n$ . Using (4) with  $s_1$  and  $h$ , we get  $(g \wedge (s_1 \vee s_2)) \vee (h' \wedge \overline{(s_1 \vee s_2)}) \in F^n$ . Therefore  $\mathbb{F}^n(f, s_1 \vee s_2) \subseteq \mathbb{F}^n(g, s_1 \vee s_2)$  and  $f \preceq_{s_1 \vee s_2} g$ . ◀

### C.3 Properties of semi-stable sets

Before proving our main theorem, we show that semi-stable sets enjoy several nice structural properties. Our first lemma shows that  $\text{SL}[\text{EG}]^b$  is closed under (syntactic) negation.

► **Lemma 16.**  *$F^n$  is semi-stable if, and only if, its complement is.*

**Proof.** Assume  $F^n$  is not semi-stable, and pick  $f$  and  $g$  in  $F^n$  and  $s \in \{0, 1\}^n$  such that none of  $\alpha = (f \wedge s) \vee (g \wedge \bar{s})$  and  $\gamma = (g \wedge s) \vee (f \wedge \bar{s})$  are in  $F^n$ . It cannot be the case that  $g = f$ , as this would imply  $\alpha = f \in F^n$ . Hence  $\alpha \neq \gamma$ . We claim that  $\alpha$  and  $\gamma$  are our witnesses for showing that the complement of  $F^n$  is not semi-stable: both of them belong to the complement of  $F^n$ , and  $(\alpha \wedge s) \vee (\gamma \wedge \bar{s})$  can be seen to equal  $f$ , hence it is not in the complement of  $F^n$ . Similarly for  $(\gamma \wedge s) \vee (\alpha \wedge \bar{s}) = g$ . ◀

► **Lemma 17.** *If  $F^n \subseteq \{0, 1\}^n$  is semi-stable, then for any  $s \in \{0, 1\}^n$  and any non-empty subset  $H^n$  of  $F^n$ , it holds that*

$$\exists f \in H^n. \forall g \in H^n. (f \wedge s) \vee (g \wedge \bar{s}) \in F^n.$$

**Proof.** For a contradiction, assume that there exist  $s \in \{0, 1\}^n$  and  $H^n \subseteq F^n$  such that, for any  $f \in H^n$ , there is an element  $g \in H^n$  for which  $(f \wedge s) \vee (g \wedge \bar{s}) \notin F^n$ . Then there must exist a minimal integer  $2 \leq \lambda \leq |H^n|$  and  $\lambda$  elements  $\{f_i \mid 1 \leq i \leq \lambda\}$  of  $H^n$  such that

$$\forall 1 \leq i \leq \lambda - 1 (f_i \wedge s) \vee (f_{i+1} \wedge \bar{s}) \notin F^n \text{ and } (f_\lambda \wedge s) \vee (f_1 \wedge \bar{s}) \notin F^n.$$

By Lemma 16, the complement of  $F^n$  is semi-stable. Hence, considering  $(f_{\lambda-1} \wedge s) \vee (f_\lambda \wedge \bar{s})$  and  $(f_\lambda \wedge s) \vee (f_1 \wedge \bar{s})$ , one of the following two vectors is not in  $F^n$ :

$$\begin{aligned} & ((f_{\lambda-1} \wedge s) \vee (f_\lambda \wedge \bar{s})) \wedge s \vee (((f_\lambda \wedge s) \vee (f_1 \wedge \bar{s})) \wedge \bar{s}) \\ & (((f_\lambda \wedge s) \vee (f_1 \wedge \bar{s})) \wedge s) \vee (((f_{\lambda-1} \wedge s) \vee (f_\lambda \wedge \bar{s})) \wedge \bar{s}) \end{aligned}$$

The second expression equals  $f_\lambda$ , which is in  $F^n$ . Hence we get that  $(f_{\lambda-1} \wedge s) \vee (f_1 \wedge \bar{s})$  is not in  $F^n$ , contradicting minimality of  $\lambda$ . ◀

For two elements  $f$  and  $g$  of  $\{0, 1\}^n$ , we write  $f \leq g$  whenever  $f(i) = 1$  implies  $g(i) = 1$  for all  $i \in [1, n]$ . Given  $B^n \subseteq \{0, 1\}^n$ , we write  $\uparrow B^n = \{g \in \{0, 1\}^n \mid \exists f \in B^n, f \leq g\}$ . A set  $F^n \subseteq \{0, 1\}^n$  is *upward-closed* if  $F^n = \uparrow F^n$ .

Fix a vector  $b \in \{0, 1\}^n$ . We define the operation  $\text{flip}_b: \{0, 1\}^n \rightarrow \{0, 1\}^n$  that maps any vector  $f$  to  $(f \wedge b) \vee (\bar{f} \wedge \bar{b})$ . In other terms,  $\text{flip}_b$  flips the  $i$ -th bit of its argument if  $b_i = 0$ , and keeps this bit unchanged if  $b_i = 1$ . In  $\text{SL}[\text{EG}]^b$ , flipping bits amounts to negating the corresponding goals. The first part of the following lemma thus indicates that our definition for  $\text{SL}[\text{EG}]^b$  is sound.

► **Lemma 18.** *If  $F^n \subseteq \{0, 1\}^n$  is semi-stable, then so is  $\text{flip}_b(F^n)$ . Moreover, for any semi-stable set  $F^n$ , there exists  $B \in \{0, 1\}^n$  such that  $\text{flip}_B(F^n)$  is upward-closed (i.e. for any  $f \in \text{flip}_B(F^n)$  and any  $s \in \{0, 1\}^n$ , we have  $f \vee s \in \text{flip}_B(F^n)$ ).*

► **Remark.** Notice that being upward-closed is not a sufficient condition for being semi-stable. For instance, the set  $F^n = \uparrow\{(0, 0, 1, 1); (1, 1, 0, 0)\}$  is not semi-stable.

**Proof.** We begin with the first statement. Assume that  $F^n$  is semi-stable, and take  $f' = \text{flip}_b(f)$  and  $g' = \text{flip}_b(g)$  in  $\text{flip}_b(F^n)$ , and  $s \in \{0, 1\}^n$ . Then

$$\begin{aligned} (f' \wedge s) \vee (g' \wedge \bar{s}) &= (((f \wedge b) \vee (\bar{f} \wedge \bar{b})) \wedge s) \vee (((g \wedge b) \vee (\bar{g} \wedge \bar{b})) \wedge \bar{s}) \\ &= (((f \wedge s) \vee (g \wedge \bar{s})) \wedge b) \vee (((\bar{f} \wedge s) \vee (\bar{g} \wedge \bar{s})) \wedge \bar{b}) \end{aligned}$$

Write  $\alpha = (f \wedge s) \vee (g \wedge \bar{s})$  and  $\beta = (\bar{f} \wedge s) \vee (\bar{g} \wedge \bar{s})$ . One can easily check that  $\beta = \bar{\alpha}$ . We then have

$$(f' \wedge s) \vee (g' \wedge \bar{s}) = (\alpha \wedge b) \vee (\bar{\alpha} \wedge \bar{b}) = \text{flip}_b(\alpha). \quad (5)$$

This computation being valid for any  $f$  and  $g$ , we also have

$$(g' \wedge s) \vee (f' \wedge \bar{s}) = (\gamma \wedge b) \vee (\bar{\gamma} \wedge \bar{b}) = \text{flip}_b(\gamma) \quad (6)$$

with  $\gamma = (g \wedge s) \vee (f \wedge \bar{s})$ . By hypothesis, at least one of  $\alpha$  and  $\gamma$  belongs to  $F^n$ , so that also at least one of  $(f' \wedge s) \vee (g' \wedge \bar{s})$  and  $(g' \wedge s) \vee (f' \wedge \bar{s})$  belongs to  $\text{flip}_b(F^n)$ .

The second statement of Lemma 18 trivially holds for  $F^n = \emptyset$ ; thus in the following, we assume  $F^n$  to be non-empty. For  $1 \leq i \leq n$ , let  $s_i \in \{0, 1\}^n$  be the vector such that  $s_i(j) = 1$  if, and only if,  $j = i$ . Applying Lemma 17, we get that for any  $i$ , there exists some  $f_i \in F^n$  such that for any  $f \in F^n$ , it holds

$$(f_i \wedge s_i) \vee (f \wedge \bar{s}_i) \in F^n. \quad (7)$$

We fix such a family  $(f_i)_{i \leq n}$  then define  $g \in \{0, 1\}^n$  as  $g = \bigvee_{1 \leq i \leq n} (f_i \wedge s_i)$ , i.e.  $g(i) = f_i(i)$  for all  $1 \leq i \leq n$ . Starting from any element of  $F^n$  and applying Equation (7) iteratively for each  $i$ , we get that  $g \in F^n$ . Since  $g \wedge s_i = f_i \wedge s_i$ , we also have

$$\forall f \in F^n \quad (g \wedge s_i) \vee (f \wedge \bar{s}_i) \in F^n$$

By Equation (6), since  $\text{flip}_g(g) = \mathbf{1}$ , we get

$$\forall f \in F^n \quad (\mathbf{1} \wedge s_i) \vee (\text{flip}_g(f) \wedge \bar{s}_i) \in \text{flip}_g(F^n). \quad (8)$$

Now, assume that  $\text{flip}_g(F^n)$  is not upward closed: then there exist elements  $f \in F^n$  and  $h \notin F^n$  such that  $\text{flip}_g(f)(i) = 1 \Rightarrow \text{flip}_g(h)(i) = 1$  for all  $i$ . Starting from  $f$  and iteratively applying Equation (8) for those  $i$  for which  $\text{flip}_g(h)(i) = 1$  and  $\text{flip}_g(f)(i) = 0$ , we get that  $\text{flip}_g(h) \in \text{flip}_g(F^n)$  and  $h \in F^n$ . Hence  $\text{flip}_g(F^n)$  must be upward closed. ◀

## C.4 Proof of Theorem 8

► **Theorem 8.** For any  $\varphi \in SL[EG]^b$  and any state  $q_0$ , it holds:  $\mathcal{G}, q_0 \models^T \varphi \Leftrightarrow \mathcal{G}, q_0 \not\models^T \neg\varphi$ . Moreover, model checking  $SL[EG]^b$  for the timeline semantics is 2-EXPTIME-complete.

**Proof.** Following Lemma 18, we assume for the rest of the proof that the set  $F^n$  of the  $SL[EG]^b$  formula  $\varphi$  is upward closed (even if it means negating some of the LTL objectives). We also assume it is non-empty, since the result is trivial otherwise.

We then have the following property:

► **Lemma 19.** Assuming  $F^n$  is upward-closed, for any  $f, g$  and  $s$  in  $\{0, 1\}^n$ , if  $f \leq g$  (i.e. for all  $i$ ,  $f(i) = 1 \Rightarrow g(i) = 1$ ), then  $f \preceq_s g$ . In particular,  $\mathbf{0}$  is a minimal element for  $\preceq_s$ , for any  $s$ .

**Proof.** Since  $f \leq g$ , then also  $(f \wedge s) \vee (h \wedge \bar{s}) \leq (g \wedge s) \vee (h \wedge \bar{s})$ , for any  $h \in \{0, 1\}^n$ . Since  $F^n$  is upward-closed, if  $(f \wedge s) \vee (h \wedge \bar{s})$  is in  $F^n$ , then so is  $(g \wedge s) \vee (h \wedge \bar{s})$ . ◀

We now develop the proof of Theorem 8. The proof is in three steps:

- we build a family of parity automata expressing the objectives that may have to be fulfilled along outcomes. A configuration is then described by a state  $q$  of the game, a vector  $d$  of states of those parity automata, and a set  $s$  of goals that are still *active* along the current outcome;
- we define formulas encoding the two ways of fulfilling a set of goals: either by fulfilling all goals along the same outcome, or by partitioning them among different branches;
- by turning the formulas above into 2-player parity games, we inductively compute optimal sets of goals (represented as vectors  $b_{q,d,s} \in \{0, 1\}^n$ ) that can be achieved from a given configuration and for each subset of active goals. By determinacy of parity games, we derive timeline maps witnessing the fact that  $b_{q,d,s}$  can be achieved, and the fact that it is optimal. If  $b_{q_0, d_0, \mathbf{1}} \in F^n$ , we get a witness map for  $\mathcal{G}, q_0 \models^T \varphi$ ; otherwise, we get one for  $\mathcal{G}, q_0 \models^T \neg\varphi$ .

### C.4.1 Automata for conjunctions of goals

We use *deterministic parity word automata* to keep track of the goals to be satisfied. Since we initially have no clue about which goal(s) will have to be fulfilled along an outcome, we use a (large) set of automata, all running in parallel.

For  $s \in \{0, 1\}^n$  and  $h \in \{0, 1\}^n$ , we let  $D_{s,h}$  be a deterministic parity automaton accepting exactly the words over  $2^{AP}$  along which the following formula  $\Phi_{s,h}$  holds:

$$\Phi_{s,h} = \bigvee_{\substack{k \in \{0,1\}^n \\ h \preceq_s k}} \bigwedge_{\substack{j \text{ s.t.} \\ (k \wedge s)(j)=1}} \varphi_j.$$

where a conjunction over an empty set (i.e., if  $(k \wedge s)(j) = 0$  for all  $j$ ) is true. Notice that in  $\Phi_{s,h}$ , we should also have imposed  $\neg\varphi_j$  for those indices  $j$  for which  $(k \wedge s)(j) = 0$ . However, using Lemma 19, if  $h \preceq_s k$  and  $k \leq k'$ , then also  $h \preceq_s k'$ , so that any conjunction containing more  $\varphi_j$ 's would also appear in  $\Phi_{s,h}$ .

As an example, take  $s \in \{0, 1\}^n$  with  $|s| = 1$  (where  $|s| = \sum_{1 \leq i \leq n} s_i$ ), writing  $j$  for the index where  $s(j) = 1$ ; for any  $h \in \{0, 1\}^n$ , if there is  $k \succeq_s h$  with  $k(j) = 0$  (which in particular is the case when  $h(j) = 0$ ), then the automaton  $D_{s,h}$  is universal; otherwise  $D_{s,h}$  accepts the set of words over  $2^{AP}$  along which  $\varphi_j$  holds.

We write  $\mathcal{D} = \{D_{s,h} \mid s \in \{0,1\}^n, h \in \{0,1\}^n\}$  for the set of automata defined above. A *vector of states of  $\mathcal{D}$*  is a function associating with each automaton  $D \in \mathcal{D}$  one of its states. We write  $\mathbf{VS}$  for the set of all vectors of states of  $\mathcal{D}$ . For any vector  $d \in \mathbf{VS}$  and any state  $q$  of  $\mathcal{G}$ , we let  $\text{succ}(d, q)$  to be the vector of states associating with each  $D \in \mathcal{D}$  the successor of state  $d(D)$  after reading  $\text{lab}(q)$ ; we extend  $\text{succ}$  to finite paths  $(q_i)_{0 \leq i \leq n}$  in  $\mathcal{G}$  inductively, letting  $\text{succ}(d, (q_i)_{0 \leq i \leq n}) = \text{succ}(\text{succ}(d, (q_i)_{0 \leq i \leq n-1}), q_n)$ .

An infinite path  $(q_i)_{i \in \mathbb{N}}$  in  $\mathcal{G}$  is accepted by an automaton  $D$  of  $\mathcal{D}$  whenever the word  $(\text{lab}(q_i))_{i \in \mathbb{N}}$  is accepted by  $D$ . We write  $\mathcal{L}(D)$  for the set of paths of  $\mathcal{G}$  accepted by  $D$ . Finally, for  $d \in \mathbf{VS}$ , we write  $\mathcal{L}(D_{s,h}^d)$  for the set of words that are accepted by  $D_{s,h}$  starting from the state  $d(D_{s,h})$  of  $D_{s,h}$ .

► **Proposition 20.** *The following holds for any  $s \in \{0,1\}^n$ :*

1.  $\Phi_{s,0} \equiv \top$  (i.e.,  $D_{s,0}$  is universal);
2. for any  $h_1, h_2 \in \{0,1\}^n$ , if  $h_1 \preceq_s h_2$ , we have  $\Phi_{s,h_2} \Rightarrow \Phi_{s,h_1}$  (i.e.,  $\mathcal{L}(D_{s,h_2}) \subseteq \mathcal{L}(D_{s,h_1})$ );
3. for any  $h \in F^n$ ,  $\Phi_{1,h} \equiv \bigvee_{k \in F^n} \bigwedge_{j \text{ s.t. } k(j)=1} \varphi_j$ .

**Proof.**  $\Phi_{s,0}$  contains the empty conjunction ( $k = \mathbf{0}$ ) as a disjunct. Hence it is equivalent to true. When  $h_1 \preceq_s h_2$ , formula  $\Phi_{s,h_1}$  contains more disjuncts than  $\Phi_{s,h_2}$ , hence the second result. Finally,  $F^n(f, \mathbf{1}) = \{0,1\}^n$  if  $f \in F^n$ , and is empty otherwise. Hence if  $h \in F^n$ , we have  $h \preceq_1 k$  if, and only if,  $k \in F^n$ , which entails the result. ◀

## C.4.2 Two ways of achieving goals

After a given history, a set of goals may be achieved either along a single outcome, in case the assignment of strategies to players gives rise to the same outcomes, or they may be split among different outcomes. We express those two ways of satisfying goals, by means of two operators parameterized by the current configuration.

The first operator covers the case where the goals currently enabled by  $s$  (those goals  $\beta_i \varphi_i$  for which  $s(i) = 1$ ) are all fulfilled along the same outcome. For any  $d \in \mathbf{VS}$  and any two  $s$  and  $h$  in  $\{0,1\}^n$ , the operator  $\Gamma_{d,s,h}^{\text{stick}}$  is defined as follows: given a context  $\chi$  with  $\mathcal{V} \subseteq \text{dom}(\chi)$  and a state  $q$  of  $\mathcal{G}$ ,

$$\mathcal{G}, q \models_{\chi} \Gamma_{d,s,h}^{\text{stick}} \Leftrightarrow \exists \rho \in \text{Play}_{\mathcal{G}}(q) \text{ s.t. } \begin{cases} - \forall j \leq n. (s(j) = 1 \Rightarrow \text{out}(q, \chi \circ \beta_j) = \rho) \\ - \rho \in \mathcal{L}(D_{s,h}^d) \end{cases}$$

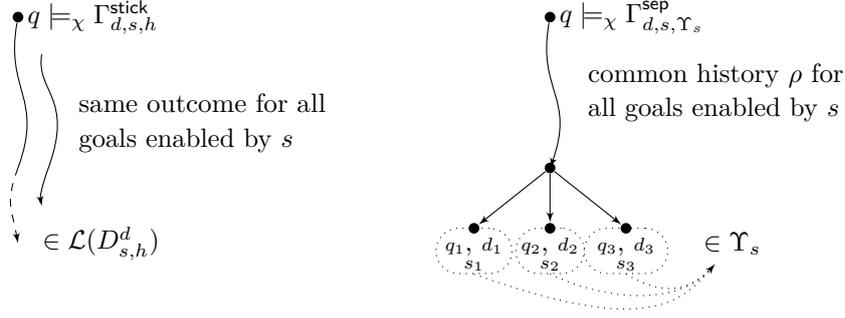
Intuitively, all the goals enabled by  $s$  must give rise to the same outcome, which is accepted by  $D_{s,h}^d$ .

We now consider the case where the active goals are partitioned among different outcomes.

► **Definition 21.** A *partition* of an element  $s \in \{0,1\}^n$  is a sequence  $(s_{\kappa})_{1 \leq \kappa \leq \lambda}$ , with  $\lambda \geq 2$ , of elements of  $\{0,1\}^n$  with  $s_1 \vee \dots \vee s_{\lambda} = s$  and where for any two  $\kappa \neq \kappa'$  and any  $1 \leq j \leq n$ , we have  $s_{\kappa}(j) = 1 \Rightarrow s_{\kappa'}(j) = 0$ .

An *extended partition* of  $s$  is a sequence  $\tau = (s_{\kappa}, q_{\kappa}, d_{\kappa})_{1 \leq \kappa \leq \lambda}$  of elements of  $\{0,1\}^n \times \mathcal{Q} \times \mathbf{VS}$  where  $(s_{\kappa})_{1 \leq \kappa \leq \lambda}$  is a partition of  $s$ ,  $q_{\kappa}$  are states of  $\mathcal{G}$ , and  $d_{\kappa}$  are vectors of states of the automata in  $\mathcal{D}$ .

We write  $\text{Part}(s)$  for the set of all extended partitions of  $s$ . Notice that we only consider non-trivial partitions; in particular, if  $|s| \leq 1$ , then  $\text{Part}(s) = \emptyset$ . For any  $d \in \mathbf{VS}$ , any  $s$  in  $\{0,1\}^n$  and any set of partitions  $\Upsilon_s$  of  $s$ , the operator  $\Gamma_{d,s,\Upsilon_s}^{\text{sep}}$  states that the goals currently



■ **Figure 9** Illustration of  $\Gamma_{d,s,h}^{\text{stick}}$  and  $\Gamma_{d,s,Y_s}^{\text{sep}}$

enabled by  $s$  all follow a common history  $\rho$  for a finite number of steps, and then partition themselves according to some partition in  $Y_s$ . The semantics of  $\Gamma_{d,s,Y_s}^{\text{sep}}$  is defined as follows:

$$\mathcal{G}, q \models_{\chi} \Gamma_{d,s,Y_s}^{\text{sep}} \Leftrightarrow \begin{cases} \exists \tau \in Y_s. \\ \exists \rho \in \text{Hist}_{\mathcal{G}}(q). \end{cases} \begin{cases} - \forall j \leq n. (s(j) = 1 \Rightarrow \rho \in \text{Pref}(\text{out}(q, \chi \circ \beta_j))) \\ - \forall \kappa \leq |\tau|. \forall j \leq n. \text{ letting } m_j(A) = \chi(\beta_j(A))(\rho). \\ (s_{\kappa}(j) = 1 \Rightarrow q_{\kappa} = \Delta(\text{last}(\rho), m_j)) \\ - \forall \kappa \leq |\tau|. \text{succ}(d, \rho \cdot q_{\kappa}) = d_{\kappa}. \end{cases}$$

Notice that  $h$  does not appear explicitly in this definition, but  $\Gamma_{d,s,Y_s}^{\text{sep}}$  will depend on  $h$  through the choice of  $Y_s$ . The operators  $\Gamma^{\text{stick}}$  and  $\Gamma^{\text{sep}}$  are illustrated on Fig. 9.

### C.4.3 Fulfilling optimal sets of goals

We now inductively (on  $|s|$ ) define new operators  $\Gamma_{d,s,h}$  combining the above two operators  $\Gamma^{\text{stick}}$  and  $\Gamma^{\text{sep}}$ , and selecting optimal ways of partitioning the goals among the outcomes.

**Base case:**  $|s| = 1$ .

When only one goal is enabled, we only have to consider a single outcome, so that we let  $\Gamma_{d,s,h} = \Gamma_{d,s,h}^{\text{stick}}$ , for any  $d \in \text{VS}$  and  $h \in \{0, 1\}^n$ . By Prop. 20, for any valuation  $\chi$  such that  $\text{Agt} \subseteq \text{dom}(\chi)$ , it holds  $\mathcal{G}, q \models_{\chi} \Gamma_{d,s,0}$ , hence also  $\mathcal{G}, q \models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,0}$ . Hence there must exist a maximal value  $b \in \{0, 1\}^n$  such that  $\mathcal{G}, q \models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,b}$ . We write  $b_{q,d,s}$  for one such value (notice that it need not be unique). By maximality, for any  $h$  such that  $b_{q,d,s} \prec_s h$ , we have  $\mathcal{G}, q \not\models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,h}$ .

**Induction step.**

We assume that for any  $d \in \text{VS}$ , any  $h \in \{0, 1\}^n$  and any  $s \in \{0, 1\}^n$  with  $|s| \leq k$ , we have defined an operator  $\Gamma_{d,s,h}$ , and that for any  $q \in \mathcal{Q}$ , we have fixed an element  $b_{q,d,s} \in \{0, 1\}^n$  for which  $\mathcal{G}, q \models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,b}$  and such that for any  $h$  such that  $b_{q,d,s} \prec_s h$ , it holds  $\mathcal{G}, q \not\models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,h}$ .

Pick  $s \in \{0, 1\}^n$  with  $|s| = k + 1$ , and an extended partition  $\tau = (s_{\kappa}, q_{\kappa}, d_{\kappa})_{1 \leq \kappa \leq \lambda}$ . Then we must have  $|s_{\kappa}| < k + 1$  for all  $1 \leq \kappa \leq \lambda$ , so that  $\Gamma_{d_{\kappa}, s_{\kappa}, h}$  and  $b_{q_{\kappa}, d_{\kappa}, s_{\kappa}}$  have been defined at previous steps. We let

$$c_{s,\tau} = \bigvee_{1 \leq \kappa \leq \lambda} (s_{\kappa} \wedge b_{q_{\kappa}, d_{\kappa}, s_{\kappa}}).$$

We then define

$$\Gamma_{d,s,h} = \Gamma_{d,s,h}^{\text{stick}} \vee \Gamma_{d,s,\Upsilon_{s,h}}^{\text{sep}} \quad \text{with } \Upsilon_{s,h} = \{\tau \in \text{Part}(s) \mid h \preceq_s c_{s,\tau}\}.$$

As previously, we claim that  $\mathcal{G}, q \models_\chi \Gamma_{d,s,\mathbf{0}}$  for any  $\chi$  such that  $\text{Agt} \subseteq \text{dom}(\chi)$ . Indeed, for a given  $\chi$ , if all the outcomes of the goals enabled by  $s$  follow the same infinite path, then this path is accepted by  $D_{s,\mathbf{0}}$  and  $\mathcal{G}, q \models_\chi \Gamma_{d,s,\mathbf{0}}^{\text{stick}}$ ; otherwise, after some common history  $\rho$ , the outcomes are partitioned following some extended partition  $\tau_0$ , which obviously satisfies  $\mathbf{0} \preceq_s c_{s,\tau_0}$  since  $\mathbf{0}$  is a minimal element of  $\preceq_s$ . Hence in that case  $\mathcal{G}, q \models_\chi \Gamma_{d,s,\Upsilon_{s,\mathbf{0}}}^{\text{sep}}$ .

In particular, it follows that  $\mathcal{G}, q \models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,\mathbf{0}}$ , and we can fix a maximal element  $b_{q,d,s}$  for which  $\mathcal{G}, q \models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,b_{q,d,s}}$  and  $\mathcal{G}, q \not\models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,h}$  for any  $h \succ_s b_{q,d,s}$ .

This concludes the inductive definition of  $\Gamma_{d,s,b_{q,d,s}}$ . We now prove that it satisfies the following lemma:

- **Lemma 22.** *For any  $q \in Q$ , any  $d \in VS$  and any  $s \in \{0,1\}^n$ ,*  
 ■ *there exists a timeline map  $\vartheta_{q,d,s}$  for  $(Q_i x_i)_{1 \leq i \leq l}$  witnessing the fact that*

$$\mathcal{G}, q \models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,b_{q,d,s}}$$

- *for any  $h \succ_s b_{q,d,s}$ , there exists a timeline map  $\bar{\vartheta}_{q,d,s,h}$  for  $(\bar{Q}_i x_i)_{1 \leq i \leq l}$  witnessing the fact that*

$$\mathcal{G}, q \models^T (\bar{Q}_i x_i)_{1 \leq i \leq l} \cdot \neg \Gamma_{d,s,h}.$$

**Proof.** The first result is a direct consequence of the construction. To prove the second part, we again turn the satisfaction of  $\Gamma_{d,s,h}$ , for  $h \succ_s b_{q,d,s}$ , into a parity game, as for the proof of Prop. 2. We only sketch the proof here, as it involves the same ingredients.

The parity game is obtained from  $\mathcal{G}$  by replacing each state by a quantification game. We also introduce two sink states,  $q_{\text{even}}$  and  $q_{\text{odd}}$ , which respectively are winning for Player  $P_\exists$  and for Player  $P_\forall$ . When arriving at a leaf  $(q, d, \mathbf{m})$  of the  $(q, d)$ -copy of the quantification game, there may be one of the following three transitions available:

- if there is a state  $q'$  such that for all  $j$  with  $s(j) = 1$ , it holds  $q' = \Delta(q, m_{\beta_j})$  (in other terms, the moves selected in the current quantification game generate the same transition for all the goals enabled by  $s$ ), then there is a single transition to  $(q', d', \varepsilon)$ , where  $d' = \text{succ}(d, q')$ .
- otherwise, if there is an extended partition  $\tau = (s_\kappa, q_\kappa, d_\kappa)_{1 \leq \kappa \leq \lambda}$  of  $s$  such that  $c_{s,\tau} \succeq_s h$  and, for all  $1 \leq \kappa \leq \lambda$ , for all  $j$  such that  $s_\kappa(j) = 1$ , we have  $\Delta(q, m_{\beta_j}) = q_\kappa$  and  $\text{succ}(d, q_\kappa) = d_\kappa$ , then there is a transition from  $(q, d, \mathbf{m})$  to  $q_{\text{even}}$ .
- otherwise, there is a transition from  $(q, d, \mathbf{m})$  to  $q_{\text{odd}}$ .

The priorities defining the parity condition are inherited from those in  $D_{s,h}$ .

Since  $\mathcal{G}, q \not\models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,h}$ , Player  $P_\exists$  does not have a winning strategy in this game, and by determinacy Player  $P_\forall$  has one. From the winning strategy of Player  $P_\forall$ , we obtain a timeline map  $\bar{\vartheta}_{q,d,s,h}$  for  $(\bar{Q}_i x_i)_{1 \leq i \leq l}$  witnessing the fact that  $\mathcal{G}, q \models^T (\bar{Q}_i x_i)_{1 \leq i \leq l} \cdot \neg \Gamma_{d,s,h}$ . ◀

► **Remark.** While the definition of  $\Gamma_{d,s,b_{q,d,s}}$  (and in particular of  $b_{q,d,s}$ ) is not effective, the parity games defined in the proof above can be used to compute each  $b_{q,d,s}$  and  $\Gamma_{d,s,b_{q,d,s}}$ . Indeed, such parity games can be used to decide whether  $\mathcal{G}, q \models^T (Q_i x_i)_{1 \leq i \leq l} \cdot \Gamma_{d,s,h}$  for all  $h$ , and selecting a maximal value for which the result holds. Then  $b_{q_0,d_0,1} \in F^n$  implies  $\mathcal{G}, q_0 \models^T (Q_i x_i)_{1 \leq i \leq l} F^n(\beta_j \varphi_j)_{1 \leq j \leq n}$ .

Each parity game has size doubly-exponential, with exponentially-many priorities; hence they can be solved in  $\mathcal{L}$ -EXPTIME. The number of games to solve is also doubly-exponential, so that the whole algorithm runs in  $\mathcal{L}$ -EXPTIME.

We now focus on the operator obtained at the end of the induction, when  $s = 1$ . Following Prop. 20,  $\mathcal{L}(D_{\mathbf{1},f})$  does not depend on the exact value of  $f$ , as soon as it is in  $F^n$ . We then let

$$\Gamma_{F^n} = \Gamma_{d_0, \mathbf{1}, f}^{\text{stick}} \vee \Gamma_{d_0, \mathbf{1}, \Upsilon_{F^n}}^{\text{sep}}$$

where  $f$  is any element of  $F^n$  (remember  $F^n$  is assumed to be non-empty),  $d_0$  is the vector of initial states of the automata in  $\mathcal{D}$ , and  $\Upsilon_{F^n} = \{\text{Part}(\mathbf{1}) \mid c_{\mathbf{1}, \tau} \in F^n\}$ . We write  $\vartheta_{\mathbf{1}}$  and  $\bar{\vartheta}_{\mathbf{1}}$  for the maps  $\vartheta_{q_0, d_0, \mathbf{1}}$  and  $\bar{\vartheta}_{q_0, d_0, \mathbf{1}, h}$  for some  $h \in F^n$ , as given by Lemma 22. From the discussion above,  $\bar{\vartheta}_{q_0, d_0, \mathbf{1}, h}$  does not depend on the choice of  $h$  in  $F^n$ , and we simply write it  $\bar{\vartheta}_{q_0, d_0, \mathbf{1}}$ .

Then:

► **Lemma 23.** *If  $\mathcal{G}, q_0 \models^T (Q_i x_i)_{1 \leq i \leq l}. \Gamma_{F^n}$ , then  $\vartheta_{\mathbf{1}}$  witnesses that  $\mathcal{G}, q_0 \models^T (Q_i x_i)_{1 \leq i \leq l}. \Gamma_{F^n}$ . Conversely, if  $\mathcal{G}, q_0 \not\models^T (Q_i x_i)_{1 \leq i \leq l}. \Gamma_{F^n}$ , then  $\bar{\vartheta}_{\mathbf{1}}$  witness that  $\mathcal{G}, q_0 \models^T (\bar{Q}_i x_i)_{1 \leq i \leq l}. \neg \Gamma_{F^n}$ .*

**Proof.** The first part directly follows from the previous lemma. For the second part,  $\mathcal{G}, q_0 \not\models^T (Q_i x_i)_{1 \leq i \leq l}. \Gamma_{F^n}$  means that  $b_{q_0, d_0, \mathbf{1}} \notin F^n$ . Hence for any  $f \in F^n$ , we have  $f \succ_s b_{q_0, d_0, \mathbf{1}}$ , so that  $\bar{\vartheta}_{q_0, d_0, \mathbf{1}}$  is a witness that  $\mathcal{G}, q \models^T (\bar{Q}_i x_i)_{1 \leq i \leq l}. \neg \Gamma_{F^n}$ . ◀

#### C.4.4 Compiling optimal maps

From Lemma 22, we have timeline maps for each  $q, d$  and  $s$ . We now compile them into two map  $\theta$  and  $\bar{\theta}$ . The construction is inductive, along histories.

Pick a history  $\rho$  starting from  $q_0$  and strategies for universally-quantified variables  $w: \mathcal{V}^\forall \rightarrow (\text{Hist} \rightarrow \text{Act})$ . Assuming  $\theta$  has been defined along all strict prefixes of  $\rho$ , a goal  $\beta_j \varphi_j$  is said *active* after  $\rho$  w.r.t.  $\theta(w)$  if the following condition holds:

$$\forall i < |\rho|. \rho(i+1) = \Delta(\rho(i), (\theta(w)(\beta_j(A))(\rho_{\leq i}))_{A \in \text{Agt}}).$$

In other terms,  $\beta_j \varphi_j$  is active after  $\rho$  w.r.t.  $\theta(w)$  if  $\rho$  is the outcome of strategies prescribed by  $\theta(w)$  under assignment  $\beta_j$ . We let  $s_{\rho, \theta(w)}$  be the element of  $\{0, 1\}^n$  such that  $s_{\rho, \theta(w)}(j) = 1$  if, and only if,  $\beta_j \varphi_j$  is active after  $\rho$  w.r.t.  $\theta(w)$ .

We now define  $\theta(w)(x_i)(\rho)$  for all  $x_i \in \mathcal{V}$ :

- if  $x_i \in \mathcal{V}^\forall$ , we let  $\theta(w)(x_i)(\rho) = w(x_i)(\rho)$ ;
- if  $x_i \in \mathcal{V}^\exists$ , we consider two cases:
  - if  $s_{\rho, \theta(w)} = 1$ , then all goals are still active, and  $\theta$  follows the map  $\vartheta_{\mathbf{1}}$ :  $\theta(w)(x_i)(\rho) = \vartheta_{\mathbf{1}}(w)(x_i)(\rho)$ .
  - otherwise, we let  $\rho_1$  be the maximal prefix of  $\rho$  for which  $s_{\rho_1, \theta(w)} \neq s_{\rho, \theta(w)}$ . We may then write  $\rho = \rho_1 \cdot \rho_2$ , and let  $q_1 = \text{last}(\rho_1)$  and  $d_1 = \text{succ}(d_0, \rho_1)$ . We then let  $\theta(w)(x_i)(\rho) = \vartheta_{q_1, d_1, s_{\rho, \theta(w)}}(w_{\bar{\rho}_1})(x_i)(\rho_2)$ .

The dual map  $\bar{\theta}$  is defined in the same way, using maps  $\bar{\vartheta}$  in place of  $\vartheta$ .

The following result will conclude our proof of Theorem 8.

► **Lemma 24.** *There exists a valuation  $\chi$  of domain  $\mathcal{V}$  such that  $\theta(\chi|_{\mathcal{V}^\forall}) = \chi$  and  $\bar{\theta}(\chi|_{\mathcal{V}^\exists}) = \chi$ . It satisfies*

$$\begin{aligned} \mathcal{G}, q_0 \models_\chi \Gamma_{F^n} &\Rightarrow \forall w \in (\text{Hist}_{\mathcal{G}} \rightarrow \text{Act})^{\mathcal{V}^\forall}. \mathcal{G}, q_0 \models_{\theta(w)} F^n(\beta_j \varphi_j)_{1 \leq j \leq n} \\ \mathcal{G}, q_0 \models_\chi \neg \Gamma_{F^n} &\Rightarrow \forall \bar{w} \in (\text{Hist}_{\mathcal{G}} \rightarrow \text{Act})^{\mathcal{V}^\exists}. \mathcal{G}, q_0 \models_{\bar{\theta}(\bar{w})} \bar{F}^n(\beta_j \varphi_j)_{1 \leq j \leq n} \end{aligned}$$

**Proof.** We use the same technique as in the proof of Prop. 6 (see Appendix B.1): from  $\theta$  and  $\bar{\theta}$ , we build a strategy valuation  $\chi$  on  $\mathcal{V}$  such that  $\theta(\chi|_{\mathcal{V}^\forall}) = \chi$  and  $\bar{\theta}(\chi|_{\mathcal{V}^\exists}) = \chi$ .

We introduce some more notations. For  $w: \mathcal{V}^\forall \rightarrow (\text{Hist}_{\mathcal{G}} \rightarrow \text{Act})$ , we let

- $\pi_j^w$  be the outcome  $\text{out}(q_0, (\theta(w)((\beta_j(A))_{A \in \text{Agt}})))$  for all  $1 \leq j \leq n$ ;
- $f^w$  be the element of  $\{0, 1\}^n$  such that  $f^w(j) = 1$  if, and only if,  $\pi_j^w \models \varphi_j$ ;
- $R^w \subseteq \{0, 1\}^n \times \text{Hist}_{\mathcal{G}}$  be the relation such that  $(s, \rho) \in R^w$  if, and only if,  $s = s_{\rho, \theta(w)}$  and  $\rho$  is minimal (meaning for any strict prefix  $\rho'$  of  $\rho$ , it holds  $(s, \rho') \notin R^w$ ).

► **Lemma 25.** *For any  $w: \mathcal{V}^\forall \rightarrow (\text{Hist}_{\mathcal{G}} \rightarrow \text{Act})$  and any  $\rho \in \text{Hist}$ , letting  $d_\rho = \text{succ}(d_0, \rho)$ , it holds*

$$\forall s \in \{0, 1\}^n. (s, \rho) \in R^w \Rightarrow b_{\text{last}(\rho), d_\rho, s} \preceq_s f^w.$$

**Proof.** Fix some  $w \in (\text{Hist}_{\mathcal{G}} \rightarrow \text{Act})^{\mathcal{V}^\forall}$ . The proof proceeds by induction on  $|s|$ .

**Base case:**  $|s| = 1$ . Assume  $(s, \rho) \in R^w$ . As  $|s| = 1$ , there is a unique goal, say  $\beta_{j_0} \varphi_{j_0}$ , active along  $\rho$  w.r.t.  $\theta(w)$ . By definition of  $\theta$ ,  $\pi_{j_0} = \rho \cdot \eta$  where  $\eta$  is the outcome of  $\vartheta_{\text{last}(\rho), d_\rho, s}(w \vec{\rho})((\beta_j(A))_{A \in \text{Agt}})$  from  $\text{last}(\rho)$ .

Because  $|s| = 1$ , we have  $\Gamma_{d_\rho, s, b_{\text{last}(\rho), d_\rho, s}} = \Gamma_{d_\rho, s, b_{\text{last}(\rho), d_\rho, s}}^{\text{stick}}$ . The map  $\vartheta_{\text{last}(\rho), d_\rho, s}$  is a witness that  $\mathcal{G}, \text{last}(\rho) \models^T (Q_i x_i)_{1 \leq i \leq l} \Gamma_{d_\rho, s, b_{\text{last}(\rho), d_\rho, s}}$ ; therefore it also witnesses that  $\mathcal{G}, \text{last}(\rho) \models^T (Q_i x_i)_{1 \leq i \leq l} \Gamma_{d_\rho, s, b_{\text{last}(\rho), d_\rho, s}}^{\text{stick}}$ . By definition of the  $\Gamma^{\text{stick}}$  operators, this implies that for any  $w$ , the outcome of  $\vartheta_{\text{last}(\rho), d_\rho, s}(w \vec{\rho})$  from  $\text{last}(\rho)$  is accepted by the automaton  $D_{s, b_{\text{last}(\rho), d_\rho, s}}^{d_\rho}$ ; in particular,  $\eta$  is accepted by  $D_{s, b_{\text{last}(\rho), d_\rho, s}}^{d_\rho}$ .

The automaton  $D_{s, b_{\text{last}(\rho), d_\rho, s}}^{d_\rho}$  accepts paths which give better results (w.r.t.  $\preceq_s$ ) for the objectives  $(\beta_j \varphi_j)_{j|s(j)=1}$  than  $b_{\text{last}(\rho), d_\rho, s}$ . In other terms, we have  $b_{\text{last}(\rho), d_\rho, s} \preceq_s f^w$ .

**Induction step.** We assume that the Proposition 25 holds for any elements  $s \in \{0, 1\}^n$  of size  $|s| < \alpha$ . We now consider for the induction step an element  $s \in \{0, 1\}^n$  such that  $|s| = \alpha$  and  $(s, \rho) \in R^w$ .

- if the enabled goals all follow the same outcome, i.e., if there exists an infinite path  $\eta$  such that  $\pi_j = \rho \cdot \eta$  for all  $j$  having  $s(j) = 1$ , then with arguments similar to those of the base case, we get  $b_{\text{last}(\rho), d_\rho, s} \preceq_s f^w$ .
- otherwise, the goals enabled by  $s$  split following an extended partition  $\tau = (s_\kappa, q_\kappa, d_\kappa)_{\kappa \leq \lambda}$ . We let  $\eta$  be the history from the last state of  $\rho$  to the point where the goals split. The map  $\vartheta_{\text{last}(\rho), d_\rho, s}$  witnesses that  $\mathcal{G}, \text{last}(\rho) \models^T \Gamma_{d_\rho, s, b_{\text{last}(\rho), d_\rho, s}}$ ; therefore  $\eta$  may only reach a partition  $\tau$  such that

$$b_{\text{last}(\rho), d_\rho, s} \preceq_s c_{s, \tau} \tag{9}$$

This partition  $\tau$  is such that for any  $1 \leq \kappa \leq \lambda$ , it holds  $(s_\kappa, \rho \cdot \eta \cdot q_\kappa) \in R^w$ ; using the induction hypothesis, we get

$$s_\kappa \wedge b_{q_\kappa, d_\kappa, s_\kappa} \preceq_{s_\kappa} f^w \tag{10}$$

Then, using Lemma 15 (see Appendix C.2) repeatedly on the  $(s_\kappa)_{1 \leq \kappa \leq \lambda}$ , and Equation (10), we obtain

$$\begin{aligned} s_1 \wedge b_{q_1, d_1, s_1} \preceq_{s_1} f^w &\Rightarrow (s_1 \wedge b_{q_1, d_1, s_1}) \Upsilon (s_2 \wedge b_{q_2, d_2, s_2}) \preceq_{s_1 \Upsilon s_2} f^w \\ &\Rightarrow \dots \\ &\Rightarrow (s_1 \wedge b_{q_1, d_1, s_1}) \Upsilon \dots \Upsilon (s_\lambda \wedge b_{q_\lambda, d_\lambda, s_\lambda}) \preceq_{s_1 \Upsilon \dots \Upsilon s_\lambda} f^w \\ &\Rightarrow c_{s, \tau} \preceq_s f^w. \end{aligned}$$

Combined with (9), we get  $b_{\text{last}(\rho), d_\rho, s} \preceq_s c_{s, \tau} \preceq_s f^w$ .  $\blacktriangleleft$

► **Lemma 26.**  $\mathcal{G}, q_0 \models_\chi \Gamma_{F^n}$  if, and only if,  $b_{q_0, d_0, 1} \in F^n$ .

**Proof.** Assume that  $b_{q_0, d_0, 1} \in \overline{F^n}$ . Then  $\mathcal{G}, q_0 \not\models^T (Q_i x_i)_{1 \leq i \leq l} . \Gamma_{F^n}$ . Applying Lemma 23, the map  $\overline{\vartheta}_1$  (and therefore  $\overline{\theta}$ , which act as  $\overline{\vartheta}_1$  before goals branch along different paths) witnesses  $\mathcal{G}, q_0 \not\models^T (Q_i x_i)_{1 \leq i \leq l} . \Gamma_{F^n}$ . This implies that  $\mathcal{G}, q_0 \not\models_\chi \Gamma_{F^n}$ , which contradicts the hypothesis.

Conversely, if  $b_{q_0, d_0, 1} \in F^n$ , then  $\mathcal{G}, q_0 \models^T (Q_i x_i)_{1 \leq i \leq l} . \Gamma_{F^n}$ , which is witnessed by map  $\vartheta_1$ . Thus  $\mathcal{G}, q_0 \models_\chi \Gamma_{F^n}$ .  $\blacktriangleleft$

We are now ready to prove the first part of Lemma 24: consider a function  $w: \mathcal{V}^\forall \rightarrow (\text{Hist}_G \rightarrow \text{Act})$ . By Lemma 25 applied to  $w$ ,  $s = 1$ , and  $\rho = q_0$ , we get that  $b_{q_0, d_0, 1} \preceq_1 f^w$ . Now, by Lemma 26,  $b_{q_0, d_0, 1} \in F^n$ , therefore the element  $f^w$ , being greater than  $b_{q_0, d_0, 1}$  for  $\preceq_1$ , must also be in  $F^n$ , which means that  $\mathcal{G}, q_0 \models_{\theta(w)} F^n(\beta_j \varphi_j)_{1 \leq j \leq n}$ .

The second implication of the lemma is proven using similar arguments.  $\blacktriangleleft$

Lemma 24 allows us to conclude that at least one of  $\varphi$  and  $\neg\varphi$  must hold on  $\mathcal{G}$  for  $\models^T$ . Lemma 6 implies that at most one can hold. Combining both we get that exactly one holds.

## C.5 Proof of Proposition 11

► **Proposition 11.** For any non-semi-stable boolean set  $F^n \subseteq \{0, 1\}^n$ , there exists a  $SL[BG]^\sharp$  formula  $\varphi$  built on  $F^n$ , a game  $\mathcal{G}$  and a state  $q_0$  such that  $\mathcal{G}, q_0 \not\models^T \neg\varphi$  and  $\mathcal{G}, q_0 \models^T \varphi$ .

**Proof.** We consider the game  $\mathcal{G}$  depicted on Figure 10 with two agents  $\blacksquare$  and  $\bullet$ . Let  $F^n$  be a non-semi-stable set over  $\{0, 1\}^n$ . Then there must exist  $f_1, f_2 \in F^n$ , and  $s \in \{0, 1\}^n$ , such that  $(f_1 \wedge s) \vee (f_2 \wedge \bar{s}) \notin F^n$  and  $(f_2 \wedge s) \vee (f_1 \wedge \bar{s}) \notin F^n$ . We then let

$$\varphi = \forall y_t^{\blacksquare} . \forall y_u^{\blacksquare} . \forall x_t^{\bullet} . \exists x_u^{\bullet} . F^n(\beta_1 \varphi_1, \dots, \beta_n \varphi_n)$$

where

$$\beta_i = \begin{cases} \text{assign}(\blacksquare \mapsto y_t^{\blacksquare}; \bullet \mapsto x_t^{\bullet}) & \text{if } s(i) = 1 \\ \text{assign}(\blacksquare \mapsto y_u^{\blacksquare}; \bullet \mapsto x_u^{\bullet}) & \text{if } s(i) = 0 \end{cases}$$

and

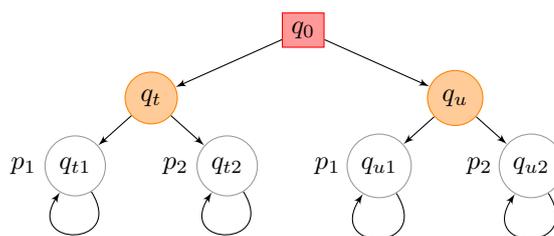
$$\varphi_i = \begin{cases} \mathbf{F} p_1 \vee \mathbf{F} p_2 & \text{if } f_1(i) = f_2(i) = 1 \\ \mathbf{F} p_1 & \text{if } f_1(i) = 1 \text{ and } f_2(i) = 0 \\ \mathbf{F} p_2 & \text{if } f_1(i) = 0 \text{ and } f_2(i) = 1 \\ \mathbf{false} & \text{if } f_1(i) = f_2(i) = 0 \end{cases}$$

It is not hard to check that the following holds:

► **Lemma 27.** Let  $\rho$  be a maximal run of  $\mathcal{G}$  from  $q_0$ . Let  $k \in \{1, 2\}$  be such that  $\rho$  visits a state labelled with  $p_k$ . Then for any  $1 \leq i \leq n$ , we have  $\rho \models \varphi_i$  if, and only if,  $f_k(i) = 1$ .

The following two lemmas conclude the proof:

► **Lemma 28.**  $\mathcal{G}, q_0 \not\models^T \varphi$



■ **Figure 10** The two-agents turn-based game  $\mathcal{G}$

**Proof.** Towards a contradiction, assume that  $\mathcal{G}, q_0 \models^T \varphi$ . We let  $\sigma_t$  (resp.  $\sigma_u$ ) be the strategy that maps history  $q_0$  to  $q_t$  (resp.  $q_u$ ). We fix strategy  $\tau_t$  such that  $\tau_t(q_0 \cdot q_t) = q_{t1}$ . There is a strategy  $\tau_u$  (with local and timeline dependences) such that

$$\mathcal{G}, q_0 \models_{\chi} F^n(\beta_1\varphi_1, \dots, \beta_n\varphi_n)$$

where  $\chi$  maps  $y_t^{\square}$  to  $\sigma_t$ ,  $y_u^{\circ}$  to  $\sigma_u$ ,  $x_t^{\circ}$  to  $\tau_t$  and  $x_u^{\circ}$  to  $\tau_u$ .

Since  $x_u^{\circ}$  is only jointly applied with  $y_u^{\square}$ , the only important information about  $\tau_u$  is its value on history  $q_0\sigma_u(q_0) = q_0q_u$ . This value is then independent on the value of  $\tau_t(q_0q_t) = \tau_t(q_0\sigma_t(q_0))$ . In particular, writing  $\chi'$  for the context obtained from  $\chi$  by replacing  $\chi(y_t^{\square}) = \tau_t$  with  $\tau'_t$ , where  $\tau'_t(q_0q_t) = q_{t2}$ , we also have

$$\mathcal{G}, q_0 \models_{\chi'} F^n(\beta_1\varphi_1, \dots, \beta_n\varphi_n)$$

Let  $v$  and  $v'$  be the vectors in  $\{0, 1\}^n$  representing the values of the goals  $(\beta_1\varphi_1, \dots, \beta_n\varphi_n)$  under  $\chi$  and  $\chi'$ . Then  $v$  and  $v'$  are in  $F^n$ . However:

- If  $\tau_u(q_0q_u) = q_{u1}$ , then  $v' = (f_1 \wedge \bar{s}) \vee (f_2 \wedge s)$ .
- If  $\tau_t(q_0q_t) = q_{t2}$ , then  $v = (f_1 \wedge s) \vee (f_2 \wedge \bar{s})$ .

In both cases, by hypothesis, this does not belong to  $F^n$ , which is a contradiction. ◀

► **Lemma 29.**  $\mathcal{G}, q_0 \not\models^T \neg\varphi$

**Proof.** Similarly, assume  $\mathcal{G}, q_0 \models^T \neg\varphi$ . Fix any three strategies  $\sigma_t$ ,  $\sigma_u$  and  $\tau_t$  respectively intended for the existentially quantified variables  $y_t^{\square}$ ,  $y_u^{\square}$  and  $x_t^{\circ}$ . Due to the nature of  $\models^T$ , these three strategies are independent from the strategy  $\tau_u$  of  $x_u^{\circ}$ . Consider then the following strategy  $\tau_u$ :

$$\tau_u(q_0 \cdot \sigma_u(q_0)) = \tau_t(q_0 \cdot \sigma_t(q_0))$$

Let  $\chi$  be the resulting context and  $v$  the vector representing the values of the goals  $(\beta_1\varphi_1, \dots, \beta_n\varphi_n)$  under  $\chi$ . Either  $v = f_1$  or  $v = f_2$ ; in both case  $v \in F^n$ , which is a contradiction. ◀