Abstract

In this paper, we study the framework of two-player Stackelberg games played on graphs in which Player 0 announces a strategy and Player 1 responds rationally with a strategy that is an optimal response. While it is usually assumed that Player 1 has a single objective, we consider here the new setting where he has several. In this context, after responding with his strategy, Player 1 gets a payoff in the form of a vector of Booleans corresponding to his satisfied objectives. Rationality of Player 1 is encoded by the fact that his response must produce a Pareto-optimal payoff given the strategy of Player 0. We study the Stackelberg-Pareto Synthesis problem which asks whether Player 0 can announce a strategy which satisfies his objective, whatever the rational response of Player 1. For games in which objectives are either all parity or all reachability objectives, we show that this problem is fixed-parameter tractable and \textsc{NEXPTIME}-complete. This problem is already \textsc{NP}-complete in the simple case of reachability objectives and graphs that are trees.

1 Introduction

Two-player zero-sum infinite-duration games played on graphs are a mathematical model used to formalize several important problems in computer science, such as reactive system synthesis. In this context, see e.g. [29], the graph represents the possible interactions between the system and the environment in which it operates. One player models the system to synthesize, and the other player models the (uncontrollable) environment. In this classical setting, the objectives of the two players are opposite, that is, the environment is adversarial. Modelling the environment as fully adversarial is usually a bold abstraction of reality as it can be composed of one or several components, each of them having their own objective.

In this paper, we consider the framework of Stackelberg games [34], a richer non-zero-sum setting, in which Player 0 (the system) called leader announces his strategy and then Player 1 (the environment) called follower plays rationally by using a strategy that is an optimal response to the leader’s strategy. This framework captures the fact that in practical applications, a strategy for interacting with the environment is committed before the interaction actually happens. The goal of the leader is to announce a strategy that guarantees him a payoff at least equal to some given threshold. In the specific case of Boolean objectives, the leader wants to see his objective being satisfied. The concept of leader and follower is also present in the framework of rational synthesis [19, 27] with the difference that
this framework considers several followers, each of them with their own Boolean objective. In that case, rationality of the followers is modeled by assuming that the environment settles to an equilibrium (e.g. a Nash equilibrium) where each component (composing the environment) is considered to be an independent selfish individual, excluding cooperation scenarios between components or the possibility of coordinated rational multiple deviations. Our work proposes a novel and natural alternative in which the single follower, modeling the environment, has several objectives that he wants to satisfy. After responding to the leader with his own strategy, Player 1 receives a vector of Booleans which is his payoff in the corresponding outcome. Rationality of Player 1 is encoded by the fact that he only responds in such a way to receive Pareto-optimal payoffs, given the strategy announced by the leader. This setting encompasses scenarios where, for instance, several components can collaborate and agree on trade-offs. The goal of the leader is therefore to announce a strategy that guarantees him to satisfy his own objective, whatever the response of the follower which ensures him a Pareto-optimal payoff. The problem of deciding whether the leader has such a strategy is called the Stackelberg-Pareto Synthesis problem (SPS problem).

Contributions. In addition to the definition of the new setting, our main contributions are the following ones. We consider the general class of $\omega$-regular objectives modelled by parity conditions and also consider the case of reachability objectives for their simplicity.\footnote{Indeed, in the classical context of two-player zero-sum games, solving reachability games is in P whereas solving parity games is only known to be in NP $\cap$ co-NP, see e.g. \cite{20}.} We provide a thorough analysis of the complexity of solving the SPS problem for both objectives. Our results are interesting and singular both from a theoretical and practical point of view.

First, we show that the SPS problem is fixed-parameter tractable (FPT) for reachability objectives when the number of objectives of the follower is a parameter (Theorem 3) and for parity objectives when, in addition, the maximal priority used in each priority function is also a parameter of the complexity analysis (Theorem 4). These are important results as it is expected that, in practice, the number of objectives of the environment is limited to a few. To obtain these results, we develop a reduction from our non-zero-sum games to a zero-sum game in which the protagonist, called Prover, tries to show the existence of a solution to the problem, while the antagonist, called Challenger, tries to disprove it. This zero-sum game is defined in a generic way, independently of the actual objectives used in the initial game, and can then be easily adapted according to the case of reachability or parity objectives.

Second, we prove that the SPS problem is NEXPTIME-complete for both reachability and parity objectives (Theorems 9, 17, and 18), and that it is already NP-complete in the simple setting of reachability objectives and graphs that are trees (Theorem 15). To the best of our knowledge, this is the first NEXPTIME-completeness result for a natural class of games played on graphs. To obtain the hardness for NEXPTIME, we present a natural succinct version of the set cover problem that is complete for this class (Theorem 20), a result of potential independent interest. We then show how to reduce this problem to the SPS problem. To obtain the NEXPTIME-membership of the SPS problem, we have shown that exponential-size solutions exist for positive instances of the SPS problem and this allows us to design a nondeterministic exponential-time algorithm. Unfortunately, it was not possible to use the FPT algorithm mentioned above to show this membership due to its too high time complexity; conversely, our NEXPTIME algorithm is not FPT.
Related Work. Rational synthesis is introduced in [19] for \( \omega \)-regular objectives in a setting where the followers are cooperative with the leader, and later in [27] where they are adversarial. Precise complexity results for various \( \omega \)-regular objectives are established in [13] for both settings. Those complexities differ from the ones of the problem studied in this paper. Indeed, for reachability objectives, adversarial rational synthesis is \( \text{PSPACE} \)-complete, while for parity objectives, its precise complexity is not settled (the problem is \( \text{PSPACE} \)-hard and in \( \text{NEXPTIME} \)). Extension to non-Boolean payoffs, like mean-payoff or discounted sum, is studied in [21, 22] in the cooperative setting and in [1, 18] in the adversarial setting.

When several players (like the followers) play with the aim to satisfy their objectives, several solution concepts exist such as Nash equilibrium [28], subgame perfect equilibrium [30], secure equilibria [11, 12], or admissibility [2, 5]. The constrained existence problem, close to the cooperative rational synthesis problem, is to decide whether there exists a solution concept such that the payoff obtained by each player is larger than some threshold. Let us mention [13, 32, 33] for results on the constrained existence for Nash equilibria and [6, 7, 31] for such results for subgame perfect equilibria. Rational verification is studied in [23, 24]. This problem (which is not a synthesis problem) is to decide whether a given LTL formula is satisfied by the outcome of all Nash equilibria (resp. some Nash equilibrium). The interested reader can find more pointers to works on non-zero-sum games for reactive synthesis in [4, 8].

Structure. The paper is structured as follows. In Section 2, we introduce the class of Stackelberg-Pareto games and the SPS problem. We show in Section 3 that the SPS problem is in \( \text{FPT} \) for reachability and parity objectives. The complexity class of this problem is studied in Section 4 where we prove that it is \( \text{NEXPTIME} \)-complete and \( \text{NP} \)-complete in case of reachability objectives and graphs that are trees. In Section 5 we provide a conclusion and discuss future work.

2 Preliminaries and Stackelberg-Pareto Synthesis Problem

This section introduces the class of two-player Stackelberg-Pareto games in which the first player has a single objective and the second has several. We present a decision problem on those games called the Stackelberg-Pareto Synthesis problem, which we study in this paper.

2.1 Preliminaries

Game Arena. A game arena is a tuple \( G = (V, V_0, V_1, E, v_0) \) where \( (V, E) \) is a finite directed graph such that: (i) \( V \) is the set of vertices and \( (V_0, V_1) \) forms a partition of \( V \) where \( V_0 \) (resp. \( V_1 \)) is the set of vertices controlled by Player 0 (resp. Player 1), (ii) \( E \subseteq V \times V \) is the set of edges such that each vertex \( v \) has at least one successor \( v' \), i.e., \( (v, v') \in E \), and (iii) \( v_0 \in V \) is the initial vertex. We call a game arena a tree arena if it is a tree in which every leaf vertex has itself as its only successor. A sub-arena \( G' \) with a set \( V' \subseteq V \) of vertices and initial vertex \( v'_0 \in V' \) is a game arena defined from \( G \) as expected.

Plays. A play in a game arena \( G \) is an infinite sequence of vertices \( p = v_0v_1\ldots \in V^\omega \) such that it starts with the initial vertex \( v_0 \) and \( (v_j, v_{j+1}) \in E \) for all \( j \in \mathbb{N} \). Histories in \( G \) are finite sequences \( h = v_0\ldots v_j \in V^+ \) defined similarly. A history is elementary if it contains no cycles. We denote by \( \text{Plays}_G \) the set of plays in \( G \). We write \( \text{Hist}_G \) (resp. \( \text{Hist}_{G,i} \)) the set of histories (resp. histories ending with a vertex in \( V_i \)). We use the notations \( \text{Plays}, \text{Hist}, \) and \( \text{Hist} \), when \( G \) is clear from the context. We write \( \text{Occ}(\rho) \) the set of vertices occurring in \( \rho \) and \( \text{Inf}(\rho) \) the set of vertices occurring infinitely often in \( \rho \).
4 Stackelberg-Pareto Synthesis (full version)

Strategies. A strategy $\sigma_i$ for Player $i$ is a function $\sigma_i : \text{Hist} \rightarrow V$ assigning to each history $h \in \text{Hist}$, a vertex $v' = \sigma_i(hv)$ such that $(v, v') \in E$. It is memoryless if $\sigma_i(hv) = \sigma_i(h'v)$ for all histories $hv, h'v$ ending with the same vertex $v \in V$. More generally, it is finite-memory if it can be encoded by a Moore machine $M$ [20]. The memory size of $\sigma_i$ is the number of memory states of $M$. In particular, $\sigma_i$ is memoryless when it has a memory size of one.

Given a strategy $\sigma_i$ of Player $i$, a play $\rho = v_0v_1 \ldots$ is consistent with $\sigma_i$ if $v_{j+1} = \sigma_i(v_0 \ldots v_j)$ for all $j \in \mathbb{N}$ such that $v_j \in V_i$. Consistency is naturally extended to histories. We denote by $\text{Plays}_{\sigma_i}$ (resp. $\text{Hist}_{\sigma_i}$) the set of plays (resp. histories) consistent with $\sigma_i$. A strategy profile is a tuple $\sigma = (\sigma_0, \sigma_1)$ of strategies, one for each player. We write $\text{out}(\sigma)$ the unique play consistent with both strategies and we call it the outcome of $\sigma$.

Objectives. An objective for Player $i$ is a set of plays $\Omega \subseteq \text{Plays}$. A play $\rho$ satisfies the objective $\Omega$ if $\rho \in \Omega$. In this paper, we focus on the two following $\omega$-regular objectives. Let $T \subseteq V$ be a subset of vertices called a target set, the reachability objective $\text{Reach}(T) = \{ \rho \in \text{Plays} \mid \text{Occ}(\rho) \cap T \neq \emptyset \}$ asks to visit at least one vertex of $T$. Let $c : V \rightarrow \mathbb{N}$ be a function called a priority function which assigns an integer to each vertex in the arena, the parity objective $\text{Parity}(c) = \{ \rho \in \text{Plays} \mid \min_{v \in \text{Inf}(\rho)}(c(v)) \text{ is even} \}$ asks that the minimum priority visited infinitely often be even.

2.2 Stackelberg-Pareto Synthesis Problem

Stackelberg-Pareto Games. A Stackelberg-Pareto game (SP game) $G = (G, \Omega_0, \Omega_1, \ldots, \Omega_t)$ is composed of a game arena $G$, an objective $\Omega_0$ for Player 0 and $t \geq 1$ objectives $\Omega_1, \ldots, \Omega_t$ for Player 1. In this paper, we focus on SP games where the objectives are either all reachability or all parity objectives and call such games reachability (resp. parity) SP games.

Payoffs in SP Games. The payoff of a play $\rho \in \text{Plays}$ corresponds to the vector of Booleans $\text{pay}(\rho) \in \{0, 1\}^t$ such that for all $i \in \{1, \ldots, t\}$, $\text{pay}_i(\rho) = 1$ if $\rho \in \Omega_i$, and $\text{pay}_i(\rho) = 0$ otherwise. Note that we omit to include Player 0 when discussing the payoff of a play. Instead we say that a play $\rho$ is won by Player 0 if $\rho \in \Omega_0$ and we write $\text{won}(\rho) = 1$, otherwise it is lost by Player 0 and we write $\text{won}(\rho) = 0$. We write $(\text{won}(\rho), \text{pay}(\rho))$ the extended payoff of $\rho$. Given a strategy profile $\sigma$, we write $\text{won}(\sigma) = \text{won}(\text{out}(\sigma))$ and $\text{pay}(\sigma) = \text{pay}(\text{out}(\sigma))$. For reachability SP games, since reachability objectives are prefix-dependant and given a history $h \in \text{Hist}$, we also define $\text{won}(h)$ and $\text{pay}(h)$ as done for plays.

We introduce the following partial order on payoffs. Given two payoffs $p = (p_1, \ldots, p_t)$ and $p' = (p'_1, \ldots, p'_t)$ such that $p, p' \in \{0, 1\}^t$, we say that $p'$ is larger than $p$ and write $p \leq p'$ if $p_i \leq p'_i$ for all $i \in \{1, \ldots, t\}$. Moreover, when it also holds that $p_i < p'_i$ for some $i$, we say that $p'$ is strictly larger than $p$ and we write $p < p'$. A subset of payoffs $P \subseteq \{0, 1\}^t$ is an antichain if it is composed of pairwise incomparable payoffs with respect to $\leq$.

Stackelberg-Pareto Synthesis Problem. Given a strategy $\sigma_0$ of Player 0, we consider the set of payoffs of plays consistent with $\sigma_0$ which are Pareto-optimal, i.e., maximal with respect to $\leq$. We write this set $P_{\sigma_0} = \{ \text{pay}(\rho) \mid \rho \in \text{Plays}_{\sigma_0} \}$. Notice that it is an antichain. We say that those payoffs are $\sigma_0$-fixed Pareto-optimal and write $|P_{\sigma_0}|$ the number of such payoffs.

A play $\rho \in \text{Plays}_{\sigma_0}$ is called $\sigma_0$-fixed Pareto-optimal if its payoff $\text{pay}(\rho)$ is in $P_{\sigma_0}$.

The problem studied in this paper asks whether there exists a strategy $\sigma_0$ for Player 0 such that every play in $\text{Plays}_{\sigma_0}$ which is $\sigma_0$-fixed Pareto-optimal satisfies the objective of Player 0. This corresponds to the assumption that given a strategy of Player 0, Player 1 will
play *rationally*, that is, with a strategy $\sigma_1$ such that $\text{out}(\langle\sigma_0,\sigma_1\rangle)$ is $\sigma_0$-fixed Pareto-optimal. It is therefore sound to ask that Player 0 wins against such rational strategies.

**Definition 1.** Given an SP game, the Stackelberg-Pareto Synthesis problem (SPS problem) is to decide whether there exists a strategy $\sigma_0$ for Player 0 (called a solution) such that for each strategy profile $\sigma = (\sigma_0,\sigma_1)$ with $\text{pay}(\sigma) \in P_{\sigma_0}$, it holds that $\text{won}(\sigma) = 1$.

**Witnesses.** Given a strategy $\sigma_0$ that is a solution to the SPS problem and any payoff $p \in P_{\sigma_0}$, for each play $\rho$ consistent with $\sigma_0$ such that $\text{pay}(\rho) = p$ it holds that $\text{won}(\rho) = 1$. For each $p \in P_{\sigma_0}$, we arbitrarily select such a play which we call a *witness* (of $p$). We denote by $\text{Wit}_{\sigma_0}$ the set of all witnesses, of which there are as many as payoffs in $P_{\sigma_0}$. In the sequel, it is useful to see this set as a tree composed of $|\text{Wit}_{\sigma_0}|$ branches. Additionally for a given history $h \in \text{Hist}$, we write $\text{Wit}_{\sigma_0}(h)$ the set of witnesses for which $h$ is a prefix, i.e., $\text{Wit}_{\sigma_0}(h) = \{p \in \text{Wit}_{\sigma_0} \mid h \text{ is prefix of } p\}$. Notice that $\text{Wit}_{\sigma_0}(h) = \text{Wit}_{\sigma_0}$ when $h = v_0$ and that $\text{Wit}_{\sigma_0}(h)$ decreases as $h$ increases, until it contains a single value or becomes empty.

**Example 2.** Consider the reachability SP game with arena $G$ depicted in Figure 1 in which Player 1 has $t = 3$ objectives. The vertices of Player 0 (resp. Player 1) are depicted as ellipses (resp. rectangles). Every objective in the game is a reachability objective defined as follows: $\Omega_0 = \text{Reach}(\{v_0,v_7\})$, $\Omega_1 = \text{Reach}(\{v_4,v_7\})$, $\Omega_2 = \text{Reach}(\{v_3\})$, $\Omega_3 = \text{Reach}(\{v_1,v_6\})$. The extended payoff of plays reaching vertices from which they can only loop is displayed in the arena next to those vertices, and the extended payoff of play $v_0v_2v_1v_5\omega$ is $(0, (0, 1), 0))$.

Consider the memoryless strategy $\sigma_0$ of Player 0 such that he chooses to always move to $v_5$ from $v_3$. The set of payoffs of plays consistent with $\sigma_0$ is $\{(0,0,1),(0,1,0),(1,0,0),(1,1,1)\}$ and the set of those that are Pareto-optimal is $P_{\sigma_0} = \{(1,0,0),(0,1,1)\}$. Notice that play $\rho = v_0v_2v_1v_5\omega$ is consistent with $\sigma_0$, has payoff $(1,0,0)$ and is lost by Player 0. Strategy $\sigma_0$ is therefore not a solution to the SPS problem. In this game, there is only one other memoryless strategy for Player 0, where he chooses to always move to $v_7$ from $v_3$. One can verify that it is again not a solution to the SPS problem.

We can however define a finite-memory strategy $\sigma'_0$ such that $\sigma'_0(v_0v_2v_1) = v_5$ and $\sigma'_0(v_0v_2v_3v_5v_1) = v_7$ and show that it is a solution to the problem. Indeed, the set of $\sigma'_0$-fixed Pareto-optimal payoffs is $P_{\sigma'_0} = \{(0,1,1),(1,1,0)\}$ and Player 0 wins every play consistent with $\sigma'_0$ whose payoff is in this set. A set $\text{Wit}_{\sigma'_0}$ of witnesses for these payoffs is $\{v_0v_2v_3v_5\omega, v_0v_2v_3v_5v_1v_7\}$ and is in this case the unique set of witnesses. This example shows that Player 0 sometimes needs memory in order to have a solution to the SPS problem.

---

2 This convention is used throughout this paper.
3 Fixed-Parameter Complexity

In this section, we show that the SPS problem is in FPT for both cases of reachability and parity SP games. The details of our proof for each type of objective are provided separately in their own subsection. We refer the reader to [15] for the concept of fixed-parameter complexity.

▶ Theorem 3. Solving the SPS problem is in FPT for reachability SP games for parameter \( t \) equal to the number of objectives of Player 1.

▶ Theorem 4. Solving the SPS problem is in FPT for parity SP games for parameters \( t \) and the maximal priority according to each parity objective of Player 1.

3.1 Challenger-Prover Game

In order to prove Theorem 3 and Theorem 4, we provide a reduction to a specific two-player zero-sum game, called the Challenger-Prover game (C-P game). This game is a zero-sum game played between Challenger (written \( C \)) and Prover (written \( P \)). We will show that Player 0 has a solution to the SPS problem in an SP game if and only if \( P \) has a winning strategy in the corresponding C-P game. In the latter game, \( P \) tries to show the existence of a strategy \( \sigma_0 \) that is solution to the SPS problem in the original game and \( C \) tries to disprove it. The C-P game is described independently of the objectives used in the SP game and its objective is described as such in a generic way. We later provide the proof of our FPT results by adapting it specifically for reachability and parity SP games.

Intuition on the C-P Game. Without loss of generality, the SP games we consider in this section are such that each vertex in their arena has at most two successors. It can be shown (see Appendix A) that any SP game \( G \) with \( n \) vertices can be transformed into an SP game \( \bar{G} \) with \( O(n^2) \) vertices such that every vertex has at most two successors and Player 0 has a solution to the SPS problem in \( G \) if and only if he has a solution to the SPS problem in \( \bar{G} \).

Let \( G \) be an SP game. The C-P game \( G' \) is a zero-sum game associated with \( G \) that intuitively works as follows. First, \( P \) selects a set \( P \) of payoffs which he announces as the set of Pareto-optimal payoffs \( P_{\sigma_0} \) for the solution \( \sigma_0 \) to the SPS problem in \( G \) he is trying to construct. Then, \( P \) tries to show that there exists a set of witnesses \( \text{Wit}_{\sigma_0} \) in \( G \) for the payoffs in \( P \). After the selection of \( P \) in \( G' \), there is a one-to-one correspondence between plays in the arenas \( G \) and \( G' \) such that the vertices in \( G' \) are augmented with a set \( W \) which is a subset of \( P \). Initially \( W \) is equal to \( P \) and after some history in \( G' \), \( W \) contains payoff \( p \) if the corresponding history in \( G \) is prefix of the witness with payoff \( p \) in the set \( \text{Wit}_{\sigma_0} \) that \( P \) is building. In addition, the objective \( \Omega_P \) of \( P \) is such that he has a winning strategy \( \sigma_P \) in \( G' \) if and only if the set \( P \) that he selected coincides with the set \( P_{\sigma_0} \) for the corresponding strategy \( \sigma_0 \) in \( G \) and the latter strategy is a solution to the SPS problem in \( G \). A part of the arena of the C-P game for Example 2 with a positional winning strategy for \( P \) highlighted in bold is illustrated in Figure 2.

Arena of the C-P Game. The initial vertex \( \bot \) belongs to \( P \). From this vertex, he selects a successor \((v_0, P, W)\) such that \( W = P \) and \( P \) is an antichain of payoffs which \( P \) announces.

---

3 We suppose the reader familiar with the concept of zero-sum games, see e.g. [20].
We write with this vertex will have this same value for their P-component of the vertices controlled by \( P \).

In the definition of projection its \( \bot \) removing as the set \( V \).

Formally, the game arena of the C-P game is the tuple \( (v, P, W) \) that belongs to \( \mathcal{P} \) or \( (v, P, (W_i, W_r)) \) that belongs to \( \mathcal{C} \). Given a play \( \rho \) (resp. history \( h \)) in \( G' \), we denote by \( \rho_V \) (resp. \( h_V \)) the play (resp. history) in \( G \) obtained by removing \( \bot \) and keeping the v-component of every vertex of \( \mathcal{P} \) in \( \rho \) (resp. \( h \)), which we call its projection.

After history \( h_m \) such that \( m = (v, P, W) \) with \( v \in V_0 \), \( \mathcal{P} \) selects a successor \( v' \) such that \( (v, v') \in E \) and vertex \( (v', P, W) \) is added to the play. This corresponds to Player 0 choosing a successor \( v' \) after history \( h_v \) in \( G \).

After history \( h_m \) such that \( m = (v, P, W) \) with \( v \in V_1 \), \( \mathcal{P} \) selects a successor \( (v, P, (W_i, W_r)) \) with \( (W_i, W_r) \) a partition of \( W \). This corresponds to \( \mathcal{P} \) splitting the set \( W \) into two parts according to the two successors \( v_i \) and \( v_r \) of \( v \). For the strategy \( \sigma_0 \) that \( \mathcal{P} \) tries to construct and its set of witnesses \( \text{Wit}_{\sigma_0} \) he is building, he asserts that \( W_i \) (resp. \( W_r \)) is the set of payoffs of the witnesses in \( \text{Wit}_{\sigma_0}(h_V v_i) \) (resp. \( \text{Wit}_{\sigma_0}(h_V v_r) \)).

From a vertex \( (v, P, (W_i, W_r)) \), \( \mathcal{C} \) can select a successor \( (v_l, P, W_l) \) or \( (v_r, P, W_r) \) which corresponds to the choice of Player 1.

Formally, the game arena of the C-P game is the tuple \( G' = (V', V'_0, V'_1, E', \bot) \) with

\[
V'_0 = \{ \bot \} \cup \{(v, P, W) | v \in V, P \subseteq \{0, 1\} \} \text{ is an antichain and } W \subseteq P \},
\]

\[
V'_1 = \{(v, P, (W_i, W_r)) | v \in V_1, P \subseteq \{0, 1\} \} \text{ is an antichain and } W_i, W_r \subseteq P \},
\]

\[
(\bot, (v, P, W)) \in E' \text{ if } v = v_0 \text{ and } P = W,
\]

\[
((v, P, W), (v', P, W)) \in E' \text{ if } v \in V_0 \text{ and } (v, v') \in E,
\]

\[
((v, P, W), (v, P, (W_i, W_r))) \in E' \text{ if } v \in V_1 \text{ and } (W_i, W_r) \text{ is a partition of } W,
\]

\[
((v, P, (W_i, W_r)), (v', P, W)) \in E' \text{ if } (v, v') \in E \text{ and } \{v' = v_i \text{ and } W = W_i\} \text{ or } \{v' = v_r \text{ and } W = W_r\}.
\]

In the definition of \( E' \), if \( v \) has a single successor \( v' \) in \( G \), it is assumed to be \( v_i \) and \( W_r \) is always equal to \( \emptyset \). We use as a convention that given the two successors \( v_i \) and \( v_j \) of vertex \( v \), \( v_i \) is the left successor if \( i < j \).

**Objective of \( \mathcal{P} \) in the C-P Game.** Let us now discuss the objective \( \Omega_\mathcal{P} \) of \( \mathcal{P} \). The \( W \)-component of the vertices controlled by \( \mathcal{P} \) has a size that decreases along a play \( \rho \) in \( G' \). We write \( \lim_W(\rho) \) the value of the \( W \)-component at the limit in \( \rho \). Recall that with this \( W \)-component, \( \mathcal{P} \) tries to construct a solution \( \sigma_0 \) to the SPS problem with associated sets.
Stackelberg-Pareto Synthesis (full version)

$P_{σ_a}$ and $Wit_{σ_a}$. Therefore, for him to win in the C-P game, $lim_W(ρ)$ must be a singleton or empty in every consistent play such that:

- $lim_W(ρ)$ must be a singleton $\{p\}$ with $p$ the payoff of $ρ_W$ in $G$, showing that $ρ_W ∈ Wit_{σ_a}$ is a correct witness for $p$. In addition, it must hold that $won(ρ_W) = 1$ as $p ∈ P$ and as $P$ wants $σ_0$ to be a solution.

- $lim_W(ρ)$ must be the empty set such that either the payoff of $ρ_W$ belongs to $P_{σ_0}$ and $won(ρ_W) = 1$, or the payoff of $ρ_W$ is strictly smaller than some payoff in $P_{σ_0}$.

These conditions verify that the sets $P = P_{σ_0}$ and $Wit_{σ_a}$ are correct and that $σ_0$ is indeed a solution to the SPS problem in $G$. They are generic as they do not depend on the actual objectives used in the SP game.

Let us give the formal definition of $Ω_P$. For an antichain $P$ of payoffs, we write $Plays^P_{G'}$, the set of plays in $G'$ which start with $⊥(v_0, P, P)$ and we define the following set

$$B_P = \{ p ∈ Plays^P_{G'} | \{ lim_W(ρ) = \{p\} ∧ pay(ρ_W) = p ∈ P ∧ won(ρ_W) = 1 \} ∨ (lim_W(ρ) = ∅ ∧ pay(ρ_W) ∈ P ∧ won(ρ_W) = 1) ∨ (lim_W(ρ) = ∅ ∧ ∃p ∈ P, pay(ρ_W) < p)) \}.$$  

Objective $Ω_P$ of $P$ in $G'$ is the union of $B_P$ over all antichains $P$. As the C-P game is zero-sum, objective $Ω_C$ equals $Plays^C_{G'} \setminus Ω_P$. The following theorem holds.

**Theorem 5.** Player 0 has a strategy $σ_0$ that is solution to the SPS problem in $G$ if and only if $P$ has a winning strategy $σ_P$ from $⊥$ in the C-P game $G'$.

**Proof of Theorem 5** Let us first assume that Player 0 has a strategy $σ_0$ that is solution to the SPS problem in $G$. Let $P_{σ_0}$ be its set of $σ_0$-fixed Pareto-optimal payoffs and let $Wit_{σ_a}$ be a set of witnesses. We construct the strategy $σ_P$ from $σ_0$ such that

- $σ_P(⊥) = (v_0, P, P)$ such that $P = P_{σ_0}$ (this vertex exists as $P_{σ_0}$ is an antichain),

- $σ_P(hm) = (v', P, W)$ if $m = (v, P, W)$ with $v ∈ V_0$ and $v' = σ_0(hv, v)$,

$σ_P(hm) = (v, P, W)$ if $m = (v, P, W)$ with $v ∈ V_0$ and for $i ∈ \{l, r\}$, $W_i = \{ pay(ρ) | ρ ∈ Wit_{σ_a}(hv, v) \}$.

It is clear that given a play $ρ$ in $G'$ consistent with $σ_P$, the play $ρ_W$ in $G$ is consistent with $σ_0$. Let us show that $σ_P$ is winning for $P$ from $⊥$ in $G'$. Consider a play $ρ$ in $G'$ consistent with $σ_P$. There are two possibilities. (i) $ρ_W$ is a witness of $Wit_{σ_a}$ and by construction $lim_W(ρ) = \{p\}$ with $p = pay(ρ_W)$; thus $won(ρ_W) = 1$ as $σ_0$ is a solution and $ρ_W$ is a witness. (ii) $ρ_W$ is not a witness and by construction $lim_W(ρ) = ∅$; as $σ_0$ is a solution, then $p = pay(ρ_W)$ is bounded by some payoff of $P_{σ_0}$ and in case of equality $won(ρ_W) = 1$. Therefore $ρ$ satisfies the objective $B_P$ of $Ω_P$ since it satisfies condition (1) in case (i) and condition (2) or (3) in case (ii).

Let us now assume that $P$ has a winning strategy $σ_P$ from $⊥$ in $G'$. Let $P$ be the antichain of payoffs chosen from $⊥$ by this strategy. We construct the strategy $σ_0$ from $σ_P$ such that $σ_0(hv, v) = v'$ given $σ_P(hm) = (v', P, W)$ with $m = (v, P, W)$ and $v ∈ V_0$. Notice that this definition makes sense since there is a unique history $hm$ ending with a vertex of $P$ associated with $hv$ showing a one-to-one correspondence between those histories.

Let us show $σ_0$ is a solution to the SPS problem with $P_{σ_0}$ being the set $P$. First notice that $P$ is not empty. Indeed let $ρ$ be a play consistent with $σ_P$. As $ρ$ belongs to $Ω_P$ and in particular to $B_P$, one can check that $P ≠ ∅$ by inspecting conditions (1) to (3). Second notice that by definition of $E'$, if $\{(v, P, W), (v, P, W_i)) \} ∈ E'$ with $W ≠ ∅$, then either $W_l$ or $W_r$ is not empty. Therefore given any payoff $p ∈ P$, there is a unique play $ρ$ consistent with $σ_P$ such that $lim_W(ρ) = \{p\}$. By construction of $σ_0$ and as $σ_P$ is winning, the play $ρ_W$ is consistent with $σ_0$, has payoff $p$, and is won by Player 0 (see (1)).
Let \( \rho \) be a play consistent with \( \sigma_0 \) and \( \rho \) be the corresponding play consistent with \( \sigma_P \).
It remains to consider (2) and (3). These conditions indicate that \( \rho \) has a payoff equal to or strictly smaller than a payoff in \( P \) and that in case of equality \( \text{won}(\rho) = 1 \). This shows that \( P_{\sigma_0} = P \) and that \( \sigma_0 \) is a solution to the SPS problem.

\[\text{Proposition 6.}\]

We now develop the proof of Theorem 3 which works by specializing the generic objective \( \Omega \) to handle reachability SP games. We extend the arena \( G' \) of the C-P game such that its vertices keep track of the objectives of \( G \) which are satisfied along a play. Given an extended payoff \( (w, p) \in \{0, 1\} \times \{0, 1\}^t \) and a vertex \( v \in V \), we define the payoff update function \( \text{upd}(w, p, v) = (w', p') \) such that

\[ w' = 1 \iff w = 1 \text{ or } v \in T_0, \]
\[ p' = 1 \iff p_i = 1 \text{ or } v \in T_i, \quad \forall i \in \{1, \ldots, t\}. \]

We obtain the extended arena \( G^* \) as follows: (i) its set of vertices is \( V' \times \{0, 1\} \times \{0, 1\}^t \), (ii) its initial vertex is \( \perp = (\perp, 0, (0, \ldots, 0)) \), and (iii) \( ((m, w, p), (m', w', p')) \) with \( m' = (v', P, W) \) or \( m' = (v', P, (W_1, W_r)) \) is an edge in \( G^* \) if \( (m, m') \in E' \) and \( (w', p') = \text{upd}(w, p, v') \).

We define the zero-sum game \( G^* = (G^*, \Omega^*_P) \) in which the three abstract conditions (1-3) detailed previously are encoded into the following Büchi objective by using the \((w, p)\)-component added to vertices. We define \( \Omega^*_P = \text{Büchi}(B^*) \) with

\[ B^* = \{(v, P, W, w, p) \in V^*_P \mid (W = \{p\} \land w = 1) \lor (W = \emptyset \land p \in P \land w = 1) \lor (W = \emptyset \land \exists p' \in P, p < p')\}. \]

\[\text{Proposition 6.}\]

Player 0 has a strategy \( \sigma_0 \) that is solution to the SPS problem in a reachability SP game \( G \) if and only if \( P \) has a winning strategy \( \sigma_P \) in \( G^* \).

The proof of this proposition is a consequence of Theorem 5. Using the one-to-one correspondence between plays in \( G \) and plays in \( G^* \) and the fact that \( G \) is a reachability SP game, the \((w, p)\)-component in vertices of \( G^* \) allows us to easily retrieve the extended payoff of a play in \( G \). Indeed, in a play \( \rho \in \text{Play}_{G^*} \), given the construction of \( G^* \) and the payoff update function, it holds that from some point on the \( W \)- and \((w, p)\)-components are constant. Therefore it holds that \( w = \text{won}(\rho) \), \( p = \text{pay}(\rho) \) and \( W = \lim w(\rho) \) for that play \( \rho \). Moreover the \( P \)-component is constant along a play in \( G^* \). It is direct to see that the plays \( \rho \) in \( G^* \) which visit infinitely often the set \( B^* \), and therefore satisfy the Büchi objective \( \Omega^*_P = \text{Büchi}(B^*) \), satisfy one of the three conditions (1-3) stated in Subsection 3.1.

The converse is also true.

We now describe a FPT algorithm for deciding the existence of a solution to the SPS problem in a reachability SP game, thus proving Theorem 3.

\[\text{Proof of Theorem 3}\]

We describe the following FPT algorithm (for parameter \( t \)) for deciding the existence of a solution to the SPS problem in a reachability SP game \( G \) by using Proposition 6. First, we construct the zero-sum game \( G^* \). Its number \( n \) of vertices is upper-bounded by \( 1 + |V| \cdot 2^{|V|} \cdot 2^{|V|} \). Indeed, except the initial vertex, vertices are of the form either \((v, P, W, w, p)\) or \((v, P, (W_1, W_r), w, p)\) such that \( P, W, W_l \) and \( W_r \) are antichains of payoffs in \( \{0, 1\}^t \), and \((w, p)\) is an extended payoff. The construction of \( G^* \) is thus in FPT for parameter \( t \). Second, By Proposition 6, deciding whether there exists
a solution to the SPS problem in $\mathcal{G}$ amounts to deciding if $\mathcal{P}$ has a winning strategy from $\perp^*$ in $\mathcal{G}^*$. Since the objective $\Omega^*_p$ of $\mathcal{P}$ in $\mathcal{G}^*$ is a Büchi objective, this game can be solved in $O(n^2)$ [11]. It follows that $\mathcal{G}^*$ can be solved in FPT for parameter $t$. ▫

3.3 Fixed-Parameter Complexity of Parity SP Games

We now turn to parity SP games and explain why solving the SPS problem in these games is in FPT, again by reduction to the C-P game. To this end, we first recall the notion of Boolean Büchi games.

Boolean Büchi games are zero-sum games which we use in our reduction to the C-P for parity SP game. Given $m$ sets $T_1, \ldots, T_m$ such that $T_i \subseteq V$, $i \in \{1, \ldots, m\}$ and $\phi$ a Boolean formula over the set of variables $X = \{x_1, \ldots, x_m\}$, the Boolean Büchi objective $\text{Boolean Büchi}(\phi, T_1, \ldots, T_m) = \{ \sigma \in \text{Plays} \mid \sigma$ satisfies $\phi, T_1, \ldots, T_m \}$ is the set of plays whose valuation of the variables in $X$ satisfy formula $\phi$. Given a play $\rho$, its valuation is such that $x_i = 1$ if and only if $\text{Inf}(\rho) \cap T_i \neq \emptyset$ and $x_i = 0$ otherwise. That is, a play satisfies the objective if the Boolean formula describing sets to be visited infinitely often by a play is satisfied. We denote by $|\phi|$ the size of $\phi$ as equal to the number of conjunctions and disjunctions in $\phi$. The following theorem on the fixed-parameter complexity of Boolean Büchi games is proved in [9].

► Theorem 7. Solving Boolean Büchi games is in FPT, with an algorithm in $O(2^M \cdot |\phi| + (M^M \cdot |V|)^3)$ time with $M = 2^n$ such that $m$ is the number of variables and $|\phi|$ is the size of $\phi$ in the Boolean Büchi objective $\mathcal{G}$.

Let $\mathcal{G} = (G, \Omega_0, \ldots, \Omega_t)$ be a parity SP game with parity objectives such that $\Omega_i = \text{Parity}(c_i)$ for a priority function $c_i : V \rightarrow \mathbb{N}$. Let $G'$ be the arena of the C-P game. In the following, we construct a Boolean Büchi objective $\Omega^*_p$ for $\mathcal{P}$ such that the following proposition holds.

► Proposition 8. Player 0 has a strategy $\sigma_0$ that is solution to the SPS problem in $\mathcal{G}$ if and only if $\mathcal{P}$ has a winning strategy $\sigma_p$ in $\mathcal{G}^* = (G', \Omega^*_p)$.

Proof of Proposition 8. Let $G' = (V', V'_p, V'_o, E', \perp)$ be the arena of the C-P game presented in Section 3.3. Recall that the objective $\Omega_p$ of this game is the union of the sets $B_P$ over all antichains $P$ such that $B_P$ is the disjunction of conditions (3). The idea of the proof is to translate this objective into a Boolean Büchi objective $\Omega^*_p$. We will proceed step by step. The required Boolean formula for defining $\Omega^*_p$ is equal to

$$\phi = \bigvee_P (x_P \land (\text{cond}_{P}^d \lor \text{cond}_{P}^d \lor \text{cond}_{P}^d))$$

such that the main disjunction is over all antichains $P$. The variable $x_P$ corresponds to the set $T_P = \{(v, P, W) \in V'_P \}$. The valuation of $x_P$ is true for a given play if and only if the set $T_P$ is visited infinitely often and therefore $P$ is the antichain chosen by $\mathcal{P}$ in $G'$. Since the $P$-component is constant along a play, only one $x_P$ is valued as true for a given play. Let us now detail each subformula $\text{cond}_{P}^d$ that is the translation of condition (i) for $i \in \{1, 2, 3\}$.

Let us begin with the encoding of payoffs. Let $d_0, \ldots, d_t$ be such that $d_i$ is the maximal even priority appearing in $G$ according to priority function $c_i$ for objective $\Omega_i$ with $i \in \{0, \ldots, t\}$.

First, we show that a parity objective $\text{Parity}(c_i)$ from $\mathcal{G}$ can be encoded as a Boolean Büchi objective. Given the parity objective $\text{Parity}(c_i)$, we construct the Boolean formula $\text{parity}_i$ over variables $\{x^i_0, x^i_1, \ldots, x^i_{d_i}\}$ such that

$$\text{parity}_i = x^i_0 \lor (x^i_2 \land \neg x^i_1) \lor \cdots \lor (x^i_{d_i} \land \neg x^i_{d_i-1} \land \neg x^i_{d_i-3} \land \cdots \land \neg x^i_1)$$
and for $j \in \{0, \ldots, d_i\}$, the set corresponding to variable $x_j$ is $T_j = \{(v, P, W) \in V_P | c_i(v) = j\}$. It is easy to show that the parity objective $\text{Parity}(c_i)$ is satisfied if and only if the Boolean Büchi objective $\text{BooleanBüchi}(\text{parity}_i, T_0, \ldots, T_{d_i})$ is satisfied.

Second, given a payoff $p = (p_1, \ldots, p_t)$ in $\mathcal{G}$, we consider the Boolean formula

$$\text{payoff}_p = C_1 \land \cdots \land C_t$$

such that $C_i = \text{parity}_i$ if $p_i = 1$ and $C_i = \neg \text{parity}_i$ otherwise. Clearly the projection $\rho_V$ of play $\rho$ realizes payoff $p$ if and only if $p$ satisfies the Boolean Büchi objective $\text{BooleanBüchi}(\text{payoff}_p, T_0, \ldots, T_{d_i}, \ldots, T_{d_t})$.

We now fix some antichain $P$. Let us detail subformula $\text{cond}_P$ encoding condition (1). For a payoff $p$, we define formula

$$\text{single}_p = x_p \land \text{payoff}_p \land \text{parity}_0$$

such that $x_p = \{(v, P, W) \in V_P | W = \{p\}\}$. Since at some point during a play, the $W$-component stabilizes and since a play $\rho$ which satisfies $\text{payoff}_p \land \text{parity}_0$ is such that $\text{pay}(\rho_V) = p$ and $\text{won}(\rho_V) = 1$, it holds that satisfying this formula corresponds exactly to satisfying condition (1) for some $p$. Formula $\text{cond}_P$ is thus the disjunction

$$\text{cond}_P = \bigvee_{p \in P} \text{single}_p.$$ 

Let us shift to subformula $\text{cond}_P$ encoding condition (2). Using similar arguments, we define for payoff $p$ formula

$$\text{empty}_p = x_p \land \text{payoff}_p \land \text{parity}_0$$

such that $x_p = \{(v, P, \emptyset) \in V_P\}$. It corresponds exactly to the set of plays $\rho$ such that $\text{lim}_W(\rho) = \emptyset$, $\text{pay}(\rho_V) = p$ and $\text{won}(\rho_V) = 1$. Therefore

$$\text{cond}_P = \bigvee_{p \in P} \text{empty}_p.$$ 

Finally, we define subformula $\text{cond}_P$ encoding condition (3). Let $\mathcal{P}$ be the set containing every payoff $p'$ such that $\exists p \in P, p' < p$. We define

$$\text{cond}_P = \bigvee_{p' \in \mathcal{P}} \text{smaller}_{p'}$$

with $\text{smaller}_{p'}$ being the formula $x_{p'} \land \text{payoff}_{p'}$.

Notice that the Boolean formula $\phi$ constructed in this proof has a number $m$ of variables and a size $|\phi|$ that only depend on $t$ and $d_i, i \in \{0, \ldots, t\}$. 

This previous construction can be used to provide the proof of Theorem 4.

**Proof of Theorem 4.** We describe the following FPT algorithm for deciding the existence of a solution to the SPS problem in a parity SP game $\mathcal{G}$ by using Proposition 8. First, we construct the zero-sum game $G^*$ of Proposition 8. Its number $n$ of vertices is upper-bounded by $1 + |V| \cdot 2^{2^t+1} + |V| \cdot 2^{2^t}$ which is in $O(|V| \cdot f(t))$ with $f$ a computable function which only depends on $t$. Moreover the number $m$ of variables and the size $|\phi|$ of the Boolean formula $\phi$ defining the Boolean Büchi objective of $G^*$ depend only on parameters $t$ and $d_i$ for $i \in \{0, \ldots, t\}$. Therefore the construction of $G^*$ is in FPT for these parameters. Deciding whether there exists a solution to the SPS problem in $\mathcal{G}$ amounts to deciding if $\mathcal{P}$ has a winning strategy from $\bot$ in $G^*$. By the latter Boolean Büchi game can be solved with an algorithm in $O(2^M \cdot |\phi| + (M^M \cdot n)^5)$ time with $M = 2^m$. It follows that $G^*$ can be solved in $O(2^M \cdot |\phi| + (M^M \cdot |V| \cdot f(t))^5)$ which is in FPT for the announced parameters. ▷
4 Complexity Class of the SPS Problem

In this section, we study the complexity class of the SPS problem and prove its \textit{NEXPTIME}-completeness for both reachability and parity SP games.

4.1 NEXPTIME-Membership

We first show the membership to \textit{NEXPTIME} of the SPS problem by providing a nondeterministic algorithm with time exponential in the size of the game $G$. By size, we mean the number $|V|$ of its vertices and the number $t$ of objectives of Player 1. Notice that the time complexity of the FPT algorithms obtained in the previous section is too high, preventing us from directly using the C-P game to show a tight membership result. Conversely, the nondeterministic algorithm provided in this section is not FPT as it is exponential in $|V|$.

\begin{itemize}
  \item Theorem 9. The SPS problem is in \textit{NEXPTIME} for reachability and parity SP games.
\end{itemize}

We show that the SPS problem is in \textit{NEXPTIME} by proving that if Player 0 has a strategy which is a solution to the problem, then he has one which is finite-memory with at most an exponential number of memory states\footnote{Recall that to have a solution to the SPS problem, memory is sometimes necessary as shown in Example 2.}. This yields a \textit{NEXPTIME} algorithm in which we nondeterministically guess such a strategy and check in exponential time that it is indeed a solution to the problem.

\begin{itemize}
  \item Proposition 10. Let $G$ be a reachability SP game or a parity SP game. Let $\sigma_0$ be a solution the the SPS problem. Then there exists another solution $\tilde{\sigma}_0$ that is finite-memory and has a memory size exponential in the size of $G$.
\end{itemize}

While the proof of Proposition 10 requires some specific arguments to treat both reachability and parity objectives, it is based on the following common principles.

\begin{itemize}
  \item We start from a winning strategy $\sigma_0$ for the SPS problem and the objectives $\Omega_0, \Omega_1, \ldots, \Omega_t$ and consider a set of witnesses $\text{Wit}_{\sigma_0}$, that contains one play for each element of the set $P_{\sigma_0}$ of $\sigma_0$-fixed Pareto-optimal payoffs.
  \item We start by showing the existence of a strategy $\hat{\sigma}_0$ constructed from $\sigma_0$, in which Player 0 follows $\sigma_0$ as long as the current consistent history is prefix of at least one witness in $\text{Wit}_{\sigma_0}$. Then when a deviation from $\text{Wit}_{\sigma_0}$ occurs, Player 0 switches to a so-called \textit{punishing strategy}. A deviation is a history that leaves the set of witnesses $\text{Wit}_{\sigma_0}$ after a move of Player 1 (this is not possible by a move of Player 0). After such a deviation, $\hat{\sigma}_0$ systematically imposes that the consistent play either satisfies $\Omega_0$ or is not $\sigma_0$-fixed Pareto-optimal, i.e., it gives to Player 1 a payoff that is strictly smaller than the payoff of a witness in $\text{Wit}_{\sigma_0}$. This makes the deviation \textit{irrational} for Player 1. We show that this can be done, both for reachability and parity objectives, with at most exponentially many different punishing strategies, each having a size bounded exponentially in the size of the game. The strategy $\hat{\sigma}_0$ that we obtain is therefore composed of the part of $\sigma_0$ that produces $\text{Wit}_{\sigma_0}$ and a punishment part whose size is at most exponential.
  \item Then, we show how to decompose each witness in $\text{Wit}_{\sigma_0}$ into at most exponentially many \textit{sections} that can, in turn, be compacted into finite elementary paths or lasso shaped paths of polynomial length. As $\text{Wit}_{\sigma_0}$ contains exactly $|P_{\sigma_0}|$ witnesses $\rho$, those compact witnesses $c\rho$ can be produced by a finite-memory strategy with an exponential size for
\end{itemize}
Then there exists a finite-memory strategy when Player \( \rho \). We express the objective
\[ \text{As explained above, a deviation is a history that leaves the set of witnesses} \]
\[ \text{First, we define the set of deviations} \]
\[ \text{Deviations and Punishing Strategies. First, we define the set of deviations} \]
\[ \text{V. Bruyère, J.-F. Raskin and C. Tamines 13} \]
\[ \text{Figure 3 The creation of strategies } \hat{\sigma}_0 \text{ and } \tilde{\sigma}_0 \text{ from a solution } \sigma_0 \text{ with } \text{Wit}_{\sigma_0} = \{r_1, r_2, r_3, r_4\}. \]
both reachability and parity objectives. This allows us to construct a strategy \( \tilde{\sigma}_0 \) that produces the compact witnesses and acts as \( \hat{\sigma}_0 \) after any deviation. This strategy is a solution of the SPS problem and has an exponential size as announced.

We now develop the details of the construction of the strategies \( \hat{\sigma}_0 \) and \( \tilde{\sigma}_0 \). Figure 3 illustrates this construction. It is done in several steps to finally get the proof of Proposition 11. For the rest of this section, we fix an SP game \( G \) with objectives \( \Omega_0, \Omega_1, \ldots, \Omega_k \), a strategy \( \sigma_0 \) that is solution to the SPS problem, a set of witnesses \( \text{Wit}_{\sigma_0} \) for the \( \sigma_0 \)-fixed Pareto-optimal payoffs in \( P_{\sigma_0} \), and we write \( \Omega^< (P_{\sigma_0}) \) the set of plays whose payoff is strictly smaller than some payoff in \( P_{\sigma_0} \).

\textbf{Deviations and Punishing Strategies.} First, we define the set of deviations \( \text{Dev}(\text{Wit}_{\sigma_0}) \) as follows:
\[ \text{Dev}(\text{Wit}_{\sigma_0}) = \{ hv \in \text{Hist}_{\sigma_0} \mid \text{Wit}_{\sigma_0}(h) \neq \emptyset \wedge \text{Wit}_{\sigma_0}(hv) = \emptyset \}. \]
As explained above, a deviation is a history that leaves the set of witnesses \( \text{Wit}_{\sigma_0} \) (by a move of Player 1).

Second, we establish the existence of canonical forms for punishing strategies. We potentially need an exponential number of them for reachability objectives and a polynomial number of them for parity objectives. In both cases, each punishing strategy has a size which can be bounded exponentially. The existence of those strategies are direct consequences of the following lemmas.

\textbf{Lemma 11 (Parity). Let } v \in V \text{ be such that there exists } hv \in \text{Dev}(\text{Wit}_{\sigma_0}). \text{ Then there exists a finite-memory strategy } \sigma^\text{Pun}_v \text{ such that for all deviations } hv \in \text{Dev}(\text{Wit}_{\sigma_0}), \text{ when Player } 0 \text{ plays } \sigma^\text{Pun}_v \text{ from } hv, \text{ all consistent plays } \rho \text{ starting in } v \text{ are such that either } h\rho \in \Omega_0 \text{ or } h\rho \in \Omega^< (P_{\sigma_0}). \text{ The size of } \sigma^\text{Pun}_v \text{ is at most exponential in the size of } G. \]

\textbf{Proof of Lemma 11} First, we note that, after a deviation \( hv \in \text{Dev}(\text{Wit}_{\sigma_0}) \), if Player 0 continues to play the strategy \( \sigma_0 \) from \( hv \), then all consistent plays \( \rho \) are such that either \( \rho \in \Omega_0 \) or \( \rho \in \Omega^< (P_{\sigma_0}) \) as \( \sigma_0 \) is a solution to the SPS problem. Therefore, we know that Player 0 has a punishing strategy for all such deviations \( hv \). Second, as parity objectives are prefix-independent, he can use one uniform strategy that only depends on \( v \) (and not on \( hv \)). There exists such a strategy with finite memory that can be constructed as follows. We express the objective \( \Omega_0 \cup \Omega^< (P_{\sigma_0}) \) as an explicit Muller objective [25] for a zero-sum game played on the arena \( G \) from initial vertex \( v \). This objective is defined by the set \( \{ C \subseteq V \mid \exists \rho \text{ such that } \inf(\rho) = C \wedge \rho \in \Omega_0 \cup \Omega^< (P_{\sigma_0}) \} \). This exactly encodes the objective of Player 0 when he plays the punishing strategy after a deviation \( hv \). It is well-known that
Then there exists a finite-memory strategy exponential in the size of \( (\Omega^t) \) with the objective \( w \). We consider the extension of \( \text{Dev}(\Omega^t) \) with \( (\text{won}(h), \text{pay}(h)) = (w, p) \), when Player 0 plays \( \sigma_{(w, p)}^\text{Pun} \) from \( h \), all consistent plays \( \rho \) starting in \( v \) are such that either \( hp \in \Omega_0 \) or \( hp \in \Omega^\text{<}(P_\sigma) \). The size of \( \sigma_{(w, p)}^\text{Pun} \) is at most exponential in the size of \( G \).

**Proof of Lemma 12** We follow the same reasoning as in the proof of Lemma 11 except that reachability objectives are not prefix-independent. We thus need to take into account the set of objectives \( \Omega_t \) already satisfied along the history \( h \), which is recorded in \( (w, p) \). The uniform finite-memory strategy \( \sigma_{(w, p)}^\text{Pun} \) that Player 0 can use from all deviations \( h \) such that \( \text{won}(h) = w \) and \( \text{pay}(h) = p \) is constructed as follows. First, notice that if \( w = 1 \), meaning that objective \( \Omega_0 \) is already satisfied, then Player 0 can play using any memoryless strategy as punishing strategy. Second, if \( w = 0 \), as done in Subsection 3.2 we consider the extension of \( G \) such that its vertices are of the form \( \langle v', w', p' \rangle \) where the \( (w', p') \)-component keeps track of the objectives that have been satisfied so far and such that its initial vertex is equal to \( \langle v, w, p \rangle \). On this extended arena, we consider the zero-sum game with the objective \( \Omega_0 \cup \Omega^\text{<}(P_\sigma) \) encoded as the disjunction of a reachability objective \( \Omega_0 \) and a safety objective \( \Omega^\text{<}(P_\sigma) \). More precisely, in the extended game, Player 0 has the objective either to reach a vertex in the set \( \{ \langle v', w', p' \rangle \mid w' = 1 \} \) or to stay forever within the set of vertices \( \{ \langle v', w', p' \rangle \mid \exists p'' \in P_\sigma : p' < p'' \} \). It is known, see e.g. [9], that there always exist memoryless winning strategies for zero-sum games with an objective which is the disjunction of a reachability objective and a safety objective. Therefore, this is the case here for the extended game, and thus also in the original game however with a winning finite-memory strategy with exponential size.

If we systematically change within \( \sigma_0 \) the behavior of Player 0 after a deviation from \( \text{Wit}_\theta \), and use the punishing strategies as defined in the proofs of Lemmas 11 and 12 we obtain a new strategy \( \sigma_0 \) that is solution to the SPS problem. The total size of the punish finite-memory strategies in \( \sigma_0 \) is at most exponential in the size of \( G \). To obtain our results, it remains to show how to compact the plays in \( \text{Wit}_\theta \). To that end, we study the histories and plays within \( \text{Wit}_\theta \).

**Compacting Witnesses.** We now show how to compact the set of witnesses in a way to produce them with a finite-memory strategy. Together with the punishing strategies this will lead to a solution \( \sigma_0 \) to SPS problem with a memory of exponential size. We first consider reachability objectives and explain later how to modify the construction for parity objectives.

Given a history \( h \) that is prefix of at least one witness in \( \text{Wit}_\theta \), we call region and we denote by \( \text{Reg}(h) \) the tuple \( \text{Reg}(h) = (\text{won}(h), \text{pay}(h), \text{Wit}_\theta(h)) \). We also use notation \( R = (w, p, W) \) for a region. Given a witness \( \rho = v_0 v_1 \ldots \in \text{Wit}_\theta \), we consider \( \rho^* = (v_0, R_0)(v_1, R_1)\ldots \) such that each \( v_j \) is extended with the region \( R_j = (w_j, p_j, W_j) = \text{Reg}(v_0 v_1 \ldots v_j) \). Similarly we define \( h^* \) associated with any history \( h \) prefix of a witness. The following properties hold for a witness \( \rho \) and its corresponding play \( \rho^* \):

- for all \( j \geq 0 \), we have \( w_j \leq w_{j+1} \), \( p_j \leq p_{j+1} \), and \( W_j \supseteq W_{j+1} \),
- the sequence \( (w_j, p_j)_{j \geq 0} \) eventually stabilizes on \( (w, p) \) equal to the extended payoff \( (\text{won}(\rho), \text{pay}(\rho)) \) of \( \rho \),
the sequence \((W_j)_{j \geq 0}\) eventually stabilizes on a set \(W\) which is a singleton such that \(W = \{p\}\).

Thanks to the previous properties, each \(\rho \in \text{Wit}_{\sigma_0}\) can be \textit{region decomposed} into a sequence of paths \(\pi[1]\pi[2] \cdots \pi[k]\) where the corresponding decomposition \(\pi^*[1]\pi^*[2] \cdots \pi^*[k]\) of \(\rho^*\) is such that for each \(\ell:\ (i)\ the region is constant along the path \(\pi^*[\ell]\) and \((ii)\) it is distinct from the region of the next path \(\pi^*[\ell + 1]\) (if \(\ell < k\)). Each \(\pi[\ell]\) is called a \textit{section} of \(\rho\), such that it is \textit{internal} (resp. \textit{terminal}) if \(\ell < k\) (resp. \(\ell = k\)).

Notice that the number of regions that are traversed by \(\rho\) is bounded by
\[
(t + 2) \cdot |\text{Wit}_{\sigma_0}|.
\]
Indeed along \(\rho\), the first two components \((w, p)\) of a region correspond to a monotonically increasing vector of \(t + 1\) Boolean values (from \((0, 0, \ldots, 0)\)) to \((1, 1, \ldots, 1)\) in the worst case, and the last component \(W\) is a monotonically decreasing set of witnesses (from \(\text{Wit}_{\sigma_0}\) to \(\{p\}\) in the worst case). So the number of regions traversed by a witness is bounded exponentially in the size of the game \(G\).

We have the following important properties for the sections of the witnesses of \(\text{Wit}_{\sigma_0}\).

Let \(\rho, \rho' \in \text{Wit}_{\sigma_0}\), with region decompositions \(\rho = \pi[1] \cdots \pi[k]\) and \(\rho' = \pi'[1] \cdots \pi'[k']\) and let \(h\) be the longest common prefix of \(\rho\) and \(\rho'\). Then there exists \(k_1 < k, k'\) such that \(h = \pi[1] \cdots \pi[k_1], \pi[\ell] = \pi'[\ell]\) for all \(\ell \in \{1, \ldots, k_1\}\) and \(\pi[k_1 + 1] \neq \pi'[k_1 + 1]\). Therefore, when \(\text{Wit}_{\sigma_0}\) is seen as a tree, the branching structure of this tree is respected by the sections.

Let \(R = (w, p, W)\) be a region and consider the set of all histories \(h\) such that \(\text{Reg}(h) = R\). Then all these histories are prefixes of each other and are prefixes of exactly \(|W|\) witnesses (as \(\text{Wit}_{\sigma_0}(h) = W\) for each such \(h\)). Therefore, the branching structure of \(\text{Wit}_{\sigma_0}\) is respected by the sections such that the associated regions are all pairwise distinct. The latter property is called the \textit{region-tree structure} of \(\text{Wit}_{\sigma_0}\).

We consider a \textit{compact} version \(c\text{Wit}_{\sigma_0}\) of \(\text{Wit}_{\sigma_0}\) defined as follows:

- each internal section \(\pi\) of \(\text{Wit}_{\sigma_0}\) is replaced by the elementary path \(c\pi\) obtained by eliminating all the cycles of \(\pi\). Each terminal section \(\pi\) of \(\text{Wit}_{\sigma_0}\) is replaced by a lasso \(c\pi = \pi_1' \cdot \pi_2'\) such that \(u\) is a vertex, \(\pi_1' \cdot \pi_2'\) is an elementary path, and \(\pi_1' u\pi_2' u\) is prefix of \(\pi\).
- each witness \(\rho\) of \(\text{Wit}_{\sigma_0}\), with region decomposition \(\rho = \pi[1] \cdots \pi[k]\) is replaced by \(c\rho = c\pi[1] \cdots c\pi[k]\) such that each \(\pi[\ell]\) is replaced by \(c\pi[\ell]\). Notice that as the region is constant inside the sections, the region decomposition of \(c\rho\) coincide with the sequence of its \(c\pi[\ell]\), \(\ell \in \{1, \ldots, k\}\).

Therefore, by construction of the compact witnesses, the region-tree structure of \(\text{Wit}_{\sigma_0}\) is kept by the set \(\{c\rho \mid \rho \in \text{Wit}_{\sigma_0}\}\) and for each \(c\rho \in c\text{Wit}_{\sigma_0}\),
\[
(\text{won}(c\rho), \text{pay}(c\rho)) = (\text{won}(\rho), \text{pay}(\rho)).
\]

We then construct the announced strategy \(\tilde{\sigma}_0\) that produces the set \(c\text{Wit}_{\sigma_0}\) of compact witnesses and after any deviation acts with the adequate punishing strategy (as mentioned in Lemma 12). More precisely, let \(g v\) be such that \(g\) is prefix of a compact witness and \(g v\) is not (Player 1 deviates from \(c\text{Wit}_{\sigma_0}\)). Then by definition of the compact witnesses, there exists a deviation \(h v\) such that \((\text{won}(g v), \text{pay}(g v)) = (\text{won}(h v), \text{pay}(h v)) = (w, p)\). Then from \(g v\) Player 0 switches to the punishing strategy \(\sigma^\text{pun}_{(v, w, p)}\).

\textbf{Lemma 13.} The strategy \(\tilde{\sigma}_0\) is a solution to the SPS problem for reachability SP games and its size is bounded exponentially in the size of the game \(G\).
Proof of Lemma 13. Let us first prove that $\tilde{\sigma}_0$ is a solution to the SPS problem. (i) By [5], the set of extended payoffs of plays in $cWit_{\sigma_0}$ is equal to the set of extended payoffs of witnesses in $Wit_{\sigma_0}$. This means that with $cWit_{\sigma_0}$, we keep the same set $P_{\sigma_0}$ and the objective $\Omega_0$ is satisfied along each compact witness. (ii) The punishing strategies used by $\tilde{\sigma}_0$ guarantee the satisfaction of the objective $\Omega_0 \cup \Omega^<_{\sigma_0}$ by Lemma 12. Therefore $\tilde{\sigma}_0$ is a solution to the SPS problem.

Let us now show that the memory size of $\tilde{\sigma}_0$ is bounded exponentially in the size of $G$. (i) By Lemma 12, each punishing strategy used by $\tilde{\sigma}_0$ is of exponential size and the number of punishing strategies is exponential. (ii) To produce the compact witnesses, $\tilde{\sigma}_0$ keeps in memory the current region and produces in a memoryless way the corresponding compact section (which is an elementary path or lasso). Thus the required memory size for producing $cWit_{\sigma_0}$ is the number of regions. By (4), every play in $cWit_{\sigma_0}$ traverses at most an exponential number of regions and there is an exponential number of such plays (equal to $|P_{\sigma_0}|$).

We now switch to parity SP games and state the following lemma whose proof follows the same line of arguments as those given for reachability objectives.

Lemma 14. The strategy $\tilde{\sigma}_0$ is a solution to the SPS problem for parity SP games and its size is bounded exponentially in the size of the game $G$.

Proof of Lemma 14. We highlight here the main differences from reachability SP games.

- As parity objectives are prefix-independent, we associate to each history $h$ of a play $\rho \in Wit_{\sigma_0}$, a singleton $Reg(h) = Wit_{\sigma_0}(h)$ instead of the triplet $(won(h), pay(h), Wit_{\sigma_0}(h))$ as in the case of reachability. This is because $(won(h), pay(h))$ does not make sense for prefix-independent objectives.

- For the definition of the compact witnesses, we proceed identically as for reachability by simply removing cycles inside each section with the exception of terminal sections. Given the terminal section $\pi[k]$ of a witness $\rho \in Wit_{\sigma_0}$, we replace it by a lasso $c\pi[k] = \pi_1^x(\pi_2^y)^\omega$ such that $c\pi[k]$ and $\pi[k]$ start at the same vertex, $Occ(c\pi[k]) = Occ(\pi[k])$, $Inf(c\pi[k]) = Inf(\pi[k])$, and $|\pi_1^x|, |\pi_2^y|$ is quadratic in $|V|$ [3] Proposition 3.1. Therefore, by construction, the objectives $\Omega_i$ satisfied by a witness $\rho$ are exactly the same as for its corresponding compact play $cp$.

We then construct the strategy $\tilde{\sigma}_0$ that produces the set $cWit_{\sigma_0}$ of compact witnesses and after any deviation $gv$ from $cWit_{\sigma_0}$ acts with the adequate punishing strategy $\sigma^\text{pun}_v$ (as mentioned in Lemma 11).

Lemmas 13 and 14 lead to Proposition 10. Using this proposition, we are now able to prove our result on the NEXPTIME-membership of reachability and parity SP games.

Proof of Theorem 9. We have established the existence of solutions to the SPS problem that use a finite memory bounded exponentially, both for reachability (Lemma 13) and for parity (Lemma 14) SP games. Let $\sigma_0$ be such a solution. As it is finite-memory, we can guess it as a Moore machine $\mathcal{M}$ with a set of memory states at most exponential in the size of $G$.

Let us explain how to verify that the guessed solution $\sigma_0$ is a solution to the SPS problem for parity objectives, i.e., every play in $Plays_{\sigma_0}$ which is $\sigma_0$-fixed Pareto-optimal satisfies the objective $\Omega_0$ of Player 0. First, we construct the cartesian product $G \times \mathcal{M}$ of the arena $G$ with the Moore machine $\mathcal{M}$ which is a graph whose infinite paths (starting from the initial vertex $v_0$ and the initial memory state) are exactly the plays consistent with $\sigma_0$. Second, to compute $P_{\sigma_0}$, we test for the existence of a play $\rho$ in $G \times \mathcal{M}$ with a given payoff $p = pay(\rho)$,
beginning with the largest possible payoff \( p = (1, \ldots, 1) \) and finishing with the smallest possible one \( p = (0, \ldots, 0) \). Verifying this corresponds to deciding whether there exists a play that satisfies an intersection of parity objectives. The latter property can be checked in polynomial time in the size of \( G \times M \) [17]. Third, to check that each Pareto optimal play in \( \text{Plays}_{\sigma_0} \) satisfies \( \Omega_0 \), we test for each \( p \in \text{Plays}_{\sigma_0} \) whether there exists a play that satisfies the objectives \( \Omega_i \) such that \( p_i = 1 \) as well as the objective \( \text{Plays}_{G \setminus \Omega_0} \). As the complement of a parity objective is again a parity objective, we use again the polynomial algorithm of [17].

As a consequence we have a \( \text{NEXPTIME} \) algorithm for parity SP games.

The case of reachability SP games is solved similarly with the following two differences. Concerning the second step, the existence of a play in \( G \times M \) that satisfies an intersection of reachability objectives can be checked in polynomial time by first extending this graph with a Boolean vector in \( \{0, 1\}^t \) keeping track of the objectives of Player 1 already satisfied. Notice that the resulting graph is still of exponential size and that the intersection of reachability objectives becomes a single reachability objective. Concerning the third step, as the complement \( \text{Plays}_{G \setminus \Omega_0} \) of \( \Omega_0 \) is not a reachability objective, we rather remove vertices of \( G \times \Omega_0 \) that contains an element of the target set \( T_0 \) before checking whether there exists a play that satisfies the objectives \( \Omega_i \) such that \( p_i = 1 \) for a given \( p \in \text{Plays}_{G \setminus \Omega_0} \).

\[\begin{align*}
\text{Figure 4} \quad & \text{The tree arena used in the reduction from the SC problem.} \\
\end{align*}\]
Player 0 has a strategy from \( v_0 \) in the corresponding SP game that is a solution to the SPS problem. The game is such that there is a solution to the SC problem if and only if Player 0 has a strategy from \( v_0 \) in \( G \) which is a solution to the SPS problem. The game is played between Player 0 with reachability objective \( \Omega_0 \) and Player 1 with \( n + 1 \) reachability objectives. The objectives are defined as follows: \( \Omega_0 = \text{Reach}(\{v_2\}) \), \( \Omega_i = \text{Reach}(\{e_i \} \cup \{S_j \mid e_i \in S_j \}) \) for \( i \in \{1, 2, \ldots, n\} \) and \( \Omega_{n+1} = \text{Reach}(\{v_2\}) \). First, notice that every play in \( G_1 \) is consistent with any strategy of Player 0 and is lost by that player. It holds that for each \( \ell \in \{1, 2, \ldots, n\} \), there is such a play with payoff \((p_1, \ldots, p_{n+1})\) such that \( p_\ell = 1 \) and \( p_j = 0 \) for \( j \neq \ell \). These payoffs correspond to the elements \( e_\ell \) we aim to cover in the SC problem. A play in \( G_2 \) visits \( v_2 \) and then a vertex \( c \) from which Player 0 selects a vertex \( S \). Such a play is always won by Player 0 and its payoff is \((p_1, \ldots, p_{n+1})\) such that \( p_{n+1} = 1 \) and \( p_\ell = 1 \) if and only if the element \( e_\ell \) belongs to the set \( S \). It follows that the payoff of such a play corresponds to a set of elements in the SC problem. It is easy to see that the following proposition holds and it follows that, as a consequence, Theorem 15 holds.

**Proposition 16.** There is a solution to an instance of the SC problem if and only if Player 0 has a strategy from \( v_0 \) in the corresponding SP game that is a solution to the SPS problem.

**Proof of Proposition 16** First, let us assume that there is a solution to the SC problem. It holds that there exists a set of \( k \) indexes \( i_1, i_2, \ldots, i_k \) such that the union of the corresponding sets equals the set \( C \) of elements we aim to cover. We define the strategy \( \sigma_0 \) as follows: \( \sigma_0(v_0v_2c_{i_j}) = S_{i_j} \). Let us show that this strategy is solution to the SPS problem by showing that any play with a \( \sigma_0 \)-fixed Pareto-optimal payoff is won by Player 0. This amounts to showing that for every play in \( G_1 \) there is a play in \( G_2 \) with a strictly larger payoff. This is sufficient as it makes sure that the payoff of plays in \( G_1 \) are not \( \sigma_0 \)-fixed Pareto-optimal and as every play in \( G_2 \) is won by Player 0. Let \( p = (p_1, \ldots, p_{n+1}) \) be the payoff of a play in \( G_1 \). It holds that \( p_\ell = 1 \) for some \( \ell \in \{1, 2, \ldots, n\} \) and \( p_j = 0 \) for \( j \neq \ell \). This corresponds to the element \( e_\ell \) in \( C \). Since the \( k \) indexes \( i_1, i_2, \ldots, i_k \) are a solution to the SC problem, it holds that there exists some index \( i_j \) such that \( e_\ell \in S_{i_j} \). It also holds that the play \( v_0v_2c_{j}(S_{i_j})^\omega \) is consistent with \( \sigma_0 \). Its payoff is \( p' = (p'_1, \ldots, p'_{n+1}) \) with \( p'_\ell = 1 \) since \( e_\ell \in S_{i_j} \) and \( p'_{n+1} = 1 \). It follows that payoff \( p' \) is strictly larger than \( p \).

Now, let us assume that Player 0 has a strategy \( \sigma_0 \) from \( v_0 \) that is a solution to the SPS problem. We can show that the set of indexes \( \{i_j \mid \sigma_0(v_0v_2c_j) = S_{i_j}, j \in \{1, \ldots, k\}\} \) is a solution to the SC problem. It is easy to see that since strategy \( \sigma_0 \) is a solution to the SPS problem, every payoff \( p \) in \( G_1 \) is strictly smaller than some payoff \( p' \) in \( G_2 \). It follows that in the SC problem, each element \( e \in C \) corresponding to \( p \) is contained in some set \( S \) corresponding to \( p' \). Since it also holds that \( S \subseteq C \) for each set \( S \), it follows that the sets mentioned above are an exact cover of \( C \).

### 4.3 NEXPTIME-Hardness

Let us come back to regular game arenas and show the NEXPTIME-hardness result for both reachability and parity SP games. Each type of objective is studied in a dedicated subsection.

**Theorem 17.** The SPS problem is NEXPTIME-hard for reachability SP games.

**Theorem 18.** The SPS problem is NEXPTIME-hard for parity SP games.

The NEXPTIME-hardness is obtained thanks to the succinct variant of the SC problem presented below.
4.3.1 Succinct Set Cover Problem

The Succinct Set Cover problem (SSC problem) is defined as follows. We are given a Conjunctive Normal Form (CNF) formula $\phi = C_1 \land C_2 \land \ldots \land C_p$ over the variables $X = \{x_1, x_2, \ldots, x_m\}$ made up of $p$ clauses, each containing some disjunction of literals of the variables in $X$. The set of valuations of the variables $X$ which satisfy $\phi$ is written $[\phi]$. We are also given an integer $k \in \mathbb{N}$ (encoded in binary) and an other CNF formula $\psi = D_1 \land D_2 \land \ldots \land D_q$ over the variables $X \cup Y$ with $Y = \{y_1, y_2, \ldots, y_n\}$, made up of $q$ clauses. Given a valuation $\text{val}_Y : Y \to \{0, 1\}$ of the variables in $Y$, called a partial valuation, we write $\psi[\text{val}_Y]$ the CNF formula obtained by replacing in $\psi$ each variable $y \in Y$ by its valuation $\text{val}_Y(y)$. We write $[\psi[\text{val}_Y]]$ the valuations of the remaining variables $X$ which satisfy $\psi[\text{val}_Y]$.

The SSC problem is to decide whether there exists a set $K = \{\text{val}_Y | \text{val}_Y : Y \to \{0, 1\}\}$ of $k$ valuations of the variables in $Y$ such that the valuations of the remaining variables $X$ which satisfy the formulas $\psi[\text{val}_Y]$ include the valuations of $X$ which satisfy $\phi$. Formally, we write this $[\phi] \subseteq \bigcup_{\text{val}_Y \in K} [\psi[\text{val}_Y]]$.

We can show that this corresponds to a set cover problem succinctly defined using CNF formulas. The set $[\phi]$ of valuations of $X$ which satisfy $\phi$ corresponds to the set of elements we aim to cover. Parameter $k$ is the number of sets that can be used to cover these elements. Such a set is described by a formula $\psi[\text{val}_Y]$, given a partial valuation $\text{val}_Y$, and its elements are the valuations of $X$ in $[\psi[\text{val}_Y]]$. This is illustrated in the following example.

**Example 19.** Consider the CNF formula $\phi = (x_1 \lor \neg x_2) \land (x_2 \lor x_3)$ over the variables $X = \{x_1, x_2, x_3\}$. The set of valuations of the variables which satisfy $\phi$ is $[\phi] = \{(1,1,1),(1,1,0),(1,0,1),(0,0,1)\}$. Each such valuation corresponds to one element we aim to cover. Consider the CNF formula $\psi = (y_1 \lor y_2) \land (x_1 \lor y_2) \land (x_2 \lor x_3 \lor y_1)$ over the variables $X \cup Y$ with $Y = \{y_1, y_2\}$. Given the partial valuation $\text{val}_Y$ of the variables in $Y$ such that $\text{val}_Y(y_1) = 0$ and $\text{val}_Y(y_2) = 1$, we get the CNF formula $\psi[\text{val}_Y] = (0 \lor 1) \land (x_1 \lor 1) \land (x_2 \lor x_3 \lor 0)$. This formula describes the contents of the set identified by the partial valuation (as a partial valuation yields a unique formula). The valuations of the variables $X$ which satisfy $\psi[\text{val}_Y]$ are the elements contained in the set. In this case, these elements are $[\psi[\text{val}_Y]] = \{(0,1,0),(0,0,1),(1,1,0),(1,0,1),(1,1,1)\}$. We can see that this set contains the elements $\{(1,1,1),(1,1,0),(1,0,1),(0,0,1)\}$ of $[\phi]$.

The following result is used in the proof of our NEXPTIME-hardness results and is of potential independent interest.

**Theorem 20.** The SSC problem is NEXPTIME-complete.

**Proof of Theorem 20.** It is easy to see that the SSC problem is in NEXPTIME. We can show that the SSC problem is NEXPTIME-hard by reduction from the Succinct Dominating Set problem (SDS problem) which is known to be NEXPTIME-complete for graphs succinctly defined using CNF formulas [13]. An instance of the SDS problem is defined by a CNF formula $\theta$ over two sets of $n$ variables $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$ and an integer $k$ (encoded in binary). The formula $\theta$ succinctly defines an undirected graph in the following way. The set of vertices is the set of all valuations of the $n$ variables in $X$ (or over the $n$ variables in $Y$) of which there are $2^n$. Let $\text{val}_X$ and $\text{val}_Y$ be two such valuations, representing two vertices. Then, there is an edge between $\text{val}_X$ and $\text{val}_Y$ if and only if $\theta[\text{val}_X, \text{val}_Y] = \text{true}$. An instance of the SDS problem is positive if there exists a set $K = \{\text{val}_X^1, \text{val}_X^2, \ldots, \text{val}_X^K\}$ of $k$ valuations of the variables in $X$, corresponding to $k$ vertices, such that all vertices in the graph are adjacent to a vertex in $K$. Formally, we write this $\bigcup_{\text{val}_X \in K} \{\text{val}_Y | \theta[\text{val}_X, \text{val}_Y] = \text{true}\} = 2^n$. 
The SDS problem can be reduced in polynomial time to the SSC problem as follows. We define the CNF formula $\phi$ over the set of variables $X$ such that the formula is empty. Therefore, the set $[\phi]$ is equal to the $2^n$ valuations of the variables in $X$. We then define the CNF formula $\psi$ over the set of variables $X$ and $Y$ such that it is the CNF equivalent to $\theta(X,Y) \lor \theta(Y,X)$. The latter formula has a size which is polynomial in the size of the CNF formula $\theta$ which defines the graph. We keep the same integer $k$. Then, it is direct to see that the instance of the SDS problem is positive if and only if the instance of SSC problem is positive. Indeed, there is a positive instance to the SDS problem if and only if there exists a set $K$ of $k$ valuations of the variables in $Y$ such that $[\phi] \subseteq \bigcup_{valY \in K} [\psi[valY]]$.

4.3.2 NEXPTIME-Hardness of Reachability SP Games

We now describe in details our reduction from the SSC problem which allows us to show the NEXPTIME-hardness of solving the SPS problem in reachability SP games.

Given an instance of the SSC problem, we construct a reachability SP game with arena $G$ consisting of a polynomial number of vertices in the number of clauses and variables in the formulas $\phi$ and $\psi$ and in the length of the binary encoding of the integer $k$. This reduction is such that there is a solution to the SSC problem if and only if Player 0 has a strategy from $v_0$ in $G$ which is a solution to the SPS problem. The arena $G$, provided in Figure 6 can be viewed as three sub-arenas reachable from $v_0$. We call these sub-arenas $G_1$, $G_2$ and $G_3$. Sub-arena $G_3$ starts with a gadget $Q_k$ whose vertices belong to Player 1 and which provides exactly $k$ different paths from $v_0$ to $v_3$.

**Gadget $Q_k$.** Parameter $k$ can be represented in binary using $r = \lfloor \log_2(k) \rfloor + 1$ bits. It also holds that the binary encoding of $k$ corresponds to the sum of at most $r$ powers of 2. Given the binary encoding $b_0 b_1 \ldots b_{r-1}$ of $k$ such that $b_i \in \{0,1\}$, let $ones = \{i \in \{0,\ldots,r-1\} \mid b_i = 1\}$. It holds that $k = \sum_{i \in ones} 2^i$. Our gadget $Q_k$ is a graph with a polynomial number of vertices (in the length of the binary encoding of $k$) such that all these vertices belong to Player 1. For each $i \in ones$ there is $2^i$ different paths from the initial vertex $\alpha$ to vertex $\beta$. Therefore, it holds that in $Q_k$ there are $k$ different paths from vertex $\alpha$ to vertex $\beta$.

**Example 21.** Let $k = 11$, it holds that it can be represented in binary using $\lfloor \log_2(11) \rfloor + 1 = 4$ bits. The binary representation of 11 is 1011 and it can be obtained by the following sum $2^3 + 2^1 + 2^0$. The gadget $Q_{11}$ is detailed in Figure 5.
**Objectives.** The game is played between Player 0 with reachability objective $\Omega_0$ and Player 1 with $t = 1 + 2 \cdot m + p + q$ reachability objectives. The payoff of a play therefore consists in a single Boolean for objective $\Omega_1$, a vector of $2 \cdot m$ Booleans for objectives $\Omega_{x_1}, \Omega_{\neg x_1}, \ldots, \Omega_{x_m}, \Omega_{\neg x_m}$, a vector of $p$ Booleans for objectives $\Omega_{C_1}, \ldots, \Omega_{C_p}$ and a vector of $q$ Booleans for objectives $\Omega_{D_1}, \ldots, \Omega_{D_q}$. The objectives are defined as follows.

- The target set for objective $\Omega_0$ of Player 0 and objective $\Omega_1$ of Player 1 is $\{v_2, v_3\}$.
- The target set for objective $\Omega_{x_i}$ (resp. $\Omega_{\neg x_i}$) with $i \in \{1, \ldots, m\}$ is the set of vertices labeled $x_i$ (resp. $\neg x_i$) in $G_1, G_2$ and $G_3$.
- The target set for objective $\Omega_{C_i}$ with $i \in \{1, \ldots, p\}$ is the set of vertices in $G_1$ and $G_3$ corresponding to the literals of $X$ which make up the clause $C_i$ in $\phi$. In addition, vertex $i_j$ in $G_2$ belongs to the target set of objective $\Omega_{C_i}$ for all $\ell \in \{1, \ldots, p\}$ such that $\ell \neq j$.
- The target set of objective $\Omega_{D_i}$ with $i \in \{1, \ldots, q\}$ is the set of vertices in $G_3$ corresponding to the literals of $X$ and $Y$ which make up the clause $D_i$ in $\psi$. In addition, vertices $v_1$ and $v_2$ satisfy every objective $\Omega_{D_i}$ with $i \in \{1, \ldots, q\}$.

**Sub-arenas $G_1$ and $G_2$.** In each sub-arena $G_1$ and $G_2$, for each variable $x_i \in X$, there is one choice vertex controlled by Player 1 which leads to $x_i$ and $\neg x_i$. These vertices have the next choice vertex as their successor, except for vertices $x_m$ and $\neg x_m$ which have a self loop. In $G_2$, there is also a vertex $v_2$ controlled by Player 1 with $p$ successors, each leading to the first choice vertex for the variables in $X$. Sub-arenas $G_1$ and $G_2$ are completely controlled by Player 1. Plays entering these sub-arenas are therefore consistent with any strategy of Player 0.

**Payoff of Plays in $G_1$.** Plays in $G_1$ do not satisfy objective $\Omega_0$ of Player 0 nor objective $\Omega_1$ of Player 1. A play in $G_1$ is of the form $v_0, v_1, z_1 \square \cdots \square (z_m)^\omega$ where $z_i$ is either $x_i$ or $\neg x_i$. It follows that a play satisfies the objective $\Omega_{x_i}$ or $\Omega_{\neg x_i}$ for each $x_i \in X$. The vector of payoffs for these objectives corresponds to a valuation of the variables in $X$, expressed as a vector of $2 \cdot m$ Booleans. In addition, due to the way the objectives are defined, objective $\Omega_{C_i}$ is satisfied in a play if and only if clause $C_i$ of $\phi$ is satisfied by the valuation this play corresponds to. The objective $\Omega_{D_i}$ for $i \in \{1, \ldots, q\}$ is satisfied in every play in $G_1$.

**Lemma 22.** Plays in $G_1$ are consistent with any strategy of Player 0. Their payoff are of the form $(0, val, sat(\phi, val), 1, \ldots, 1)$ where $val$ is a valuation of the variables in $X$ expressed
as a vector of payoffs for objectives $\Omega_x$ to $\Omega_{\neg x_m}$ and $\text{sat}(\phi, \text{val})$ is the vector of payoffs for objectives $\Omega_{C_i}$ to $\Omega_{C_p}$ corresponding to that valuation. All plays in $G_1$ are lost by Player 0.

**Payoff of Plays in $G_2$.** Plays in $G_2$ satisfy the objectives $\Omega_0$ of Player 0 and $\Omega_2$ of Player 1. A play in $G_2$ is of the form $v_0 v_2 i_2 \square z_1 \square \cdots \square (z_m) \omega$ where $z_\ell$ is either $x_\ell$ or $\neg x_\ell$. It follows that a play satisfies either the objective $\Omega_x$ or $\Omega_{\neg x}$ for each $x \in X$ which again corresponds to a valuation of the variables in $X$. The objective $\Omega_D_i$ for $i \in \{1, \ldots, q\}$ is satisfied in every play in $G_2$. Compared to the plays in $G_1$, the difference lies in the objectives corresponding to clauses of $\phi$ which are satisfied. In any play in $G_2$, a vertex $i_j$ with $j \in \{1, \ldots, p\}$ is first visited, satisfying all the objectives $\Omega_{C_i}$, with $\ell \in \{1, \ldots, p\}$ and $\ell \neq j$. All but one objective corresponding to the clauses of $\phi$ are therefore satisfied.

**Lemma 23.** Plays in $G_2$ are consistent with any strategy of Player 0. Their payoff are of the form $(1, \text{val}, \text{vec}, 1, \ldots, 1)$ where $\text{val}$ is a valuation of the variables in $X$ expressed as a vector of payoffs for objectives $\Omega_x$ to $\Omega_{\neg x_m}$ and $\text{vec}$ is a vector of payoffs for objectives $\Omega_{C_1}$ to $\Omega_{C_p}$ in which all of them except one are satisfied. All plays in $G_2$ are won by Player 0.

From the two previous lemmas, we can state the following lemma when considering the payoffs of plays in $G_1$ and $G_2$.

**Lemma 24.** For every play in $G_1$ which corresponds to a valuation of the variables in $X$ that does not satisfy $\phi$, there is a play in $G_2$ with a strictly larger payoff.

**Proof of Lemma 24** Let $\rho$ be a play in $G_1$ which corresponds to a valuation of the variables in $X$ that does not satisfy $\phi$. It follows that at least one objective, say $\Omega_{C_1}$, is not satisfied in $\rho$ as at least one clause of $\phi$ (clause $C_1$) is not satisfied by that valuation. Let us consider the play $\rho'$ in $G_2$ which visits vertex $i_\ell$ and after visits the vertices corresponding to the same valuation of the variables in $X$ as $\rho$. By Lemmas 22 and 23 it follows that the payoff of $\rho'$ is strictly larger than that of $\rho$ (as we have $(0, \text{val}, \text{sat}(\phi, \text{val}), 1, \ldots, 1) < (1, \text{val}, \text{vec}, 1, \ldots, 1)$ with $\text{sat}(\phi, \text{val}) \leq \text{vec}$).

The following lemma is a consequence of Lemma 24.

**Lemma 25.** The set of payoffs of plays in $G_1$ that are $\sigma_0$-fixed Pareto-optimal when considering $G_1 \cup G_2$ for any strategy $\sigma_0$ of Player 0 is equal to the set of payoffs of plays in $G_1$ whose valuation of $X$ satisfy $\phi$.

**Proof of Lemma 25** This property stems from the following observations. First, any play in $G_1$ which satisfies every objective $\Omega_{C_i}$ with $i \in \{1, \ldots, p\}$, and therefore corresponds to a valuation of $X$ which satisfies $\phi$, has a payoff that is incomparable to every possible payoff in $G_2$. This is because such a play satisfies more objectives in $\Omega_{C_1}, \ldots, \Omega_{C_p}$ than the plays in $G_2$ but does not satisfy objective $\Omega_i$ while the plays in $G_2$ do. Second, every other play in $G_1$ has a strictly smaller payoff than at least one play in $G_2$ due to Lemma 24 and its payoff is therefore not $\sigma_0$-fixed Pareto-optimal.

**Problematic Payoffs in $G_1$.** The plays described in the previous lemma correspond exactly to the valuations of $X$ which satisfy $\phi$ and therefore to the elements we aim to cover in the SSC problem. They are $\sigma_0$-fixed Pareto-optimal when considering $G_1 \cup G_2$ and are lost by Player 0. All other $\sigma_0$-fixed Pareto-optimal payoffs in $G_1 \cup G_2$ are only realized by plays in $G_2$ which are all won by Player 0. It follows that in order for Player 0 to find a strategy $\sigma_0$ from $v_0$ that is solution to the SPS problem, it must hold that those payoffs are not $\sigma_0$-fixed.

[Note: The content continues with more detailed explanations and proofs related to the Stackelberg-Pareto synthesis and the related lemmas and theorems.]

---

22 Stackelberg-Pareto Synthesis (full version)
Pareto-optimal when considering $G_1 \cup G_2 \cup G_3$. Otherwise, a play consistent with $\sigma_0$ with a $\sigma_0$-fixed Pareto-optimal payoff is lost by Player 0. We call those payoffs problematic payoffs.

In order for Player 0 to find a strategy $\sigma_0$ which is a solution to the SPS problem, this strategy must be such that for each problematic payoff in $G_1$, there is a play in $G_3$ consistent with $\sigma_0$ and with a strictly larger payoff. Since the plays in $G_3$ are all won by Player 0, this would ensure that the strategy $\sigma_0$ is a solution to the problem. This corresponds in the SSC problem to selecting a series of sets in order to cover the valuations of $X$ which satisfy $\phi$.

**Sub-arena $G_3$.** Sub-arena $G_3$ starts with gadget $Q_k$ whose vertices are controlled by Player 1. Then, for each variable $y_i \in Y$, there is one choice vertex controlled by Player 0 which leads to $y_i$ and $\neg y_i$. These vertices have the next choice vertex as their successor, except for $y_n$ and $\neg y_n$ which lead to the first choice vertex for the variables in $X$.

**Payoff of Plays in $G_3$.** Plays in $G_3$ satisfy the objectives $\Omega_0$ of Player 0 and $\Omega_1$ of Player 1. A play in $G_3$ consistent with a strategy $\sigma_0$ is of the form $v_0 \Box \ldots \Box v_3 r_1 \Box \ldots \Box r_n \Box z_1 \Box \ldots \Box (z_m)\Box$ where $r_i$ is either $y_i$ or $\neg y_i$ and $z_i$ is either $x_i$ or $\neg x_i$. Since only the vertices leading to $y$ or $\neg y$ for $y \in Y$ belong to Player 0, it holds that $v_3 r_1 \Box \ldots \Box r_n$ is the only part of any play in $G_3$ which is directly influenced by $\sigma_0$. That part of a play comes after a history from $v_0$ to $v_3$ of which there are $k$, provided by gadget $Q_k$. By definition of a strategy, this can be interpreted as Player 0 making a choice of valuation of the variables in $Y$ after each of those $k$ histories. After this, the play satisfies either the objective $\Omega_x$ or $\Omega_{\neg x}$ for each $x \in X$ which corresponds to a valuation of $X$. Due to the way the objectives are defined, the objective $\Omega_{C_i}$ (resp. $\Omega_{D_i}$) is satisfied if and only if clause $C_i$ of $\phi$ (resp. $D_i$ of $\psi$) is satisfied by the valuation of the variables in $X$ (resp. $X$ and $Y$) the play corresponds to.

**Creating Strictly Larger Payoffs in $G_3$.** In order to create a play with a payoff $r'$ that is strictly larger than a problematic payoff $r$, $\sigma_0$ must choose a valuation of $Y$ such that there exists a valuation of the remaining variables $X$ which together with this valuation of $Y$ satisfies $\psi$ and $\phi$ (since in $r$ every objective $\Omega_{C_i}$ for $i \in \{1, \ldots, p\}$ and $\Omega_{D_i}$ for $i \in \{1, \ldots, q\}$ is satisfied). Since the plays in $G_3$ also satisfy the objective $\Omega_1$ and plays in $G_1$ do not, this ensures that $r < r'$.

We can finally establish that our reduction is correct.

**Proposition 26.** Player 0 has a strategy $\sigma_0$ from $v_0$ in $G$ that is a solution to the SPS problem if and only if there is a solution to the corresponding instance of the SSC problem.

**Proof of Proposition 26.** Let us assume that $\sigma_0$ is a solution to the SPS problem in $G$ and show that there is a solution to the SSC problem. Let $val_X$ be a valuation of the variables in $X$ which satisfies $\phi$. This valuation corresponds to a play in $G_1$ with a problematic payoff $r$. Since the objective of Player 0 is not satisfied in that play and since $\sigma_0$ is a solution to the SPS problem, it holds that $r$ is not $\sigma_0$-fixed Pareto-optimal. It follows that there exists a play in $G_3$ that is consistent with $\sigma_0$ and whose payoff is strictly larger than $r$. As described above, such a play corresponds to a valuation $val_Y$ of the variables in $Y$ such that $val_X \in [\psi[val_Y]]$. Since this can be done for each $val_X \in [\phi]$ and since there is a set $K$ of $k$ possible valuations $val_Y$ in $G_3$, it holds that $[\phi] \subseteq \bigcup_{val_Y \in K} [\psi[val_Y]]$.

Let us now assume that there is a solution to the SSC problem and show that we can construct a strategy $\sigma_0$ that is solution to the SPS problem. Let $K$ be the set of $k$ valuations $val_Y$ of the variables in $Y$ which is a solution to the SSC problem. Since there are $k$ possible
histories from $v_0$ to $v_3$ in $G_3$ provided by the gadget $Q_k$ described previously, we define $\sigma_0$ such that the $n$ vertices $y_i$ or $\neg y_i$ for $i \in \{1, \ldots, n\}$ visited after each history correspond to a valuation in $K$. We can now show that this strategy is a solution to the SPS problem. We do this by showing that each play $\rho$ with problematic payoff $r$ in $G_1$ has a strictly smaller payoff than that of some play $\rho'$ with payoff $r'$ in $G_3$. Such a payoff $r$ corresponds to a valuation $\text{val}_X \in K$. Since $K$ is a solution to the SSC problem, it holds that there exists some valuation $\text{val}_Y \in K$ such that $\text{val}_X \in [\text{val}_Y]$. It follows, given the definition of $\sigma_0$, that there exists a play $\rho'$ in $G_3$ corresponding to that valuation $\text{val}_Y$ and which visits the vertices $x$ or $\neg x$ for each $x \in X$ such that it corresponds to the valuation $\text{val}_X$. Given the properties mentioned before, the payoff $r'$ of this play is such that $r < r'$.

The previous proof yields our result on the NEXPTIME-hardness of the SPS problem in reachability SP games.

4.3.3 NEXPTIME-Hardness of Parity SP Games

We now provide the NEXPTIME-hardness result for parity SP games. The proof of this result follows the same ideas used in the proof for reachability SP games. It again uses a reduction from the SSC problem in which we construct an arena $G_i$ and its structure of three sub-arenas $G_1$, $G_2$, and $G_3$ is kept. We describe the main difficulties that we encounter when adapting this proof for parity objectives and how to overcome them by modifying each sub-arena $G_i$ into $G'_i$. The modified arena $G'$ is depicted in Figure 8.

Two Particular Objectives. Remember that for the case of reachability SP games, the objective $\Omega_0$ of Player 0 and the first objective $\Omega_1$ of Player 1 were either always satisfied or always not satisfied in all plays of a given sub-arena $G_1$, $G_2$, or $G_3$. This property holds in the modified arena $G'$, with these objectives being expressed using parity conditions.

On the Choice of Valuations. Recall that for reachability objectives, we encoded the choice of valuations for the variables $x_i \in X$ made by Player 1 by using sub-arena $G_1$ of Figure 6 (which was also reused as part of $G_2$ and $G_3$). A play in this sub-arena encodes a valuation by visiting one literal $l_i \in \{x_i, \neg x_i\}$ for each $x_i \in X$ thus satisfying the objectives $\Omega_i$, $i \in \{1, \ldots, m\}$.

This simple schema cannot be reused in the case of parity objectives as they are prefix-independent objectives. Instead, we ask Player 1 to repeatedly produce the same choice along loops (from $v_1$ back to $v_1$) in the adapted gadget of Figure 7.

We associate with each variable $x_i \in X$ two parity objectives with priority function $c_i$ with $l_i \in \{x_i, \neg x_i\}$ defined as follows: $c_i(l_i) = 2$, $c_i(\neg l_i) = 1$ and $c_i(v) = 3$ for all the other

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{figure7}
\caption{The repeating structure used in the reduction from the SSC problem for parity SP games.}
\end{figure}
Figure 8 The arena $G'$ used in the reduction from the SSC problem for parity SP games.
Thus with the proposed encoding with priority functions, there are several ways to observe valuation vertices. It is easy to see that plays in which the valuation changes infinitely many times (for example, visiting $x_i$ and $\neg x_i$ infinitely often), have a payoff strictly smaller than some other play which settles on a choice of valuation for each variable. Indeed, the payoff for objectives $\Omega_{x_1}$ and $\Omega_{\neg x_1}$ is $(0,0)$ in the first case and $(1,0)$ or $(0,1)$ in the second. We say that a valuation is properly encoded if Player 1 eventually repeats the same choice along the loops to settle on a valuation.

Notice that the gadget of Figure 7 appears nearly identical as part of the three sub-arenas $G'_1$, $G'_2$, and $G'_3$ of Figure 8. In all of these sub-arenas, we define the values of each priority function $c_i$ with $l_i \in \{x_i, \neg x_i\}$ exactly as explained above.

**On the Satisfied Clauses.** Recall that in the case of reachability SP games, we associated one reachability objective with each clause $C_j$ (resp. $D_j$) of $\phi$ (resp. $\psi$). As encoding valuations is made more complex by the prefix-independency of parity objectives, we also need to adapt the way we check which clauses are satisfied by a given valuation.

Let us first explain our encoding for clauses $C_j$ of $\phi$. We associate one parity objective with priority function $c_i^{C_j}$ with each literal $l$ of each clause $C_j$. Therefore, if $C_j = l_1^{C_j} \lor \ldots \lor l_{n_{C_j}}^{C_j}$, there are $n_{C_j}$ parity objectives for clause $C_j$. Priority function $c_i^{C_j}$ is defined as follows for each vertex $l_i \in \{x_i, \neg x_i\}$ that appears in $G'_1$ and $G'_2$ (we will define it later for $G'_3$): $c_i^{C_j}(l_i) = 2$ if $l = l_i$ and $c_i^{C_j}(l_i) = 1$ if $l = \neg l_i$, and for all the other vertices $v$, we have $c_i^{C_j}(v) = 3$. This encoding of clauses has the following important property: given a valuation $\val_X$ of the variables in $X$ properly encoded by Player 1, a clause $C_j$ is satisfied by $\val_X$ if and only if the parity condition $\Parity(c_i^{C_j})$ is satisfied for at least one of the literals $l$ of $C_j$. Thus with the proposed encoding with priority functions, there are several ways to observe that a clause $C_j$ is satisfied (the corresponding payoff is a non-null vector of $n_{C_j}$ Booleans).

In the sub-arena $G'_3$, we see a part resembling the gadget of Figure 7 however made of vertices $y_i$, $i \in \{1, \ldots, n\}$. This part is related to the choice of a valuation of the variables in $Y$ made by Player 0 with respect to $\psi$. To encode clauses $D_j$ of $\psi$, we proceed exactly as we did previously with clauses $C_j$ of $\phi$. We associate one priority function $c_i^{D_j}$ with each literal $l$ of each clause $D_j$ (recall that such a literal uses both sets of variables $X$ and $Y$). We similarly define the values of $c_i^{D_j}$ for vertices of $G'_3$: for $l' \in \{x_i, \neg x_i \mid i \in \{1, \ldots, m\}\} \cup \{y_i, \neg y_i \mid i \in \{1, \ldots, n\}\}$, we define $c_i^{D_j}(l') = 2$ if $l = l'$ and $c_i^{D_j}(l') = 1$ if $l = \neg l'$, and for all the other vertices $v$ of $G'_3$, we define $c_i^{D_j}(v) = 3$. Notice that the definition of $c_i^{D_j}$ is given for $G'_3$ only. We will later give its definition for $G'_1$ and $G'_2$.

**Modifications Needed on $G_1$.** Let us now explain how to modify $G_1$ into $G'_1$. Remember that in the case of reachability SP games, the objective associated with each clause $D_j$ of $\psi$ is satisfied by all plays in $G_1$. Indeed the purpose of $G_1$ (in combination with $G_2$) is to isolate all the encodings of the valuations of $X$ that satisfy $\phi$ independently of $\psi$. In case of a positive instance of the SSC problem, the payoff of these encodings are strictly smaller than that of some play in $G_3$, which have to satisfy all clauses of $\psi$ by definition of this problem.

We proceed similarly in $G'_1$. However as there are several ways to satisfy $D_j$ in $\psi$ (at least one of its literals has to be satisfied), we let Player 0 choose which way to do it. This is encoded by the part of $G'_1$ made with vertices $D_j$, $j \in \{1, \ldots, q\}$, controlled by Player 0, and their successors $l_1^{D_j}, \ldots, l_{n_{D_j}}^{D_j}$ such that $D_j = l_1^{D_j} \lor \ldots \lor l_{n_{D_j}}^{D_j}$. Given a properly encoded $X$ valuation $\val_X$ made by Player 1, Player 0 chooses a $Y$ valuation $\val_Y$ such that if $\val_X \in [\phi]$, then $\val_Y \in [\psi[\val_X]]$. Player 0 makes such a choice by selecting at least one literal $l_{D_j}$ of $D_j$, for each clause $D_j$ of $\psi$, such that $l$ is satisfied by the valuation made of $\val_X$ and
We then turned to the complexity class of the SPS problem and provided a proof of its membership, which relied on showing that any solution to the SPS problem in any arena \(G \in \{\psi\} \) can be transformed into a solution with an exponential encoding \(G' \in \{\psi[\psi]\} \) whenever \(val_X \in \{\psi\} \). As a consequence, the objectives that are satisfied by the plays in \(G \) are exactly (i) those associated with a valuation \(val_X \), (ii) the objectives associated with all clauses \(D_j \) of \(\psi \), and (iii) the objectives associated with all clauses \(C_j \) of \(\phi \) except one.

We achieve the same requirement for parity SP games by using \(G_2 \) in place of \(G_2 \). In this sub-arena, Player 1 first selects one clause \(C' \) in \(\phi \) and then in the selecting part of \(G_2' \), the priority functions are defined as follows.

- We use the priority functions \(c_i \) with \(l = \{x_i, \neg x_i \mid i \in \{1, \ldots, m\}\} \) as defined above for encoding \(X \) valuations.
- The priority functions \(c_i^{D_j} \) are all defined such that the associated objective \(\text{Parity}(c_i^{D_j}) \) is satisfied.
- Similarly the priority functions \(c_i^{C_j} \) are defined such that the associated objective \(\text{Parity}(c_i^{C_j}) \) is satisfied, except for all priority functions \(c_i^{C_j} \) such that \(C_j = C \) for which this objective is not satisfied.

Modifications Needed on \(G_3 \). We have modified \(G_1 \) into \(G_1' \) and \(G_2 \) into \(G_2' \) such that the only plays in \(G_1' \) with a Pareto-optimal payoff when considering \(G_1' \cup G_2' \), given any strategy of Player 0 are those that encode valuations \(val_X \in \{\psi\} \). In \(G_1' \), after such a valuation chosen by Player 1, Player 0 indicates which valuation \(val_Y \) to use such that \(val_Y \in \{\psi[\psi]\} \). He chooses this valuation \(val_Y \) by indicating each clause \(D_j \) which literals of \(D_j \) he has chosen such that the valuation made of \(val_X \) and \(val_Y \) satisfies \(D_j \).

Let us now explain how to modify \(G_3 \) into \(G_3' \). After each of the \(k \) histories produced by gadget \(Q_k \), both players have to choose some valuation (resp. \(val_X \) and \(val_Y \)) for the variables that they control. In case of a positive instance of SSC problem, Player 0 will be able to select one of the valuations \(val_Y \) that he used in \(G_1' \) such that \(val_X \in \{\psi[\psi]\} \) whenever \(val_X \in \{\psi\} \). It follows that plays in \(G_3' \) have a larger payoff than the Pareto-optimal payoffs of \(G_1' \cup G_2' \). In case of a negative instance of SSC problem, Player 0 will not be able to do so.

Clearly there exists a solution to the SPS problem in the modified arena \(G' \) if and only if the instance of the SSC problem is positive.

## Conclusion

We have introduced in this paper the class of two-player SP games and the SPS problem in those games. We provided a reduction from SP games to a two-player zero-sum game called the C-P game in order to provide FPT results on solving this problem. We then showed how the arena and the generic objective of this C-P game can be adapted to specifically handle reachability and parity SP games. This allowed us to prove that reachability (resp. parity) SP games are in FPT when the number \(t \) of objectives of Player 1 (resp. when \(t \) and the maximal priority according to each priority function in the game) is a parameter. We then turned to the complexity class of the SPS problem and provided a proof of its \(\text{NEXPTIME} \)-membership, which relied on showing that any solution to the SPS problem in a reachability or parity SP game can be transformed into a solution with an exponential
memory. We provided a proof of the \( \text{NP} \)-completeness of the problem in the simple setting of reachability \( SP \) games played on tree arenas. We then came back to regular game arenas and provided the proof of the \( \text{NEXPTIME} \)-hardness of the \( SPS \) problem in reachability and parity \( SP \) games. This proof relied on a reduction from the \( \text{SSC} \) problem which we proved to be \( \text{NEXPTIME} \)-complete, a result of potential independent interest.

In future work, we want to study other \( \omega \)-regular objectives as well as quantitative objectives such as mean-payoff in the framework of \( SP \) games and the \( SPS \) problem. It would also be interesting to study whether other works, such as rational synthesis, could benefit from the approaches used in this paper.

References

1. Mrudula Balachander, Shibashis Guha, and Jean-François Raskin. Fragility and robustness in mean-payoff adversarial Stackelberg games. CoRR, abs/2007.07209, 2020. URL: https://arxiv.org/abs/2007.07209, arXiv:2007.07209.
2. Dietmar Berwanger. Admissibility in infinite games. In Wolfgang Thomas and Pascal Weil, editors, STACS 2007, 24th Annual Symposium on Theoretical Aspects of Computer Science, Aachen, Germany, February 22-24, 2007, Proceedings, volume 4393 of Lecture Notes in Computer Science, pages 188–199. Springer, 2007. doi:10.1007/978-3-540-70918-3_17.
3. Patricia Bouyer, Romain Brenguier, Nicolas Markey, and Michael Ummels. Pure Nash equilibria in concurrent deterministic games. Log. Methods Comput. Sci., 11(2), 2015. doi:10.2168/LMCS-11(2:9)2015.
4. Romain Brenguier, Lorenzo Clemente, Paul Hunter, Guillermo A. Pérez, Mickael Randour, Jean-François Raskin, Ocan Sankur, and Mathieu Sassolas. Non-zero sum games for reactive synthesis. In Adrian-Horia Dediu, Jan Janousek, Carlos Martín-Vide, and Bianca Truthe, editors, Language and Automata Theory and Applications - 10th International Conference, LATA 2016, Prague, Czech Republic, March 14-18, 2016, Proceedings, volume 9618 of Lecture Notes in Computer Science, pages 3–23. Springer, 2016. doi:10.1007/978-3-319-30000-9\_1.
5. Romain Brenguier, Jean-François Raskin, and Ocan Sankur. Assume-admissible synthesis. In Luca Aceto and David de Frutos-Escrig, editors, 26th International Conference on Concurrency Theory, CONCUR 2015, Madrid, Spain, September 1-4, 2015, volume 42 of LIPIcs, pages 100–113. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015. doi:10.4230/LIPIcs.CONCUR.2015.100.
6. Léonard Brice, Jean-François Raskin, and Marie van den Bogaard. Subgame-perfect equilibria in mean-payoff games. CoRR, abs/2101.10685, 2021. URL: https://arxiv.org/abs/2101.10685, arXiv:2101.10685.
7. Thomas Brihaye, Véronique Bruyère, Aline Goeminne, Jean-François Raskin, and Marie van den Bogaard. The complexity of subgame perfect equilibria in quantitative reachability games. Log. Methods Comput. Sci., 16(4), 2020. URL: https://lmcs.episciences.org/6883.
8. Véronique Bruyère. Computer aided synthesis: A game-theoretic approach. In Emilie Charlier, Julien Leroy, and Michel Rigo, editors, Developments in Language Theory - 21st International Conference, DLT 2017, Liège, Belgium, August 7-11, 2017, Proceedings, volume 10396 of Lecture Notes in Computer Science, pages 3–35. Springer, 2017. doi:10.1007/978-3-319-62809-7\_1.
9. Véronique Bruyère, Quentin Hautem, and Jean-François Raskin. Parameterized complexity of games with monotonically ordered omega-regular objectives. In Sven Schewe and Lijun Zhang, editors, 29th International Conference on Concurrency Theory, CONCUR 2018, September 4-7, 2018, Beijing, China, volume 118 of LIPIcs, pages 29:1–29:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.CONCUR.2018.29.
10. Krishnendu Chatterjee and Monika Henzinger. Efficient and dynamic algorithms for alternating Büchi games and maximal end-component decomposition. J. ACM, 61(3):15:1–15:40, 2014. doi:10.1145/2597631.
A Useful Result on SP Games

Proposition 27. Every parity (resp. reachability) SP game $G$ with arena $G$ containing $n$ vertices can be transformed into a parity (resp. reachability) SP game $\bar{G}$ with arena $\bar{G}$ containing at most $n^2$ vertices such that any vertex in $\bar{G}$ has at most 2 successors and Player 0 has a strategy $\sigma_0$ that is solution to the SPS problem in $G$ if and only if Player 0 has a strategy $\bar{\sigma}_0$ that is solution to the problem in $\bar{G}$. 
Proof of Proposition 27. Let $G$ be an SP game with arena $\tilde{G}$. Let us first describe the arena $\tilde{G}$ of $\tilde{G}$. Let $v \in V$ be a vertex of $G$, then $v$ is also a vertex of $\tilde{G}$ such that it belongs to the same player and is the root of a complete binary tree with $\ell = |\{v' \mid (v, v') \in E\}|$ leaves if $(v, v') \not\in E$. Otherwise, $v$ has a self loop and its other successor is the root of such a tree with $\ell - 1$ leaves. The internal vertices of the tree (that is vertices which are not $v$, nor the leaves) belong to the same player as $v$. Each leaf vertex $v'$ of this tree is such that $(v, v') \in E$, belongs to the same player as in $G$ and is again the root of its own tree. The initial vertex $v_0$ of $G$ remains unchanged in $\tilde{G}$. Since every vertex in $\tilde{G}$ is part of a binary tree or has a self loop and a single successor, it holds that it has at most two successors. Since $G$ is a game arena, this transformation is such that each vertex in $\tilde{G}$ has at least one successor. It follows that $\tilde{G}$ is a game arena containing $n$ vertices $v \in V$ and at most $n - 1$ internal vertices per tree in the case where $v \in V$ has $n$ successors in $G$. It follows that the number of vertices in $\tilde{G}$ is at most $n + n \cdot (n - 1) = n^2$. If $G$ is a reachability SP game, the target sets remain unchanged in $\tilde{G}$. For parity SP games, the priority function $c$ remains unchanged for vertices $v \in V$ and we define $c(v') = c(v)$ for $v' \in \tilde{V} \setminus V$ such that $v'$ is an internal vertex of a tree whose root is $v$.

Let us now show that there is a solution to the SPS problem in $G$ if and only if there is a solution in $\tilde{G}$. From each root $v$ of a tree in $\tilde{G}$ (corresponding to a vertex $v$ of Player $i$ in $G$) there is a set of $\ell = |\{v' \mid (v, v') \in E\}|$ different paths controlled by Player $i$, each leading to a vertex $v'$. It follows that there exists a play $\rho = v_0v_1v_2 \ldots \in \text{Plays}_G$ if and only if there exists a play $\rho' = a_0a_1a_2 \ldots b_0v_0v_1b_1 \ldots b_{n-1}v_2 \ldots \in \text{Plays}_{\tilde{G}}$ such that every vertex $a_i$ (resp. $b_i$) belongs to the same player as $v_0$ (resp. $v_1$) and so on. Given the way the objectives are defined, in the case of reachability or parity SP games, it holds that $\text{pay}(\rho) = \text{pay}(\rho')$ and $\text{won}(\rho) = \text{won}(\rho')$. Therefore, a strategy $\sigma_0$ that is solution to the SPS problem in $G$ can be transformed into a strategy $\tilde{\sigma}_0$ which is a solution in $\tilde{G}$ and vice-versa. ◀