Semiperfect-Information Games

Krishnendu Chatterjee[1] and Thomas A. Henzinger[1,2]

[1] University of California, Berkeley, USA
[2] EPFL, Switzerland
{c_krish,tah}@eecs.berkeley.edu

Abstract. Much recent research has focused on the applications of games with ω-regular objectives in the control and verification of reactive systems. However, many of the game-based models are ill-suited for these applications, because they assume that each player has complete information about the state of the system (they are "perfect-information" games). This is because in many situations, a controller does not see the private state of the plant. Such scenarios are naturally modeled by "partial-information" games. On the other hand, these games are intractable; for example, partial-information games with simple reachability objectives are 2EXPTIME-complete.

We study the intermediate case of "semiperfect-information" games, where one player has complete knowledge of the state, while the other player has only partial knowledge. This model is appropriate in control situations where a controller must cope with plant behavior that is as adversarial as possible, i.e., the controller has partial information while the plant has perfect information. As is customary, we assume that the controller and plant take turns to make moves. We show that these *semiperfect-information turn-based games* are equivalent to *perfect-information concurrent games*, where the two players choose their moves simultaneously and independently. Since the perfect-information concurrent games are well-understood, we obtain several results of how semiperfect-information turn-based games differ from perfect-information turn-based games on one hand, and from partial-information turn-based games on the other hand. In particular, semiperfect-information turn-based games can benefit from randomized strategies while the perfect-information variety cannot, and semiperfect-information turn-based games are in NP $\cap$ coNP for all parity objectives.

1 Introduction

Games on graphs. Games played on graphs play a central role in many areas of computer science. In particular, when the vertices and edges of a graph represent the states and transitions of a reactive system, then the synthesis problem (Church's problem) asks for the construction of a winning strategy in a game played on a graph [2, 17, 16, 15]. Game-theoretic formulations have also proved useful for the verification [1], refinement [11], and compatibility checking [6] of reactive systems. Games played on graphs are dynamic games that proceed for an infinite number of rounds. In each round, the players choose moves; the moves, together with the current state, determine the successor state. An outcome of the game, called a *play*, consists of the infinite sequence of states that are visited.

Strategies and objectives. A strategy for a player is a recipe that describes how the player chooses a move to extend a play. Strategies can be classified as follows: *pure* strategies, which always deterministically choose a move to extend the play, vs. *randomized* strategies, which may choose at a state a probability distribution over the available moves; *memoryless* strategies, which depend only on the current state of the play, vs. *memory* strategies, which may depend on the history of the play up to the current state. Objectives are generally Borel measurable functions [14]: the objective for a player is a Borel set B in the Cantor topology on S^ω (where S is the set of states), and the player satisfies the objective iff the outcome of the game is a member of B. In verification, objectives are usually *ω-regular languages.* The ω-regular languages generalize the classical regular languages to infinite strings; they occur in the low levels of the Borel hierarchy (they lie in $\Sigma_3 \cap \Pi_3$) and they form a robust and expressive language for determining payoffs for commonly used specifications. The simplest ω-regular objectives correspond to "safety" (the closed sets in the topology of S^ω) and "reachability" (the open sets).

Classification of games. Games played on graphs can be classified according to the knowledge of the players about the state of the game, and the way of choosing moves. Accordingly, there are (a) *perfect-information* games, where each player has complete knowledge about the history of the play up to the current state, and (b) *partial-information* (or *incomplete-information*) games, where a player may not have complete knowledge about the current state of the game and the past moves played by the other player. According to the way of choosing moves, the games on graphs can be classified into *turn-based* and *concurrent* games. In turn-based games, in any given round only one player can choose among multiple moves; effectively, the set of states can be partitioned into the states where it is player 1's turn to play, and the states where it is player 2's turn. In concurrent games, both players may have multiple moves available at each state, and the players choose their moves simultaneously and independently.

Perfect-information versus partial-information games. The perfect-information turn-based (**PT**) games have been widely studied in the computer-science community, and also have deep connections with mathematical logic. For the algorithmic analysis of **PT** games with ω-regular objectives see, for example, [9, 10, 19, 12, 20]. On the other hand, the perfect-information concurrent (**PC**) games (also known as Blackwell games) have been studied mainly in the game-theory community. Only recently has the algorithmic analysis of **PC** games caught interest [7, 5, 8, 3]. It is, however, the *partial-information* games which provide the most natural framework for modular verification and control. In practice, a process or a controller does not have access to the internal or private variables of the other processes or the plant, and partial-information games are the adequate model for such scenarios. Nonetheless, partial-information games have received little attention in computer science, perhaps be due to the high computational complexity of such games. Reif [18] showed that the decision problem for partial-information turn-based games, even for simple reachability objectives, is 2EXPTIME-complete (the same problem can be solved in linear time for **PT** games, and lies in NP $\cap$ coNP for **PC** games [4]).

Semiperfect-information turn-based games. In this paper, we study a subclass of partial-information turn-based games, namely, the *semiperfect-information* turn-based (**ST**) games, where one player (player 1) has incomplete knowledge about the state of the game and the moves of player 2, while player 2 has complete knowledge about the state and player 1 moves. The semiperfect-information games are asymmetric, because one player has partial information and the other player has perfect information. These games provide a better model for controller synthesis than the perfect-information games. In controller synthesis, the controller cannot observe the private variables of the plant and hence has limited knowledge about the state of the game. However, the controller ought to achieve its objective against *all* plant behaviors, and this unconditionally adversarial nature of the plant is modeled most adequately by allowing the plant to have complete knowledge about the game.

Semiperfect-information versus perfect-information games. The **ST** games differ considerably from the **PT** games. In the case of **PT** games, for every Borel objective Φ for player 1 and complementary objective $\overline{\Phi}$ for player 2, the determinacy result of Martin [13] establishes that for every state in the game graph, either player 1 has a pure strategy to satisfy the objective Φ with certainty against all strategies of player 2; or player 2 has a pure strategy to satisfy the objective $\overline{\Phi}$ with certainty against all strategies of player 1. We show that, in contrast, in **ST** games, in general the players cannot guarantee to win with certainty, and randomized strategies are more powerful than pure strategies.

*Example 1 (***ST*** games).* Consider the game shown in Fig. 1. The game is a turn-based game, where the $\Box$ states are player 1 states (where player 1 moves), and the $\Diamond$ states are player 2 states (where player 2 moves); we will follow this convention in all figures. The game is a semiperfect-information game, as player 1 cannot distinguish between the two states in $P_1 = \{ s_1, s_2 \}$. Informally, if the current state of the game is in P_1, then player 1 knows that the game is in P_1 but does not know whether the current state is s_1 or s_2. At state s_0 player 2, can choose between s_1 and s_2, which is indicated by the edge from s_0 to P_1. The objective for player 1 is to reach the state s_3, and the set of available moves for player 1 at the states in P_1 is $\{ a, b \}$. Consider a pure strategy σ for player 1. Consider the counter-strategy π for player 2 as follows: each time player 1 plays move a, player 2 places player 1 at state s_2 in the previous round, and the play reaches s_0; and each time player 1 plays move b, player 2 places player 1 at state s_1 in the previous round, and the play again reaches s_0. Hence for every pure strategy σ for player 1 there is a counter-strategy π for player 2 such that the state s_3 is never reached.

Now consider a randomized memoryless strategy σ_m for player 1 as follows: σ_m plays the moves a and b each with probability $1/2$. Given any strategy π for player 2, every time P_1 is reached, it reaches s_3 with probability $1/2$ and goes

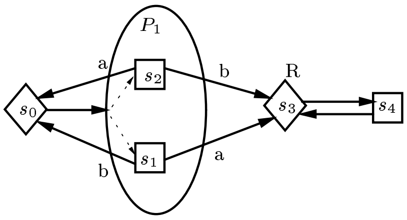

Fig. 1. A semiperfect-information turn-based game.

back to s_0 with probability $1/2$. Hence state s_3 is reached with probability $1/2$ for each visit to P_1, and thus s_3 is eventually reached with probability 1. Given the strategy σ_m, consider a counter-strategy π for player 2 which always places player 1 at state s_1 every time the play reaches s_0. Given the strategies σ_m and π, there exist paths that never reach s_3; however, the measure for the set of those paths is 0. Hence, although player 1 can win with probability 1 from state s_0, she cannot win with certainty. ∎

Semiperfect-information versus partial-information games. The class of **ST** games is considerably simpler than the full class of partial-information turn-based games. While the decision problem for partial-information turn-based games with reachability objectives is 2EXPTIME-complete, we show that for **ST** games the corresponding decision problem is in NP $\cap$ coNP for reachability and also for general parity objectives (the parity objectives are a canonical representation for ω-regular objectives). This shows that the **ST** games can be solved considerably more efficiently than general partial-information turn-based games.

Outline of our main results. We show that though **ST** games differ from **PT** games, there is a close connection between **ST** (turn-based) games and **PC** (concurrent) games. In fact, we establish the equivalence of **ST** games and **PC** games: we present reductions of **ST** games to **PC** games, and vice versa. The **PC** games have been proposed as a framework for modeling synchronous interactions in reactive systems [7, 1]. Our reductions show that such games also provide a framework for analyzing **ST** games. We obtain several results on **ST** games from the equivalence of **ST** games and **PC** games. The main results are as follows:

- The *optimum value* for a player for an objective Φ is the maximal probability with which the player can ensure that Φ is satisfied. We establish the quantitative determinacy for **ST** games with arbitrary Borel objectives: for all **ST** games with objective Φ for player 1 and complementary objective $\overline{\Phi}$ for player 2, the sum of the optimum values for the players at all states is 1.
- The optimum values of **ST** games with parity objectives can be approximated, for any given error bound, in NP $\cap$ coNP. We give an example showing that the optimum values may be irrational in **ST** games with reachability objectives; this indicates that optimum values can only be approximated.

– We analyze, for various classes of parity objectives, the precise memory requirements for strategies to ensure the optimal values of **ST** games within a given error bound.

2 Definitions

In this section we define semiperfect-information turn-based games, strategies, objectives, and values in such games. We later define perfect-information concurrent games and the corresponding notion of strategies, objectives, and values.

2.1 Semiperfect-information turn-based games

A *turn-based game* is played over a finite state space by two players (player 1 and player 2), and the players make moves in turns. In games with *perfect information* each player has complete knowledge about the state and the sequence of moves made by both players. In contrast, in games with *semiperfect information* player 1 has partial knowledge about the state and the moves of player 2, whereas player 2 has complete knowledge about the state and the moves of player 1.

Turn-based game structures. A turn-based game structure $G = ((S_1, S_2), \approx, M, \Gamma_1, \Gamma_2, \delta)$ is a tuple with the following components:

1. Two finite, disjoint sets S_1 and S_2 of states. The state space S of the game structure is their union, i.e., $S = S_1 \cup S_2$. The states in S_1 are player 1 states and the states in S_2 are player 2 states.
2. An equivalence relation $\approx$ on S. The restriction of $\approx$ to S_1 induces a partition $\mathcal{P}_1$ of the set S_1 of player 1 states, and the restriction of $\approx$ to S_2 induces a partition $\mathcal{P}_2$ of S_2. Let $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$ be the corresponding partition of the entire state space S.
3. A finite set M of moves for the two players.
4. Two functions $\Gamma_1 \colon S_1 \to 2^M \backslash \emptyset$ and $\Gamma_2 \colon S_2 \to 2^M \backslash \emptyset$. Each function Γ_i, for $i = 1, 2$, assigns to every state $s \in S_i$ a nonempty set $\Gamma_i(s) \subseteq M$ of moves that are available to player i at state s.
5. Two functions $\delta_1 \colon S_1 \times M \to S_2$ and $\delta_2 \colon S_2 \times M \to S_1$. The transition function δ_1 for player 1 gives for every state $s \in S_1$ and available move $a \in \Gamma_1(s)$ a successor state $\delta_1(s, a) \in S_2$. The transition function δ_2 for player 2 gives for every state $s \in S_2$ and move $b \in \Gamma_2(s)$ a successor state $\delta_2(s, b) \in S_1$.

Semiperfect-information turn-based games. In semiperfect-information turn-based (**ST**) games player 1's view of the game structure is only partial: player 1 knows the $\approx$-equivalence class of the current state, but not the precise state in that class. We formalize this by separating the visible and invisible parts of player 2's moves and transitions.

– The move assignment Γ_2 for player 2 consists of two parts: Γ_2^{vis} assigns to every state $s \in S_2$ a nonempty set $\Gamma_2^{\mathrm{vis}}(s) \subseteq M$ of available *visible* moves for player 2 at state s; and Γ_2^{inv} assigns to every equivalence class $P \in \mathcal{P}_1$

a nonempty set $\Gamma_2^{\mathrm{inv}}(P) \subseteq M$ of available *invisible* moves for player 2 at P. Intuitively, player 1 can observe the set of visible moves of player 2, but she cannot observe the invisible moves of player 2.
– The transition function δ_2 for player 2 consists of two parts: the *visible* transition function $\delta_2^{\mathrm{vis}}\colon S_2 \times M \to \mathcal{P}_1$ gives for every state $s \in S_2$ and move $a \in \Gamma_2^{\mathrm{vis}}(s)$ a successor class $\delta_2^{\mathrm{vis}}(s,a) \in \mathcal{P}_1$; and the *invisible* transition function $\delta_2^{\mathrm{inv}}\colon \mathcal{P}_1 \times M \to S_1$ gives for every equivalence class $P \in \mathcal{P}_1$ and move $a \in \Gamma_2^{\mathrm{inv}}(P)$ a successor state $\delta_2^{\mathrm{inv}}(P,a) \in P$.

Note that the definition of **ST** games reduces to classical perfect-information turn-based games if $\approx$ is equality, i.e., if each equivalence class of $\approx$ is a singleton.

Example 2 (Games with variables). A game with variables between player 1 and player 2 consists of a four-tuple $(V_1^{\mathrm{pvt}}, V_1^{\mathrm{pub}}, V_2^{\mathrm{pvt}}, V_2^{\mathrm{pub}})$ of boolean variables. The set V_i^{pvt}, for $i = 1, 2$, is the set of *private* variables for player i, which player i can observe and update, but the other player can neither observe nor update. The set V_i^{pub} is the set of *public* variables for player i, which both players can observe but only player i can update. A state of the game is a valuation of all variables in $V = V_1^{\mathrm{pvt}} \cup V_1^{\mathrm{pub}} \cup V_2^{\mathrm{pvt}} \cup V_2^{\mathrm{pub}}$. Player 1 and player 2 alternately update the variables in $V_1 = V_1^{\mathrm{pvt}} \cup V_1^{\mathrm{pub}}$ and in $V_2 = V_2^{\mathrm{pvt}} \cup V_2^{\mathrm{pub}}$, respectively. For player 1, a nondeterministic update function u_1 is given by its private component $u_1^{\mathrm{pvt}}\colon [V_1 \cup V_2^{\mathrm{pub}}] \to 2^{[V_1^{\mathrm{pvt}}]}$ and its public component $u_1^{\mathrm{pub}}\colon [V_1 \cup V_2^{\mathrm{pub}}] \to 2^{[V_1^{\mathrm{pub}}]}$, where $[U]$ is the set of valuations for the variables in U. The nondeterministic player 2 update function is given similarly. We consider the special case that $V_1^{\mathrm{pvt}} = \emptyset$, i.e., player 1 has no private variables, and hence player 2 has complete knowledge of the state. This special class of games with variables can be mapped to **ST** games as follows:

– Two states are equivalent if they agree on the valuation of the variables in $V_1^{\mathrm{pub}} \cup V_2^{\mathrm{pub}}$. This defines the equivalence relation $\approx$.
– The update function u_1^{pub} is represented by the move assignment Γ_1 and transition function δ_1 for player 1. The update function u_2^{pub} is represented by the visible move assignment Γ_2^{vis} and visible transition function δ_2^{vis} for player 2. The update function u_2^{pvt} is represented by the invisible move assignment Γ_2^{inv} and invisible transition function δ_2^{inv} for player 2.

These games provide a model for controller synthesis: player 1 is the controller and player 2 represents the plant. The private plant state is unknown to the controller, but the plant has complete knowledge about the state. If both V_1^{pvt} and V_2^{pvt} are empty, we have perfect-information turn-based games. ∎

Remarks. For technical and notational simplicity we make two simplifying assumptions. First, we assume that for all equivalence classes $P \in \mathcal{P}_1$, if $s, s' \in P$, then $\Gamma_1(s) = \Gamma_1(s')$, i.e., player 1 has the same set of moves available in all states of an equivalence class. This restriction does not cause any loss of generality. Suppose that choosing a move $a \notin \Gamma_1(s)$ at a state s causes player 1 to lose immediately. For $P \in \mathcal{P}_1$, if the sets of available moves are not identical for all states $s \in P$, let $A = \bigcup_{s \in P} \Gamma_1(s)$. Then the equivalence class P can be replaced by P' such that the sets of states in P and P' are the same, and the set of available moves for all states in P' is A. For a state $s \in P$ and a move $a \in A$ with $a \notin \Gamma_1(s)$, in the new equivalence class P', the successor state $\delta_1(s,a)$ is losing for player 1.

Second, we assume that for all equivalence classes $P \in \mathcal{P}_1$ and all states $s \in P$, there exists a move $a \in \Gamma_2^{\text{inv}}(P)$ such that $\delta_2^{\text{inv}}(P,a) = s$. In other words, in each equivalence class P player 2 has the choice to move to any state in P. Hence, if $P = \{ s_1, \ldots, s_k \}$, then $\Gamma_2^{\text{inv}}(P) = \{ 1, \ldots, k \}$ and $\delta_2^{\text{inv}}(P,j) = s_j$ for all $j \in \{ 1, \ldots, k \}$. In games with variables, this corresponds to the assumption that player 2 can update the variables in V_2^{pvt} in all possible ways, i.e., player 1 has no knowledge about the moves of player 2. We now argue that also this restriction does not result in a loss of generality in the model. Given a **ST** game structure G, suppose that for some state $s \in S_2$, the possible transitions to states in an equivalence class $P \in \mathcal{P}_1$ target a strict subset $Z \subsetneq P$. We transform the game structure G as follows: (a) add a copy of the subset Z of states; (b) the states in the copy Z are player 1 states and Z is an equivalence class; (c) the visible transition of player 2 goes from state s to Z instead of P; and (d) the transition function for player 1 for the states in Z follow the transition function for the corresponding states of the original structure G. Observe that the number of subsets of states that are added by this transformation is bounded by the size of the transition function of the original game structure. Hence the blow-up caused by the transformation, to obtain an equivalent **ST** game structure that satisfies the restriction, is at worst quadratic in the size of the original game structure.

Notation. The *partial-information* (or *hiding*) function $\rho \colon S \to \mathcal{P}$ maps every state $s \in S$ to its equivalence class, i.e., $\rho(s) = P \in \mathcal{P}$ if $s \in P$. The set E of *edges* is defined as follows:

$$\begin{aligned} E = {} & \{ (s,s') \mid s \in S_1, s' \in S_2, (\exists a \in \Gamma_1(s))(\delta_1(s,a) = s') \} \\ & \cup \{ (s,s') \mid s \in S_2, s' \in S_1, (\exists a \in \Gamma_2^{\text{vis}}(s))(\delta_2^{\text{vis}}(s,a) = P \text{ and } s' \in P) \}. \end{aligned}$$

A *play* $\omega = \langle s_0, s_1, s_2, \ldots \rangle$ is an infinite sequence of states such that for all $j \geq 0$, we have $(s_j, s_{j+1}) \in E$. We denote by Ω the set of all plays. Given a finite sequence $\langle s_0, s_1, \ldots, s_k \rangle$ of states, we write $\rho(\langle s_0, s_1, \ldots, s_k \rangle)$ for the corresponding sequence $\langle \rho(s_0), \rho(s_1), \ldots, \rho(s_k) \rangle$ of equivalence classes. The notation for infinite sequence of states is analogous.

For a countable set A, a *probability distribution* on A is a function $\mu \colon A \to [0,1]$ such that $\sum_{a \in A} \mu(a) = 1$. We denote the set of probability distributions on A by $\mathcal{D}(A)$. Given a distribution $\mu \in \mathcal{D}(A)$, we denote by $\text{Supp}(\mu) = \{x \in A \mid \mu(x) > 0\}$ the support of μ.

Strategies. A strategy for player 1 is a recipe of how to extend a play. Player 1 does not have perfect information about the states in the play; she only knows the sequence of equivalence classes where the given play has been. Hence, for a finite sequence $\langle s_0, s_1, \ldots, s_k \rangle$ of states representing the history of the play so far, the view for player 1 is given by $\rho(\langle s_0, s_1, \ldots, s_k \rangle)$. Given this view of the history, player 1's strategy is to prescribe a probability distribution over the set of available moves. Formally, a *strategy σ for player 1* is a function $\sigma\colon \rho(S^* \cdot S_1) \to \mathcal{D}(M)$, such that for all finite sequences $\langle s_0, s_1, \ldots, s_k \rangle$ of states such that $s_k \in S_1$, and for all moves $a \in M$, if $\sigma(\rho(\langle s_0, s_1, \ldots, s_k \rangle))(a) > 0$, then $a \in \Gamma_1(s_k)$. The strategy σ for player 1 is *pure* if for all $\langle s_0, s_1, \ldots, s_k \rangle$ such that $s_k \in S_1$, there is a move $a \in \Gamma_1(s_k)$ with $\sigma(\rho(\langle s_0, s_1, \ldots, s_k \rangle))(a) = 1$, i.e., for all histories the strategy deterministically chooses a move. The strategy σ is *memoryless* if it is independent of the history of the play and only depends on the current state. Formally, a memoryless strategy σ for player 1 is a function $\sigma\colon \rho(S_1) \to \mathcal{D}(M)$. A strategy is *pure memoryless* if it is both pure and memoryless, i.e., it can be represented as a function $\sigma\colon \rho(S_1) \to M$.

A strategy for player 2 is a recipe for player 2 to extend the play. In contrast to player 1, player 2 has perfect information about the history of the play and precisely knows every state in the history. Given the history of a play such that the last state is a player 2 state, player 2 chooses a probability distribution over the set of available visible moves to select an equivalence class $P \in \mathcal{P}_1$, and also chooses a probability distribution over the set of available invisible moves to select a state in P. Formally, a *strategy π for player 2* consists of two components:

- a function $\pi^{\mathrm{vis}}\colon S^* \cdot S_2 \to \mathcal{D}(M)$ such that for all $\langle s_0, s_1, \ldots, s_k \rangle$ with $s_k \in S_2$, if $\pi^{\mathrm{vis}}(\langle s_0, s_1, \ldots, s_k \rangle)(a) > 0$, then $a \in \Gamma_2^{\mathrm{vis}}(s_k)$;
- a function $\pi^{\mathrm{inv}}\colon S^* \cdot S_2 \cdot \mathcal{P}_1 \to \mathcal{D}(M)$ such that for all $\langle s_0, s_1, \ldots, s_k, P_{k+1} \rangle$ with $s_k \in S_2$ and $P_{k+1} \in \mathcal{P}_1$, if $\pi^{\mathrm{inv}}(\langle s_0, s_1, \ldots, s_k, P_{k+1} \rangle)(a) > 0$, then $a \in \Gamma_2^{\mathrm{inv}}(P_{k+1})$.

The strategy π for player 2 is *pure* if both component strategies π^{vis} and π^{inv} are pure. Similarly, the strategy π is *memoryless* if both component strategies π^{vis} and π^{inv} are memoryless; and it is *pure memoryless* if it is pure and memoryless. We denote by Σ and Π the sets of strategies for player 1 and player 2, respectively. We write Σ^P, Σ^M, and Σ^{PM} for the sets of pure, memoryless, and pure memoryless strategies for player 1, respectively. The analogous classes of strategies for player 2 are defined similarly.

Objectives. We specify objectives for the two players by providing sets $\Phi_i \subseteq \Omega$ of *winning plays* for each player i. In this paper we study only zero-sum games, where the objectives of the two players are strictly competitive. In other words, it is implicit that if the objective of one player 1 is Φ_1, then the objective of player 2 is $\Phi_2 = \Omega \setminus \Phi_1$. In the case of semi-perfect information games, the objective Φ_1 of player 1 is specified as a subset of $\mathcal{P}^\omega$, rather than an arbitrary subset of S^ω; this is because player 1 cannot distinguish between the states of an equivalence class. In the setting of games with variables (Example 2), this means that the objective of player 1 gives a property of the traces over the public variables of both players. Given an objective $\Phi \subseteq \mathcal{P}^\omega$, we write $\Omega \setminus \Phi$, short for the complementary objective $\{\rho(\omega) \mid \omega \in \Omega\} \setminus \Phi$.

A general class of objectives are the Borel objectives [13]. A *Borel objective* $\Phi \subseteq \mathcal{P}^\omega$ is a Borel set in the Cantor topology on $\mathcal{P}^\omega$. In this paper we consider *ω-regular objectives* [19], which lie in the first $2^1/_2$ levels of the Borel hierarchy. The ω-regular objectives, and subclasses thereof, can be specified in the following forms. For a play $\omega = \langle s_0, s_1, s_2, \ldots \rangle \in \Omega$, we define $\mathrm{Inf}(\rho(\omega)) = \{\, \rho(s) \in \mathcal{P} \mid s_k = s$ for infinitely many $k \geq 0 \,\}$ to be the set of equivalence classes that occur infinitely often in ω.

– *Reachability and safety objectives.* Given a set $T \subseteq \mathcal{P}$ of "target" equivalence classes, the reachability objective requires that some equivalence class in T be visited. The set of winning plays is $\mathrm{Reach}(T) = \{\, \rho(\omega) \mid \omega = \langle s_0, s_1, s_2, \ldots \rangle \in \Omega$, and $\rho(s_k) \in T$ for some $k \geq 0 \,\}$. Given a set $F \subseteq \mathcal{P}$, the safety objective requires that only equivalence classes in F be visited. Thus, the set of winning plays is $\mathrm{Safe}(F) = \{\, \rho(\omega) \mid \omega = \langle s_0, s_1, s_2, \ldots \rangle \in \Omega$, and $\rho(s_k) \in F$ for all $k \geq 0 \,\}$.
– *Büchi and coBüchi objectives.* Given a set $B \subseteq \mathcal{P}$ of "Büchi" equivalence classes, the Büchi objective requires that B is visited infinitely often. Formally, the set of winning plays is $\text{Büchi}(B) = \{\, \rho(\omega) \mid \omega \in \Omega$ and $\mathrm{Inf}(\rho(\omega)) \cap B \neq \emptyset \,\}$. Given $C \subseteq \mathcal{P}$, the coBüchi objective requires that all equivalence classes that are visited infinitely often, are in C. Hence, the set of winning plays is $\text{coBüchi}(C) = \{\, \rho(\omega) \mid \omega \in \Omega$ and $\mathrm{Inf}(\rho(\omega)) \subseteq C \,\}$.
– *Parity objectives.* For $c, d \in \mathbb{N}$, let $[c..d] = \{\, c, c+1, \ldots, d \,\}$. Let $p \colon \mathcal{P} \to [0..d]$ be a function that assigns a *priority* $p(P)$ to every equivalence class $P \in \mathcal{P}$, where $d \in \mathbb{N}$. The *even-parity objective* is defined as $\mathrm{Parity}(p) = \{\, \rho(\omega) \mid \omega \in \Omega$ and $\min\big(p(\mathrm{Inf}(\rho(\omega)))\big)$ is even $\}$, and the *odd-parity objective* is $\mathrm{coParity}(p) = \{\, \rho(\omega) \mid \omega \in \Omega$ and $\min\big(p(\mathrm{Inf}(\rho(\omega)))\big)$ is odd $\}$. Note that for a priority function $p \colon \mathcal{P} \to \{\, 0, 1 \,\}$, the even-parity objective $\mathrm{Parity}(p)$ is equivalent to the Büchi objective $\text{Büchi}(p^{-1}(0))$, i.e., the Büchi set consists of the equivalence class with priority 0.

We say that a play ω *satisfies* an objective $\Phi \subseteq \mathcal{P}^\omega$ if $\rho(\omega) \in \Phi$. Given a state $s \in S$ and strategies $\sigma \in \Sigma, \pi \in \Pi$ for the two players, the *outcome* of the game is a probability distribution over the set Ω of plays, and every Borel objective Φ is a measurable subset. The probability that the outcome of the game satisfies the Borel objective Φ starting from state s following the strategies σ and π is denoted $\mathrm{Pr}_s^{\sigma,\pi}(\Phi)$.

Values of the game. Given an objective Φ for player 1 and a state s, the maximal probability with which player 1 can ensure that Φ is satisfied from s, is called the *value* of the game at s for player 1. Formally, we define the value functions $\langle\langle 1 \rangle\rangle_{val}$ and $\langle\langle 2 \rangle\rangle_{val}$ for players 1 and 2 as follows: $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = \sup_{\sigma \in \Sigma} \inf_{\pi \in \Pi} \mathrm{Pr}_s^{\sigma,\pi}(\Phi)$; and $\langle\langle 2 \rangle\rangle_{val}(\Omega \backslash \Phi)(s) = \sup_{\pi \in \Pi} \inf_{\sigma \in \Sigma} \mathrm{Pr}_s^{\sigma,\pi}(\Omega \backslash \Phi)$. A strategy σ for player 1 is *optimal* from state s for objective Φ if $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = \inf_{\pi \in \Pi} \mathrm{Pr}_s^{\sigma,\pi}(\Phi)$. The strategy σ for player 1 is *ε-optimal*, for a real $\varepsilon \geq 0$, from state s for objective Φ if $\inf_{\pi \in \Pi} \mathrm{Pr}_s^{\sigma,\pi}(\Phi) \geq \langle\langle 1 \rangle\rangle_{val}(\Phi)(s) - \varepsilon$. The optimal and ε-optimal strategies for player 2 are defined analogously.

Sure, almost-sure, and limit-sure winning strategies. Given an objective Φ, a strategy σ is a *sure* winning strategy for player 1 from a state s for Φ if for every strategy π of player 2, every play ω that is possible when following the strategies σ and π from s, belongs to Φ. The strategy σ is an *almost-sure* winning strategy for player 1 from s for Φ if for every strategy π

of player 2, $\Pr_s^{\sigma,\pi}(\Phi) = 1$. A family $\Sigma^{\mathcal{C}}$ of strategies is *limit-sure* winning for player 1 from s for Φ if $\sup_{\sigma \in \Sigma^{\mathcal{C}}} \inf_{\pi \in \Pi} \Pr_s^{\sigma,\pi}(\Phi)(s) = 1$. See [7, 5] for formal definitions. The sure, almost-sure, and limit-sure winning strategies for player 2 are defined analogously. The sure winning set $\langle\langle 1 \rangle\rangle_{sure}(\Phi)$, the almost-sure winning set $\langle\langle 1 \rangle\rangle_{almost}(\Phi)$, and the limit-sure winning set $\langle\langle 1 \rangle\rangle_{limit}(\Phi)$ for player 1 for objective Φ are the sets of states from which player 1 has sure, almost-sure, and limit-sure winning strategies, respectively. The sure winning set $\langle\langle 2 \rangle\rangle_{sure}(\Omega \setminus \Phi)$, the almost-sure winning set $\langle\langle 2 \rangle\rangle_{almost}(\Omega \setminus \Phi)$, and the limit-sure winning set $\langle\langle 2 \rangle\rangle_{limit}(\Omega \setminus \Phi)$ for player 2 are defined analogously.

Observe that the limit-sure winning set is the set of states with value 1, which is the classical notion of *qualitative* winning. It follows from the definitions that for all game structures and all objectives Φ, we have $\langle\langle 1 \rangle\rangle_{sure}(\Phi) \subseteq \langle\langle 1 \rangle\rangle_{almost}(\Phi) \subseteq \langle\langle 1 \rangle\rangle_{limit}(\Phi)$ and $\langle\langle 2 \rangle\rangle_{sure}(\Omega \setminus \Phi) \subseteq \langle\langle 2 \rangle\rangle_{almost}(\Omega \setminus \Phi) \subseteq \langle\langle 2 \rangle\rangle_{limit}(\Omega \setminus \Phi)$. Computing sure, almost-sure, and limit-sure winning sets and strategies is referred to as the qualitative analysis of games; computing values, as the quantitative analysis.

Sufficiency of a family of strategies. Given a family $\Sigma^{\mathcal{C}}$ of player 1 strategies, we say that the family $\Sigma^{\mathcal{C}}$ *suffices* with respect to an objective Φ on a class $\mathcal{G}$ of game structures for

- *sure winning* if for every game structure $G \in \mathcal{G}$ and state $s \in \langle\langle 1 \rangle\rangle_{sure}(\Phi)$, there is a player 1 sure winning strategy $\sigma \in \Sigma^{\mathcal{C}}$ from s for Φ;
- *almost-sure winning* if for every structure $G \in \mathcal{G}$ and state $s \in \langle\langle 1 \rangle\rangle_{almost}(\Phi)$, there is a player 1 almost-sure winning strategy $\sigma \in \Sigma^{\mathcal{C}}$ from s for Φ;
- *limit-sure winning* if for every structure $G \in \mathcal{G}$ and state $s \in \langle\langle 1 \rangle\rangle_{limit}(\Phi)$, $\sup_{\sigma \in \Sigma^{\mathcal{C}}} \inf_{\pi \in \Pi} \Pr_s^{\sigma,\pi}(\Phi) = 1$;
- *ε-optimality*, for $\varepsilon \geq 0$, if for every game structure $G \in \mathcal{G}$ and state s of G, there is a player 1 strategy $\sigma \in \Sigma^{\mathcal{C}}$ such that $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) - \varepsilon \leq \inf_{\pi \in \Pi} \Pr_s^{\sigma,\pi}(\Phi)$. Sufficiency for optimality is the special case of sufficiency for ε-optimality with $\varepsilon = 0$.

Theorem 1 (Perfect-information turn-based games). *The following assertions hold for all perfect-information turn-based (**PT**) games:*

1. *[13] For all Borel objectives Φ, the sets $\langle\langle 1 \rangle\rangle_{sure}(\Phi)$ and $\langle\langle 2 \rangle\rangle_{sure}(\Omega \setminus \Phi)$ form a partition of the state space.*
2. *[13] The family Σ^P of pure strategies suffices for sure winning with respect to all Borel objectives.*
3. *[9] The family Σ^{PM} of pure memoryless strategies suffices for sure winning with respect to all parity objectives.*

It follows from Theorem 1 that in the case of **PT** games the values can be either 1 or 0. We show that, in contrast, **ST** games can have values other than 1 and 0. Example 1 shows that in general we have $\langle\langle 1 \rangle\rangle_{sure}(\Phi) \subsetneq \langle\langle 1 \rangle\rangle_{almost}(\Phi)$ in **ST** games, even for reachability objectives Φ. The next example shows that in general $\langle\langle 1 \rangle\rangle_{almost}(\Phi) \subsetneq \langle\langle 1 \rangle\rangle_{limit}(\Phi)$ in **ST** games, again for reachability objectives Φ. We also show that sure determinacy (Part 1 of Theorem 1) does not

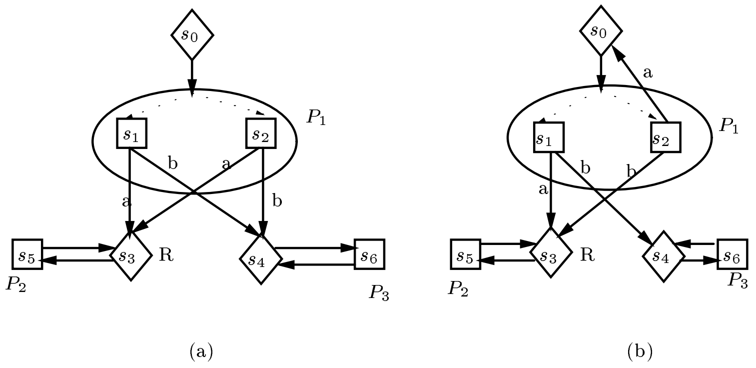

Fig. 2. Values and limit-sure winning states in **ST** games.

hold for **ST** games, and that randomized strategies are more powerful than pure strategies in **ST** games.

Example 3 (Values and limit-sure winning in **ST** *games).* Consider the two games shown in Fig. 2(a) and Fig. 2(b). The two state partitions for player 1 are $\mathcal{P}_1^a = \mathcal{P}_1^b = \{ P_1, P_2, P_3 \}$ with $P_1 = \{ s_1, s_2 \}$, $P_2 = \{ s_5 \}$, and $P_3 = \{ s_6 \}$. In both games, the set of moves available to player 1 in $P_1 = \{ s_1, s_2 \}$ is $\{ a, b \}$. The transitions are shown in the figures. The game starts at the state s_0 and the objective for player 1 is to reach the state s_3, i.e., Reach($\{ s_3 \}$).

Values. Consider the game shown in Fig. 2(a). For every pure strategy $\sigma \in \Sigma^P$ for player 1, consider a counter-strategy π for player 2 as follows: if player 1 chooses move a, then player 2 places player 1 at state s_2; and if player 1 chooses move b, then player 2 places player 1 at state s_1. Hence the game reaches s_4 and player 1 looses. The player 1 strategy $\sigma \in \Sigma^M$ that plays move a and b with probability $1/2$, reaches state s_3 with probability $1/2$ against all strategies for player 2. For every player 1 strategy σ that chooses move a with greater probability than move b, the counter-strategy for player 2 places player 1 at state s_2; and for every player 1 strategy σ that chooses move b with greater probability than move a, the counter-strategy for player 2 places player 1 at state s_1. It follows that the value for player 1 at state s_0 is $1/2$. Thus the sure-determinacy result for **PT** games does not extend to **ST** games.

Limit-sure winning. Consider the game shown in Fig. 2(b). For $\varepsilon > 0$, consider the memoryless player 1 strategy $\sigma_\varepsilon \in \Sigma^M$ that plays move a with probability $1 - \varepsilon$, and move b with probability ε. The game starts at s_0, and in each round, if player 2 chooses state s_2, then the game reaches s_3 with probability ε and comes back to s_0 with probability $1 - \varepsilon$; whereas if player 2 chooses state s_1, then the game reaches state s_3 with probability $1 - \varepsilon$ and state s_4 with probability ε. Hence, given the strategy σ_ε for player 1, the game reaches s_3 with probability at least $1 - \varepsilon$ against all strategies π for player 2. Therefore $s_0 \in \langle\langle 1 \rangle\rangle_{limit}(\text{Reach}(\{ s_3 \}))$. However, we now argue that

$s_3 \notin \langle\langle 1 \rangle\rangle_{almost}(\text{Reach}(\{ s_3 \}))$, and thus also $s_0 \notin \langle\langle 1 \rangle\rangle_{sure}(\text{Reach}(\{ s_3 \}))$. To prove this claim, given a strategy σ for player 1, consider the following counter-strategy π for player 2: for $k \geq 0$, in round $2k+1$, if player 1 plays move a with probability 1, then at round $2k$ player 2 chooses state s_2 and ensures that s_3 is reached with probability 0, and the game reaches s_0 in round $2k+2$; otherwise, if player 1 plays move b with positive probability in round $2k+1$, then player 2 in round $2k$ chooses state s_1, and the game reaches s_4 with positive probability. It follows that $s_0 \notin \langle\langle 1 \rangle\rangle_{almost}(\text{Reach}(\{ s_3 \}))$. ∎

2.2 Perfect-information concurrent games

In contrast to turn-based games, where the players make their moves in turns, in concurrent games both players choose their moves simultaneously and independently of each other.

Perfect-information concurrent game structures. A perfect-information concurrent (**PC**) game structure $G = (S, M, \Gamma_1, \Gamma_2, \delta)$ is a tuple that consists of the following components:

- A finite state space S and a finite set M of moves.
- Two move assignments $\Gamma_1, \Gamma_2 \colon S \to 2^M \setminus \emptyset$. For $i = 1, 2$, the move assignment Γ_i associates with each state $s \in S$ a nonempty set $\Gamma_i(s) \subseteq M$ of moves available to player i at state s.
- A deterministic transition function $\delta \colon S \times M \times M \to S$ which gives the successor state $\delta(s, a, b)$ from state s when player 1 chooses move $a \in \Gamma_1(s)$ and player 2 chooses move $b \in \Gamma_2(s)$.

Strategies, objectives, and values. A strategy σ for player 1 is a function $\sigma \colon S^+ \to \mathcal{D}(M)$ such that for all $\langle s_0, s_1, \ldots, s_k \rangle$ if $\sigma(\langle s_0, s_1, \ldots, s_k \rangle)(a) > 0$, then $a \in \Gamma_1(s_k)$. The strategies for player 2 are defined similarly. The classes of pure, memoryless, and pure memoryless strategies are defined as in the case of **ST** games. The definitions for objectives and values are also analogous to the definitions for **ST** games. Concurrent games satisfy a quantitative version of determinacy formalized in the next theorem.

Theorem 2 (Quantitative determinacy [14]). *For all* **PC** *games, Borel objectives Φ, and states s, we have* $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) + \langle\langle 2 \rangle\rangle_{val}(\Omega \setminus \Phi)(s) = 1$.

3 Equivalence of ST Games and PC Games

In this section we show the equivalence of **ST** games and **PC** games. We first present a reduction from **ST** games to **PC** games.

From ST games to PC games. Consider an **ST** game structure $G = ((S_1, S_2), \approx, M, \Gamma_1, \Gamma_2, \delta)$. We construct a **PC** game structure $\alpha(G) = (\widehat{S}, \widehat{M}, \widehat{\Gamma}_1, \widehat{\Gamma}_2, \widehat{\delta})$ as follows:

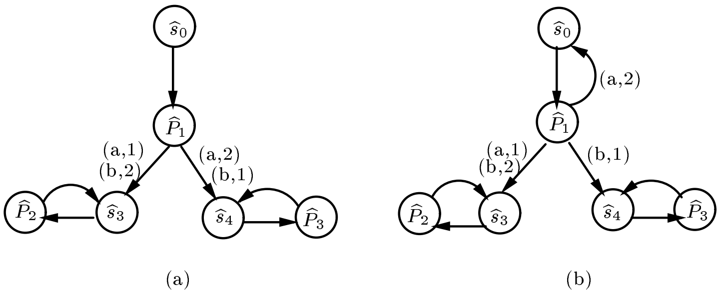

Fig. 3. PC games for the **ST** games of Fig. 2.

- *State space.* Let $\widehat{S} = \{ \widehat{s} \mid s \in S_2 \} \cup \{ \widehat{P} \mid P \in \mathcal{P}_1 \}$. For a state $s \in S$, we write $\alpha(s)$ for $\widehat{\rho(s)} \in \widehat{S}$, i.e., if $s \in S_2$, then $\alpha(s) = \widehat{s}$; and if $s \in S_1$ and $s \in P \in \mathcal{P}_1$, then $\alpha(s) = \widehat{P}$. Also, given a state $\widehat{s} \in \widehat{S}$, we define a map $\beta(\widehat{s})$ as follows: if $s \in S_2$ and $\alpha(s) = \widehat{s}$, then $\beta(\widehat{s}) = s$; else $\beta(\widehat{s}) = s'$ for some $s' \in P$ with $\widehat{s} = \widehat{P}$.
- *Move assignments.* For every state $s \in S_2$, let $\widehat{\Gamma}_2(\widehat{s}) = \Gamma_2^{\mathrm{vis}}(s)$ and $\widehat{\Gamma}_1(\widehat{s}) = \{ \perp \}$. For every equivalence class $P \in \mathcal{P}_1$ with $P = \{ s_1, \ldots, s_k \}$, let $\widehat{\Gamma}_1(\widehat{P}) = \Gamma_1(s_1)$ and $\widehat{\Gamma}_2(\widehat{P}) = \Gamma_2^{\mathrm{inv}}(P) = \{ 1, \ldots, k \}$. Let $\widehat{M} = \bigcup_{\widehat{s} \in \widehat{S}} \big(\widehat{\Gamma}_1(\widehat{s}) \cup \widehat{\Gamma}_2(\widehat{s}) \big)$.
- *Transition function.* For every state $\widehat{s} \in \widehat{S}$ and all moves $(a, b) \in \widehat{\Gamma}_1(\widehat{s}) \times \widehat{\Gamma}_2(\widehat{s})$, let

$$\widehat{\delta}(\widehat{s}, a, b) \;=\; \begin{cases} \alpha(\delta_2^{\mathrm{vis}}(\beta(\widehat{s}), b)) & \text{if } \beta(\widehat{s}) \in S_2, \\ \alpha(\delta_1(s_b, a)) & \text{if } \widehat{s} = \widehat{P} \text{ for } P = \{ s_1, \ldots, s_k \} \in \mathcal{P}_1. \end{cases}$$

  Intuitively the concurrent state $\widehat{P}$ captures the following idea: player 2 chooses the move $b \in \Gamma_2^{\mathrm{inv}}(P)$ to place player 1 at the state $s_b \in P$; and player 1 chooses a move from $\Gamma_1(s)$ for $s \in P$. The joint moves a for player 1 and b for player 2, together with the player 1 transition function δ_1, determines the transition function $\widehat{\delta}$ of the concurrent game.

Example 4. Fig. 3 shows the **PC** game structures that correspond to the **ST** game structures of Fig. 2, mainly illustrating the reduction for the equivalence class $P_1 = \{ s_1, s_2 \}$. ∎

Strategy maps. Let $\widehat{\Sigma}$ and $\widehat{\Pi}$ be the sets of player 1 and player 2 strategies in the game structure $\alpha(G)$. Given two strategies $\sigma \in \Sigma$ and $\pi \in \Pi$ in the **ST** structure G, we define corresponding strategies $\alpha(\sigma) \in \widehat{\Sigma}$ and $\alpha(\pi) \in \widehat{\Pi}$ in the **PC** structure $\alpha(G)$ as follows:

$$\alpha(\sigma)(\langle \widehat{s}_0, \widehat{s}_1, \ldots, \widehat{s}_k \rangle) \;=\; \begin{cases} \text{play } \perp \text{ with probability } 1 & \text{if } \beta(\widehat{s}_k) \in S_2, \\ \sigma(\rho(\langle \beta(\widehat{s}_0), \beta(\widehat{s}_1), \ldots, \beta(\widehat{s}_k) \rangle)) & \text{otherwise;} \end{cases}$$

$$\alpha(\pi)(\langle \widehat{s}_0, \widehat{s}_1, \ldots, \widehat{s}_k \rangle) \;=\; \begin{cases} \pi^{\mathrm{vis}}(\langle \beta(\widehat{s}_0), \beta(\widehat{s}_1), \ldots, \beta(\widehat{s}_k) \rangle) & \text{if } \beta(\widehat{s}_k) \in S_2, \\ \pi^{\mathrm{inv}}(\langle \beta(\widehat{s}_0), \beta(\widehat{s}_1), \ldots, \rho(\beta(\widehat{s}_k)) \rangle) & \text{otherwise.} \end{cases}$$

Similarly, given strategies $\widehat{\sigma} \in \widehat{\Sigma}$ and $\widehat{\pi} \in \widehat{\Pi}$ for the two players in the concurrent structure $\alpha(G)$, we define corresponding strategies $\beta(\widehat{\sigma}) \in \Sigma$ and $\beta(\widehat{\pi}) \in \Pi$ in the turn-based structure G as follows:

$$\begin{aligned} \beta(\widehat{\sigma})(\rho(\langle s_0, s_1, \ldots, s_k \rangle)) &= \widehat{\sigma}(\langle \alpha(s_0), \alpha(s_1), \ldots, \alpha(s_k) \rangle) \text{ if } s_k \in S_1; \\ \beta(\widehat{\pi})^{\mathrm{vis}}(\langle s_0, s_1, \ldots, s_k \rangle) &= \widehat{\pi}(\langle \alpha(s_0), \alpha(s_1), \ldots, \alpha(s_k) \rangle) \text{ if } s_k \in S_2; \\ \beta(\widehat{\pi})^{\mathrm{inv}}(\langle s_0, s_1, \ldots, s_k, P_{k+1} \rangle) &= \widehat{\pi}(\langle \alpha(s_0), \alpha(s_1), \ldots, \alpha(s_k), \widehat{P}_{k+1} \rangle). \end{aligned}$$

Given an objective $\Phi \subseteq \mathcal{P}^\omega$ for the **ST** game structure G, we denote by $\alpha(\Phi) \subseteq \widehat{S}^\omega$ the corresponding objective for the **PC** game structure $\alpha(G)$, which is formally defined as $\alpha(\Phi) = \{ \langle \widehat{s}_0, \widehat{s}_1, \widehat{s}_2, \ldots \rangle \mid \rho(\langle \beta(\widehat{s}_0), \beta(\widehat{s}_1), \beta(\widehat{s}_2), \ldots \rangle) \in \Phi \}$.

Lemma 1 (ST games to PC games). *For all* **ST** *game structures G with Borel objectives Φ,*

1. *for all player 1 strategies σ and player 2 strategies π in G, and for all states s of G, we have $\mathrm{Pr}_s^{\sigma,\pi}(\Phi) = \mathrm{Pr}_{\alpha(s)}^{\alpha(\sigma),\alpha(\pi)}(\alpha(\Phi))$;*
2. *for all player 1 strategies $\widehat{\sigma}$ and player 2 strategies $\widehat{\pi}$ in the* **PC** *game structure $\alpha(G)$, and all states $\widehat{s}$ of $\alpha(G)$, we have $\mathrm{Pr}_{\widehat{s}}^{\widehat{\sigma},\widehat{\pi}}(\alpha(\Phi)) = \mathrm{Pr}_{\beta(\widehat{s})}^{\beta(\widehat{\sigma}),\beta(\widehat{\pi})}(\Phi)$.*

From PC games to ST games. Consider a **PC** game structure $G = (S, M, \Gamma_1, \Gamma_2, \delta)$. We construct an **ST** game structure $\gamma(G) = ((\widetilde{S}_1, \widetilde{S}_2), (\widetilde{\mathcal{P}_1}, \widetilde{\mathcal{P}_2}), \widetilde{M}, \widetilde{\Gamma}_1, \widetilde{\Gamma}_2, \widetilde{\delta})$ as follows:

> Every state $s \in S$ with $\Gamma_1(s) = A$ and $\Gamma_2(s) = \{ 1, \ldots, k \}$ is replaced by a gadget consisting of a player 2 state $\widetilde{s}$ with an edge to an equivalence class $\widetilde{P} \in \widetilde{\mathcal{P}_1}$ such that $\widetilde{P} = \{ \widetilde{s}_1, \ldots, \widetilde{s}_k \}$ and
> 1. $\widetilde{\Gamma}_2^{\mathrm{vis}}(\widetilde{s}) = \{ b \}$, $\widetilde{\Gamma}_2^{\mathrm{inv}}(\widetilde{P}) = \{ 1, \ldots, k \}$, and $\widetilde{\Gamma}_1(\widetilde{s}_j) = A$ for all $\widetilde{s}_j \in \widetilde{P}$;
> 2. $\widetilde{\delta}_2^{\mathrm{vis}}(\widetilde{s}, b) = \widetilde{P}$, $\widetilde{\delta}_2^{\mathrm{inv}}(\widetilde{P}, j) = \widetilde{s}_j$, and $\widetilde{\delta}_1(\widetilde{s}_j, a) = \gamma(\delta(s, a, j))$, where given a state $s \in S$, we denote by $\gamma(s)$ the state $\widetilde{s} \in \widetilde{S}_2$.

For a state pair $(\widetilde{s}, \widetilde{s}') \in \widetilde{S}^2$, let $\lambda(\widetilde{s}, \widetilde{s}')$ be the state $s \in S$ with $\gamma(s) = \widetilde{s}$.

Example 5. Consider the **PC** game shown in Fig. 4(a). The set of available moves for player 1 at the states s_2 and s_3 is $\{ a, b \}$, and for player 2, it is $\{ 1, 2 \}$. Fig. 4(b) shows an equivalent **ST** game, illustrating the translation of the concurrent states s_2 and s_3. ∎

Given an objective $\Phi \subseteq S^\omega$ for the **PC** game structure G, we define the corresponding objective $\gamma(\Phi) \subseteq \widetilde{\mathcal{P}}^\omega$ for the **ST** game structure $\gamma(G)$ as $\gamma(\Phi) = \{ \rho(\langle \widetilde{s}_0, \widetilde{s}_1, \widetilde{s}_2, \ldots \rangle) \mid \langle \lambda(\widetilde{s}_0, \widetilde{s}_1), \lambda(\widetilde{s}_2, \widetilde{s}_3), \ldots \rangle \in \Phi \}$. Similar to the previous reduction, there exist simple translations $\gamma \colon \Sigma \to \widetilde{\Sigma}$ and $\gamma \colon \Pi \to \widetilde{\Pi}$ mapping strategies in the game structure G to strategies in $\gamma(G)$, and reverse translations $\lambda \colon \widetilde{\Sigma} \to \Sigma$ and $\lambda \colon \widetilde{\Pi} \to \Pi$ mapping strategies in $\gamma(G)$ to strategies in G such that the following lemma holds.

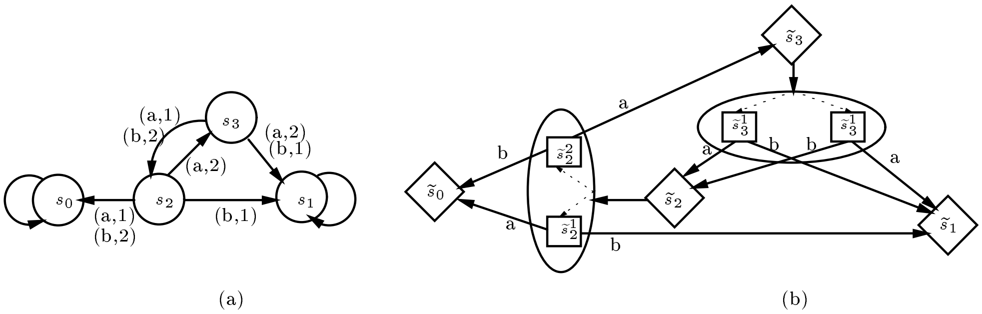

Fig. 4. A **PC** game with irrational values and the corresponding **ST** game.

Lemma 2 (PC games to ST games). *For all* **PC** *game structures G with Borel objectives Φ,*

1. *for all player 1 strategies σ and player 2 strategies π in G, and for all states s of G, we have $\Pr_s^{\sigma,\pi}(\Phi) = \Pr_{\gamma(s)}^{\gamma(\sigma),\gamma(\pi)}(\gamma(\Phi))$;*
2. *for all player 1 strategies $\widetilde{\sigma}$ and player 2 strategies $\widetilde{\pi}$ in the* **ST** *game structure $\gamma(G)$, and for all states $\widetilde{s}$ of $\gamma(G)$, we have $\Pr_{\widetilde{s}}^{\widetilde{\sigma},\widetilde{\pi}}(\gamma(\Phi)) = \Pr_{\lambda(\widetilde{s},\widetilde{s})}^{\lambda(\sigma),\lambda(\pi)}(\Phi)$.*

The following theorem follows from Lemma 1 and Lemma 2.

Theorem 3 (Equivalence of ST and PC games).

1. *For every* **ST** *game structure G, there is a* **PC** *game structure $\alpha(G)$ such that for all Borel objectives Φ and all states s of G, we have $\langle\!\langle 1 \rangle\!\rangle_{val}(\Phi)(s) = \langle\!\langle 1 \rangle\!\rangle_{val}(\alpha(\Phi))(\alpha(s))$.*
2. *For every* **PC** *game structure G, there is an* **ST** *game structure $\gamma(G)$ such that for all Borel objectives Φ and all states s of S, we have $\langle\!\langle 1 \rangle\!\rangle_{val}(\Phi)(s) = \langle\!\langle 1 \rangle\!\rangle_{val}(\gamma(\Phi))(\gamma(s))$.*

*Example 6 (***ST** *games with irrational values).* Consider the **PC** game shown in Fig. 4(a). The objective of player 1 is to reach the state s_0. Recall that the set of available moves for player 1 at the states s_2 and s_3 is $\{a, b\}$, and for player 2, it is $\{1, 2\}$. Let the value for player 1 at state s_2 be x. The player 1 strategy σ that plays the moves a and b with probability $1/2$ at s_3 ensures that the value for player 1 at state s_3 is at least $x/2$. Similarly, the player 2 strategy π that plays the moves 1 and 2 with probability $1/2$ at s_3 ensures that the value for player 1 at state s_3 is at most $x/2$. Hence, the value for player 1 at state s_3 is $x/2$. It follows from the characterization of the values of concurrent games as fixpoints of values of matrix games [8] that

$$x = \min \max \begin{bmatrix} 1 & \frac{x}{2} \\ 0 & 1 \end{bmatrix}$$

where the operator min max denotes the optimal value in a matrix game. The solution for x is achieved by solving the following optimization problem:

$$\text{minimize } x \text{ subject to } c + \big((1-c)\cdot x\big)/2 \le x \text{ and } 1-c \le x.$$

Intuitively, c is the probability to choose move a in an optimal strategy. The solution to the optimization problem is achieved by setting $x = 1-c$. Hence, $c + (1-c)^2/2 = (1-c)$, which implies $(1+c)^2 = 2$. Since c must lie in the interval $[0,1]$, we conclude that $c = \sqrt{2}-1$. Thus the value for player 1 at state s_2 is $x = 2-\sqrt{2}$. By Theorem 3 it follows that the player 1 value at state $\widetilde{s}_3$ is also $x/2$, which is irrational. ∎

Values and determinacy of ST games. Example 3 shows that the sure determinacy of **PT** games does not extend to **ST** games. Example 6 shows that the values in **ST** games can be irrational even for reachability objectives. Theorem 2 and Theorem 3 establish the quantitative determinacy for **ST** games.

Corollary 1 (Values and determinacy of ST games).

1. *There exists an* **ST** *game with a reachability objective Φ and a state s such that $s \notin (\langle\langle 1\rangle\rangle_{sure}(\Phi) \cup \langle\langle 2\rangle\rangle_{sure}(\Omega \setminus \Phi))$.*
2. *There exists an* **ST** *game with a reachability objective Φ and a state s such that the value $\langle\langle 1\rangle\rangle_{val}(\Phi)(s)$ for player 1 at s for Φ is irrational.*
3. *For all* **ST** *games, all Borel objectives Φ, and all states s, we have $\langle\langle 1\rangle\rangle_{val}(\Phi)(s) + \langle\langle 2\rangle\rangle_{val}(\Omega \setminus \Phi)(s) = 1$.*

Computational complexity of ST games. The result of [18] shows that computing sure winning sets in the general case of partial-information turn-based games, which correspond to games with variables (Example 2) where all four variable sets V_1^{pvt}, V_1^{pub}, V_2^{pvt}, and V_2^{pub} are nonempty, is 2EXPTIME-complete for reachability objectives. Even in the simpler case when $V_1^{\text{pub}} = \emptyset$ or $V_2^{\text{pub}} = \emptyset$, the problem is still EXPTIME-complete. We show that **ST** games, which correspond to the subclass of games with variables with $V_1^{\text{pvt}} = \emptyset$, can be solved considerably more efficiently. The approach to solve a **ST** game by a reduction to an exponential-size **PT** game, using a subset construction, only yields the sure winning sets. However, solving **ST** games by our reduction to **PC** games allows the arbitrarily precise and more efficient computation of values.

Corollary 2 (Complexity of ST games). *For all* **ST** *games, all parity objectives Φ, and all states s,*

1. *whether $s \in \langle\langle 1\rangle\rangle_{sure}(\Phi)$ or $s \in \langle\langle 1\rangle\rangle_{almost}(\Phi)$ or $s \in \langle\langle 1\rangle\rangle_{limit}(\Phi)$ can each be decided in NP $\cap$ coNP;*
2. *for all rational constants r and $\varepsilon > 0$, whether $\langle\langle 1\rangle\rangle_{val}(\Phi)(s) \in [r-\varepsilon, r+\varepsilon]$ can be decided in NP $\cap$ coNP.*

Proof. For all **ST** games the reduction to **PC** games is achieved in linear time. The complexity for computing qualitative winning sets (Part 1 of the corollary) follows from the results of [5]. The complexity for approximating values (Part 2) follows from the results of [3]. ∎

Table 1. Family of strategies for various objectives, where Σ^{PM} denotes the family of pure memoryless strategies, Σ^{M} denotes the family of randomized memoryless strategies, and Σ^{HI} denotes the family of randomized history-dependent, infinite-memory strategies.

objective	sure	almost-sure	limit-sure	ε-optimal
safety	Σ^{PM}	Σ^{PM}	Σ^{PM}	Σ^{M}
reachability	Σ^{PM}	Σ^{M}	Σ^{M}	Σ^{M}
coBüchi	Σ^{PM}	Σ^{M}	Σ^{M}	Σ^{M}
Büchi	Σ^{PM}	Σ^{M}	Σ^{HI}	Σ^{HI}
parity	Σ^{PM}	Σ^{HI}	Σ^{HI}	Σ^{HI}

Sufficiency of strategies for ST games. Our reduction of **ST** games to **PC** games and the characterization of memory requirements for **PC** games with parity objectives [5, 3] gives the following corollary.

Corollary 3 (ε-optimal strategies for ST games). *The most restrictive family of strategies that suffices for sure, almost-sure, and limit-sure winning, and for ε-optimality with $\varepsilon > 0$, for* **ST** *games with respect to different classes of parity objectives is given in Table 1.*

4 Conclusion

We introduced and analyzed **ST** (semiperfect-information turn-based) games, the subclass of partial-information turn-based games where one player has partial knowledge about the state of the game and the other player has complete knowledge. These games provide a better model for controller synthesis than **PT** (perfect-information turn-based) games, by allowing the plant to have private variables that are inaccessible to the controller, and they can be solved at much lower computational costs than the full class of partial-information turn-based games. We established the equivalence of **ST** games and **PC** (perfect-information concurrent) games and thus precisely characterize the class of **ST** games.

Semiperfect-information turn-based stochastic games. The class of **ST** *stochastic* games is the generalization of **ST** games where the transition function is probabilistic rather than deterministic. Similarly, the **PC** stochastic games are the generalization of **PC** games with probabilistic transition functions. The equivalence of **ST** games and **PC** games extends to the stochastic case in a straight-forward manner, i.e., the **ST** stochastic games can be reduced to **PC** stochastic games, and vice versa. The reductions are similar to the reductions for the nonstochastic case. Consequently, results analogous to Theorem 3, Corollary 1, Corollary 2, and Corollary 3 follow for **ST** stochastic games.

Acknowledgments. This research was supported in part by the AFOSR MURI grant F49620-00-1-0327 and the NSF ITR grant CCR-0225610.

References

1. R. Alur, T.A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49:672–713, 2002.
2. J.R. Büchi and L.H. Landweber. Solving sequential conditions by finite-state strategies. *Transactions of the AMS*, 138:295–311, 1969.
3. K. Chatterjee, L. de Alfaro, and T.A. Henzinger. The complexity of quantitative concurrent parity games. In *SODA 06*. ACM Press, 2006.
4. K. Chatterjee, R.Majumdar, and M. Jurdziński. On Nash equilibria in stochastic games. In *CSL 04*, volume 3210 of *LNCS*, pages 26–40. Springer, 2004.
5. L. de Alfaro and T.A. Henzinger. Concurrent omega-regular games. In *LICS 00*, pages 141–154. IEEE Computer Society Press, 2000.
6. L. de Alfaro and T.A. Henzinger. Interface theories for component-based design. In *EMSOFT 01*, volume 2211 of *LNCS*, pages 148–165. Springer, 2001.
7. L. de Alfaro, T.A. Henzinger, and O. Kupferman. Concurrent reachability games. In *FOCS 98*, pages 564–575. IEEE Computer Society Press, 1998.
8. L. de Alfaro and R. Majumdar. Quantitative solution of omega-regular games. In *STOC 01*, pages 675–683. ACM Press, 2001.
9. E.A. Emerson and C. Jutla. The complexity of tree automata and logics of programs. In *FOCS 88*, pages 328–337. IEEE Computer Society Press, 1988.
10. E.A. Emerson and C. Jutla. Tree automata, mu-calculus, and determinacy. In *FOCS 91*, pages 368–377. IEEE Computer Society Press, 1991.
11. T.A. Henzinger, O. Kupferman, and S. Rajamani. Fair simulation. *Information and Computation*, 173:64–81, 2002.
12. M. Jurdzinski. Small progress measures for solving parity games. In *STACS 00*, volume 1770 of *LNCS*, pages 290–301. Springer, 2000.
13. D.A. Martin. Borel determinacy. *Annals of Mathematics*, 102:363–371, 1975.
14. D.A. Martin. The determinacy of Blackwell games. *The Journal of Symbolic Logic*, 63:1565–1581, 1998.
15. R. McNaughton. Infinite games played on finite graphs. *Annals of Pure and Applied Logic*, 65:149–184, 1993.
16. A. Pnueli and R. Rosner. On the synthesis of a reactive module. In *POPL 89*, pages 179–190. ACM Press, 1989.
17. P.J. Ramadge and W.M. Wonham. Supervisory control of a class of discrete-event processes. *SIAM Journal of Control and Optimization*, 25:206–230, 1987.
18. J.H. Reif. Universal games of incomplete information. In *STOC 79*, pages 288–308. ACM Press, 1979.
19. W. Thomas. Languages, automata, and logic. In *Handbook of Formal Languages*, volume 3, chapter 7, pages 389–455. Springer, 1997.
20. J. Vöge and M. Jurdziński. A discrete strategy improvement algorithm for solving parity games. In *CAV 00*, volume 1855 of *LNCS*, pages 202–215. Springer, 2000.